**

**HUNAN UNIVERSITY**

《DM&BID》

Report 15

|  |  |
| --- | --- |
| **报告名称：** | 智能推荐系统 |
| **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：** | 2023.5.9 |

目录

[一、智能推荐系统-协同过滤算法 3](#_Toc1807294701)

[（一）相似度计算的三种常见方法 3](#_Toc1193336539)

[1.1欧式距离 3](#_Toc1057431640)

[1.2余弦相似度 3](#_Toc1826394555)

[1.3皮尔逊相关系数 3](#_Toc82035667)

[（二）案例：电影智能推荐系统 4](#_Toc88953895)

[2.1案例背景 4](#_Toc399494953)

[2.2读取数据 4](#_Toc1277794549)

[2.3数据分析 6](#_Toc1056515043)

[2.4数据处理 6](#_Toc1453534305)

[2.5智能推荐 7](#_Toc1924579510)

[二、 知识小记 9](#_Toc1009133456)

[（一） 协同过滤算法 9](#_Toc1827634633)

[1.1算法原理 9](#_Toc1596673790)

[1.2基于用户的协同过滤算法 9](#_Toc340735618)

[1.3基于物品的协同过滤算法 9](#_Toc1552128824)

[（二） 相似度计算 10](#_Toc1145284859)

[2.1欧式距离 10](#_Toc906697152)

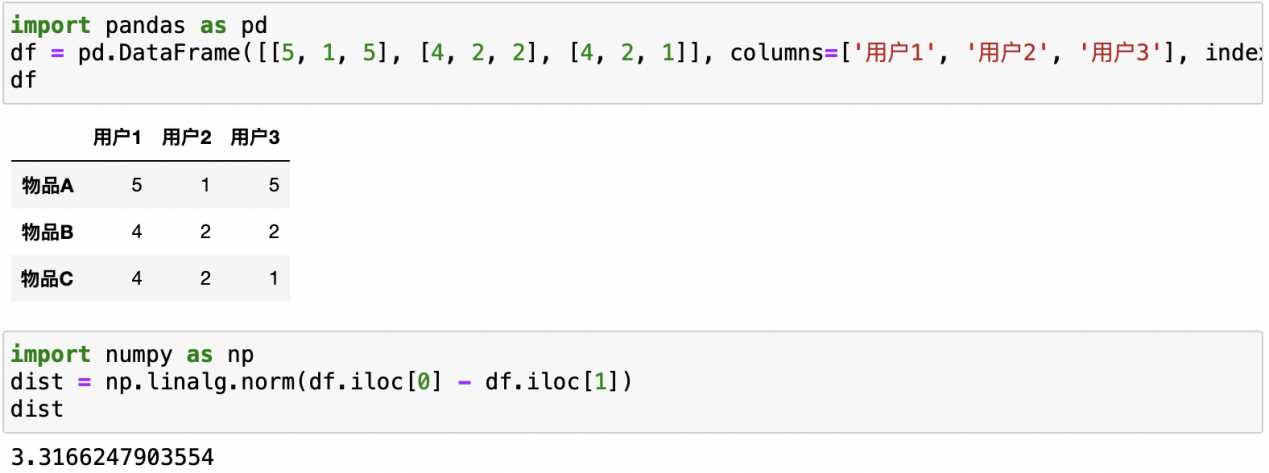
[2.2余弦相似度 11](#_Toc315074552)

[2.3皮尔逊相关系数 12](#_Toc1910805609)

