# 试探共享经济的信誉系统：以 Airbnb “星级评分”的影响因素为切入点

Python 课程 · 津南校区第一组

1612555 顾　容　菱
1613641 张　靖　昱
1711342 李　　纪
1812972 艾 合 买 提
1811144 李　　岳

2019 年 12 月 26 日

# 目录

# 目录

# 共享经济


- 定义：共享经济是指利用互联网等现代信息技术，以使用权分享为主要特征，整合海量、分散化资源，满足多样化需求的经济活动总和。
- 信誉系统是当今互联网共享平台的一个基本组成部分，特别是对于具有不同用户、商品和服务的 P2P（peer-to-peer）平台，而 Airbnb 作为最成功和最有影响力的 P2P 平台之一，其信誉系统具有较高的调查价值。

# Airbnb 及其信誉系统介绍

- **双边信誉系统**
  - 该平台上的用户可分为房东（卖方）和房客（买方）两类。该平台允许房东和房客互相进行评估和评分。
  - 为了促进评价的公正性和真实性，只有在双方都完成了评价后或 14 天的评价期结束后，评价才会显示在页面上。
  - 与其他主要旅游点评平台不同，Airbnb 不会公开与单个点评相关的星级评级，只会公开文本评论以及每个房源的点评汇总统计数据。
  - 我们今天探讨的重点是四舍五入到最近的半星的平均房源评级；我们称之为是“星级评分”。它仅仅代表了一个是聚合的（即，平均值和四舍五入）的值，而无法检索哪个客人提交了哪个等级。

# Airbnb 及其信誉系统介绍


图 1: Airbnb 信誉系统

# 目录

# Airbnb 数据来源

- 本作业的基础数据来源于 InsideAirbnb.com。为增加样本量以及时间跨度、从而使结论更具普遍意义，本文选取了美国 New York、San Francisco、Austin、Los Angles 等 4 个城市在 2018 年 4 月-2019 年 9 月共 126887 个房源的月度数据。

表 1: 2018 年 4 月-2019 年 9 月 4 座城市的 Airbnb 房源的原始数据说明
数据来源：http://insideairbnb.com/get-the-data.html

| 城市 | 起始时间 | 结束时间 | 文件数目（真实） | 文件数目（扩充后） | 单文件大小（随机采样值） |
|---|---|---|---|---|---|
| New York | 2018/4 | 2019/9 | 18 | 18 | 180MB |
| San Francisco | 2018/4 | 2019/9 | 17 | 18 | 30MB |
| Los Angeles | 2018/4 | 2019/9 | 18 | 18 | 170MB |
| Austin | 2018/4 | 2019/9 | 17 | 18 | 45MB |

# TripAdvisor 数据来源

- 作为一个直接与酒店竞争的住宿平台，Airbnb 与酒店的评分比较或许可以为我们的主题提供线索。因此，我们爬取了另一家主要旅游网站TripAdvisor（中文名：猫途鹰）下的相关数据。该网站的主要特点为：
  - 房源类型多样;
  - 该网站自全球超过 1000 万注册会员的海量评价信息及超过 3 亿条旅游评论，使得房源的信息更具有真实性。
- 爬取过程说明
  1. 导入 Requests（用于请求网页）、Time（用于获取当前时间以及等待多长时间后执行程序）、Re（正则表达式，获取网页中特定内容）、BeautifulSoup（通过 XPath 获取网页中指定内容）、Pandas（数据读取与储存）、Numpy(数据处理)等工具包;
  2. 通过定义 get_hotel_url () 函数，传入酒店目录链接，从而获取该页面中所有的酒店链接;
  3. 用 get_hotel_rental_inf () 函数，传入具体的 hotel 链接通过正则表达式及 beautifulsoup 的 selector 获取该酒店的具体信息;
  4. 运行主函数，获取当前时间，然后循环爬取从该城市的第一页目录到最后一页，最后保存为 csv 文件。

# TripAdvisor 数据来源

- 我们共爬取了该网站上在 2019 年 12 月 6 日的共2340 家酒店（Hotel）（包括民宿（B& B））和 4286 家度假出租房（Vocational Rental）数据:

表 2: 2019 年 12 月 6 日爬取的度假出租屋的原始数据变量说明
数据来源：https://www.tripadvisor.com/Hotels

| 变量名 | 含义 | 变量名 | 含义 |
|---|---|---|---|
| title | 酒店名称 | owner_name | 房东 |
| rate | 评分 | owner_inf | 房东信息 |
| review | 评论数 | amenities | 设施 |
| id_locid | 位置 id | overview | 概述 |
| latitude | 纬度 | description | 描述 |
| longitude | 经度 | rules | 住房规则 |

# TripAdvisor 数据来源

表 3: 2019 年 12 月 6 日爬取的酒店和民宿的原始数据变量说明
数据来源：https://www.tripadvisor.com/Hotels

| 变量名 | 含义 | 变量名 | 含义 |
|---|---|---|---|
| title | 酒店名称 | good | 好 |
| rate | 评分 | ordinary | 一般 |
| Location_rate | 位置评分 | bad | 较差 |
| Cleanliness_rate | 卫生评分 | verybad | 很差 |
| Service_rate | 服务评分 | features | 房间特点 |
| Value_rate | 性价比评分 | types | 房间类型 |
| description | 描述 | low_price | 最低价 |
| location | 位置 | high_price | 最高价 |
| rank | 排名 | latitude | 纬度 |
| review | 评论数 | longitude | 经度 |
| great | 很棒 | locationid | 位置 id |

