

本科毕业论文

题 目：基于随机森林的电力

能耗预测

院 （部）： 计算机科学与技术学院

专 业： 软件工程

班 级： 软件142班

姓 名： 王震

学 号： 20141114048

指导教师： 宋玲

完成日期： 2018年5月25日

目 录

[摘 要 III](#_Toc515219898)

[ABSTRACT IV](#_Toc515219899)

[1前 言 1](#_Toc515219900)

[1.1研究意义 1](#_Toc515219901)

[1.2本文主要内容 1](#_Toc515219902)

[1.3本文主要工作 1](#_Toc515219903)

[1.4本章小结 2](#_Toc515219904)

[2相关研究 3](#_Toc515219905)

[2.1随机森林算法的相关研究 3](#_Toc515219906)

[2.2基于随机森林建模与应用相关研究 3](#_Toc515219907)

[3基于机器学习的随机森林算法 6](#_Toc515219908)

[3.1数据预处理 6](#_Toc515219909)

[3.2决策树 8](#_Toc515219910)

[3.3信息熵 12](#_Toc515219911)

[3.4随机森林算法 12](#_Toc515219912)

[3.5评价函数 16](#_Toc515219913)

[3.6本章小结 18](#_Toc515219914)

[4基于RandomForest的模型设计与实现 19](#_Toc515219915)

[4.1环境配置 19](#_Toc515219916)

[4.2数据预处理 19](#_Toc515219917)

[4.2.1数据集的合成 20](#_Toc515219918)

[4.2.2数据集标准化 20](#_Toc515219919)

[4.2.3数据集切分 21](#_Toc515219920)

[4.3随机森林模型 22](#_Toc515219921)

[4.3.1 应用RandomForestRegressor建立模型 22](#_Toc515219922)

[4.3.2参数调优 23](#_Toc515219923)

[4.3.3计算正确率 24](#_Toc515219924)

[4.3.4评价函数 24](#_Toc515219925)

[4.3.5预测结果的比较与分析 25](#_Toc515219926)

[4.4 本章小结 25](#_Toc515219927)

[5结 论 26](#_Toc515219928)

[谢 辞 27](#_Toc515219929)

[参考文献 28](#_Toc515219930)

# 摘 要

“加快生态文明体制改革、建设美丽中国”是十九大报告的重要议题，而建筑领域绿色低碳发展是实现“美丽中国”目标的重要途径；房屋建筑在全寿命周期中消费了大量资源和能源，对生态文明建设产生巨大影响。

为了在改善住宅建筑的能源性能方面进行高效且有效的城市规划，我们需要对其影响特征有一个清晰的认识。以往我们对影响特征与能耗之间建模的研究仍存在些许空白和限制。建筑环境作为能源消耗的主要载体，通过科学、合理、准确地预测建筑运行能耗是设定合理、明确的节能目标，制定建筑节能政策、法规，以及开展建筑节能工作的重要前提条件，也是实现建筑能耗需求同其他经济领域协调、可持续发展的重要保障。在这种宏观趋势下,加强对建筑能耗的预测和分析,具有重要的理论和实际意义。

本文工作的主要概括如下：了解随机森林算法，针对固定建筑物每日的气象数据进行数据提取和处理，使用随机森林算法对已收集数据集进行训练，获得预测正确率的模型，之后使用其他多种算法训练出该方法下预测正确率最高的模型；然后针对两种模型分析得出所有所用算法中更适合于解决此类问题的算法；最终得出最佳的预测模型。

关键词：建筑数据；数据处理；随机森林；

**Power consumption prediction based on Random Forest**

# ABSTRACT

"Accelerating the reform of the ecological civilization system and building a beautiful China" is an important topic in the nineteenth report. House buildings consume a lot of resources and energy in the whole life cycle, which has a great impact on the construction of ecological civilization.

Efficient and effective city planning in improving the energy performance of residential buildings requires a clear understanding of the influential features. Previous studies on modeling the relationships between influential features and the energy consumption have several gaps and limitations, such as the linear modeling methodology and insufficient consideration of particular features. Building environment as the main carrier of energy consumption, through the scientific, reasonable, accurately predict building energy consumption is to set reasonable and clear energy saving goal, building energy conservation policies and regulations, and carry out the work of building energy efficiency and important premise condition, is to realize building energy consumption demand with the rest of the economy to coordinate, the sustainable development important guarantee. Under this macro trend, it is of great theoretical and practical significance to strengthen the prediction and analysis of building energy consumption.

Understand random forest algorithm In this paper, the main work summarized as follows: in view of the fixed building daily meteorological data extraction and data processing, using the random forest algorithm to collect training data set, get the prediction accuracy of the model, then using a variety of other algorithm training the model with highest accuracy under the method; Then, based on the analysis of two models, all the algorithms used are more suitable for solving such problems. Finally, the best prediction model is obtained.

**Key words**: user activity; data processing; clustering; Random Forests

# 1前 言

目前，发达国家建筑能耗占国家总能耗30％--40％，其中建筑运行能耗占30％以上。我国建筑能耗占社会商品总能耗的27％左右，暖通空调能耗约又占建筑能耗的50％～70％，建筑环境设备优化操作和管理可实现20％-30％的能源节约。有研究指出，建筑消耗了约40％的全球能源，其温室气体排放量约占全球温室气体排放的33％。随着全球气候变化、能源短缺、大气污染日益严重，特别是近几年，全国大范围出现的雾霾天气和PM2．5值严重超标，给我国能源生产和消费方式提出了严峻挑战。

1.1研究意义

“加快生态文明体制改革、建设美丽中国”是十九大报告的重要议题，而建筑领域绿色低碳发展是实现“美丽中国”目标的重要途径。

当前，房屋建筑在全寿命周期中消费了大量资源和能源，2015年全国建筑能耗占全国能源消费总量的20%，对生态文明建设产生巨大影响。同时，新时代下建筑还承载着人民对更加健康舒适的美好居住空间的需求。因此，加快推进建筑节能工作是实现绿色低碳发展的重要举措，而建筑能耗数据是科学推进建筑节能工作的基础，但当前建筑能耗权威数据缺失，相关研究缺乏系统性。

建筑环境作为能源消耗的主要载体，通过科学、合理、准确地预测建筑运行能耗是设定合理、明确的节能目标，制定建筑节能政策、法规，以及开展建筑节能工作的重要前提条件，也是实现建筑能耗需求同其他经济领域协调、可持续发展的重要保障。在这种宏观趋势下,加强对建筑能耗的预测和分析,具有重要的理论和实际意义。