# 一、智能推荐系统-协同过滤算法

## （一）相似度计算的三种常见方法

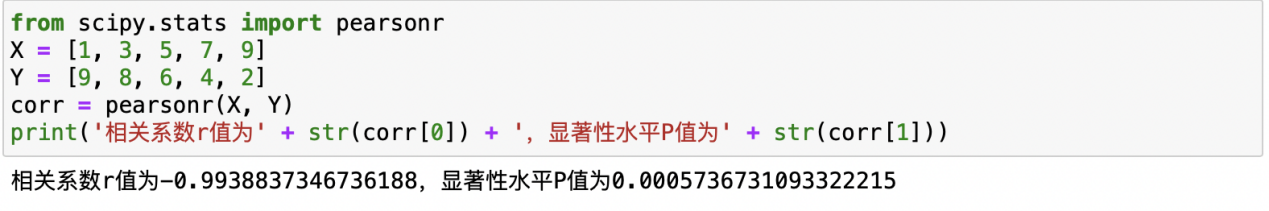
**1.1欧式距离**

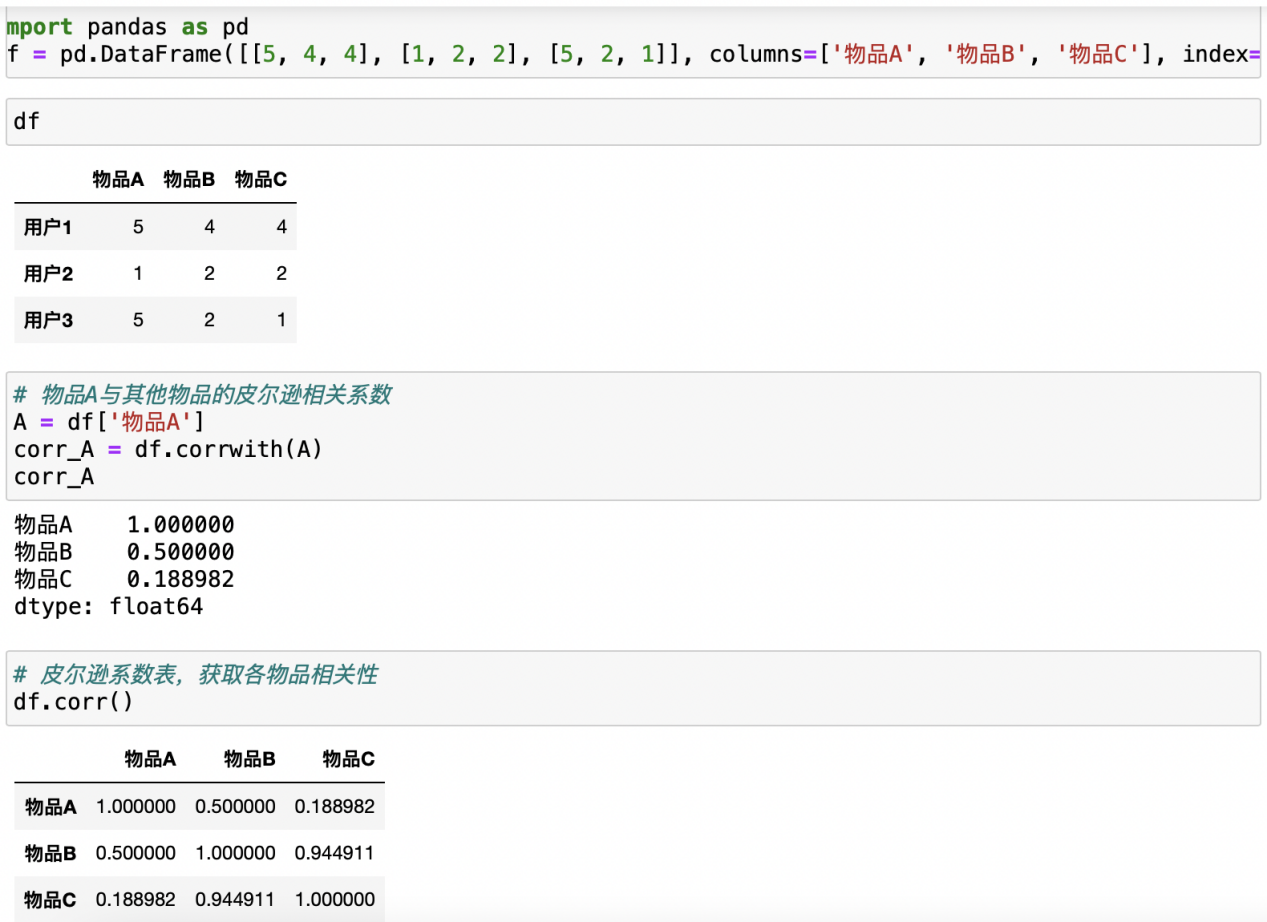
****

**1.2余弦相似度**

****

**1.3皮尔逊相关系数**

****

****

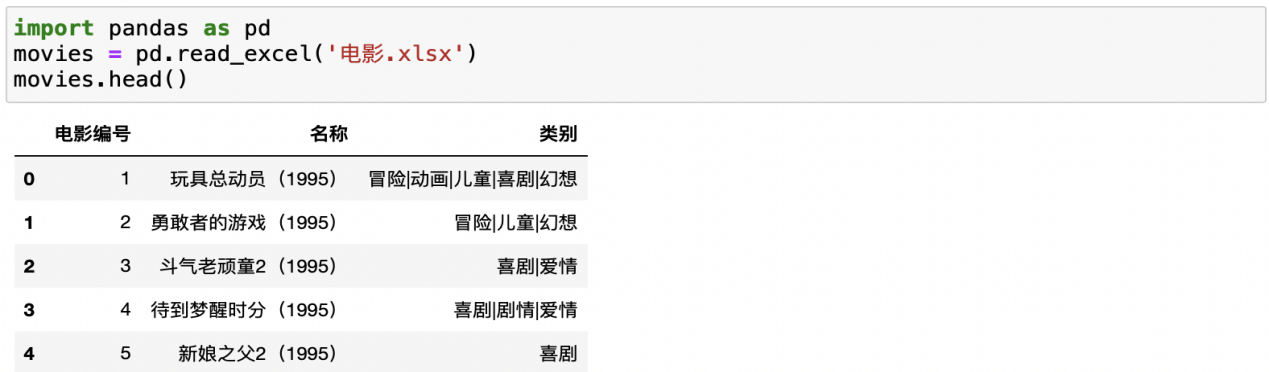
## （二）案例：电影智能推荐系统

**2.1案例背景**

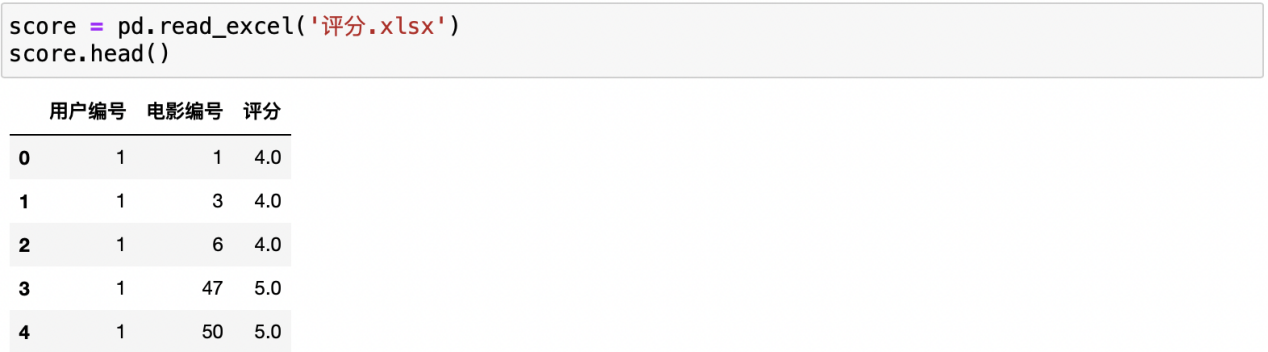
人们经常会在视频平台上观看影片，有时目标明确，想要观看某部电影，但有时仅仅是随机搜寻。如果视频平台可以利用基于物品的智能推荐系统，有效地从用户对其观看过的电影的评分中挖掘数据，便可以根据用户偏好的电影个性化地推荐更多类似的电影，优化用户体验，提高用户粘性，创造额外收入。（此案例基于皮尔逊系数）

