**

**HUNAN UNIVERSITY**

《DM&BID》

Report 9

|  |  |
| --- | --- |
| **报告名称：** | 随机森林模型 |
| **学生姓名：** | 杨超然 |
| **学生学号：** | 202106060220 |
| **专业班级：** | 电商2102班 |
| **学 院：** | 工商管理学院 |
| **指导老师：** | 江资斌 |
| **日 期：** | 2023.3.27 |

目录

[一、 随机森林模型的代码实现 3](#_Toc2122494784)

[（一）随机森林分类模型 3](#_Toc918974371)

[（二）随机森林回归模型 3](#_Toc499864173)

[二、 量化金融-股票数据获取 3](#_Toc261128547)

[（一）股票数据获取 4](#_Toc1478398608)

[1.1股票基本数据获取 4](#_Toc1059608866)

[1.2获得日线行情数据 5](#_Toc1911809938)

[1.3获得分钟级别的数据 5](#_Toc1139301552)

[1.4获得实时行情数据 6](#_Toc1276987812)

[1.5获得分笔数据 6](#_Toc382588166)

[1.6获得指数信息 6](#_Toc593266844)

[（二）股票衍生变量生成 7](#_Toc269274087)

[2.1生成股票基本数据 7](#_Toc941535980)

[2.2衍生变量计算 7](#_Toc1735704764)

[2.3获得移动平均线指标MA值 8](#_Toc572107700)

[三、 量化金融-股票涨跌预测模型搭建 9](#_Toc1129826281)

[（一） 多因子模型搭建 9](#_Toc939897993)

[1.1引入库 9](#_Toc2123344666)

[1.2股票数据处理与衍生变量生成 9](#_Toc170555616)

[1.3提取特征变量和目标变量 10](#_Toc1785053014)

[1.4划分测试集与训练集 10](#_Toc1039457708)

[1.5模型搭建 11](#_Toc386230011)

[（二） 模型使用与评估 11](#_Toc1672213643)

[2.1预测下一天涨跌情况 11](#_Toc776209612)

[2.2模型准确度评估 11](#_Toc1939277006)

[2.3分析数据特征重要性 12](#_Toc1069329323)

[（三） 参数调优 12](#_Toc2074773565)

[（四） 绘制收益回测曲线 12](#_Toc2027330616)

[四、学习小记 13](#_Toc1370119810)

[（一）集成模型 13](#_Toc136499889)

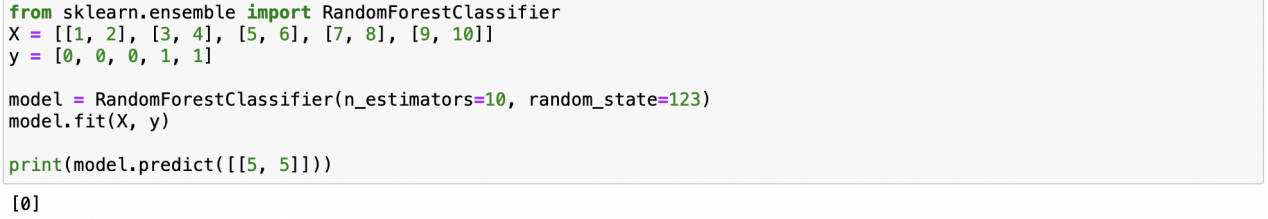
[（二）Bagging算法与Boosting算法的解释 13](#_Toc641099427)

[（三）随机森林模型原理 14](#_Toc1032612590)

[（四） get\_k\_data()函数 15](#_Toc1304448723)