有结果之后有什么应用 预测的意义？？？电能的调度

1.2本文主要内容

本文的主要内容是对随机森林算法的发展和其实际应用情况进行总结，之后对随机森林算法的原理进行研究，然后记录数据集的分析过程，对所要分析的数据集进行预处理。最后使用多种训练出多种不同回归算法的预测模型并选择其中最优秀的预测模型。回归模型基于变量和趋势之间的关系的分析，以便做出关于连续变量的预测。本设计的主要思想是对建筑的电力能耗的数据进行分析，构建预测模型，对能量消耗进行预测。

概括 三四行

第一章

第二章

第三章

1.3本文主要工作

本文工作的主要概括如下：针对固定建筑物每日的气象数据进行数据提取和处理，使用随机森林算法对已收集数据集进行训练，获得预测正确率的模型，之后使用其他多种算法训练出该方法下预测正确率最高的模型；然后针对两种模型分析得出所有所用算法中更适合于解决此类问题的算法；最终得出最佳的预测模型。

具体的工作流程为:

1. 学习并总结随机森林算法使用案例和相关研究
2. 研究随机森林算法原理
3. 数据预处理
4. 使用随机森林算法以及其他算法进行建模，得到回归模型
5. 对得到的回归模型进行调参，得到准确率最高的模型

画一个框架图processon？？？

1.4本章小结

本章主要介绍了当今建筑能耗在国家能源消耗中的重要性，进一步通过对建筑能耗进行预测并且进行分析得出如何进行建筑能耗的节能的最佳方法，这也是这项研究的意义所在。介绍了整篇文章的主要内容和流程，最后对本文中主要的工作进行了概括和总结。

# 2相关研究

每章开头的帽子简介

下文针对这个本文主题的论文 10-20篇进行综述 参考文献

国外相关研究 国内相关研究

加上2．3节基于随机森林的建筑能耗预测相关研究

2.1中

2.2节 描述的详细一点 包括什么作者在什么中提到了什么 去掉一

2.1随机森林算法的相关研究

近年来，随机森林模型在界内的关注度与受欢迎程度有着显著的提升，这多半归功于它可以快速地被应用到几乎任何的数据科学问题中去，从而使人们能够高效快捷地获得第一组基准测试结果。在各种各样的问题中，随机森林一次又一次地展示出令人难以置信的强大，而与此同时它又方便实用。需要大家注意的是，随机森林方法有自己的局限性。

谈及随机森林算法的产生与发展，我们必须回溯到20世纪80年代。可以说，该算法是Leo Breiman, Adele Cutler, Ho Tin Kam, Dietterich, Amit和Geman这几位大师呕心沥血的共同结晶，他们中的每个人都对随机森林算法的早期发展作出了重要的贡献。Leo Breiman和 Adele Cutler最早提出了执行随机森里的关键算法，这一算法也成为了他们的专利之一。Amit, Gemen和Ho Tim Kam各自独立地介绍了特征随即选择的思想，并且运用了Breiman的“套袋”思想构建了控制方差的决策树集合。在此之后，Deitterich在模型中引入了随机节点优化的思想，对随机森林进行了进一步完善。

随机森林是一种多功能的机器学习算法，能够执行回归和分类的任务。同时，它也是一种数据降维手段，用于处理缺失值、异常值以及其他数据探索中的重要步骤，并取得了不错的成效。另外，它还担任了集成学习中的重要方法，在将几个低效模型整合为一个高效模型时大显身手。

在随机森林中会生成很多的决策树，并不像在CART模型里一样只生成唯一的树。当在基于某些属性对一个新的对象进行分类判别时，随机森林中的每一棵树都会给出自己的分类选择，并由此进行“投票”，森林整体的输出结果将会是票数最多的分类选项；而在回归问题中，随机森林的输出将会是所有决策树输出的平均值。

2.2基于随机森林建模与应用相关研究

随着机器学习在技术和商业的发展，随机森林的在实际问题的应用场景已经很多，下面举几个使用随机森林系统的案例：

一、基于随机森林的股权众筹项目风险评估研究，内容是依托于大众融资的互联网金融，正适合目前我国的创业经济发展潮流，为中小微企业提供了一个崭新的融资方式. 但是，正在探索发展的众筹行业也面临诸多困境，我国的众筹体制及相应的法律法规还不完善，众筹平台在快速增长吸引了大量项目的同时，其风险和危机也与日俱增，违约、 非法集资、携款潜逃等问题频频出现在众筹项目中.随着互联网金融的发展，金融交易数据必将激增，信息更新速度将会越来越快，如何有效地利用现有数据改善当前的众筹风险问题至是个值待关注和探讨的问题。因此，对众筹项目进行风险评估与管理，为我国众筹投资者提供投资依据，具有重要的现实意义。

二、基于随机森林的太阳能辐射预测模型原因是随着环境污染问题的加重，近年来太阳能作为最理想的清洁新能源备受关注。准确的掌握太阳能辐射情况对于太阳能的利用尤为关键。为了高效利用太阳能，准确的预测太阳能辐射情况极其重要。针对太阳能辐射的 预测问题，研究了基于随机森林的太阳能辐射预测方法，根据影响太阳能辐射的因子建立了随机森林树型分类器，构建了一种基于随机森林的太阳能辐射量预测模型。准确有效的太阳能辐射预测方法对于合 理开发太阳能资源、提高光伏发电的效率具有重要 的实际意义。

三、基于随机森林的月貌面向对象分类，针对月球地貌分类的研究较少且没有相对简单的方法，提出将面向对象和随机森林相结合的方法对月貌进行分类。月貌能直观地反映月表特征及其目前的状态，也记录着月球形成和演变的历史信息。月表地貌学 的研究有利于加深认识和理解月表形貌，对月表年龄的估算、月壤厚度的反演、月球起源及演化历史的探索等具有重要意义，同时可为月球资源的开发利用提供基础。

四、基于随机森林方法的地铁车门故障诊断，针对现有地铁车门故障诊断方法存在的诊断速度慢以及大量故障检修数据未得到合理利用等问题，提出一种基于信息增益率的随机森林故障诊断方法。针对现有地铁车门故障诊断方法存在的诊断速度慢以及大量故障检修数据未得到合理利用等问题，提出一种基于信息增益率的随机森林故障诊断方法。该方法将地铁车门历史故障数据集转化成决 策表，通过Bootstrap重抽样，建立多棵基于信息增益率的决策树，形成随机森林故障诊断模型，实现地铁车门故障的快速诊断。

五、基于信息增益和随机森林分类器的入侵检测系统研究目的是解决互联网以及其他网络上的安全数据通信总是会受到入侵以及滥用等威胁。目前，许多误用检测系统无法检测未知攻击，而异常检测系统虽然能够精确检测未知攻击，但由于入侵检测固有的特性，入侵事件与正常事件类间存在极大的不平衡性，这导致很难利用机器学习的方法高效地进行入侵行为检测。为此提出了一种基于信息增益和随机森林分类器的入侵检测系统.为了解决类之间的不平衡性，对训练数据集应用了合成少数过采样算法。