**2.2读取数据**

通过pandas库导入9712条包含电影名称和类型的数据和关于这9712部电影的100836条评分数据：



使用score.head()展示电影评分：

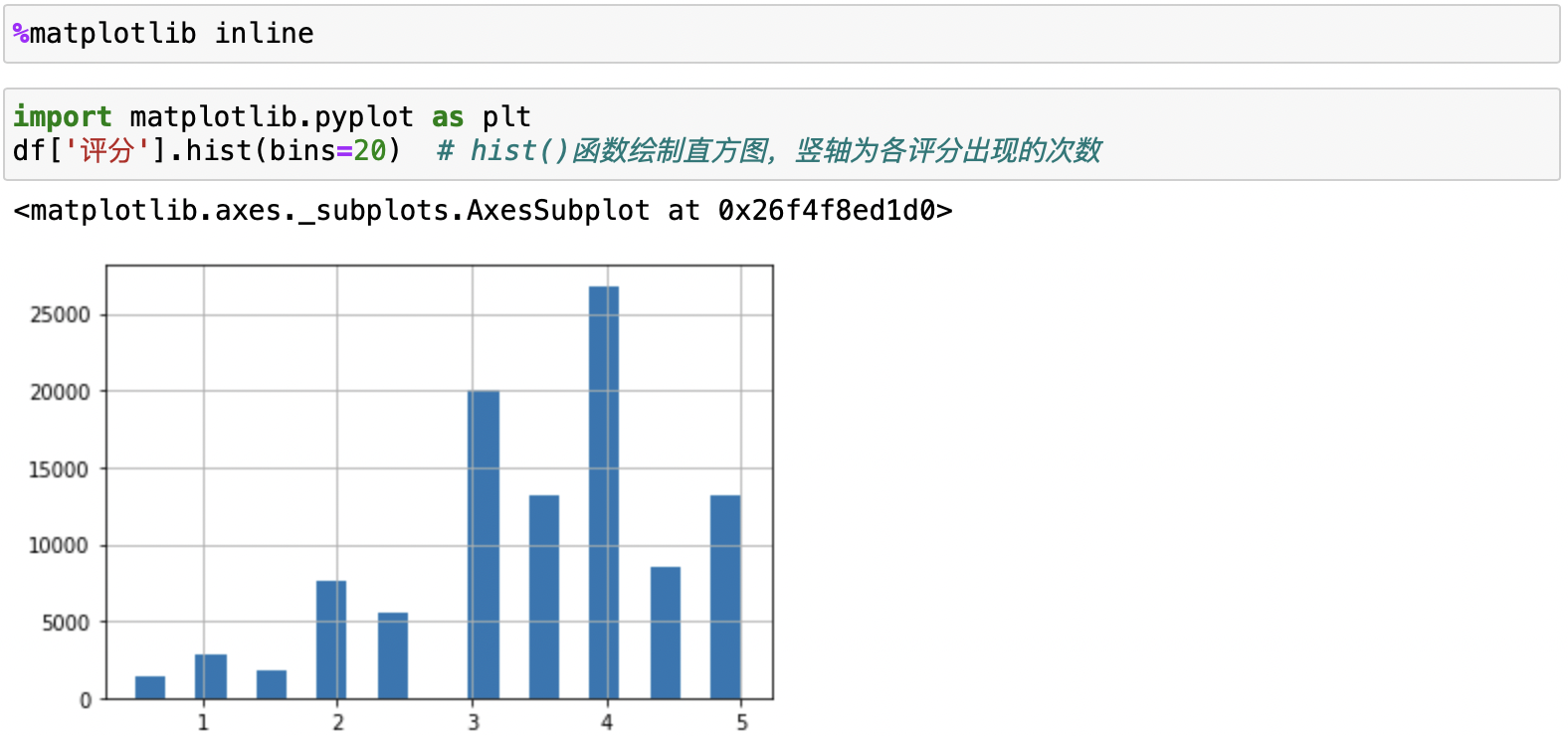


使用merge()函数通过电影编号连接起来:

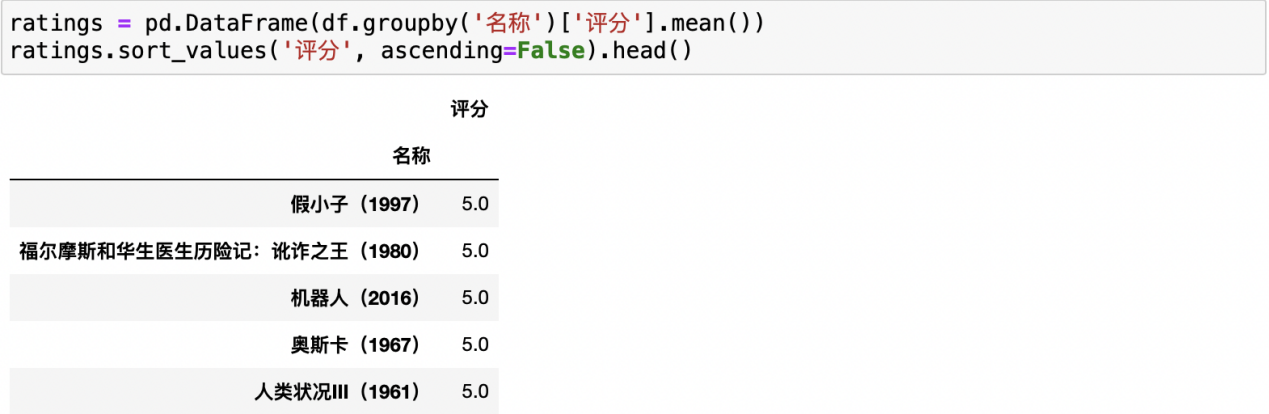


将该汇总表导出为一个新的Excel文件，并建立一个直方图来查看评分的分布：



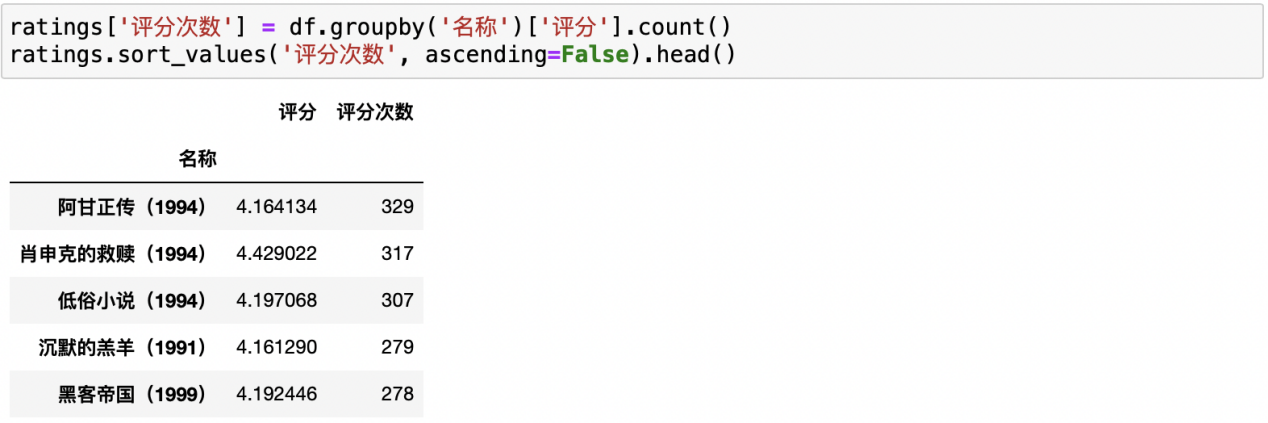


**2.3数据分析**



可以看到这5部电影的观影平均分为5分，这可能是因为打分人数较少且普遍打分较高的缘故。

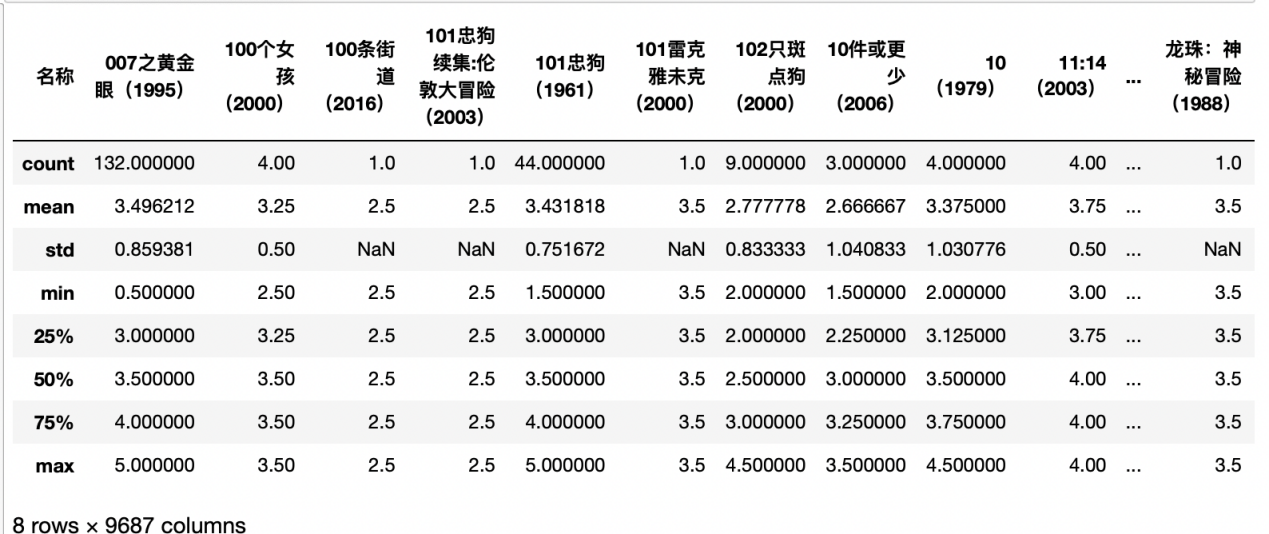
通过pandas的groupby()分类函数按名称进行归类，并通过count()函数对每部电影的评分进行计数，然后为每部电影的评分的数量创建新的一列：“评分次数”：



**2.4数据处理**

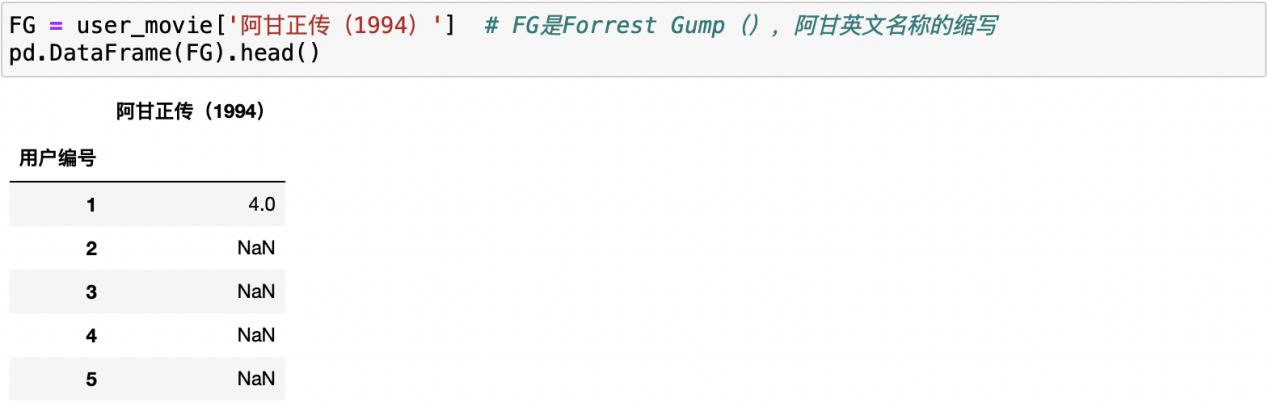
使用pandas中的pivot\_table()函数对变量df创建数据透视表并赋值给变量user\_movie：此处将pivot\_table()函数的参数index设为用户编号，使用户编号成为数据透视表的索引；将参数columns设为名称，使电影名称成为数据透视表的列；将参数values设为评分，使电影评分作为数据透视表显示的数、