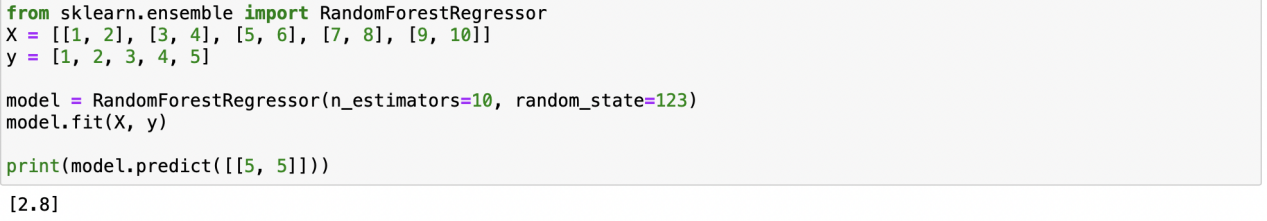
# 随机森林模型的代码实现

## （一）随机森林分类模型



模型预测（5，5）将为0

## （二）随机森林回归模型



模型预测（5，5）将为2.8

# 量化金融-股票数据获取

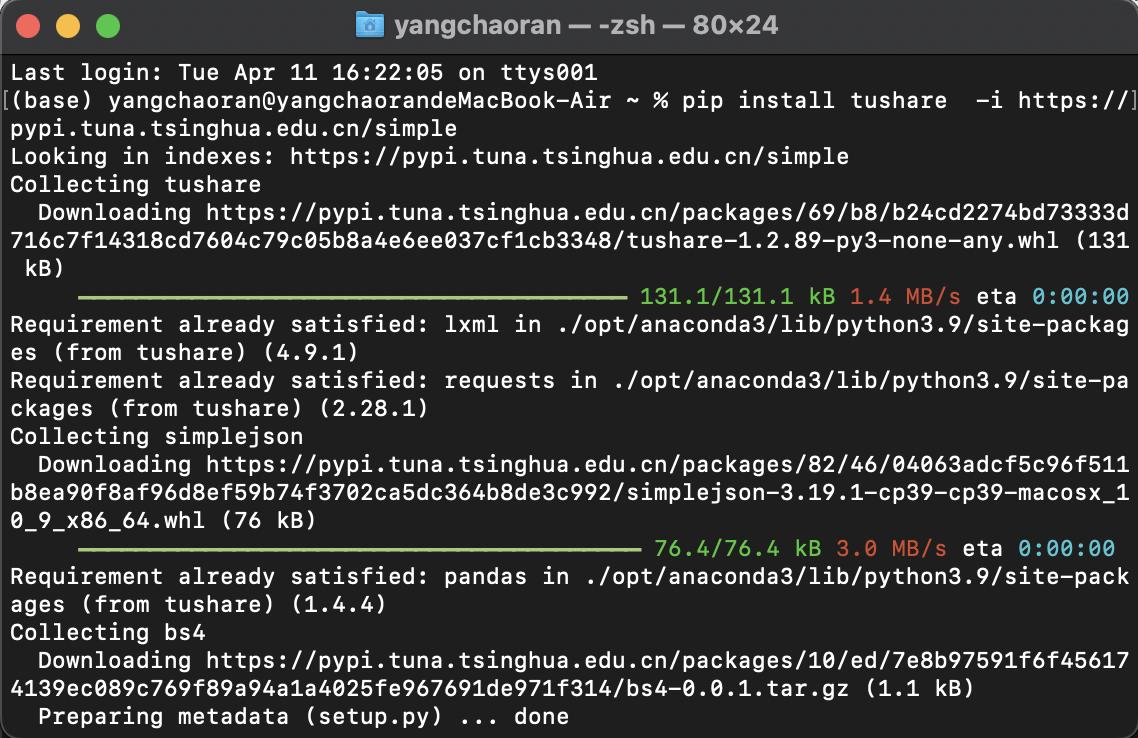
量化金融有两个主要方向：通过大数据分析进行智能择时（选择合适的交易时机）与智能择股（选择合适的交易股票），这两种方向的本质都是预测股票涨跌的情况。在实战应用中，股票涨跌预测非常难，因为影响股票涨跌的因素不仅有众多技术指标（股价以及股价的衍生数据），还有公司的基本面指标（例如公司的资产，收入等信息），以及宏观经济数据等各种各样的数据，因此在这也只是简单演示一下量化金融模型的基本思路，实际应用的股票涨跌预测模型则复杂的多。