六、利用随机森林和纹理特征的森林类型识别，结合遥感判读样地、植被指数、纹理信息以及地形因子等多源数据，构建最小距离分类模型、支持向量机分类模型和随机森林分类模型，对黑龙江凉水自然保护区森林优势树种进行分类。

七、随机森林方法在玉米-大豆精细识别中的应用，研究基于遥感影像的作物精确识别技术方法, 对获取作物分布信息具有重要意义。

通过随机森林算法及其思想，解决了许多实现起来困难的现实问题，这说明随机森林算法针对某些特定问题有着很好的性能，也说明了随机森林适合解决某一类特殊问题。

2.3 本章小结

在机器学习中，随机森林算法的使用场景非常广泛，技术本身是一种多功能的机器学习算法，能够执行回归和分类的任务。可以快速地被应用到几乎任何的数据科学问题中去，从而使人们能够高效快捷地获得第一组基准测试结果。在实际问题的解决上，完美或者很好的完成了诸如基于随机森林的股权众筹项目风险评估研究、基于随机森林的太阳能辐射预测模型、基于随机森林的月貌面向对象分类等困难问题，这也展示出了随机森林算法的优势所在。

# 3基于机器学习的预测

3.1数据预处理

每种方法补充相应的话，写的更加详细

属性构造

加进来的都是能看的懂的 理解的 用不到的不用写

现实世界中数据大体上都是不完整，不一致的脏数据，无法直接进行数据挖掘，或挖掘结果差强人意。为了提高数据挖掘的质量产生了数据预处理技术。数据预处理有多种方法：数据清理，数据集成，数据变换，数据归约等。这些数据处理技术在数据挖掘之前使用，大大提高了数据挖掘模式的质量，降低实际挖掘所需要的时间。

1. 数据清洗
2. 缺失值处理

处理缺失值分为三类：删除记录、数据补差和不处理。

数据补插方法：

1. 补插均值/中位数/众数

2. 使用固定值

3. 最近邻补插

4. 回归方法

5. 插值法：拉格朗日插值法、牛顿插值法、Hermit插值法、分段插值、样条插值

1. 异常值处理
2. 删除有异常值的记录
3. 视为缺失值
4. 平均值修正
5. 不处理

要分析异常值的原因，再决定取舍。

1. 数据集成

将多个数据源放在一个统一的数据仓库中。

1. 实体识别

同名异义

异名同义

单位不统一

1. 冗余属性识

同一属性多次出现

同一属性命名不一致

1. 数据变换

对数据进行规范化处理

1. 简单函数变换

原始数据进行数学函数变换，平方、开方、取对数、差分运算。用来将不具有正太分布的数据变换成具有正太性的数据。

时间序列分析中，对数变换或者差分运算可以将非平稳序列转换为平稳序列。

1. 规范化

消除指标间规范化

1. 最小-最大规范化
2. 零-均值规范化
3. 小数指定标规范化
4. 连续属性性离散化

将连续属性变为分类属性，即连续属性离散化。数据离散化本质上通过断点集合将连续的属性空间划分为若干区，最后用不同的符号或者整数值代表落在每个子区间中的数据。离散化涉及两个子任务：确定分类以及如何将连续属性值映射到这些分类值。

1. 等宽法
2. 等频法

基于聚类分析的方法

1. 属性构造

利用已有的属性构造新的属性，并加到现有的属性中。

1. 小波变换

非平稳序列的分析手段。

基于小波变换的特征提取方法：

1. 基于小波变换的多尺度空间能量分布特征提取法
2. 基于小波变换的多尺度空间的模极大特征值提取法
3. 基于小波变换的特征提取方法
4. 基于适应性小波神经网络的特征提取方法
5. 数据规约

降低无效，错误数据对建模的影响，提高建模的准确性。   
少量且代表性的数据将大幅缩减数据挖掘所需时间。   
降低存储数据成本。

1. 属性规约
2. 合并属性
3. 逐步向前选择
4. 逐步向后删除
5. 决策树归纳
6. 主成分分析
7. 数值规约

通过选择替代的、较小的数据来减少数据量，包含有参数方法和无参数方法两类；有参数方法使用模型评估数据，不需要存放真实数据，只需要存放参数，例如回归、对数线性模型。无参数需要数据，例如直方图、聚类、抽样。

3.2特征重要性

3.3机器学习模型

3.5交叉验证

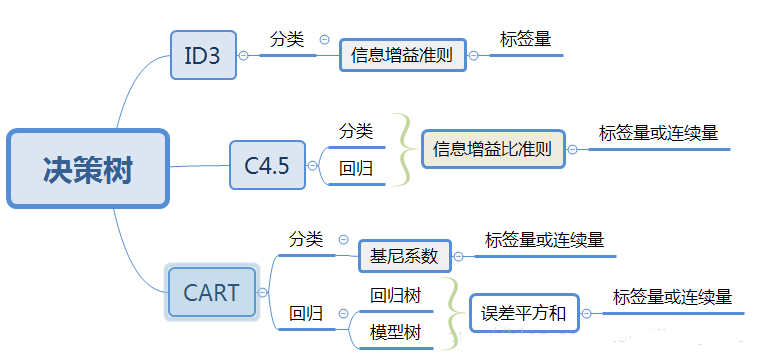
3.4评价函数

3.2机器学习模型

3.2.1决策树

3.2.1svm

3.2.1决策树3.2.2svm



1. 决策树模型
2. 定义

决策树是一种监督学习算法。它适用于类别和连续输入（特征）和输出（预测）变量。基于树的方法把特征空间划分成一系列矩形，然后给每一个矩形安置一个简单的模型（像一个常数）。从概念上来讲，它们是简单且有效的。分类决策树模型是一种描述对实例进行分类的树形结构。决策树由结点和有向边组成。结点有两种类型：内部节点和叶节点，内部节点表示一个特征或属性，叶节点表示一个类。