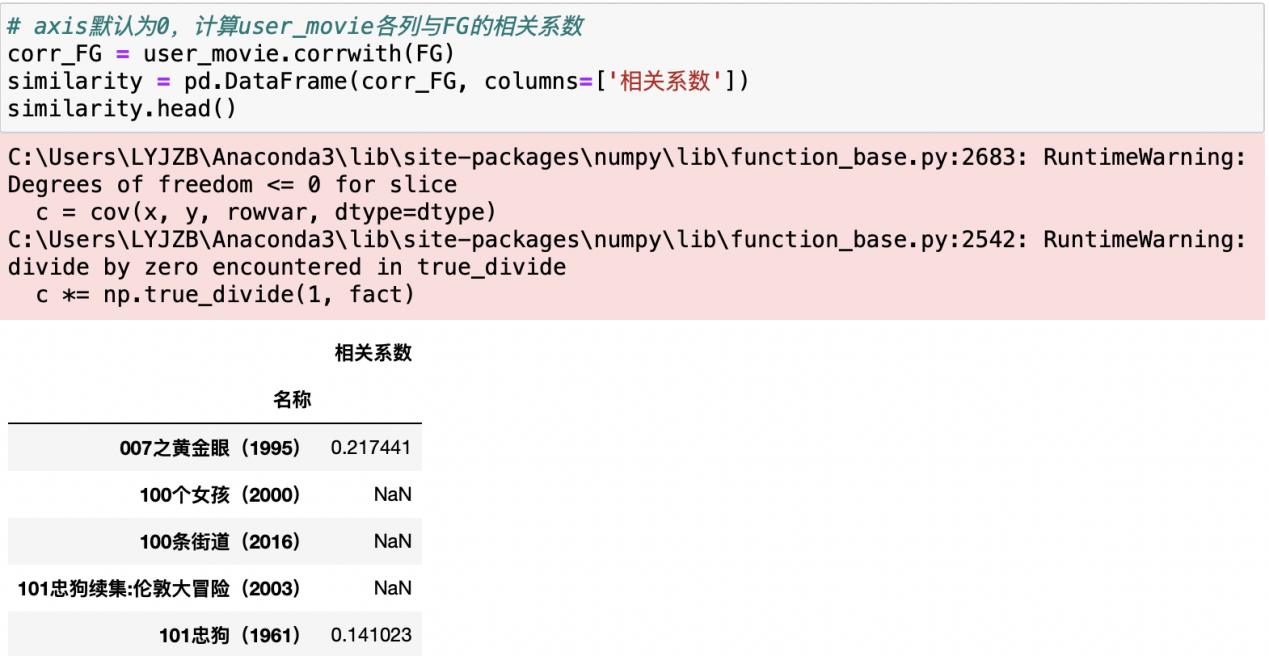


**2.5智能推荐**

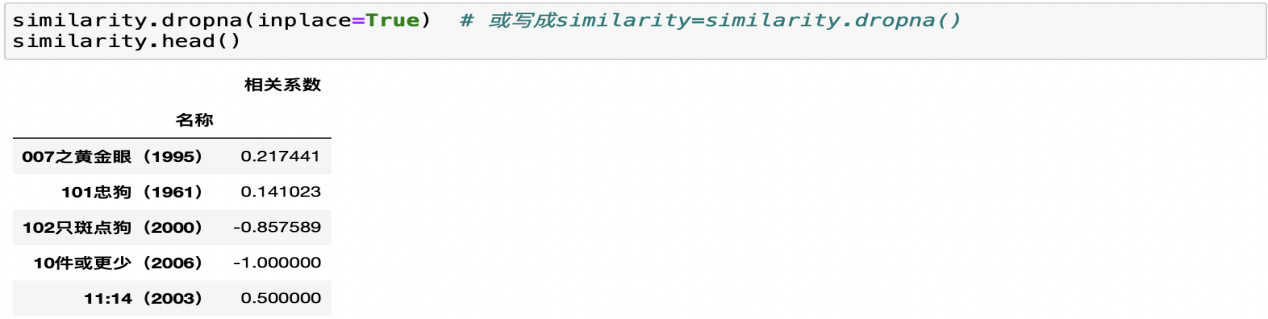
这里利用之前处理好的数据来进行相关性分析，作为演示，选取《阿甘正传》作为案例，来分析应该向观看了《阿甘正传》的观众推荐什么样的电影。首先展示各用户对《阿甘正传》的评分，使用head()函数显示前五行，其中FG是阿甘正传的英名称：Forrest Gump的首字母缩写，代码如下：