# 数据预处理

- **整合 72 张分表**

  为整合 4 座城市历时 18 个月的分表，我们先在每张表上都创建两个新变量，分别为 date（比如 201909）与 city（比如 Austin）；同时除去重复变量：即 listing_id 相同的记录只取一条。此时，我们便以 id、date 和 city 作为某一时间段、某一地点某个房源的唯一标识。

- **剔除无关变量及残缺变量**

  考虑到本文实证的需要，我们剔除了无关变量，并记录了剩余 43 个所需变量及其残缺值数量。同时去除 listing_id、room_type、property_type、price、latitude、longitude 等变量中有残缺的记录。

# 数据预处理

- 处理异常值

考虑到各类评分的数值范围，我们分别剔除了 review_scores_rating 的变量值不属于 [20,100] 的记录以及 review_scores_accuracy、review_scores_cleanliness、review_scores_checkin、review_scores_communication、review_scores_location、review_scores_value 的变量值不属于 [0,10] 的记录。最后发现有关评分的所有数值都无异常值。

同时考虑到后文对房源匹配是需要准确的位置信息，所以去除 is_location_exact 中非“t”（t 表示正确）的记录。

# 数据预处理

- 创建“星级评分”变量（star_rating）

参考 Teubner et al.（2018）对 review_scores_rating 到 star_rating 的转换方法，我们创建了新变量 star_rating。以下是具体转换规则：

表 4: review_scores_rating 转化为 star_rating 的规则说明

| star_rating | review_scores_rating | star_rating | review_scores_rating |
|---|---|---|---|
| 1 | [20,24] | 3.5 | [65,74] |
| 1.5 | [25,34] | 4 | [75,84] |
| 2 | [35,44] | 4.5 | [85,94] |
| 2.5 | [45,54] | 5 | [95,100] |
| 3 | [55,64] | | |

# 数据预处理

- **形成总表和分表**

  在上述的基础上，我们形成表“总 _ listings.csv”，共 1268644 个房源。之后，我们提取 2019 年 9 月的房源数据，成表“201909_ listings.csv”。

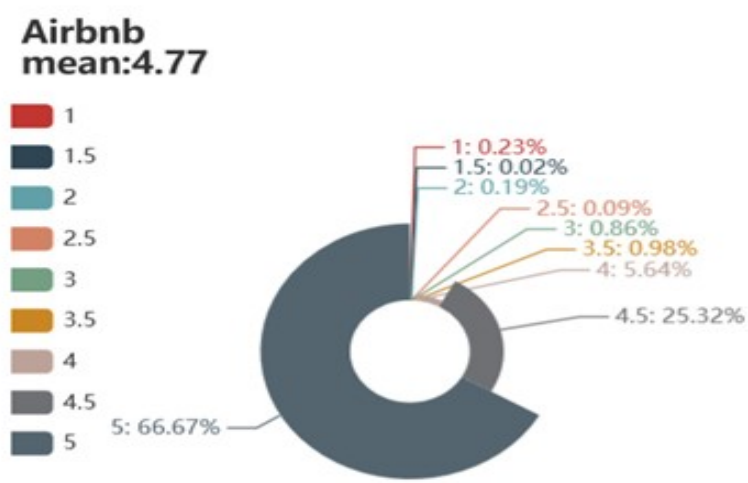
表 5: “201909_ listings.csv”数值变量的描述性统计

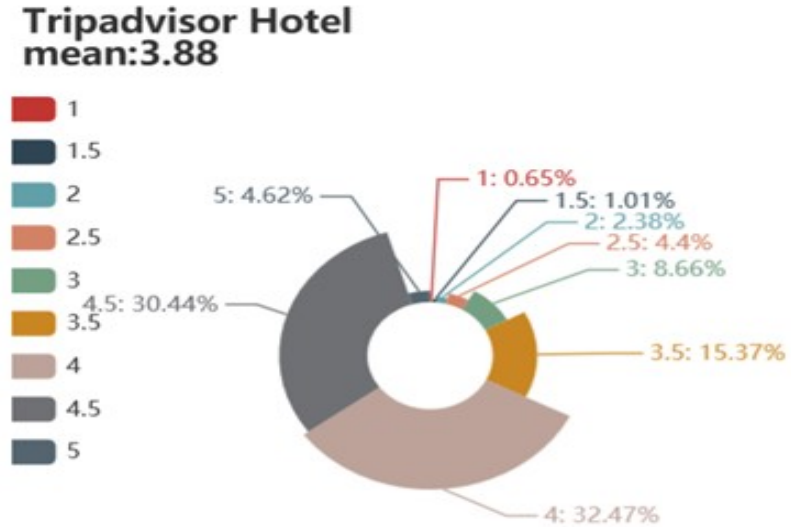
| Variable | Mean | Std | Min | 25%Q | Median | 75%Q | Max |
|---|---|---|---|---|---|---|---|
| latitude | 36.843 | 3.728 | 30.119 | 34.046 | 37.731 | 40.712 | 40.915 |
| longitude | -97.451 | 21.386 | -122.513 | -118.356 | -97.759 | -73.96 | -73.717 |
| accommodates | 3.45 | 2.425 | 1 | 2 | 2 | 4 | 32 |
| bathrooms | 1.307 | 0.735 | 0 | 1 | 1 | 1 | 15.5 |
| bedrooms | 1.345 | 0.976 | 0 | 1 | 1 | 2 | 22 |
| beds | 1.864 | 1.572 | 0 | 1 | 1 | 2 | 50 |
| number_of_reviews | 39.54 | 62.927 | 0 | 4 | 14 | 47 | 900 |
| review_scores_rating | 94.548 | 8.077 | 20 | 93 | 97 | 100 | 100 |
| review_scores_accuracy | 9.661 | 0.79 | 2 | 10 | 10 | 10 | 10 |
| review_scores_cleanliness | 9.4 | 0.988 | 2 | 9 | 10 | 10 | 10 |
| review_scores_checkin | 9.783 | 0.669 | 2 | 10 | 10 | 10 | 10 |
| review_scores_communication | 9.774 | 0.695 | 2 | 10 | 10 | 10 | 10 |
| review_scores_location | 9.655 | 0.706 | 2 | 9 | 10 | 10 | 10 |
| review_scores_value | 9.445 | 0.875 | 2 | 9 | 10 | 10 | 10 |
| reviews_per_month | 1.687 | 1.898 | 0.01 | 0.27 | 1 | 2.55 | 62.5 |
| star_rating | 4.767 | 0.429 | 1 | 4.5 | 5 | 5 | 5 |