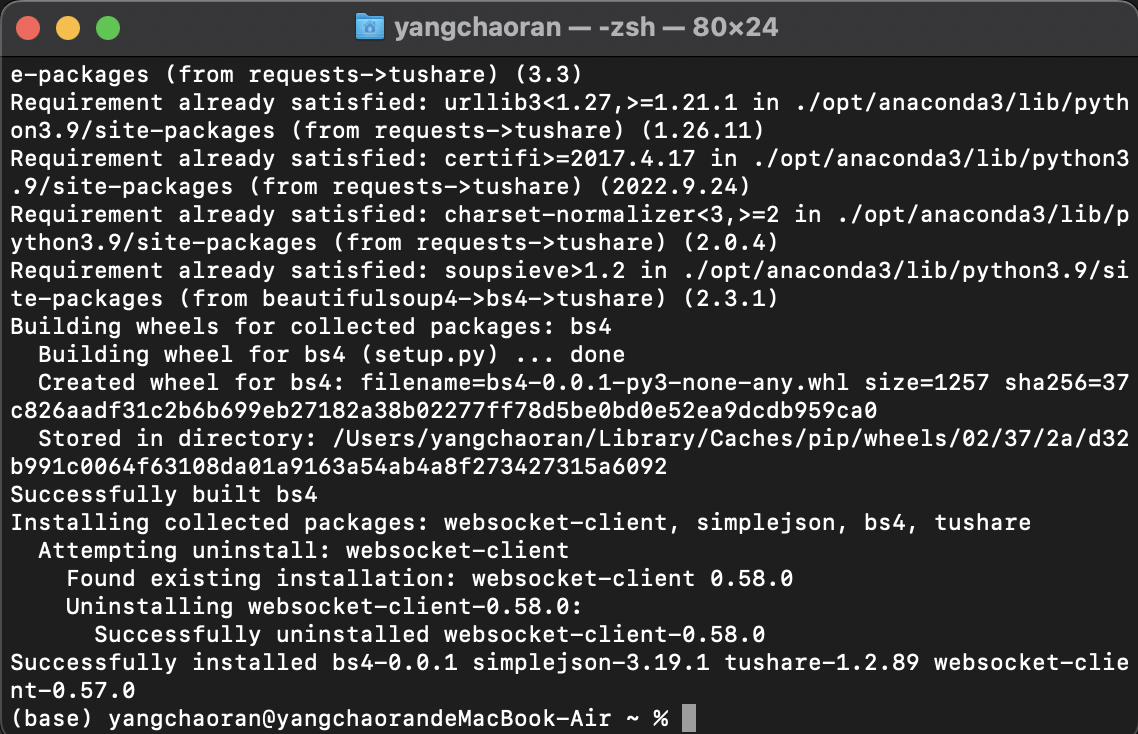
## （一）股票数据获取

1.1股票基本数据获取

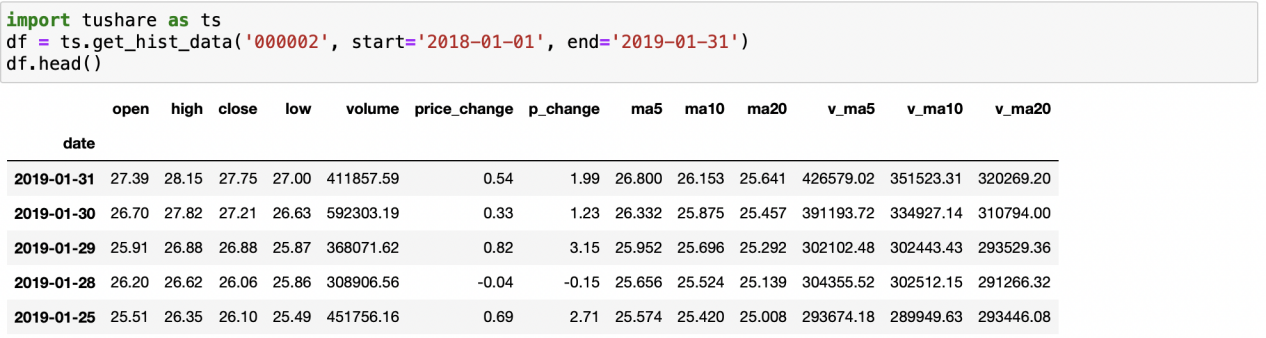
想要搭建机器学习模型，首先得有合适的数据，我们将使用免费的财经数据Python接口包：Tushare库，通过它我们能够免费地调用历史行情数据来进行分析。其官方地址为：<http://tushare.org/>，如果是想查看股价行情数据，可以访问相应网址：<http://tushare.org/trading.html>。

推荐通过PIP安装法来安装Tushare库，以MAC系统为例：terminal中输入pip install tushare  -i https://pypi.tuna.tsinghua.edu.cn/simple安装tushare清华镜像，安装成功后显示如下：