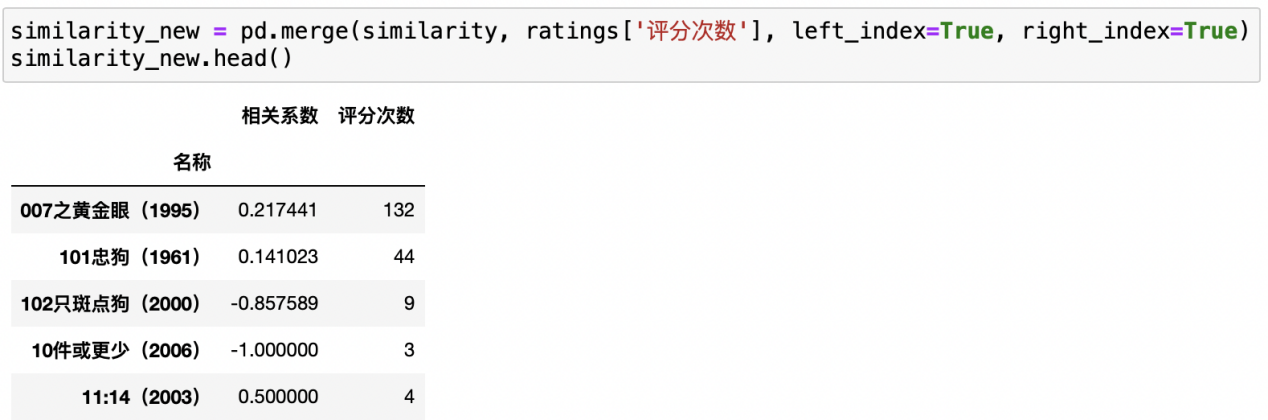


利用corrwith函数计算《阿甘正传》与其他电影间的皮尔逊相关系数：



利用merge()函数或join()函数，按行索引对齐合并的方式把表格similarity和表格ratings进行合并，这样就可以把每部电影与《阿甘正传》的皮尔逊相关系数的信息和每部电影对应的评分次数信息放在一张表格中显示。





设置阈值，只有当电影评分次数大于该阈值时才认为该电影的总体用户评分有效，此处可以简单地设置为20。之后通过sort\_values()函数根据相关系数这列数值进行倒序排列：



# 知识小记

## 协同过滤算法

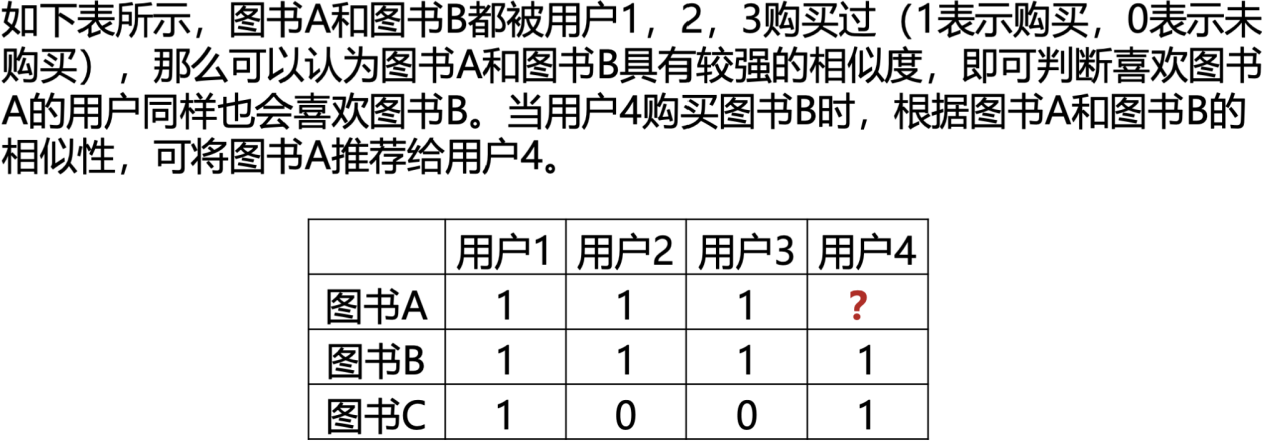
**1.1算法原理**

协同过滤算法的原理是根据用户群体对产品偏好的数据，发现用户之间的相关性或者物品之间的相关性，并基于这些相关性为用户作推荐。根据原理的不同，协同过滤算法分为如下两大类：基于用户的协同过滤算法和基于物品的协同过滤算法，而其本质都是寻找相似度。（基于物品的协同过滤算法使用较多）

**1.2基于用户的协同过滤算法**



**1.3基于物品的协同过滤算法**



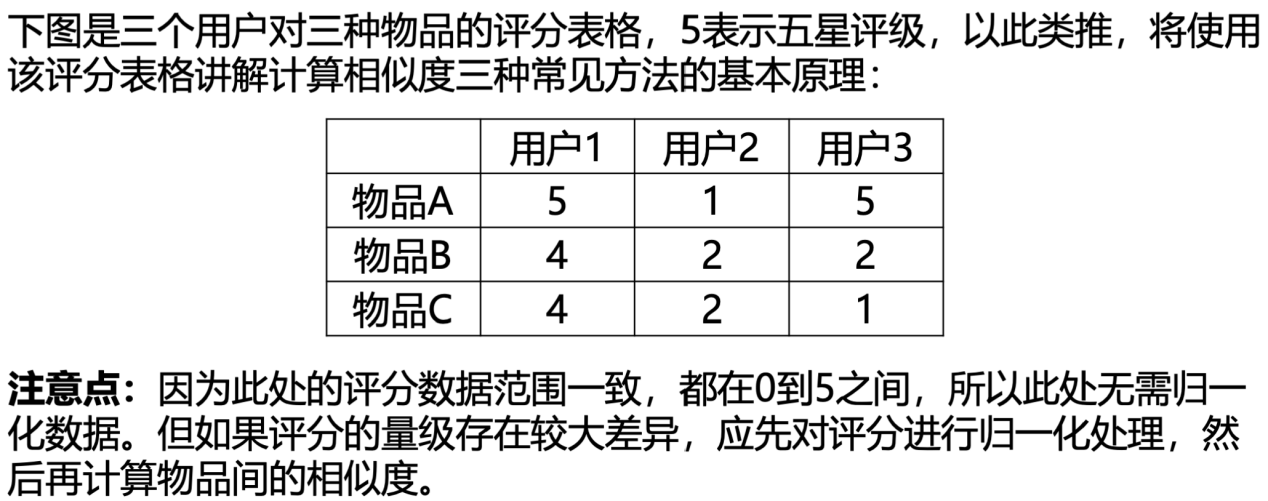
在商业实战中，大多应用场景偏向于使用基于物品的协同过滤算法，主要有如下两个原因：