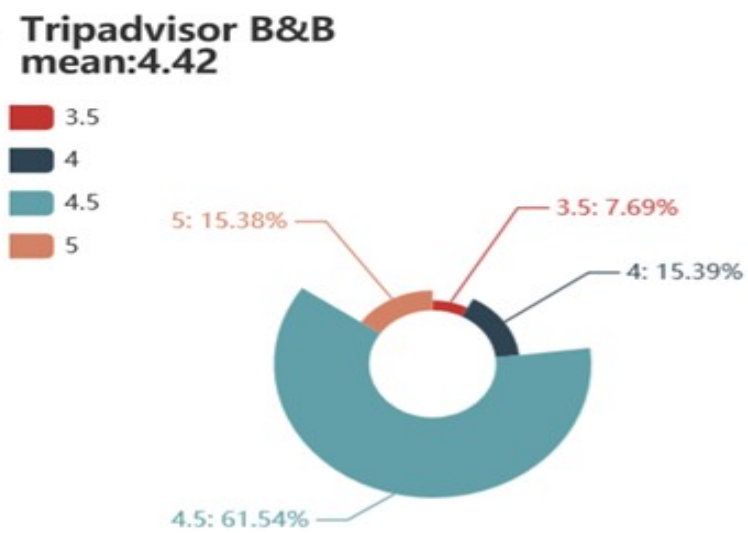
# 目录

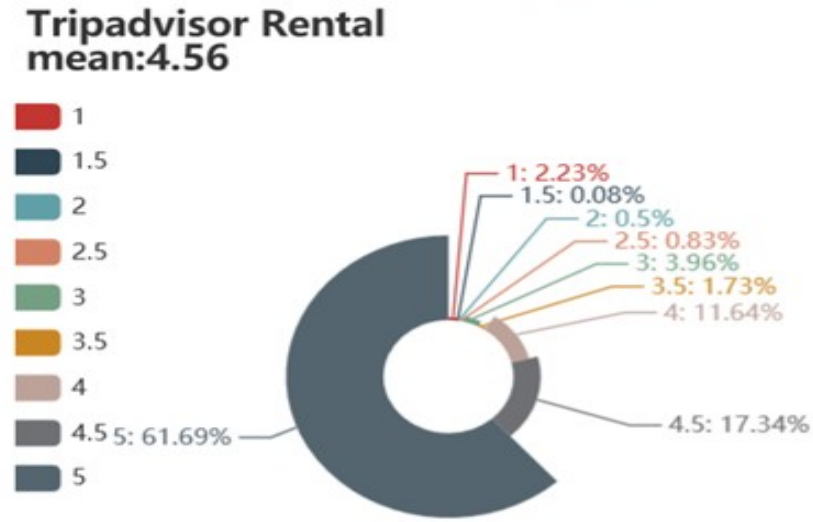
# TripAdvisor 与 Airbnb 房源的“星级评分”的总体比较

- 用 Pyecharts 库中的 Pie 函数做“radius 型玫瑰图”:



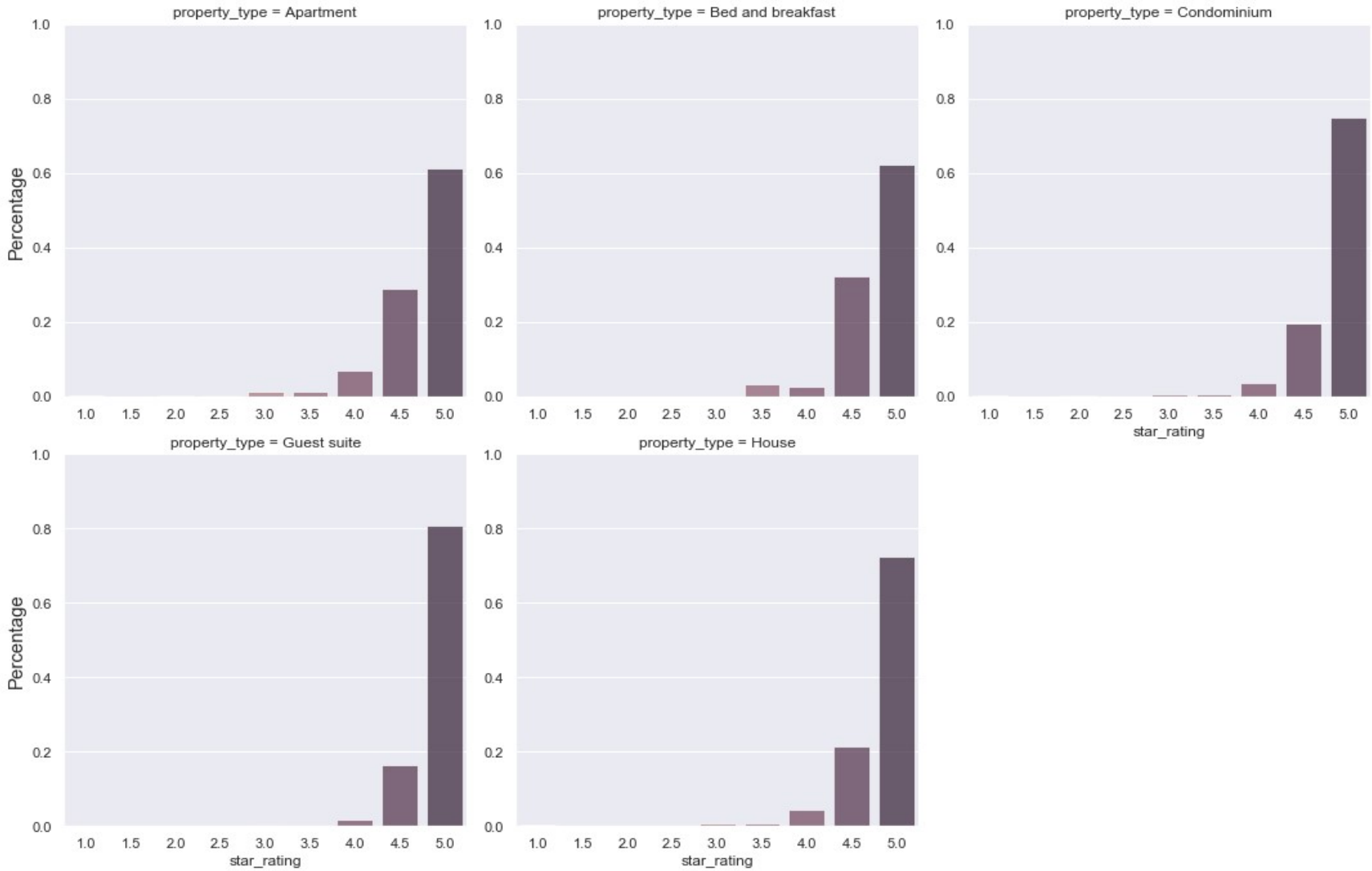



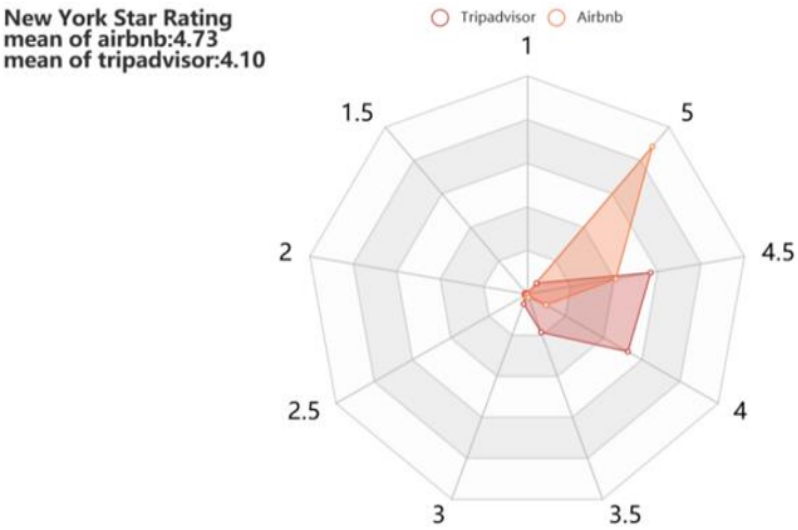
# 不同房源类型对评分的影响

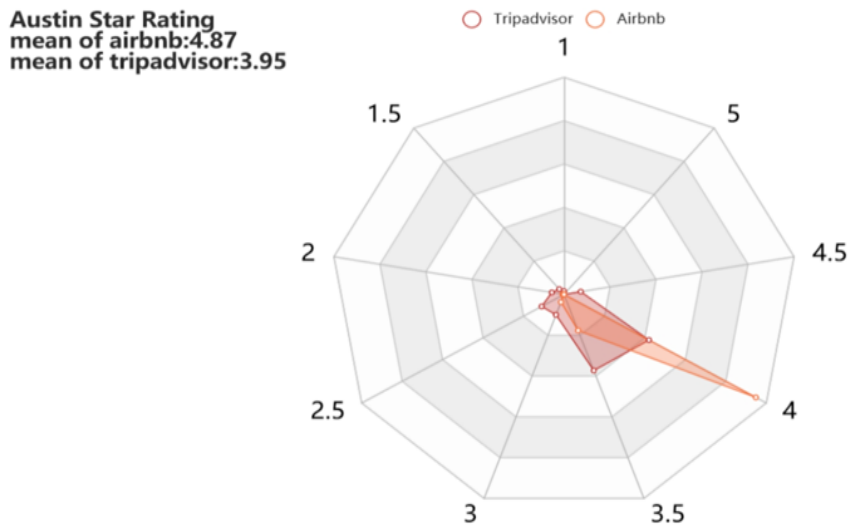
- 用 Seaborn 中的 Catplot 函数绘图：

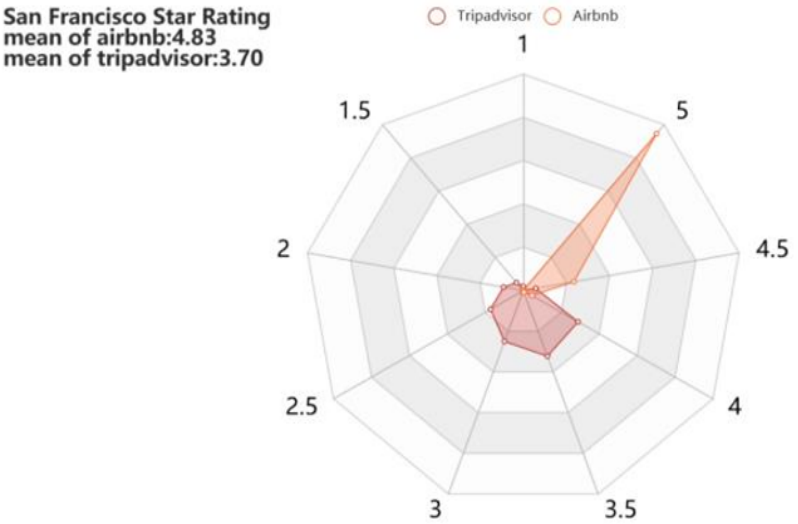


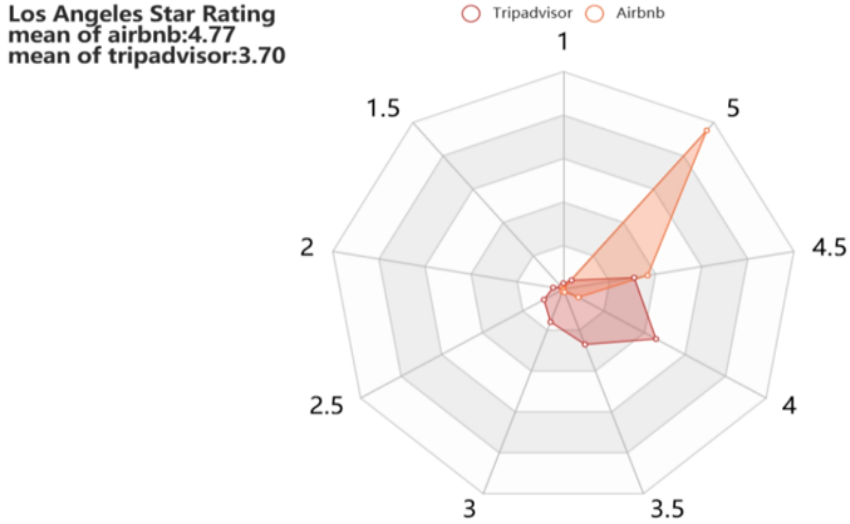
# 不同地理位置对评分的影响

- 用 Seaborn 中的 Catplot 函数绘图：



# Tripadvisor 和 Airbnb 上相同房源的匹配

- **根据经纬度匹配初步匹配地理位置相近的房源**

首先采用经纬度 <=500m 的匹配模式。经验证，Airbnb 的经纬度精确度为小数点后 12 位左右，TripAdvisor 的经纬度精确度为小数点后 5-6 位。若小数点后前 5-6 位相同，Airbnb 和 TripAdvisor 的距离最多相差 1.4168m，可以忽略不计。同时考虑到运算时间问题，匹配时使用了每个城市的分表而没有使用总表。

使用 Geopy 工具包中的 distance 可以通过经纬度来计算距离，再筛选出两房源位置小于 500 米的做为一组匹配数据。

# Tripadvisor 和 Airbnb 上相同房源的匹配

- 根据字符串匹配在地理位置相近的房源中精确匹配相同房源

在房源地理位置相近的基础上，通过对房源的描述关键词的相似度进行计算，从而确定是否为统一房源。

利用 Jaccard 算法来计算两相近房源关键词字符串的 Jaccard 系数。Jaccard 值的定义为 A 与 B 交集的大小与并集大小的比值：

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Jaccard 值越大说明相似度越高，当相似度超过一定程度基本可以确定是两者是同一房源。