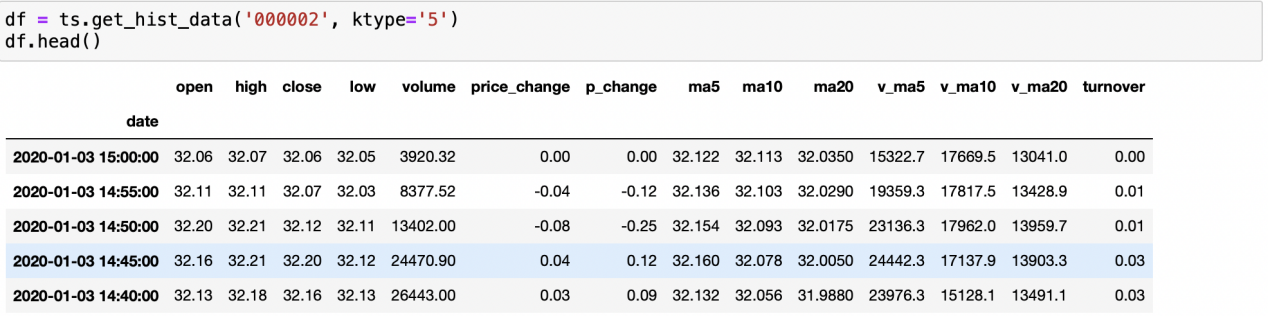
1.2获得日线行情数据



若不写开始与结束日期，直接写ts.get\_hist\_data(‘00002’)会直接获取当天往前三年的数据。

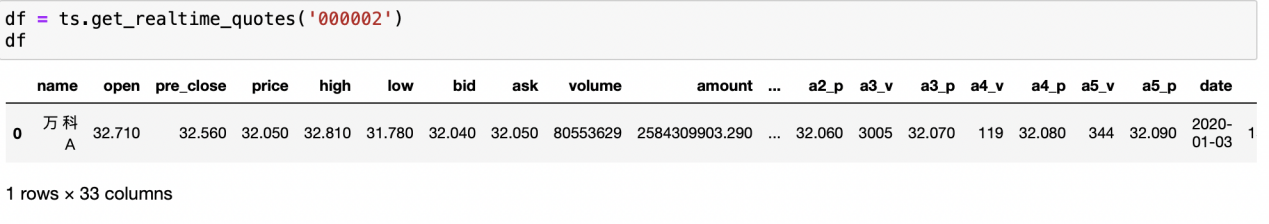
1.3获得分钟级别的数据

通过设置ktype参数可以获得分钟级别的数据，代码如下：



1.4获得实时行情数据

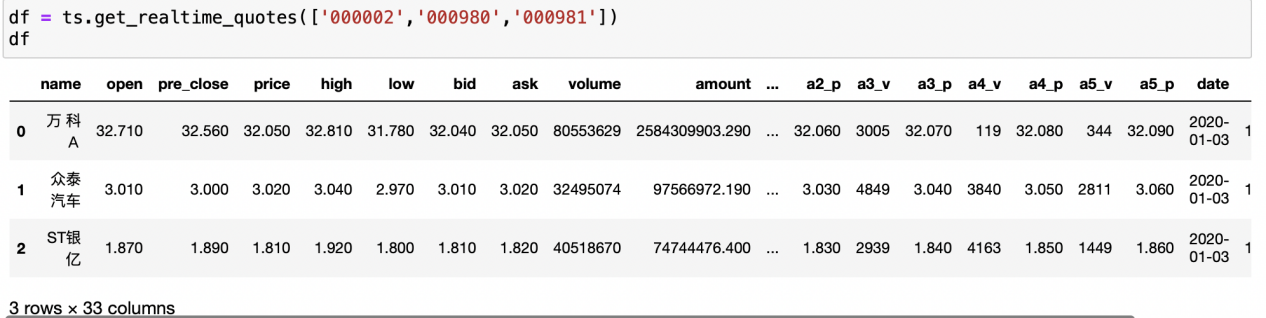
获取股票当前报价与成交信息，代码如下：



其运行结果就是当时的股价信息，如果收盘后运行的话获得的就是当日收盘价相关信息。如果觉得列数过多，可以通过DataFrame选取列的方法选取相应的列，代码如下：



获取多个股票实时数据，可用如下代码：



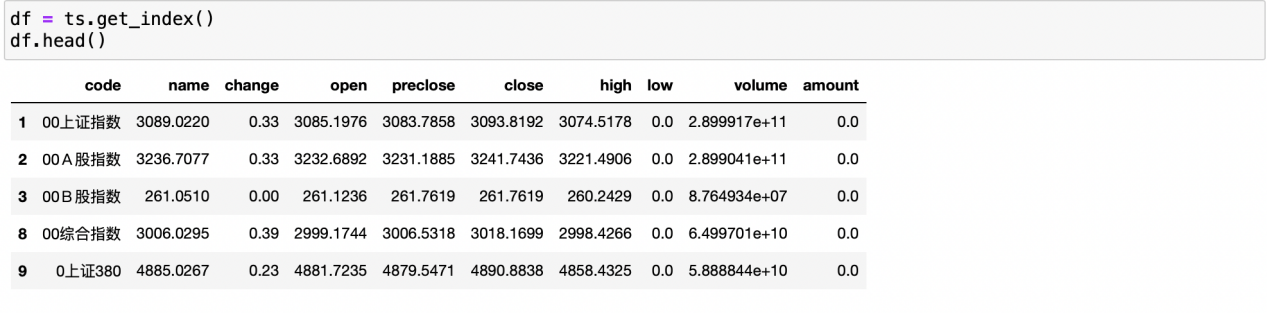
1.5获得分笔数据