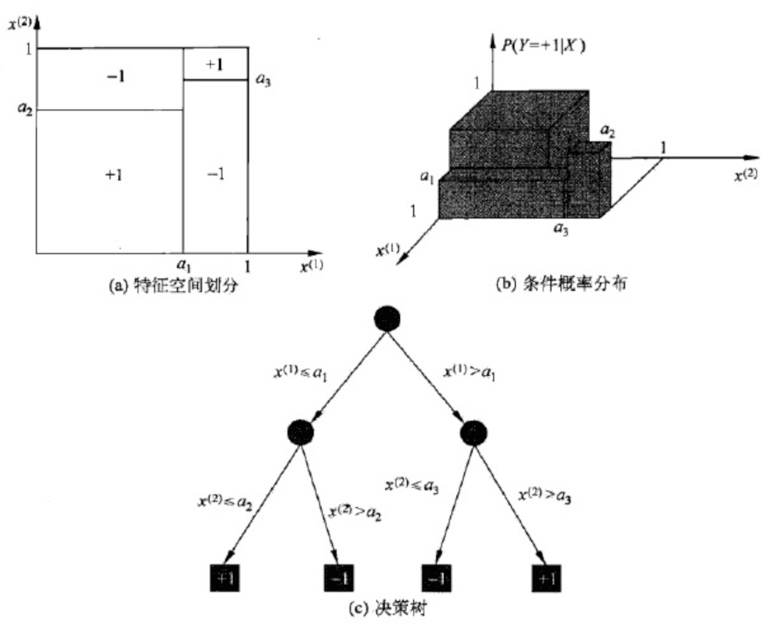
分类的时候，从根节点开始，对实例的某一个特征进行测试，根据测试结果，将实例分配到其子结点；此时，每一个子结点对应着该特征的一个取值。如此递归向下移动，直至达到叶结点，最后将实例分配到叶结点的类中。

1. 决策树与if-then规则

现在我们可以更抽象一些。决策树可以看成一个if-then规则的集合：由决策树的根结点到叶结点的每一条路径构建一条规则；路径上的内部结点的特征对应着规则的条件，而叶结点对应着分类的结论。决策树的路径和其对应的if-then规则集合是等效的，它们具有一个重要的性质：互斥并且完备。这里的意思是说：每一个实例都被一条路径或一条规则所覆盖，而且只被一条规则所覆盖。

1. 决策树与条件概率分布

决策树还是给定特征条件下类的条件概率分布的一种表示。该条件分布定义在特征空间的划分（partition）上，特征空间被划分为互不相交的单元（cell），每个单元定义一个类的概率分布就构成了一个条件概率分布。决策树的一条路径对应于划分中的一个单元。决策树所表示的条件概率分布由各个单元给定条件下类的条件概率分布组成。给定实例的特征X，一定落入某个划分，决策树选取该划分里最大概率的类作为结果输出。如图：



1. 决策树的学习

决策树学习算法包含特征选择、决策树的生成与剪枝过程。决策树的学习算法通常是递归地选择最优特征，并用最优特征对数据集进行分割。开始时，构建根结点，选择最优特征，该特征有几种值就分割为几个子集，每个子集分别递归调用此方法，返回结点，返回的结点就是上一层的子结点。直到所有特征都已经用完，或者数据集只有一维特征为止。

1. 特征选择

在下一节熵中会详细解释

1. 决策树的生成

此处主要介绍两种决策树学习的算法：ID3和C4.5。

1、ID3算法由Ross Quinlan发明，建立在“奥卡姆剃刀”的基础上：越是小型的决策树越优于大的决策树（be simple简单理论）。ID3算法中根据信息增益评估和选择特征，每次选择信息增益最大的特征作为判断模块建立子结点。ID3算法可用于划分标称型数据集，没有剪枝的过程，为了去除过度数据匹配的问题，可通过裁剪合并相邻的无法产生大量信息增益的叶子节点（例如设置信息增益阀值）。使用信息增益的话其实是有一个缺点，那就是它偏向于具有大量值的属性。就是说在训练集中，某个属性所取的不同值的个数越多，那么越有可能拿它来作为分裂属性，而这样做有时候是没有意义的，另外ID3不能处理连续分布的数据特征，于是就有了C4.5算法。

2、C4.5算法用信息增益率来选择属性，继承了ID3算法的优点。并在以下几方面对ID3算法进行了改进：

（1）克服了用信息增益选择属性时偏向选择取值多的属性的不足

（2）在树构造过程中进行剪枝

（3）能够完成对连续属性的离散化处理

（4）能够对不完整数据进行处理

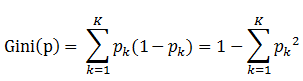
C4.5算法产生的分类规则易于理解、准确率较高；但效率低，因树构造过程中，需要对数据集进行多次的顺序扫描和排序。也是因为必须多次数据集扫描，C4.5只适合于能够驻留于内存的数据集。在实现过程中，C4.5算法在结构与递归上与ID3完全相同，区别只在于选取决决策特征时的决策依据不同，二者都有贪心性质：即通过局部最优构造全局最优。

1. CART算法

CART（Classification And Regression Trees，分类回归树）算法，CART是一个独立于其他经典决策树算法的算法，所以导致CART相对来说较为复杂。因为它不仅仅可以作为分类树，还可以作为回归树。采用的是Gini指数（选Gini指数最小的特征s）作为分裂标准,同时它也是包含后剪枝操作。ID3算法和C4.5算法虽然在对训练样本集的学习中可以尽可能多地挖掘信息，但其生成的决策树分支较大，规模较大。为了简化决策树的规模，提高生成决策树的效率，就出现了根据GINI系数来选择测试属性的决策树算法CART。

基尼指数：表示在样本集合中一个随机选中的样本被分错的概率。

基尼指数（基尼不纯度）= 样本被选中的概率 \* 样本被分错的概率



CART分类生成算法

输入：训练数据集D，停止计算的条件；

输出：CART决策树；

根据训练数据集，从根结点开始，递归地对每个结点进行以下操作，构建二叉决策树：

1、设结点的训练数据集为D，计算现有特征对该数据集的基尼指数。此时，对每一个特征A，对其可能取的每个值a，根据样本点对A=a的测试为“是”或“否”将D分割成D1和D2两部分，之后计算基尼指数。

2、在所有可能的特征A以及它们所有可能的切分点a中，选择基尼指数最小的特征及其对应的切分点作为最有特征与最优切分点。依照最优特征与最优切分点，从现有结点生成两个子节点，将训练数据集按照特征分配到两个子节点中去。

3、对两个子结点递归地调用两个子结点，将训练数据集按特征分配到两个子节点中去。

4、生成CART决策树。

五、剪枝

在决策树学习中将已生成的树进行简化的过程称为剪枝。决策树的剪枝往往通过极小化决策树的损失函数或代价函数来实现。实际上剪枝的过程就是一个动态规划的过程：从叶结点开始，自底向上地对内部结点计算预测误差以及剪枝后的预测误差，如果两者的预测误差是相等或者剪枝后预测误差更小，当然是剪掉的好。但是如果剪枝后的预测误差更大，那就不要剪了。剪枝后，原内部结点会变成新的叶结点，其决策类别由多数表决法决定。不断重复这个过程往上剪枝，直到预测误差最小为止。

3.3信息熵

1、信息增益（ID3）

信息增益：以某特征划分数据集前后的熵的差值。

在熵的理解那部分提到了，熵可以表示样本集合的不确定性，熵越大，样本的不确定性就越大。因此可以使用划分前后集合熵的差值来衡量使用当前特征对于样本集合D划分效果的好坏。

