Numpy

用于科学计算的工具包 主要用来处理多维矩阵 Cpython开发 打破了GIL锁，效率比较高

安装numpy

pip install numpy

使用Numpy

import numpy as np

在numpy中 使用Ndarray对象来代表多维数组（N维）

1. 创建一个Ndarry对象

np.array(list)

nd=np.array([[1,2],[3,4]],ndmin=2,dtype=np.int) *# object:list ndmin指定最小维度 dtype指定元素的类型(int,float)*

属性：

print(nd.T) *# 转置*

*#nd.dtype=np.int16 改变数组中元素的类型  
#print(nd)  
#print(nd.shape) #(2, 2) 查看数组的形状  
#nd.shape=(4,4) # 修改数组的形状 要求元素个数不能改变  
#print(nd.size) # 查看元素个数  
# nd.size=8 #！！！！错误 不能修改  
# print(nd.ndim) #查看数组的维度*

方法：

矩阵的秩：矩阵的维度

轴：行列的顺序 行0轴 列1轴 更高维度往后查

推荐系统

为了提高用户的体验感

协同过滤算法

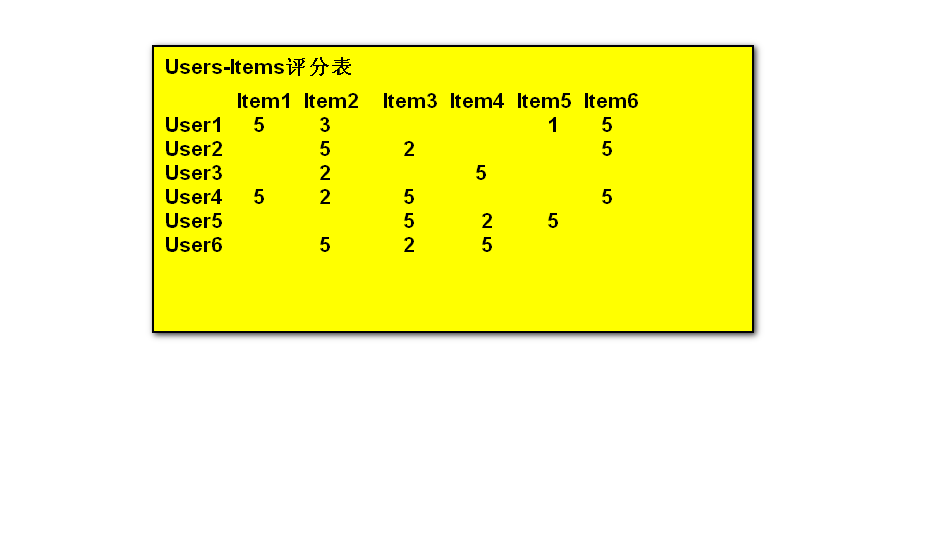
协同：找相似用户

过滤：过滤掉用户消费过的商品

基于用户的推荐系统 给人推荐商品

1. 计算该用户与所有用户之间的距离 找到距离该用户最近的那些用户
2. 找到相似用户的喜好物品
3. 过滤掉该用户已经消费过的商品
4. 最终输出剩下的商品

用户—商品评分表



代码实现思路：

1. 加载数据
2. 计算用户之间的距离 1. 欧式距离 2.余弦距离 3.杰卡德距离 计算两个集合之间的距离 交集/并集
3. 取距离最近的前k个用户
4. 求k个用户对所有商品的平均评分
5. 过滤掉用户已经消费过的商品
6. 取出平均评分最高的前n个商品或者取出所有满足阈值的商品
7. 输出所有取到的商品

基于商品的推荐系统 给商品推荐人