# Tripadvisor 和 Airbnb 上相同房源的匹配

- 测试相似度阈值

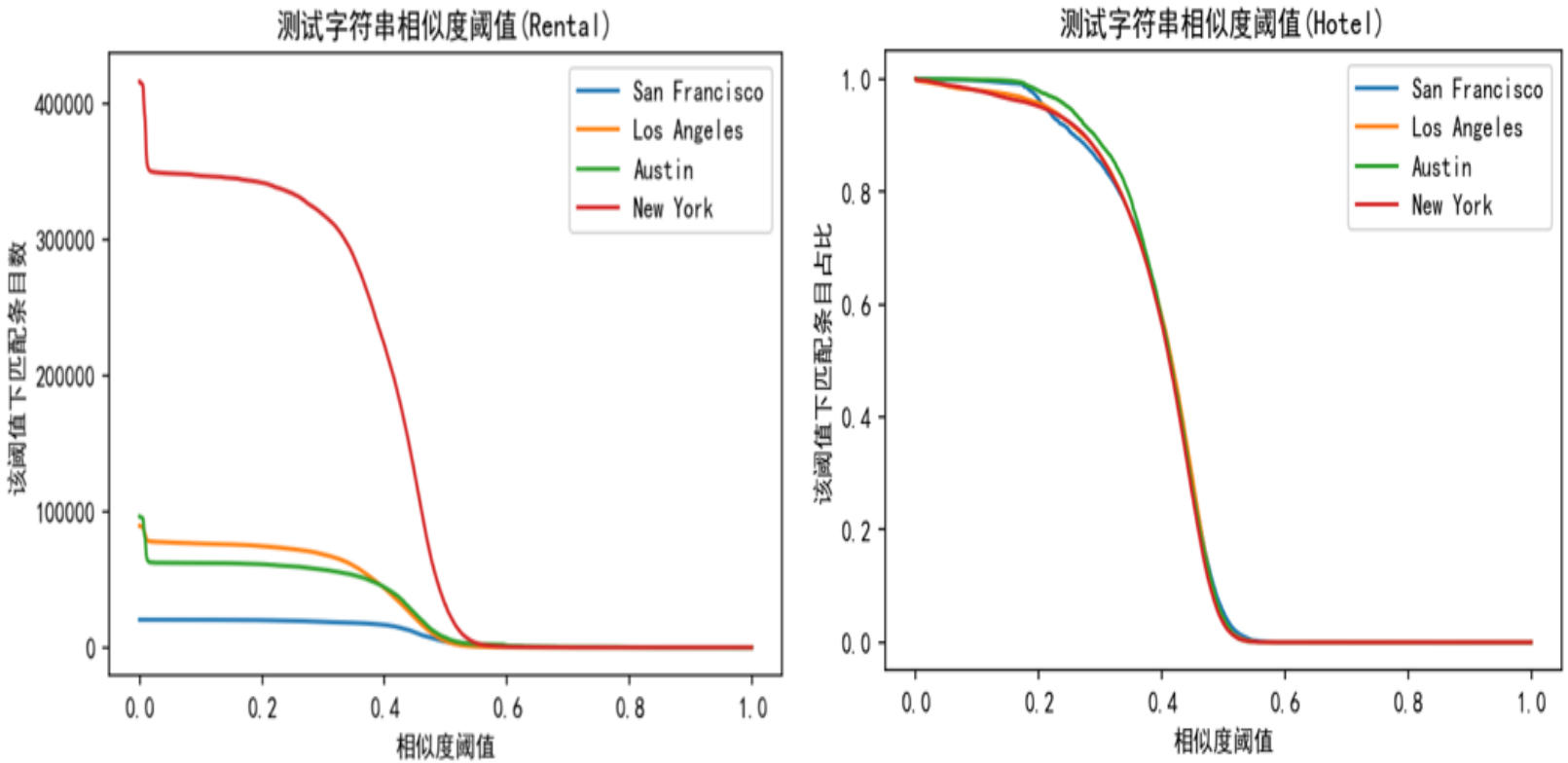
相似度的阈值需要通过测试来确定，且阈值的确定对数据结果的影响极大：设置大了，容忍度提高，造成误判；设置小了，造成漏判。

