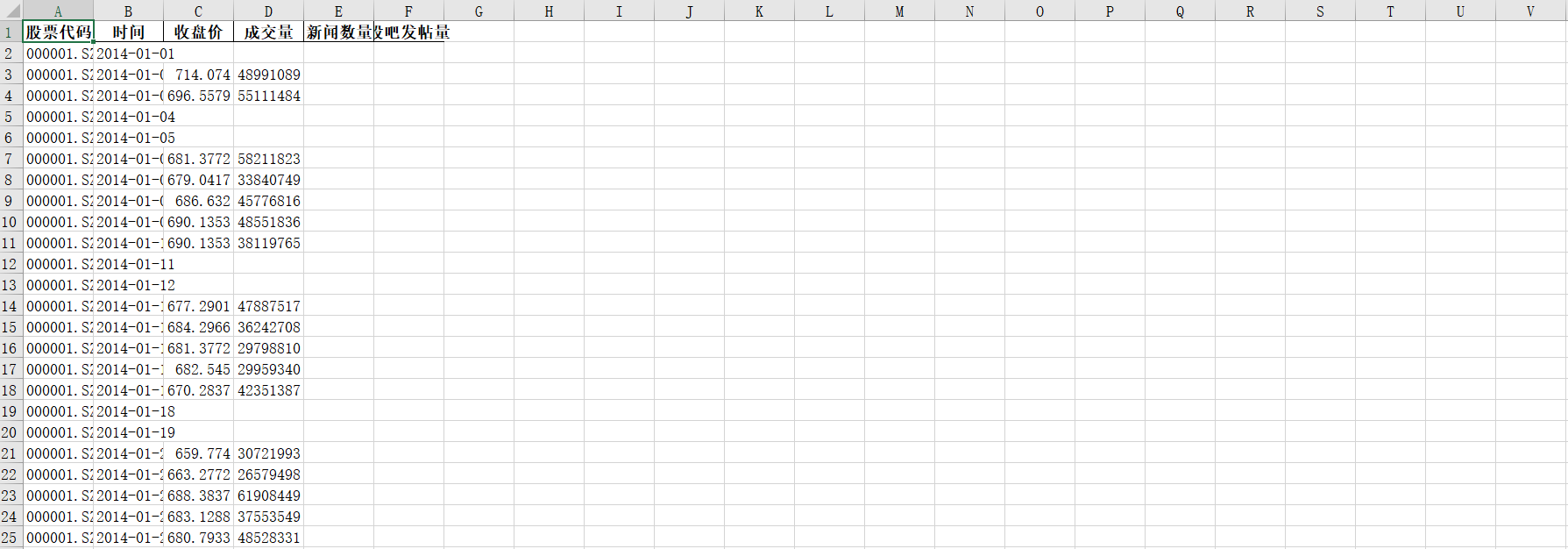