获得历史分笔数据（每笔成交信息）的代码如下：



1.6获得指数信息

通过如下代码可获得上证指数等指数信息：



## （二）股票衍生变量生成

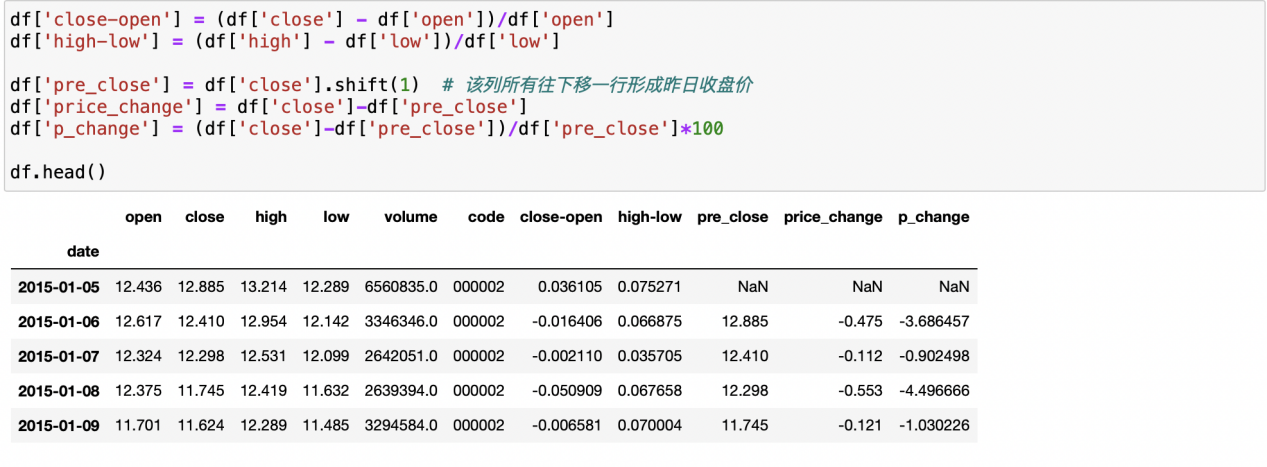
2.1生成股票基本数据

首先通过get\_k\_data()函数获取从2015-01-01到2019-12-31的股票基本数据：



2.2衍生变量计算

通过如下变量构造一些衍生变量：



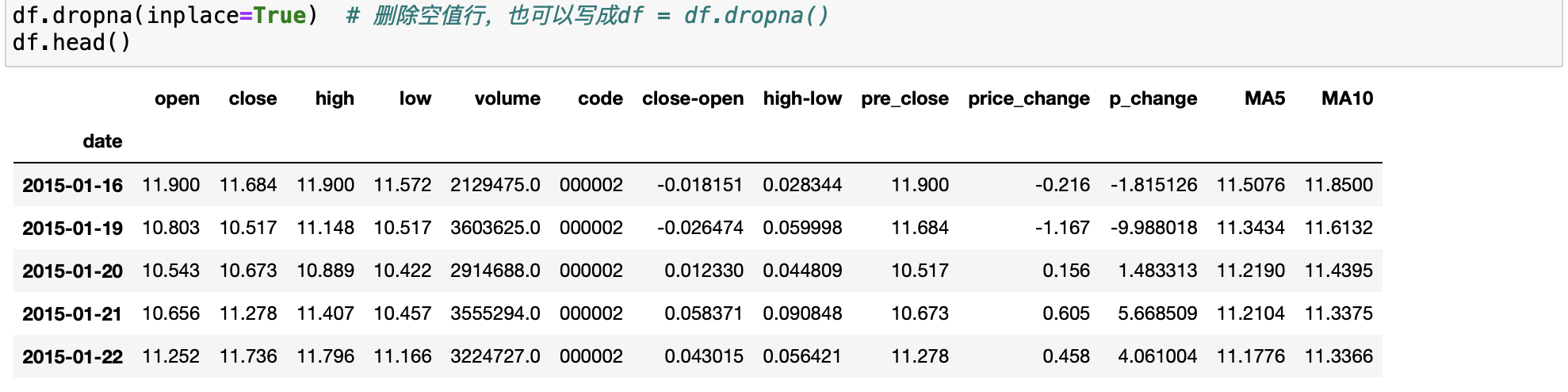
2.3获得移动平均线指标MA值

通过如下代码可以获得股价的5日移动平均值和10日移动平均值：

需要注意的是：当我们在计算像MA5这样的数据时，数据前四天对应的平均值是无法计算出来的（因为最开始的4天数据量不够去计算5日均值），所以会产生空值NaN，通常会通过dropna()函数删除空值，以免在后续计算中出现空值造成的问题。