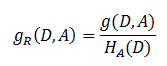
使用某个特征A划分数据集D，计算划分后的数据子集的熵 entroy(后)

信息增益 = entroy(前) - entroy(后)



2、信息增益比（C4.5算法）

信息增益比 = 惩罚参数 \* 信息增益



信息增益比本质是在信息增益的基础之上乘上一个惩罚参数。特征个数较多时，惩罚参数较小；特征个数较少时，惩罚参数较大。

3.4随机森林算法

1、概念

随机森林是一种多功能的机器学习算法，能够执行回归和分类的任务。同时，它也是一种数据降维手段，用于处理缺失值、异常值以及其他数据探索中的重要步骤，并取得了不错的成效。另外，它还担任了集成学习中的重要方法，在将几个低效模型整合为一个高效模型时大显身手。

在随机森林中，我们将生成很多的决策树，并不像在CART模型里一样只生成唯一的树。当在基于某些属性对一个新的对象进行分类判别时，随机森林中的每一棵树都会给出自己的分类选择，并由此进行“投票”，森林整体的输出结果将会是票数最多的分类选项；而在回归问题中，随机森林的输出将会是所有决策树输出的平均值。

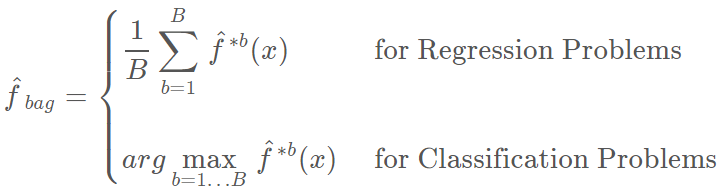
在随机森林算法中，用到了许多个决策树。决策树是一种监督学习算法，它适用于类别和连续输入（特征）和输出（预测）变量。基于树的方法把特征空间划分成一系列矩形，然后给每一个矩形安置一个简单的模型（像一个常数）。从概念上来讲，它们是简单且有效的。首先我们通过一个例子来理解决策树。然后用一种正规分析方法来分析创建决策树的过程。分类和回归树（简称 CART）是 Leo Breiman 引入的术语，指用来解决分类或回归预测建模问题的决策树算法。

2、袋装（Bootstrap Aggregating——Bagginng）

在统计学中，Bootstrap 是依靠替换随机采样的任意试验或度量。我们从上文可以看见，决策树会受到高方差的困扰。这意味着如果我们把训练数据随机分成两部分，并且给二者都安置一个决策树，我们得到的结果可能就会相当不同。Bootstrap 聚集，或者叫做袋装，是减少统计学习方法的方差的通用过程。

给定一组 n 个独立的样本观测值 Z\_1，Z\_2，...，Z\_n，每一个值的方差均为 \*σ^\*2，样本观测值的均值方差为 \*σ^\*2/\*n\*。换句话说，对一组观测值取平均会减小方差。因此一种减小方差的自然方式，也就是增加统计学习方法预测精度的方式，就是从总体中取出很多训练集，使用每一个训练集创建一个分离的预测模型，并且对预测结果求取平均值。

这里有一个问题，即我们不能获取多个训练数据集。相反，我们可以通过从（单一）训练数据集提取重复样本进行自助法（bootstrap）操作。在这种方法中，我们生成了 B 个不同的自助训练数据集。我们随后在第 b 个自助训练数据集得到了一个预测结果，从而获得一个聚集预测（aggregate prediction）。



这就叫做袋装（bagging）。注意，聚集（aggregating）在回归和分类问题中可能有不同的均值。当平均预测值在回归问题中的效果很好时，我们将会需要使用多数票决（majority vote）：由于分类问题中的聚集机制，整体预测就是在 B 个预测值中最常出现的那个主要类别。

3、Out-of-Bag（OOB）误差

Bagging 方法最大的优势是我们可以不通过交叉验证而求得测试误差。回想一下，Bagging 方法的精髓是多棵树可以重复地拟合观察样本的自助子集。平均而言，每一个袋装树可以利用 2/3 的观察样本。而剩下的 1/3 观察样本就可以称为 out-of-bag (OOB) 观察样本，它们并不会拟合一一棵给定袋装树。我们可以使用每一棵树的 OOB 观察样本而计算第 i 个观察样本的预测值，这将会导致大约有 B/3 的预测值可以预测第 i 个观察样本。现在我们可以使用和 Bagging（平均回归和大多数投票分类）类似的聚集技术，我们能获得第 i 个观察样本的单一预测值。我们可以用这种方式获得 n 个观察样本的 OOB 预测，因此总体的 OOB MSE（回归问题）和分类误差率（分类问题）就能计算出来。OOB 误差结果是 Bagging 模型测试误差的有效估计，因为每一个样本的预测值都是仅仅使用不会进行拟合训练模型的样本。

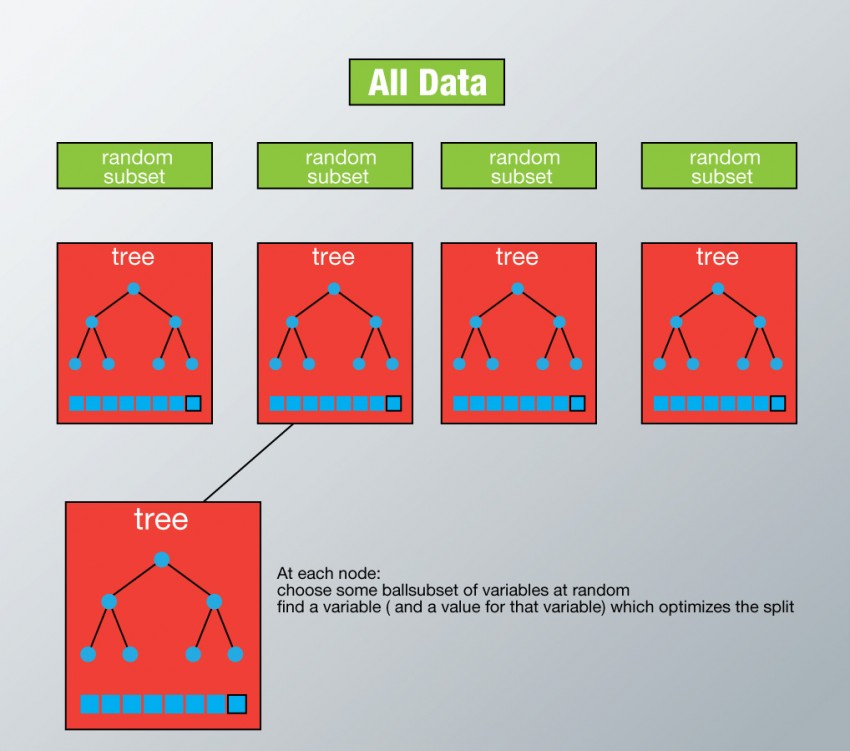
4、特征重要性度量

通过使用单一树，Bagging 通常会提升预测的精确度。但是，解释最终的模型可能很困难。当我们袋装大量的树时，就不再可能使用单一的树表征最终的统计学习流程，因此，Bagging 是以牺牲阐释性能力为代价来提升预测精确度的。有趣的是，一个人可使用 RSS（用于 bagging 回归树）或者基尼指数（用于 bagging 分类树）得到每一个预测器的整体总结。在 bagging 回归树的情况中，我们可以记录由于所有的 B 树上平均的给定预测分子分裂而造成的 RSS 减少的所有数量。一个大的值表示一个重要的预测器。相似地，在 bagging 分类树的情况下，我们可以添加由于所有的 B 树上平均的给定预测分子分裂而造成的基尼系数降低的所有数量。一旦训练完成，sklearn 模块的不同袋装树（bagged tree）学习方法可直接访问特征的重要性数据作为属性。