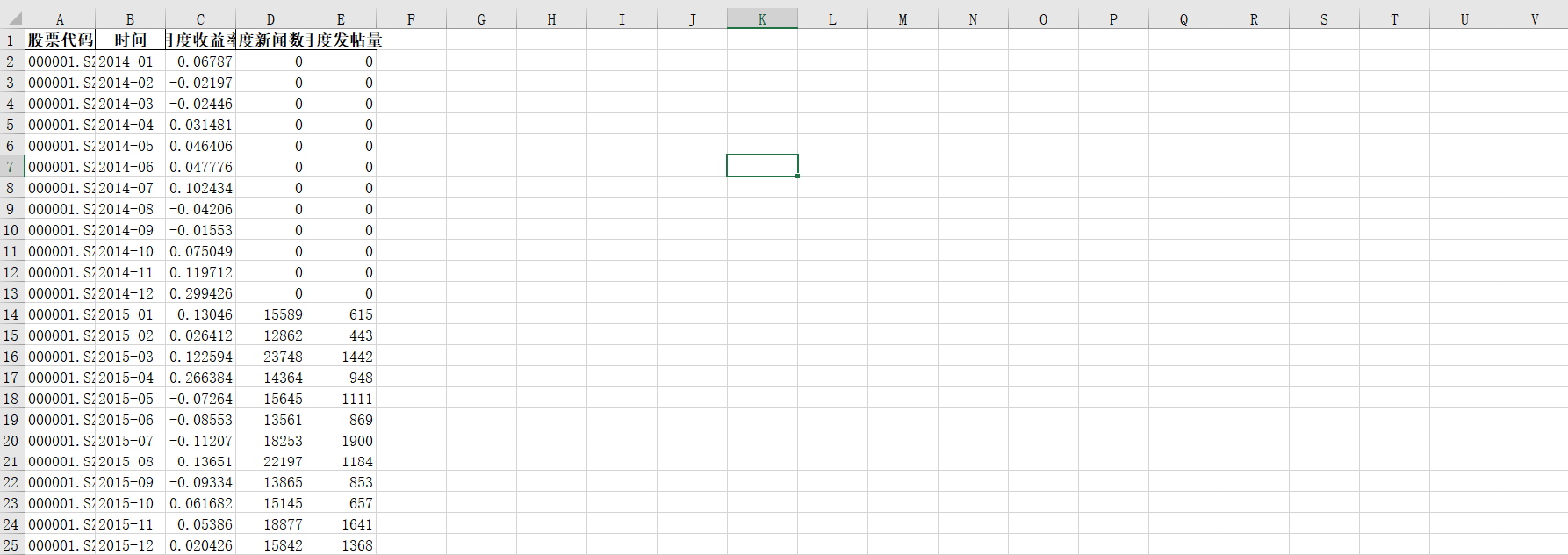
1. 数据整合结果：