再删除表中的空值：



# 量化金融-股票涨跌预测模型搭建

## 多因子模型搭建

模型中相关参数的意义如下：

max\_depth：每个决策树的最大深度，这里设置树的最大深度为3，也即共3层；

n\_estimators：设置基模型：决策树模型的个数，这里设置为数字10，也就是

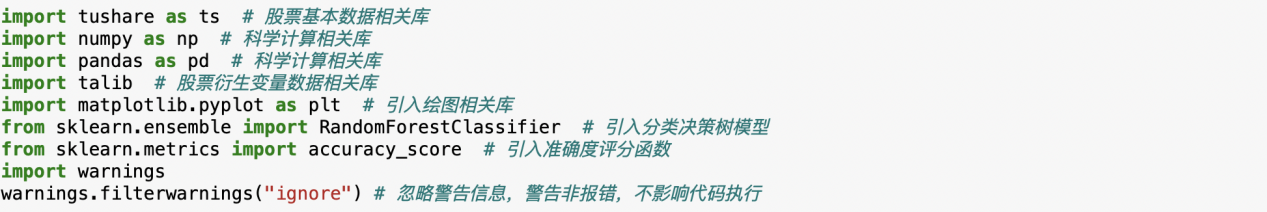
说该随机森林中共有10个决策树；

min\_samples\_leaf：在叶节点处所需的最小样本数，这里设置为数字10，也

就是如果叶子节点的样本数小于10则停止分裂；

random\_state：随机状态参数，使得每次运行的结果都是一样的，数字1没有特殊含义，可以换成其他数字。

1.1引入库



1.2股票数据处理与衍生变量生成

将之前的代码进行汇总，方便之后预测模型的搭建





1.3提取特征变量和目标变量

原理是：以今天的股价信息预测下一天的股价涨跌情况，因此变量y应该是下一天的股票涨跌情况。

截屏2023-04-11 17.13.08

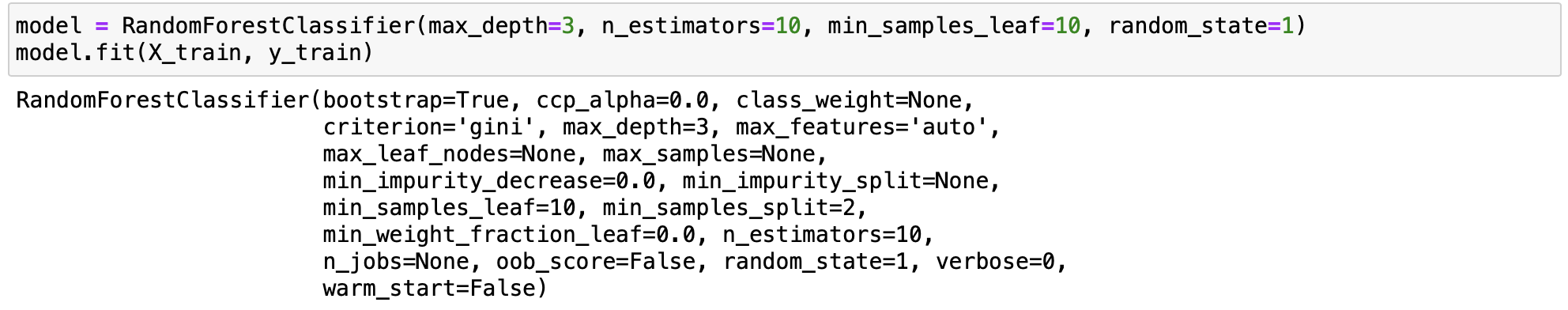
1.4划分测试集与训练集

接下来，我们要将原始数据集进行分割，我们要注意到一点，训练集与测试集的划分要按照时间序列划分，而不是像之前利用train\_test\_split()函数进行划分。原因在于股票价格的变化趋势具有时间性，如果我们随机划分，则会破坏时间性特征，因为我们是根据当天数据来预测下一天的股价涨跌情况，而不是任意一天的股票数据来预测下一天的股价涨跌情况。

