**二分网络上的链路预测实验报告**

17030140014 张笑天

1. 实验内容

1、采用二分网络模型，对ml-1m文件夹中的“用户---电影”打分数据进行建模，考虑将用户信息、电影详细信息、以及打分分值作为该网络上的边、点的权重；

2、根据网络结构特征给出节点相似性度量指标；

3、基于相似性在二分网络上进行链路预测；

4、采用交叉验证的方法验证预测结果；

5、画出ROC曲线来度量预测方法的准确性。

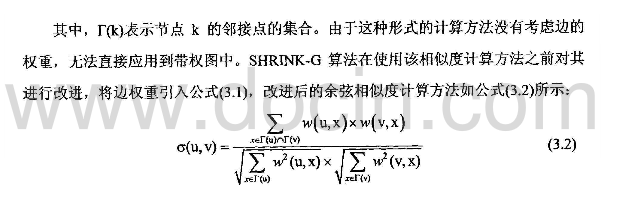
1. 分析与设计

参考：<https://people.maths.ox.ac.uk/porterm/research/mariano_thesis_final.pdf>

**注意：本算法自己设计，因此可能存在很大缺陷。**

将用户ID，电影ID作为二分图的两类结点，同时打分作为边的权重。

相似度指标：使用对于和某个用户共同评分的电影的带权重余弦相似度作为指标。



预测方式：取最为相似的前SEARCH\_NUM个节点，取众数，如果众数>=3则喜爱，否则不喜爱。

1. 详细实现

使用Python编写代码

使用networkx作为实现网络的包

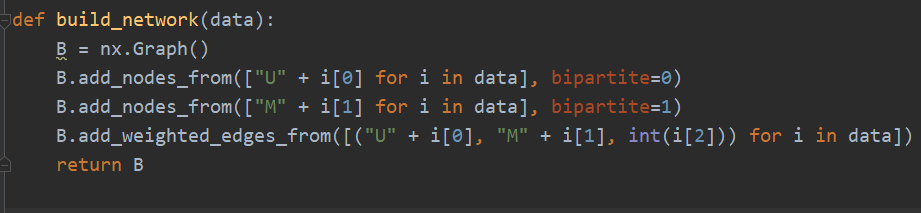
使用sklearn计算一些统计数据，进行数据生成。

使用matplotlib绘制ROC曲线

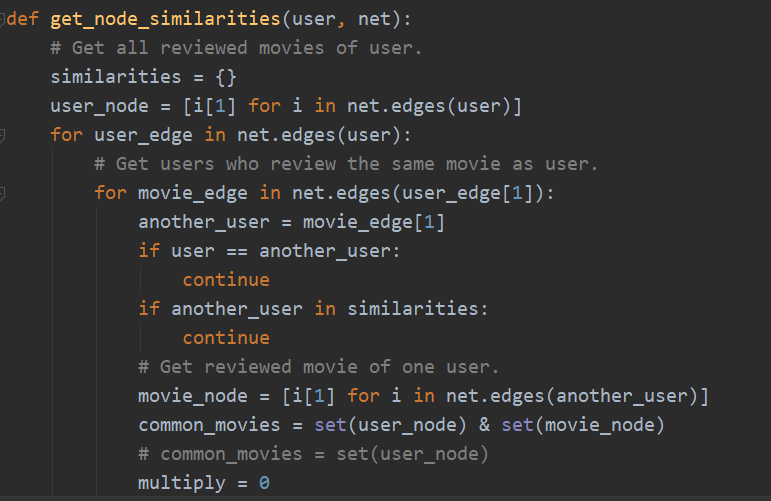
使用numpy构建矩阵