5、随机森林算法是如何工作的？

在随机森林中，每一个决策树“种植”和“生长”的规则如下所示：

1、假设我们设定训练集中的样本个数为N，然后通过有重置的重复多次抽样来获得这N个样本，这样的抽样结果将作为我们生成决策树的训练集；



2、如果有M个输入变量，每个节点都将随机选择m(m<M)个特定的变量，然后运用这m个变量来确定最佳的分裂点。在决策树的生成过程中，m的值是保持不变的； 3.每棵决策树都最大可能地进行生长而不进行剪枝； 4.通过对所有的决策树进行加总来预测新的数据（在分类时采用多数投票，在回归时采用平均）。

随机森林的优点与缺点

优点：

1、正如上文所述，随机森林算法能解决分类与回归两种类型的问题，并在这两个方面都有相当好的估计表现。

2、随机森林对于高维数据集的处理能力令人兴奋，它可以处理成千上万的输入变量，并确定最重要的变量，因此被认为是一个不错的降维方法。此外，该模型能够输出变量的重要性程度，这是一个非常便利的功能。

3、在对缺失数据进行估计时，随机森林是一个十分有效的方法。就算存在大量的数据缺失，随机森林也能较好地保持精确性。

4、当存在分类不平衡的情况时，随机森林能够提供平衡数据集误差的有效方法

5、模型的上述性能可以被扩展运用到未标记的数据集中，用于引导无监督聚类、数据透视和异常检测。

6、随机森林算法中包含了对输入数据的重复自抽样过程，即所谓的bootstrap抽样。这样一来，数据集中大约三分之一将没有用于模型的训练而是用于测试，这样的数据被称为out of bag samples，通过这些样本估计的误差被称为out of bag error。研究表明，这种out of bag 方法的与测试集规模同训练集一致的估计方法有着相同的精确程度，因此在随机森林中我们无需再对测试集进行另外的设置。

缺点：

1、随机森林在解决回归问题时并没有像它在分类中表现的那么好，这是因为它并不能给出一个连续型的输出。当进行回归时，随机森林不能够作出超越训练集数据范围的预测，这可能导致在对某些还有特定噪声的数据进行建模时出现过度拟合。

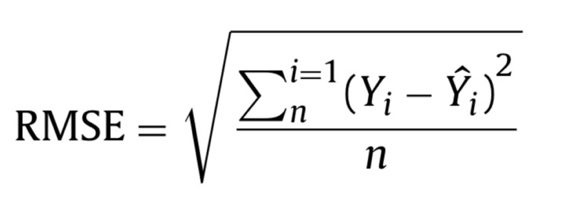
2、对于许多统计建模者来说，随机森林给人的感觉像是一个黑盒子——你几乎无法控制模型内部的运行，只能在不同的参数和随机种子之间进行尝试。

3.5评价函数

1、均方根误差（RMSE）

RMSE是预测值与真实值的误差平方根的均值。

均方根误差RMSE(root-mean-square error)， 均方根误差亦称标准误差,它是观测值与真值偏差的平方与观测次数比值的平方根。均方根误差是用来衡量观测值同真值之间的偏差。标准误差对一组测量中的特大或特小误差反映非常敏感，所以，标准误差能够很好地反映出测量的精密度。可用标准误差作为评定这一测量过程精度的标准。计算公式如下：



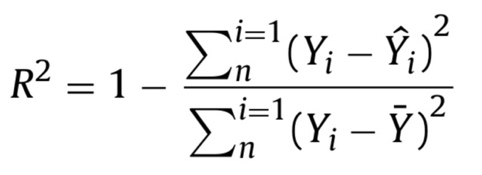
优点：标准化平均方差对均方差进行了标准化改进，通过计算拟评估模型与以均值为基础的模型之间准确性的比率，标准化平均方差取值范围通常为0～1，比率越小，说明模型越优于以均值进行预测的策略，NMSE的值大于1，意味着模型预测还不如简单地把所有观测值的平均值作为预测值，

缺点：但是通过这个指标很难估计预测值和观测值的差距，因为它的单位也和原变量不一样了，综合各个指标的优缺点，我们使用三个指标对模型进行评估。

2、R方（R^2）

R^2方法是将预测值跟只使用均值的情况下相比，看能好多少。其区间通常在（0,1）之间。0表示还不如什么都不预测，直接取均值的情况，而1表示所有预测跟真实结果完美匹配的情况。

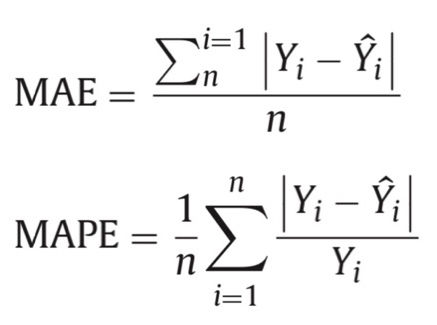
计算公式如下：



3、平均绝对误差（MAPE）平均绝对误差（MAE）

相对百分误差绝对值的平均值MAPE(mean absolute percentage error):可以用来衡量一个模型预测结果的好坏。

MAE平均绝对误差=︱原值-估计值︱/n



3.6本章小结

本章主要内容详细的介绍了数据集预处理的方法和随机森林算法原理，包括随机森林的基础决策树、决策树的构建原理、随机森林算法以及四种评价函数。

# 

# 4基于RandomForest的模型设计与实现

4.1环境配置

操作系统：macOS High Sierra

IDE: PyCharm CE

语言：

4.2数据集介绍

按照3章中的顺序进行分析

做一个表

4.2数据预处理

数据集使用的是源于蒙斯大学的电量能耗数据energydata.csv

数据集包含每十分钟提取一次的信息，共约4.5个月，采用ZigBee无线传感器网络对室内温湿度条件进行监测。最近的气象站(比利时Chievres机场)的天气信息可以从公共数据集网站下载，并与使用日期和时间列的实验数据集合并在一起。