因此，我们将前90%的数据作为训练集，后10%的数据作为测试集，代码如下：

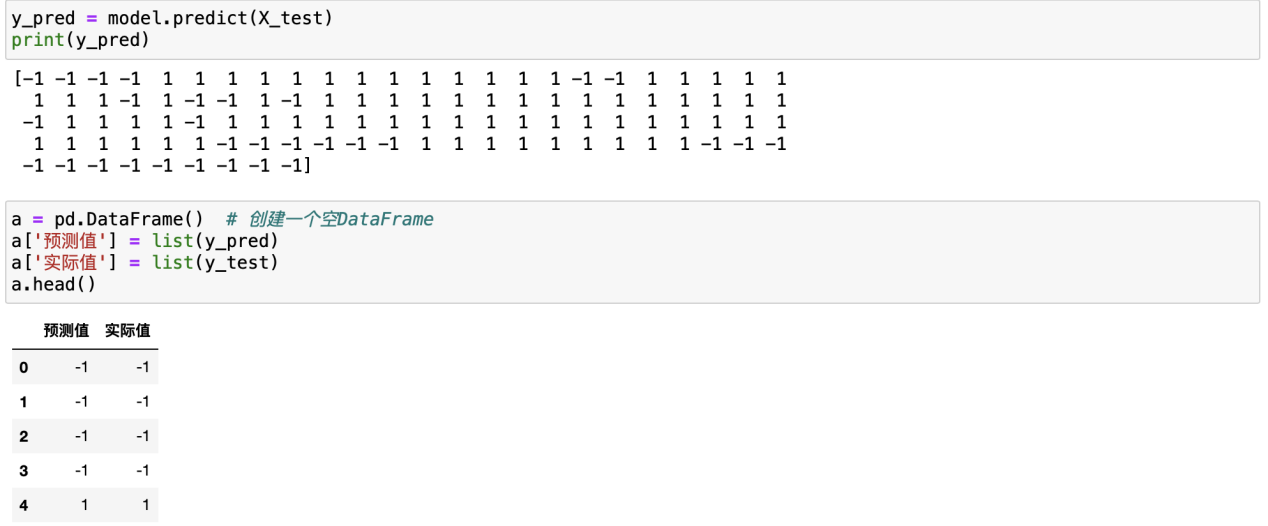


1.5模型搭建

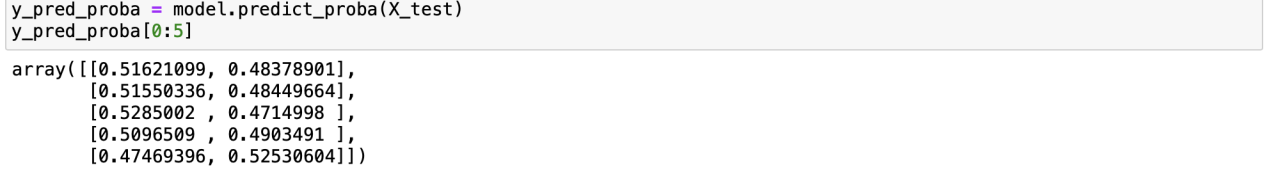


## 模型使用与评估

2.1预测下一天涨跌情况



用如下代码查看预测概率：



2.2模型准确度评估



2.3分析数据特征重要性



## 参数调优

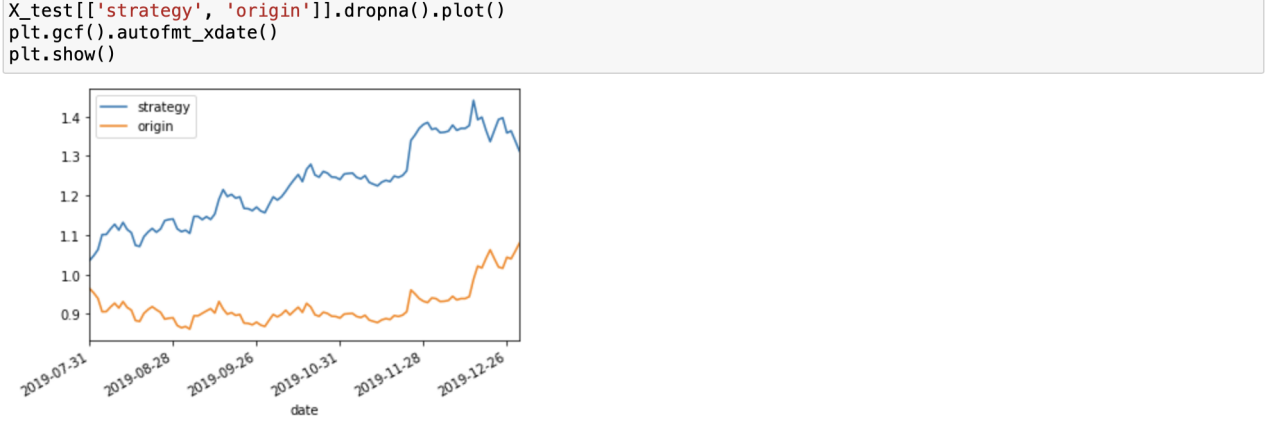


## 绘制收益回测曲线

首先通过如下代码构建相关参数：



再将数据中的空值删除后，进行可视化，代码如下（X轴刻度已设置为自动倾斜）：



# 四、学习小记

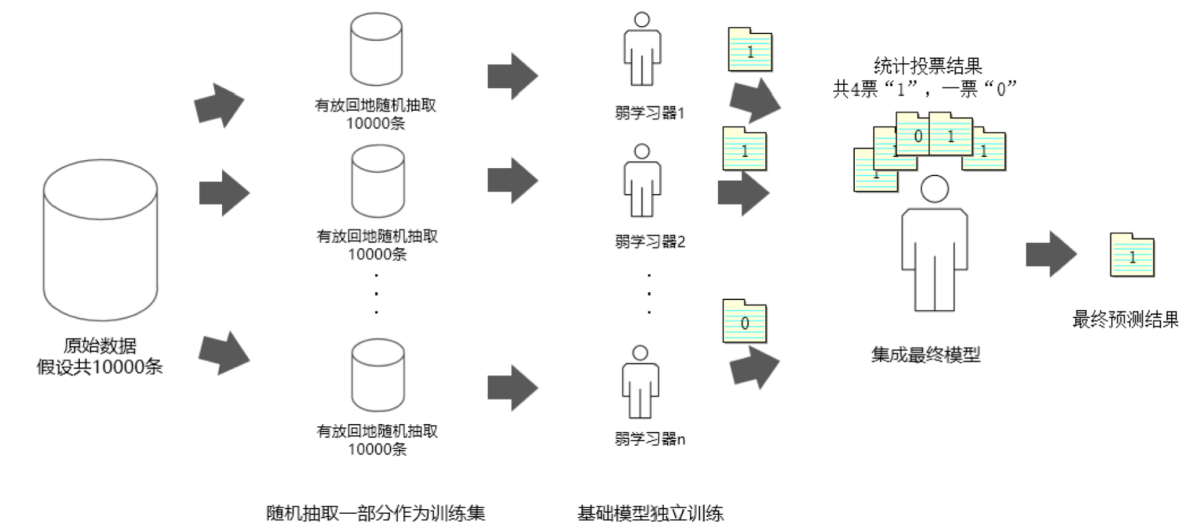
## （一）集成模型

集成学习：使用一系列的**弱学习器**（也称**基础模型**）进行学习，并将各弱学习器的结果进行整合从而获得比单个学习器更好的学习效果

集成学习的两种常见算法：Bagging算法（随机森林模型）与Boosting算法（AdaBoost、GBDT、XGBoost和LightGBM模型）

## （二）Bagging算法与Boosting算法的解释

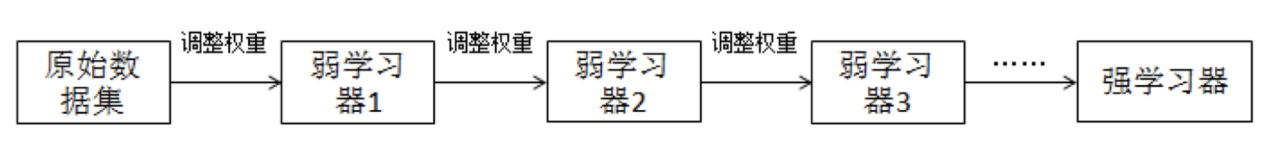
Bagging算法：采用类似于**“民主投票”**的方式，即每一个基础模型都有一票，最终结果通过所有基础模型投票，少数服从多数的原则产生预测结果。