在确定阈值时，分别计算各个城市两组数据的中位数和平均值等统计数据，并进行比较，通过比较两组数据统计信息的一致性来判断两组数据集合中的数据是否一致。

# Tripadvisor 和 Airbnb 上相同房源的匹配

- 测试相似度阈值

经过仔细筛查过后，选取了 0.7 作为阈值，接着人工验证了 50 个左右的匹配数据，确认取相似度阈值为 0.7 符合要求且较为准确。



# Tripadvisor 和 Airbnb 上相同房源的匹配

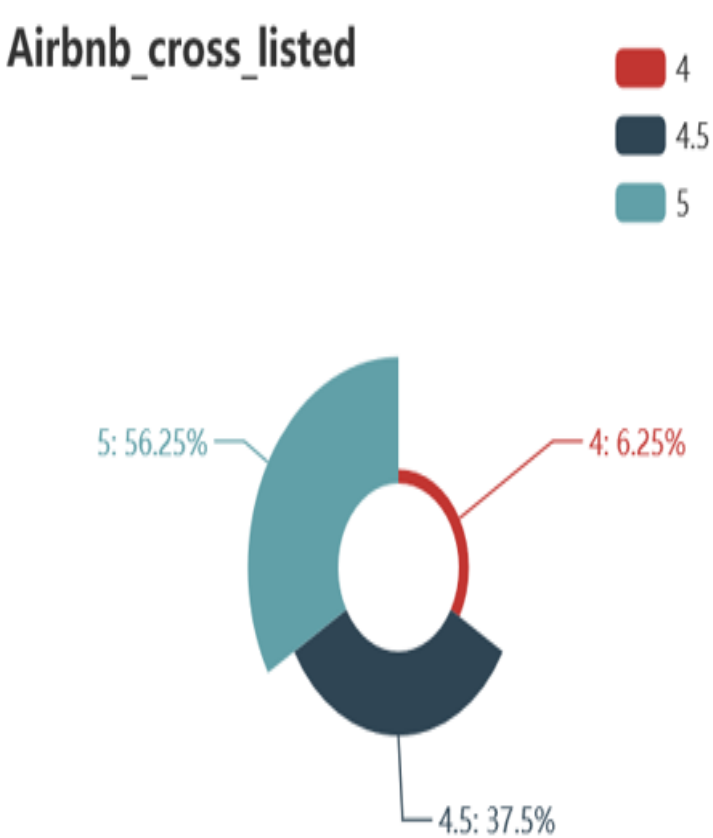
- 筛选过程及结果总结

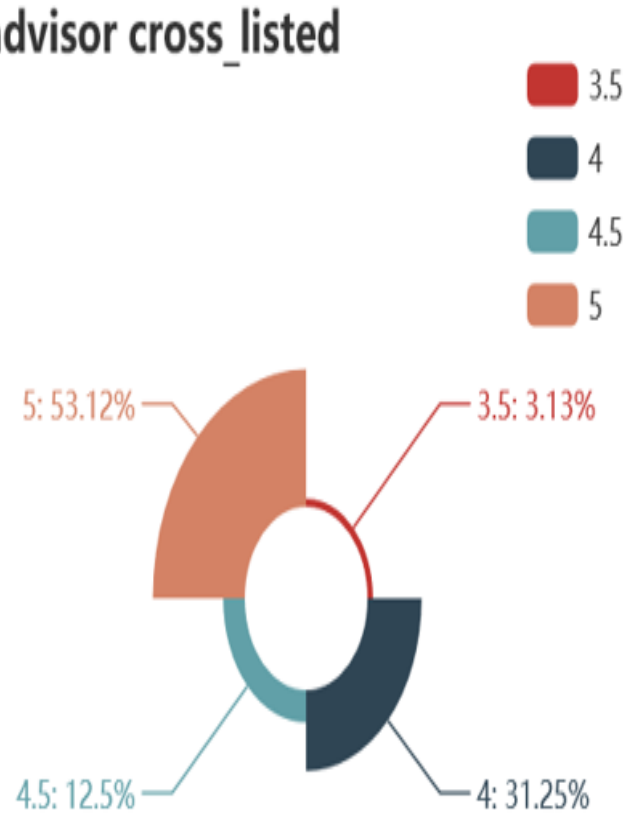
表 6: 每个筛选阶段剩余房源数

| | 坐标匹配 (Hotel) | 描述匹配 (Hotel) | 坐标匹配 (Rental) | 描述匹配 (Rental) | 去除没有评论的房源后 (总剩余房源数) |
|---|---|---|---|---|---|
| Austin | 9880 | 0 | 96763 | 447 | 15 |
| Los Angeles | 38318 | 0 | 90370 | 7 | 5 |
| New York | 133122 | 0 | 421154 | 105 | 11 |
| San Francisco | 44077 | 0 | 20376 | 111 | 1 |

# Tripadvisor 和 Airbnb 上相同房源的评分比较

- 用 Pyecharts 库中的 Pie 函数来做“area 型玫瑰图”：

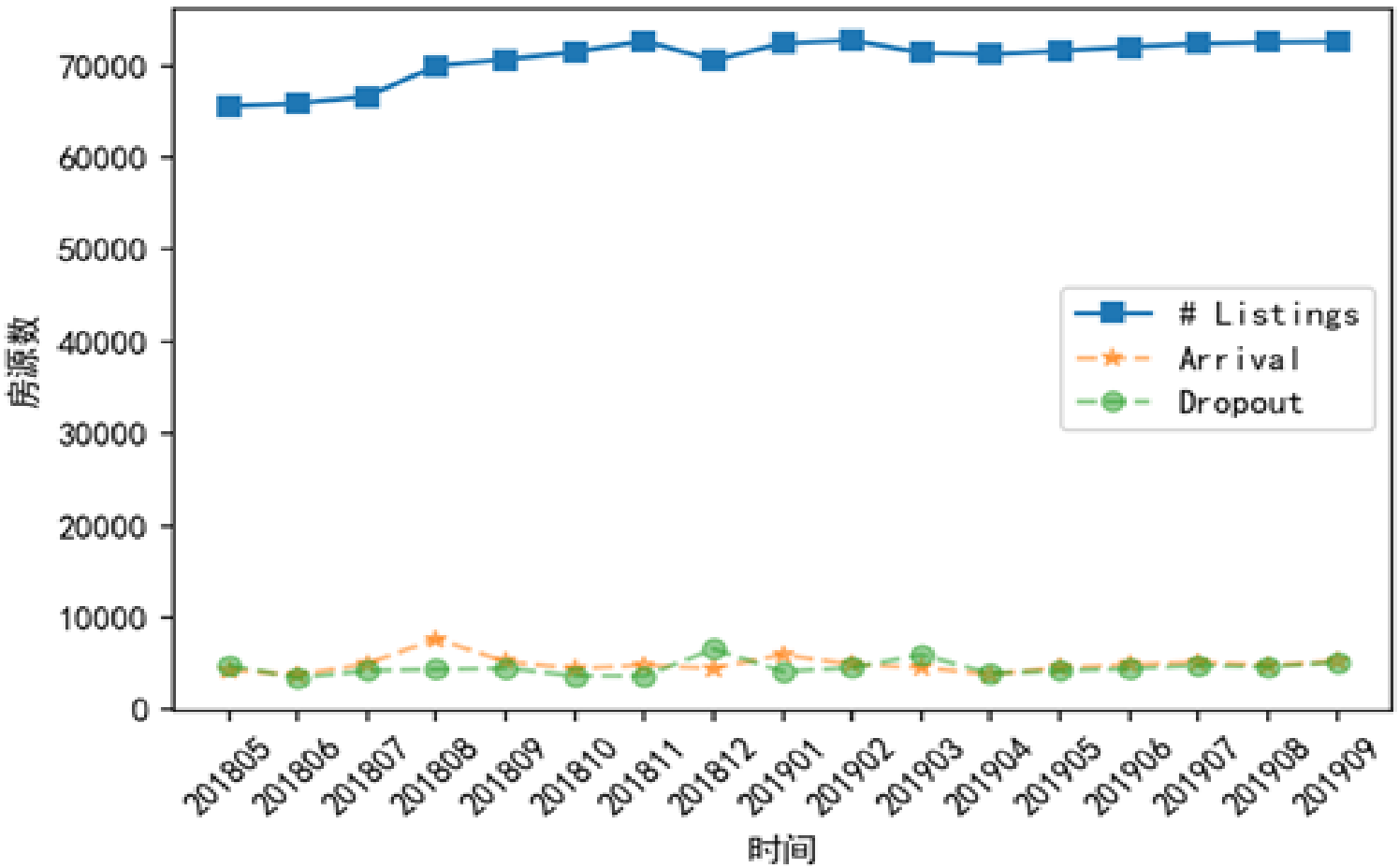


# 目录

# Airbnb 上房源进入退出的基本概况

- 用 Pandas 中的 Merge 函数，合并同一房源在不同时间点的信息，形成 2018 年 4 月-2019 年 9 月所有房源的面板数据；
- 在此表的基础上，我们作出每个月 Airbnb 平台上房源的数量以及进入、退出的情况：