数据集共19735个实例；

数据集共有29个特征，分别为：时间、能源使用、电灯、厨房温度、厨房湿度、客厅温度、客厅湿度、洗衣房温度、洗衣房湿度、办公室温度、办公室湿度、浴室温度、浴室湿度、建筑物北侧温度、建筑物北侧湿度、熨烫室温度、熨烫室湿度、青少年房间温度、青少年房间湿度、家长室温度、家长室湿度、气象站温度、气象站湿度、气象站压力、气象站风速、能见度、露点。

没有空缺值

4.2.1数据集的合成

我将原数据集划分为两个数据集：

1. 除预测结果外的特征值(data.csv)
2. 电能使用数值（target.csv）

之后使用load\_csv方法导入上述csv文件

def load\_csv(filename):

dataset = list()

with open(filename, 'r') as file:

csv\_reader = reader(file)

for row in csv\_reader:

if not row:

continue

dataset.append(row)

return dataset

4.2.2数据集标准化

时间特征的处理，共使用了4种方法：

1. 将每天的144个特征转化为数值格式1-144：

for i in range(len(X)):

chuli = X[i][0].split(':')

X[i][0] = int(chuli[0]) \* 6 + (int(chuli[1]) / 10) + 1

1. 将每天的每个特征转换为秒数：

for i in range(len(train\_data)):

# print(X[i][0])

chuli = train\_data[i][0].split(':')

train\_data[i][0] = int(chuli[0]) \* 3600 + (int(chuli[1])\*60)

1. 将每天的每个特征转换为分钟数：

for i in range(len(X)):

chuli = X[i][0].split(':')

X[i][0] = int(chuli[0]) \* 60 + (int(chuli[1]))

1. 将每天的每个特征归一化（0-1）：

for i in range(len(X)):

chuli = X[i][0].split(':')

X[i][0] = (int(chuli[0]) \* 6 + (int(chuli[1]) / 10) + 1)/len(X)

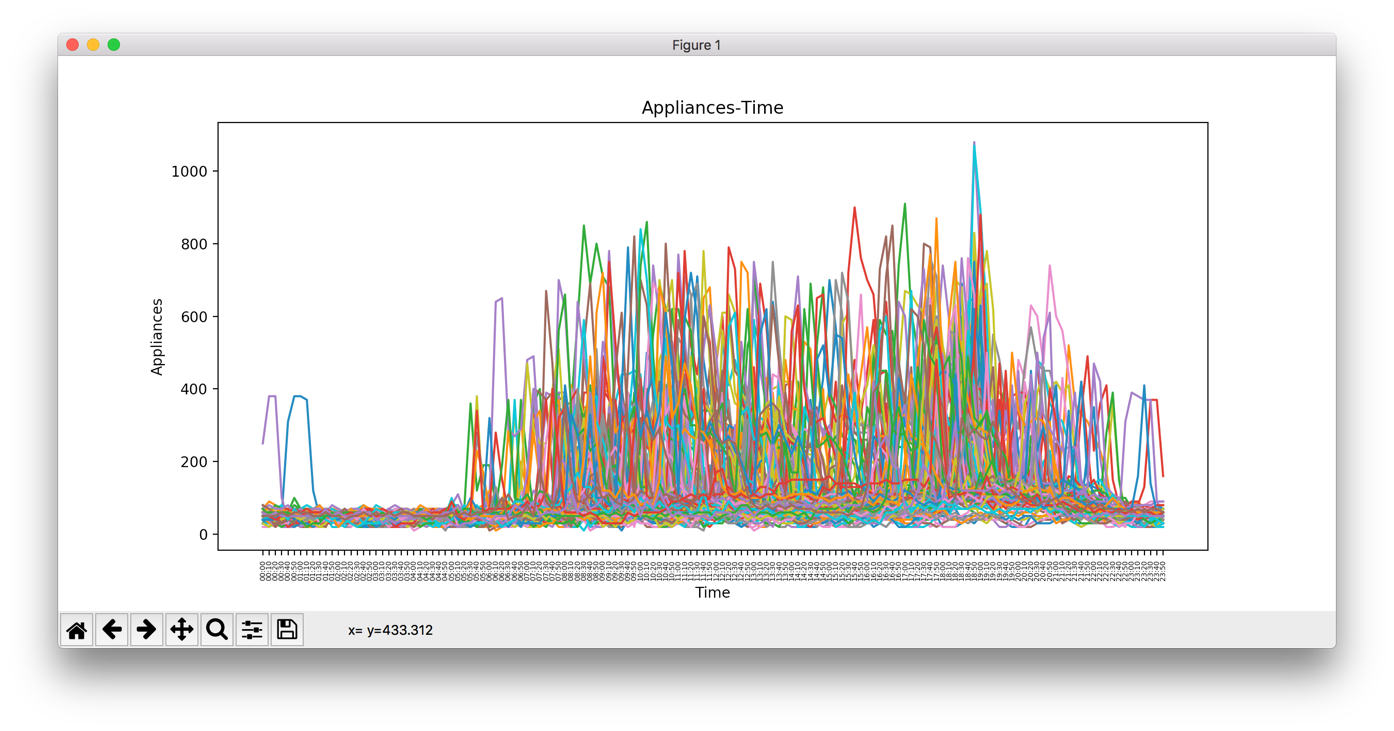


图4.1 时间-能耗折线图

4.2.3数据集切分

使用2/3数据集用来训练，1/3数据集用来测试

训练数据-特征：

X = train\_data[:int(len(train\_data)\*2/3)]

训练数据-结果：

a = []

for i in train\_target:

a.append(float(i[0]))

train\_target = a

y = train\_target[:int(len(train\_target)\*2/3)]

4.3随机森林模型

4.3.1 应用RandomForestRegressor建立模型

特征重要性

选择训练模型

clf = RandomForestRegressor(n\_estimators=100,oob\_score = 'true')

拟合模型方法

def fit(self, X, y, sample\_weight=None):

X = check\_array(X, accept\_sparse="csc", dtype=DTYPE)

y = check\_array(y, accept\_sparse='csc', ensure\_2d=False, dtype=None)

if sample\_weight is not None:

sample\_weight = check\_array(sample\_weight, ensure\_2d=False)

if issparse(X):

X.sort\_indices()

n\_samples, self.n\_features\_ = X.shape

y = np.atleast\_1d(y)

if y.ndim == 2 and y.shape[1] == 1:

warn("A column-vector y was passed when a 1d array was"

" expected. Please change the shape of y to "