Boosting算法：注重“培养精英”和“重视错误”。

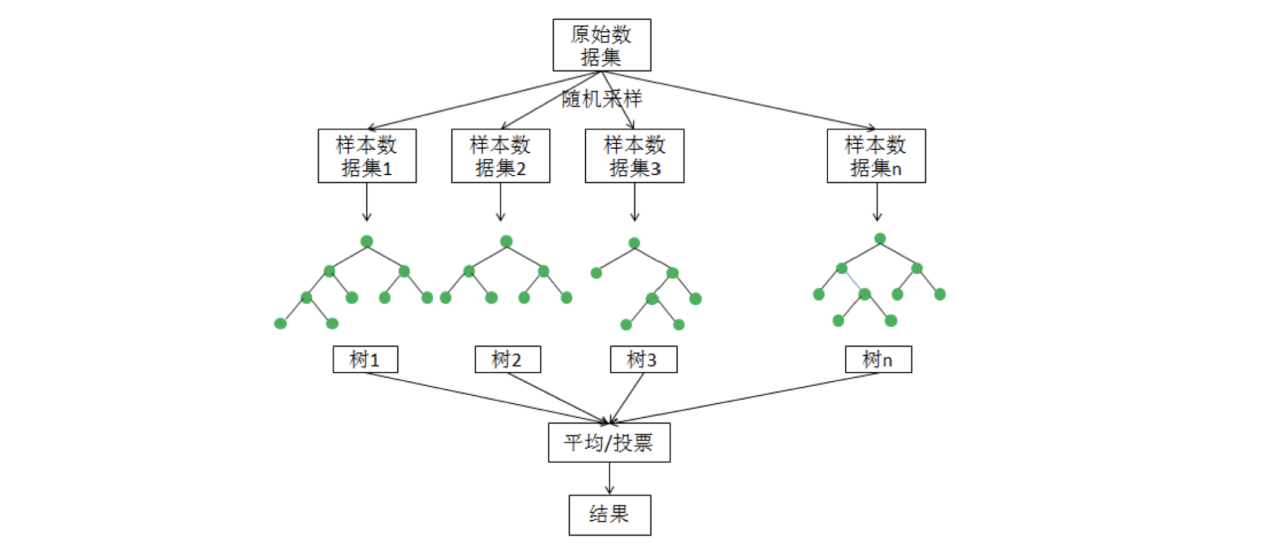
**“培养精英”**，即每一轮对于预测结果较为准确的基础模型，会给予它一个较大的权重，表现不好的基础模型则会降低它的权重。这样在最终预测时，“优秀模型”的权重是大的，相当于它可以投出多票，而“一般模型”只能在投票时投出一票或不能投票。

**“重视错误”**，即在每一轮训练后改变训练数据的权值或概率分布，通过提高那些在前一轮被基础模型预测错误样例的权值，减小前一轮预测正确样例的权值，来使得分类器对误分的数据有较高的重视程度，从而提升模型的整体效果。



## （三）随机森林模型原理

随机森林（Random Forest）是一种**经典的Bagging模型**，其弱学习器为决策树模型。如下图所示，随机森林模型会在原始数据集中**随机抽样**，构成n个不同的样本数据集，然后根据这些数据集**搭建n个不同的决策树模型**，最后根据这些**决策树模型的平均值**（针对回归模型）或者**投票**（针对分类模型）情况来获取最终结果。



为保证模型泛化能力（通用能力），随机森林在建立每棵树时会遵循以下两个原则：

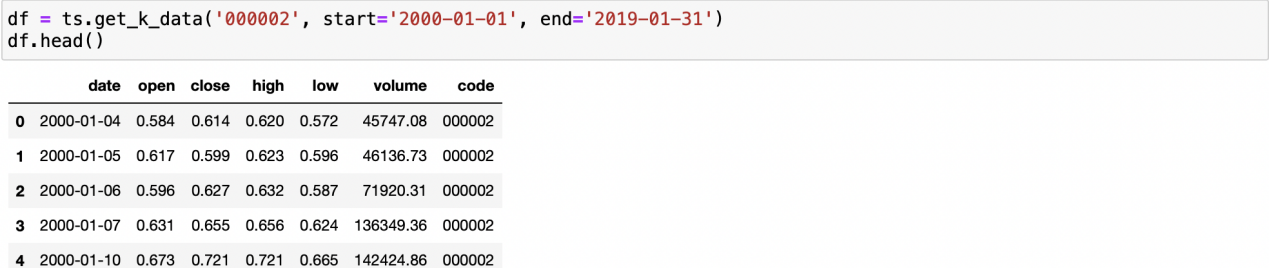
**数据随机**：随机地从所有数据当中有放回地抽取数据作为其中一棵决策树 的数据进行训练。举例来说，有1000个原始数据，有放回的抽取1000次，构成一组新的数据（因为是有放回抽取，有些数据可能被选中多次，有些数据可能不被选上），作为某一个决策树的数据来进行模型的训练。

**特征随机**：如果每个样本的特征为M，则选定k<M(一般选择k为M的平方根)，即在M个特征中随机抽取k个特征。

## get\_k\_data()函数

在获取股票数据的过程中，我们采用了get\_hist\_data()函数，该函数不仅获得了 股票的基本价格信息，还获取了价格变化、均线价格等衍生变量，所以它最多也只能获得当天往前三年的数据。如果想获取超过三年的日线级数据，需使用

ts.get\_k\_data()函数,它只获取股价的基本数据，代码如下：



通过get\_k\_data()函数获取的数据没有像get\_hist\_data()函数那样将日期默认设为行索引，这里的日期还是作为一个普通的列（date列），如果想把这里的date列转为行索引，可以使用设置索引的set\_index()函数，代码如下：

