一、机器学习实战p256， 推荐系统。二 、bilibili上关于推荐系统，看看。

1. 协同过滤和聚类可以组合起来使用。如果数据规模非常大，协同过滤中计算KNN（在全局user或item中找K个最近的邻居）是非常耗时的。但如果先对user （或item）做一个聚类，然后在目标user（或item）所属聚类簇中找KNN，可以极大地节省计算开销
2. 种计算相似度的方法，欧式距离，皮尔逊相关系数，余弦相似度（<http://blog.csdn.net/google19890102/article/details/28112091）。（http://blog.csdn.net/yimingsilence/article/details/54934302>）

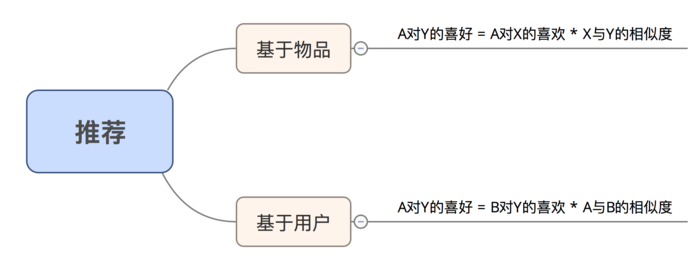


图1

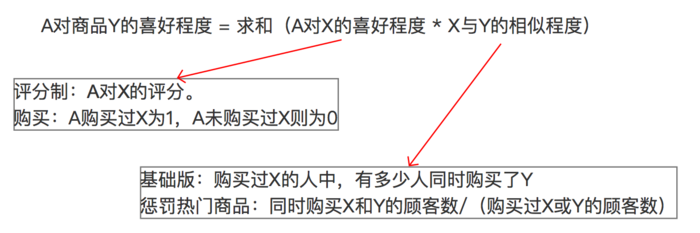


图2

1. **基于用户的协同过滤**，计算两个用户之间的相似性，看A,B,C,D听同一首歌的次数，找出4个用户的听歌记录，然后搜查听相同歌曲的数目以及每首歌的被听次数，计算二者的相似性。具体怎么实现看看简书（<https://www.jianshu.com/p/e56665c54df8>）。
2. **基于物品的协同过滤**，举例说明，A购买了H、I两种商品。商城还有J、K、L、M、N五种商品，我们要判断用户A对JKLMN的喜好程度。

A对J的喜好程度 = （A对H的喜好程度\*H与J的相似度）+ （A对I的喜好程度\*I与J的相似度）。

对于KLMN都是同样的算法。然后对JKLMN进行排序，就可以推荐排序前两名的给用户A了。推荐列表就是这么生成。

用户A对商品X的喜好程度怎么算，两种：1.打分制，2.是否购买这件物品，购买为1，不购买为0. 计算X与Y的相似度见图2

1. 基于用户相似度的推荐中，一般采用Pearson相关系数；

基于物品相似度的推荐中，改进的余弦相似度效果更好。

1. 推荐引擎的评价。如何对推荐引擎进行评价呢？此时我们既没有预测的目标值，也没有用户来调查他们对预测的满意程度。这里采用**交叉测试**的方法。具体的做法是将某些已知的评分值去掉，然后对他们进行预测，最后计算预测值与真实值之间的差异。

通常用于推荐引擎评价的指标称为**最小均方根误差**（Root Mean Squared Error, **RMSE**），它首先计算均方误差的平均值然后取其平方根。如果评级在1星到5星这个范围内，而我们的RMSE为1.0，那么意味着我们的预测值和用户给出的真实评价相差一个星级。