k-NN 计算问题（前 k 个最近的用户）

User-based 的传统实现方式是对于用户 u_i, 计算与 u_i 最为相似的前 k 个用户。然后根据这 k 个用户对 u_i 未打分的 item 进行预测评分。这里着重强调的是：“计算与 u_i 最为相似的前 k 个用户”.

这个思路是存在问题的：1. 前 k 个用户可能对于某个 $item_i$ 都没有与之对应的评分记录，那么 $item_i$ 相对于 u_i 依然是缺失的，除非选择的 k 比较大，能够 cover 住 $item_i$ 的历史记录; 2. 前 k 个用户都对 $item_i$ 进行评分也是不太可能的，除非这个 item 十分火热。那么采用极少的 $item_i$ 评分记录来预测 u_i 对 $item_i$ 的评分，虽然能够得到评分，但参考的背景信息太少，准确度不一定能够得到保证。

这里的方案是改为对每个 $item$ 寻找存在打分记录的所有用户 U, 在 U 中找出与 u_i 最为相似的前 k 个用户，然后对 $item_i$ 的评分进行预测。但是这样做复杂度明显高，或许存在折中的方案。

用户的背景差异

每个用户之间的打分喜好是不同的，有的人倾向于一直打高分，有的人倾向于总是打低分。有的人只会对特别喜欢的打出高分，其他的总是一直低分，甚至是不评分，就好比淘宝评论，大多数人在遇到非常好，或者非常差的购物体验的时候才会去评分。对于这样差异很大的情况，第一种方案是对每个用户的打分记录进行正则化。正则化后还需要注意的是在获取真正的评分记录时，能够反转回原来的评分区间。

item 的评分区间

$item$ 的评分区间如果是比较连续的情况，采用权重的方式（通过计算用户间的相似度，然后累加权重评分求平均的方式）来计算是比较合适的。对于评分区间离散（比如只有两项：“好”，“坏”），那么更适合分类的方式进行处理。也就是回归（Regression，求权重均值）还是分类（Classification，求 item 应该打到哪一类）问题的区别。同时，一个 item 所有的打分记录背后会满足某个分布（比如正态分布），当打分记录数量不断增多时，这个 item 的打分均值趋于稳定，直白了讲，可能借助这信息发现了大多数人都不喜欢的 item，是否可以结合这个信息，做一些有意义的事情?