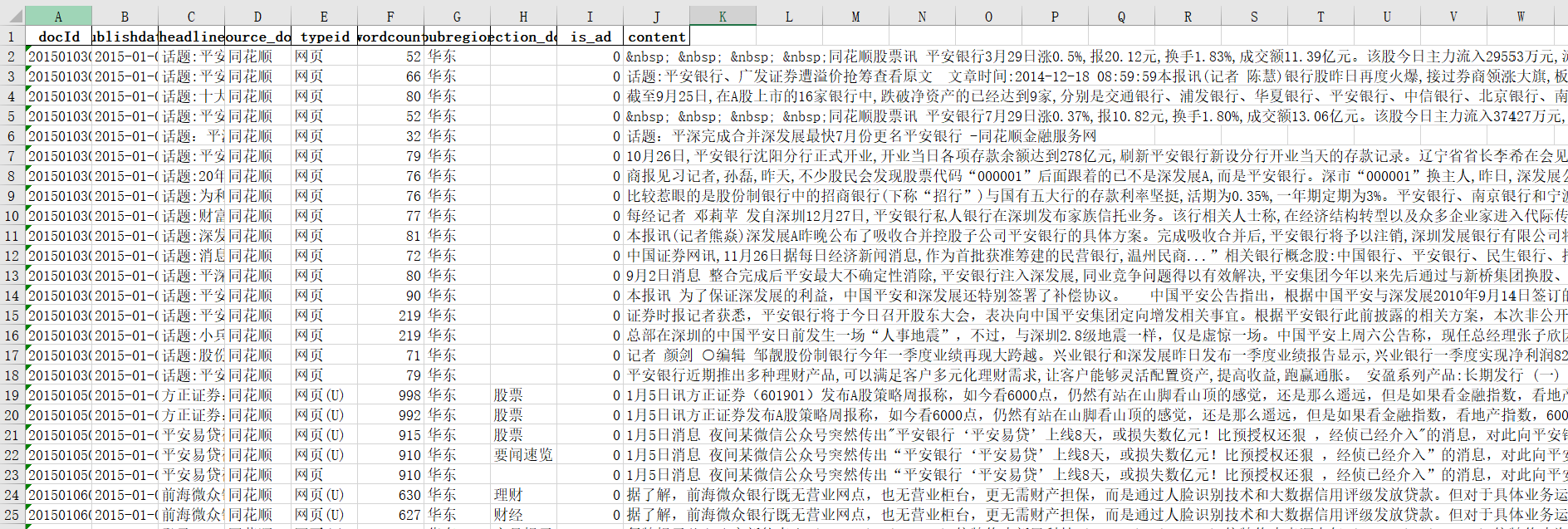
（1）第一道题答案



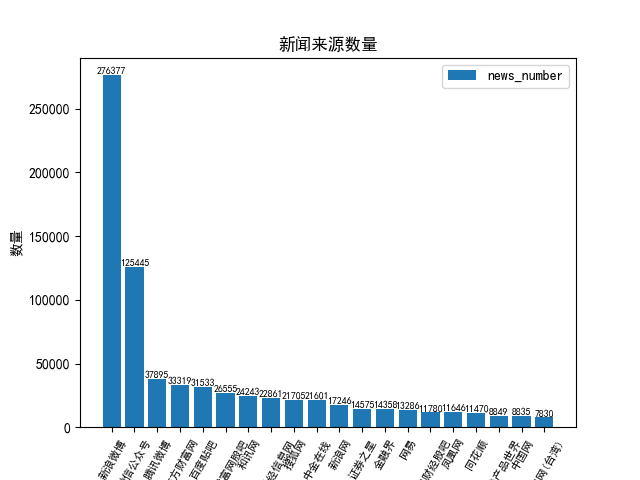
（2）第五题答案



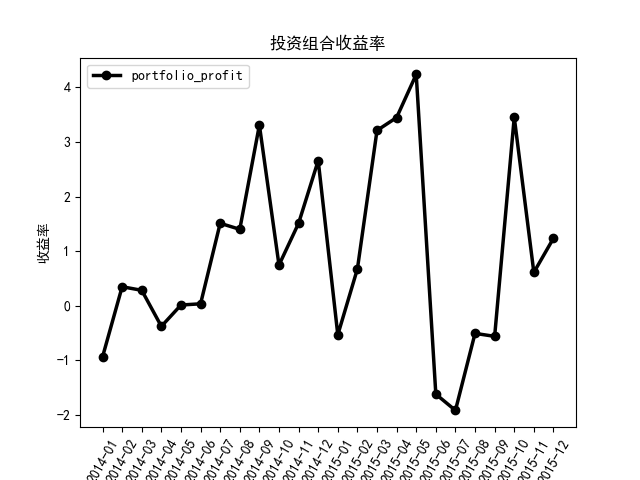
（3）第八题答案



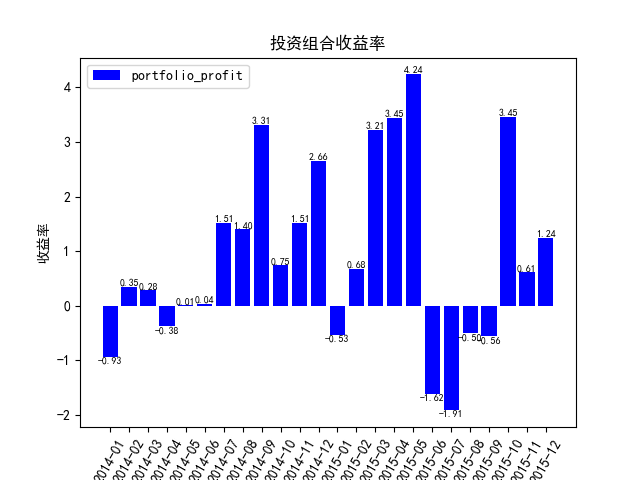
1. 可视化结果：
2. 新闻来源数量



1. 月度收益率折线图



1. 月度收益率条形图



1. 教学建议

（1），改变教学方式：

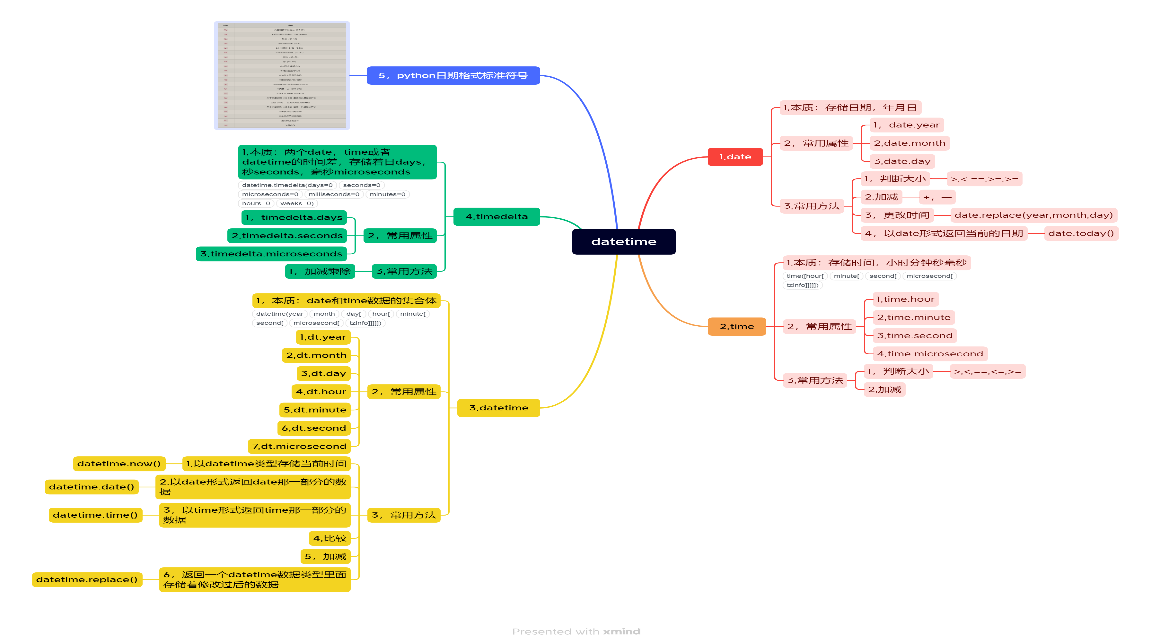
目前主流的教学方法，似乎都是以函数为主，但是却忽略了类的本质，但个人认为类才应该是教学的重点。

以类和目的为导向的学习方法是在学习过程中总结出来的，由于尚且处在实践中，因此想跟老师交流分享一下。

类实质上就是数据的集合，用于存储数据的，如list就是一维的有顺序的数据集合体，dict就是键和值的数据集合体，还有Data Frame类就是一组二维数据以表格形式存储的信息集合体。那么类中的方法和属性又居于什么地位呢？属性就是用来描述存储在这个类中数据的概况或者说信息，如数组array.size描述的就是该数组中存储的数据个数，方法（函数）是用来处理存储的数据的一种方法，如，各种函数是为了满足开发主体的需求从而创建的处理信息的一种方式，正是因为函数只是数据处理的一种方式，如果学习python的人并不想要掌握处理该信息的方式，那么完全没有必要学，如一个文学类的学生学习，并不需要求方差和或者标准差，那么就可以不用掌握np.std()这个方法。