·原因一：通常用户的数量是非常庞大的（如淘宝数亿的用户群体），而物品的数量相对则比较有限，因此计算不同物品之间的相似度往往比计算不同用户的相似度容易很多。

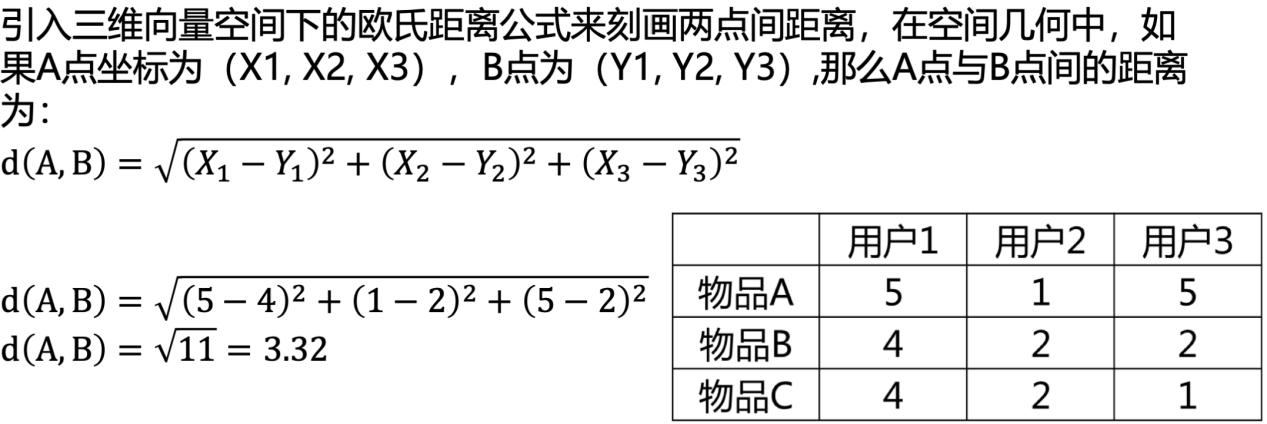
·原因二：用户的喜好较为多变，而物品属性较明确不随时间变化，过去用户对物品的评分长期有效，所以物品间的相似度比较固定，因此可以预先离线计算好物品间的相似度，把结果存在表中，向客户进行推荐时再使用。