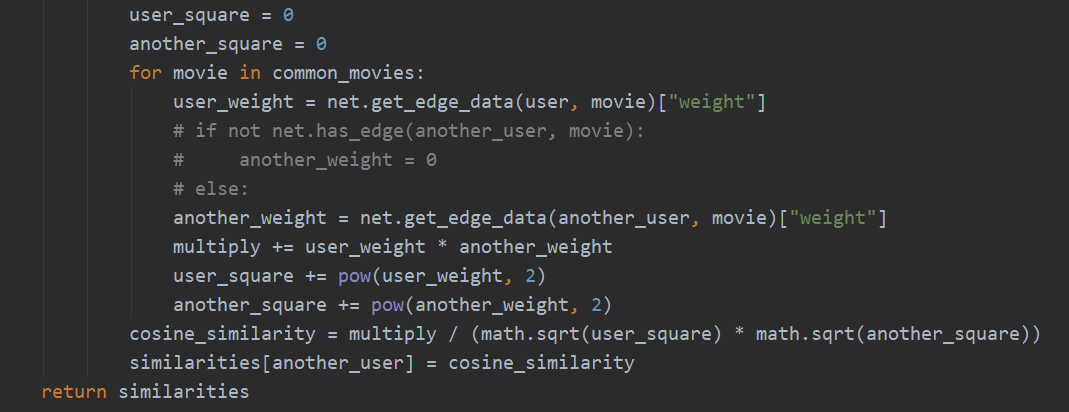
主要的函数：

build\_network(data)：由于用户和电影的ID存在重合，因此在前面加上字母作为区别，然后构建二分图。

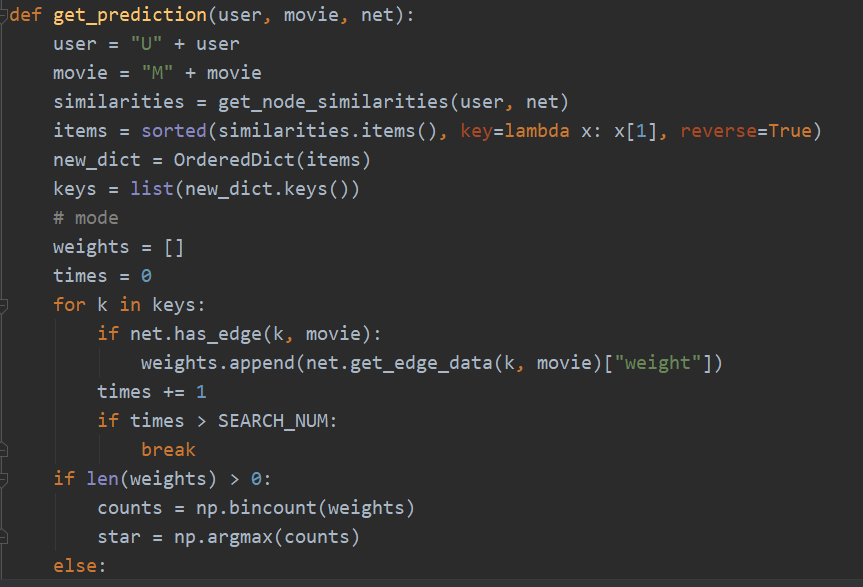


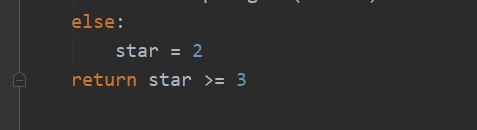
get\_node\_similarities(user, net)：用来计算对于某个用户在图中相似的节点，思路：寻找这个用户看过的所有电影，针对每个电影，寻找同样看过的用户，并记录下来。取两个用户共同看过的电影，并且计算相似度，得到相似度字典。





get\_prediction(user, movie, net)：针对某个用户预测其对电影是否喜爱。思路：得到用户的相似度字典后，求最为相似并且评论过要求电影的前SEARCH\_NUM名用户的评分众数作为返回结果。



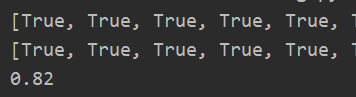


1. 实验结果

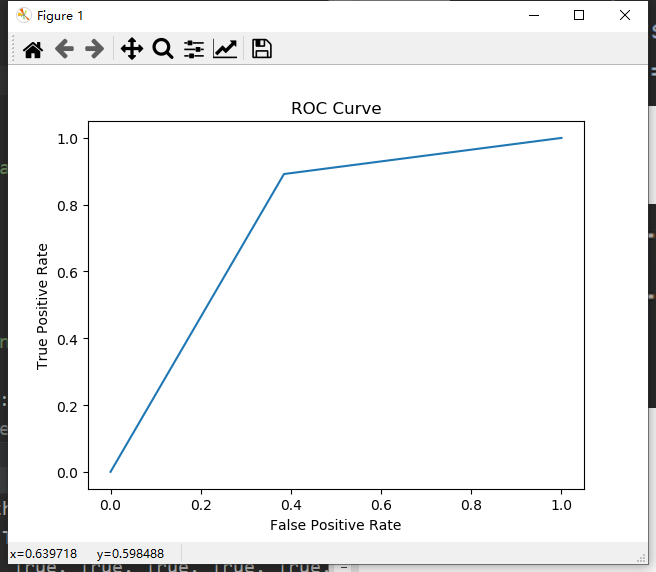
算法缺陷

仅仅使用很少测试样本，因为本算法搜索过慢，一定程度提高SEARCH\_NUM有利于提高准确度。

准确度：



ROC：



1. 心得体会

本次算法均为自己设计并且寻找相似度。

设计中遇到了很多阻碍，包括对于链路分析概念不清楚，对于算法时间估计有问题。

本算法依旧存在很多缺陷，例如计算相似度时没有考虑共同评论电影的数量，搜索时过慢等等。