知道了这些之后，我们可以以类和目的为导向这个方法来分析一下如何学习datetime这一模块。首先，datetime这一模块中主要有四个类，分别是date，time，datetime，time delta。我们以date作为分析的对象，其余三个类可以参考下面的思维导图。首先，先明晰date的本质：存储日期，年月日，也就是说，如果我们有需要将年月日这一数据（如2022-1-10）存储起来，我们可以使用date。其次date的属性是描述数据的信息的，其中的date.year用于获取存储在date中的年份这一数据信息，date.month,date.day用于获取月和日这一数据信息。最后，也就是最重要的一步——方法的学习，根据本人的学习需求（目的），我只学习了四个层面的方法，也就是掌握了四种处理date数据的方式，第一个层面是比较两个date数据的大小，第二个层面是用于处理两个date数据的差或和，第三个层面处理将date中的数据信息进行替换，第四个层面也是最重要的一个，用于获取当前的时间的年月日并以date存储起来。方法根据需求来决定是否去学习，往小了看，一个方法中有很多个参数，如果我们不需要使用或者很少用到其中的一些参数，那么我们也可以适当舍弃，如plt.bar（）中的一个参数data基本很少用到；往大的方面来看，函数的学习取决于我们的需求，那么类呢？如果我们不需要掌握某种数据存储方式，如set集合，虽然是基础的数据类型之一，但是如果用处不多，不学习似乎也是可以的（从一个非计算机专业的视角来看）。类是这样，那么模块呢？模块也可以按照自己的需求进行处理。因此，根据需求（目的）来决定自己是否学习某个模块，某个类，某个函数，当开始学之后就以类为导向进行学习，这就是以需求和类为导向的学习方法，这个方法最重要的一点在于框架的构建。

同时，还需要跟学习者讲明一点：由于函数是根据你学习时候的需求决定的，而需求会随着时间的变化而变化，当我们产生了新的数据处理需求怎么办？如一个文科生突然开始想学习如何计算方差和标准差了，因为之前没有学过，这时候该怎么办？此时，可以去网上根据自己的需求搜索，并丰富自己的思维导图。这种方法就是边学边用，边学边忘，可能对于一个非计算机类的学生而言有点难，但是如果我们知道了框架是什么，为什么要去在意这些细节呢？毕竟人脑的存储空间是有限的，而我们所需要的信息都可以在网络上面找到。



现在再回来看一下主流的教学方式，是以函数为主，意思就是说，不管自己的需求如何？尝试着把每个类都学一遍，尝试着每个函数都去了解一下，尝试着把函数的每个参数都了解一遍。

这个学习方式的后果是什么呢？以自身的感受来说，这种学习方式有“三难“。

第一难，函数数量多，函数参数多，函数学习难。由于python是开源的，当有人产生了一种新的数据处理方式的需求时，便会开发出一个新的类，新的函数，因此里面的函数数量多如牛毛，而且函数的难度也会随着数据处理需求的提高而提高，因此凭借个人力量想把这些函数学完，基本不太现实。

第二难，心情难。由于函数多，难度大，在学习的时候，不免会产生“此路漫长无尽头“的感觉，同时由于缺乏明确的需求，似乎大部分学习的函数也不知道可以用来干嘛，就会产生学习python也没有用的想法，从而造成了学习者沮丧的心情；

第三难，学习效率难。由于沮丧，那么学习者便很难投入热情去学习python，那么python的学习效率也就可想而知了。

这三难是相互循环促进的，由于难度高，导致了心情难，最终影响了学习效率，而学习效率的低下又会导致学习难度变高，如此循环往复，不断加深。



其实金融学生去学习python，最重要的不是学习开发函数，而是学习如何使用函数，就像工具的开发和工具的使用一样，我们不会开发，但是不影响我会使用，因而不用像计算机专业一样专研的很深。但学习python，最终还是要落到学习函数这一方面，只是个人觉得不应该只专注于函数，不应该让大家产生一种每个函数都要学的感觉，只需要令学习者明白学习python的框架和思路，剩下的交给他们自己去做就行了，每个人学习python都有自己的需求，应该让python成为一个根据自己需求而使用的工具，而不是沦为python的一个学习工具。

（2）教学时间太短，而教学内容太多，感觉时间不够。