"(n\_samples,), for example using ravel().",

DataConversionWarning, stacklevel=2)

if y.ndim == 1:

y = np.reshape(y, (-1, 1))

self.n\_outputs\_ = y.shape[1]

y, expanded\_class\_weight = self.\_validate\_y\_class\_weight(y)

if getattr(y, "dtype", None) != DOUBLE or not y.flags.contiguous:

y = np.ascontiguousarray(y, dtype=DOUBLE)

if expanded\_class\_weight is not None:

if sample\_weight is not None:

sample\_weight = sample\_weight \* expanded\_class\_weight

else:

sample\_weight = expanded\_class\_weight

self.\_validate\_estimator()

if not self.bootstrap and self.oob\_score:

raise ValueError("Out of bag estimation only available"

" if bootstrap=True")

random\_state = check\_random\_state(self.random\_state)

if not self.warm\_start or not hasattr(self, "estimators\_"):

self.estimators\_ = []

n\_more\_estimators = self.n\_estimators - len(self.estimators\_)

if n\_more\_estimators < 0:

raise ValueError('n\_estimators=%d must be larger or equal to '

'len(estimators\_)=%d when warm\_start==True'

% (self.n\_estimators, len(self.estimators\_)))

elif n\_more\_estimators == 0:

warn("Warm-start fitting without increasing n\_estimators does not "

"fit new trees.")

else:

if self.warm\_start and len(self.estimators\_) > 0:

random\_state.randint(MAX\_INT, size=len(self.estimators\_))

trees = []

for i in range(n\_more\_estimators):

tree = self.\_make\_estimator

(append=False, random\_state=random\_state)

trees.append(tree)

trees = Parallel(n\_jobs=self.n\_jobs, verbose=self.verbose,

backend="threading")(

delayed(\_parallel\_build\_trees)(

t, self, X, y, sample\_weight, i, len(trees),

verbose=self.verbose, class\_weight=self.class\_weight)

for i, t in enumerate(trees))

self.estimators\_.extend(trees)

if self.oob\_score:

self.\_set\_oob\_score(X, y)

if hasattr(self, "classes\_") and self.n\_outputs\_ == 1:

self.n\_classes\_ = self.n\_classes\_[0]

self.classes\_ = self.classes\_[0]

return self

预测的代码！！！ 只加核心代码！！！

4.3.2参数调优

一、RandomForestRegressor()常用内置参数

1、max\_features：随机森林允许单个决策树使用特征的最大数量，增加max\_features一般能提高模型的性能，因为在每个节点上，我们有更多的选择可以考虑。然而，这未必完全是对的，因为它降低了单个树的多样性，而这正是随机森林独特的优点。 但是，可以肯定，你通过增加max\_features会降低算法的速度。因此，你需要适当的平衡和选择最佳max\_features。

2、n\_estimators：建立子树的数量，较多的子树可以让模型有更好的性能，但同时让你的代码变慢。你应该选择尽可能高的值，只要你的处理器能够承受的住，因为这使你的预测更好更稳定。

3、min\_sample\_leaf：最小样本叶片大小。

n\_jobs这个参数告诉引擎有多少处理器是它可以使用。“-1”意味着没有限制，而“1”值意味着它只能使用一个处理器。

4、random\_state此参数让结果容易复现。一个确定的随机值将会产生相同的结果，在参数和训练数据不变的情况下。我曾亲自尝试过将不同的随机状态的最优参数模型集成，有时候这种方法比单独的随机状态更好。

5、random\_state是随机数生成器使用的种子;如果是RandomState实例，random\_state就是随机数生成器;如果为None，则随机数生成器是np.random使用的RandomState实例。

6、oob\_score这是一个随机森林交叉验证方法。它和留一验证方法非常相似，但这快很多。这种方法只是简单的标记在每颗子树中用的观察数据。然后对每一个观察样本找出一个最大投票得分，是由那些没有使用该观察样本进行训练的子树投票得到。

7、criterion string, optional (default=”gini”) 字符串，可选择(默认值为“gini”)。 衡量分裂质量的性能（函数）。

8、max\_depth（决策）树的最大深度。

9、min\_samples\_split分割内部节点所需要的最小样本数量。

10、min\_samples\_leaf需要在叶子结点上的最小样本数量。

11、max\_leaf\_nodes以最优的方法使用max\_leaf\_nodes来生长树。

12、bootstrap建立决策树时，是否使用有放回抽样。

13、estimators\_决策树分类器的序列。

14、feature\_importances\_特征的重要性（值越高，特征越重要）。

15、oob\_score\_使用袋外估计获得的训练数据集的得分。

二、调优过程

1、n\_estimators

n\_estimators = 10时，时间：14秒，准确率：75.5%。

n\_estimators = 100时，时间：43秒，准确率：76.3%。

n\_estimators = 1000时，时间：598秒，准确率：76.8%。

2、oob\_score

oob\_score = 'true',正确率:75.5%

oob\_score = 'false',正确率:75.5%

3、max\_depth

max\_depth=5，正确率：30%

max\_depth=10，正确率：38%

max\_depth=100，正确率：75%

max\_depth=1000，正确率：76%

4.3.3计算正确率

计算误差在N Wh内正确率：

a = 0

wucha = N

for i in range(len(predict\_data)):

if abs((predict\_target - correct\_target)[i])<wucha:

a += 1

acc = a/len(predict\_data)

4.3.4评价函数

1、RMSE

rmse = np.sqrt(((predict\_target - correct\_target) \*\* 2).mean())

2、R方

average = np.sum(correct\_target)/len(correct\_target) # 平均值

a = []

for i in range(len(correct\_target)):

a.append(average)

r = 1 - (((predict\_target - correct\_target)\*\*2).sum()/(((predict\_target - a)\*\*2).sum()))

1. MAE

mae = (abs((predict\_target - correct\_target)).sum())/len(correct\_target)

1. MAPE

mape = (abs((predict\_target - correct\_target))/predict\_target).sum()/len(correct\_target)

4.3.5预测结果的比较与分析

预测结果，

4.4 本章小结

本章主要

# 5结 论

结论。。。

# 谢 辞

衷心感谢我的指导老师。她严肃的科学态度，严谨的治学精神，精益求精的工作作风，深深地感染和激励着我。老师学识渊博，品德高尚，平易近人，在我学习期间不仅仅传授了做学问的秘诀，还传授了做人的准则，这些都将使我终生受益。在我毕业论文的写作过程中，老师始终给予我精心的指导和不懈的支持。她循循善诱的教导和不拘一格的思路给予我无尽的启迪。在此谨向老师致以诚挚的谢意和崇高的敬意。

　　同时，我也要向身边的同学表示感谢，在整个毕业设计的过程中他们帮我解决了许多棘手的问题，并且论文中某些观点提出和修正与他们的讨论分不开。

　　路漫漫其修远兮，吾将上下而求索。我愿在未来的学习和研究过程中，以更加丰厚的成果来答谢以前关心、帮忙和支持过我的老师和同学。

# 参考文献