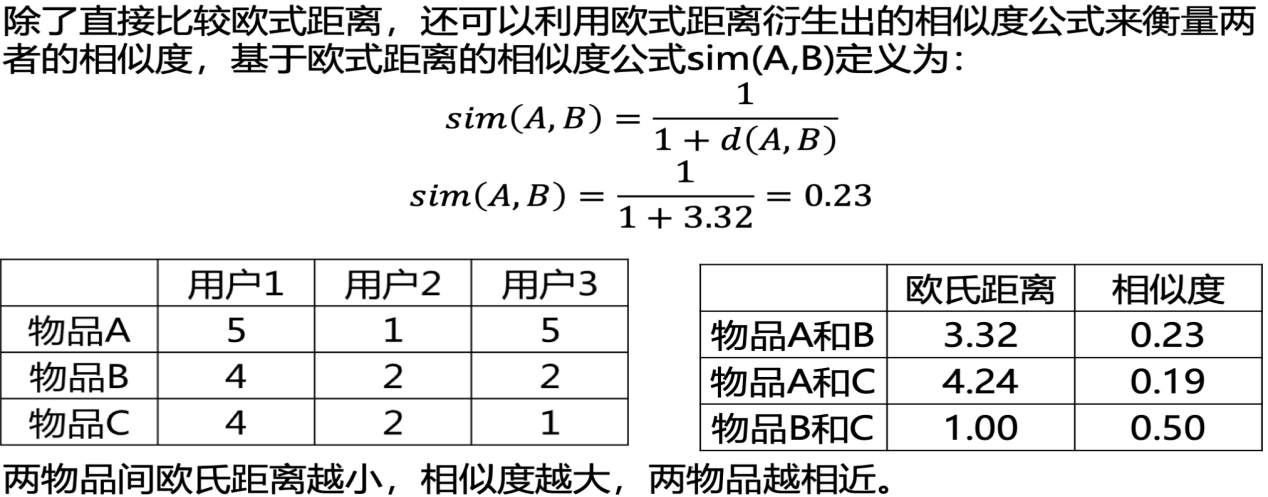
## 相似度计算

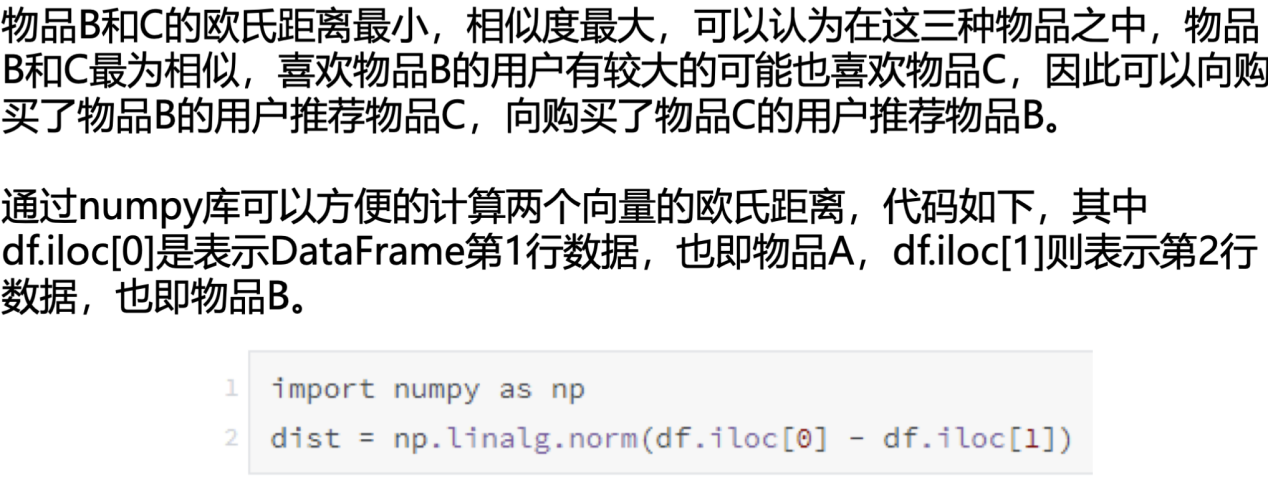
相似度计算有三种常见的方法：欧式距离、余弦值、皮尔逊相关系数



**2.1欧式距离**

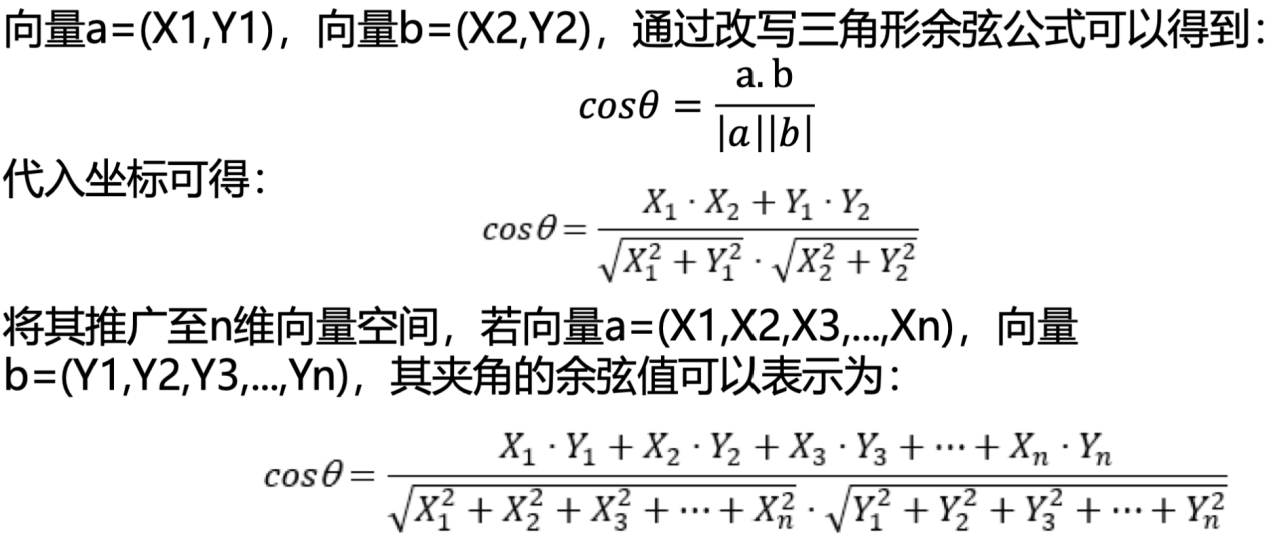




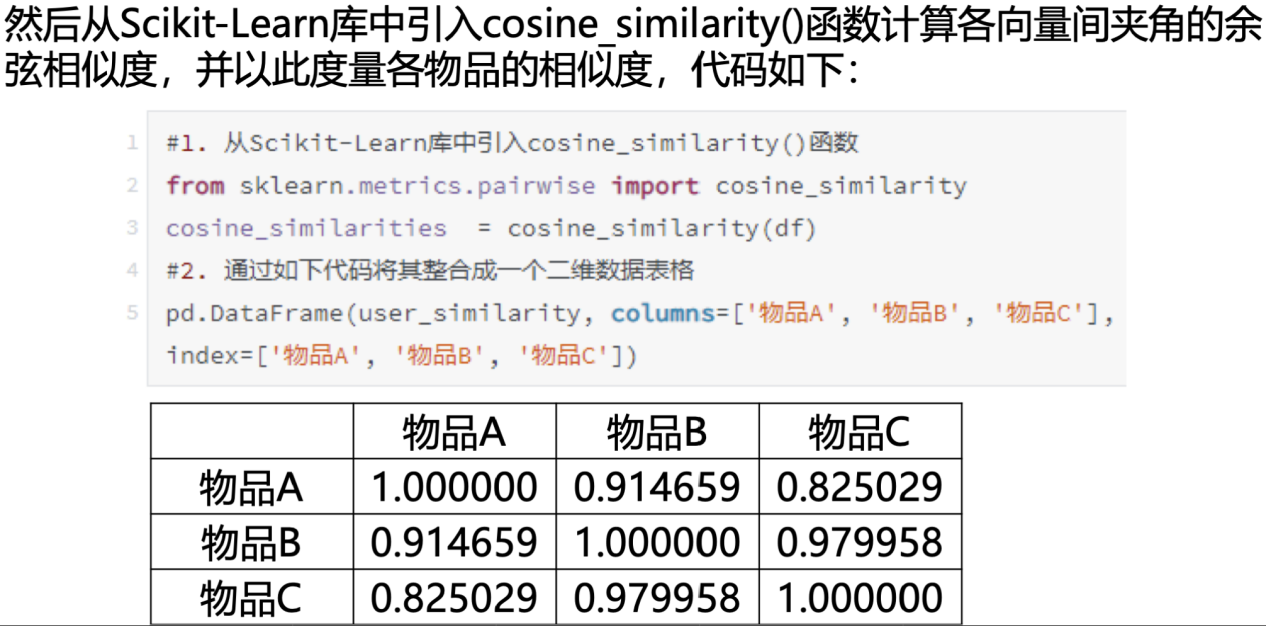


**2.2余弦相似度**

在二维向量空间下可以使用两向量夹角（θ）的余弦值（cosθ）来表示两个向量的相似度，称为余弦相似度。余弦相似度的范围在0-1之间，夹角越小，余弦值越接近于1，两个向量越靠近，两者越相似。







**2.3皮尔逊相关系数**

皮尔逊相关系数r是用来描述两个变量间线性相关强弱程度的统计量，r的绝对值越大表明相关性越强。r取值范围为[-1,1]，为正代表两个变量存在正相关，为负代表两个变量存在负相关。其计算公式如下