# Airbnb“星级评分”的 Markov 模型的动态模拟

- Markov 链（马尔可夫链）
  - 为状态空间中经过从一个状态到另一个状态的转换的随机过程。该过程要求具备“无记忆”的性质：下一状态的概率分布只能由当前状态决定，在时间序列中它前面的事件均与之无关。这种特定类型的“无记忆性”称作马尔可夫性质。
  - 在马尔可夫链的每一步，系统根据概率分布，可以从一个状态变到另一个状态，也可以保持当前状态。状态的改变叫做转移，与不同的状态改变相关的概率叫做转移概率。
- 本作业的“状态”指的是一个房源的“星级评分”，所有 3.5 星或以下的评分都被聚合为一个联合类别，所以共有 4 类状态，分别是：$\leq 3.5$，4，4.5,5.

# Airbnb“星级评分”的 Markov 链的动态模拟

- 利用 Pandas、Operator 和 Numpy 三个工具包，计算 2018 年 4 月-2019 年 9 月 4 个城市所有房源的状态转移概率：

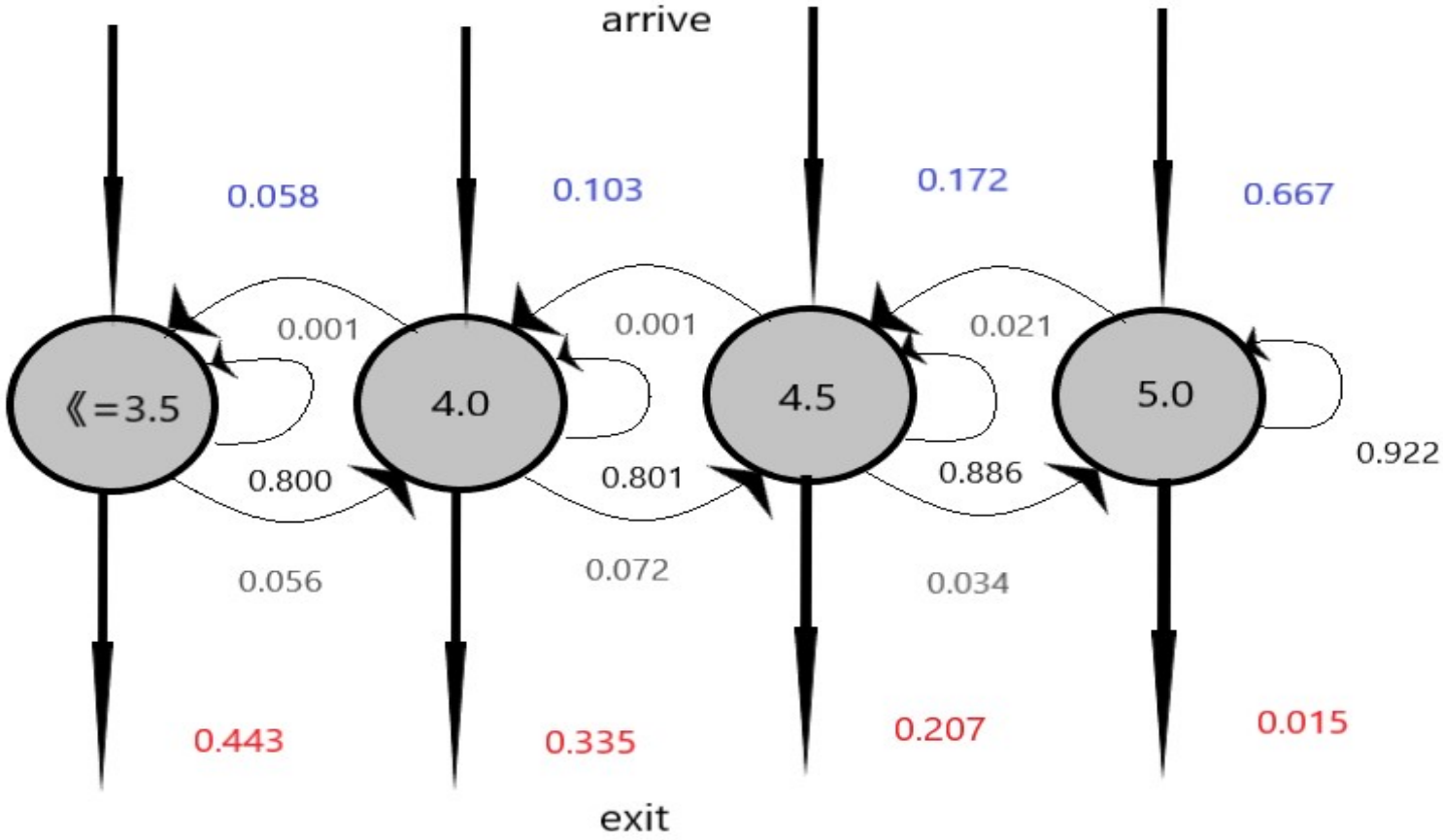


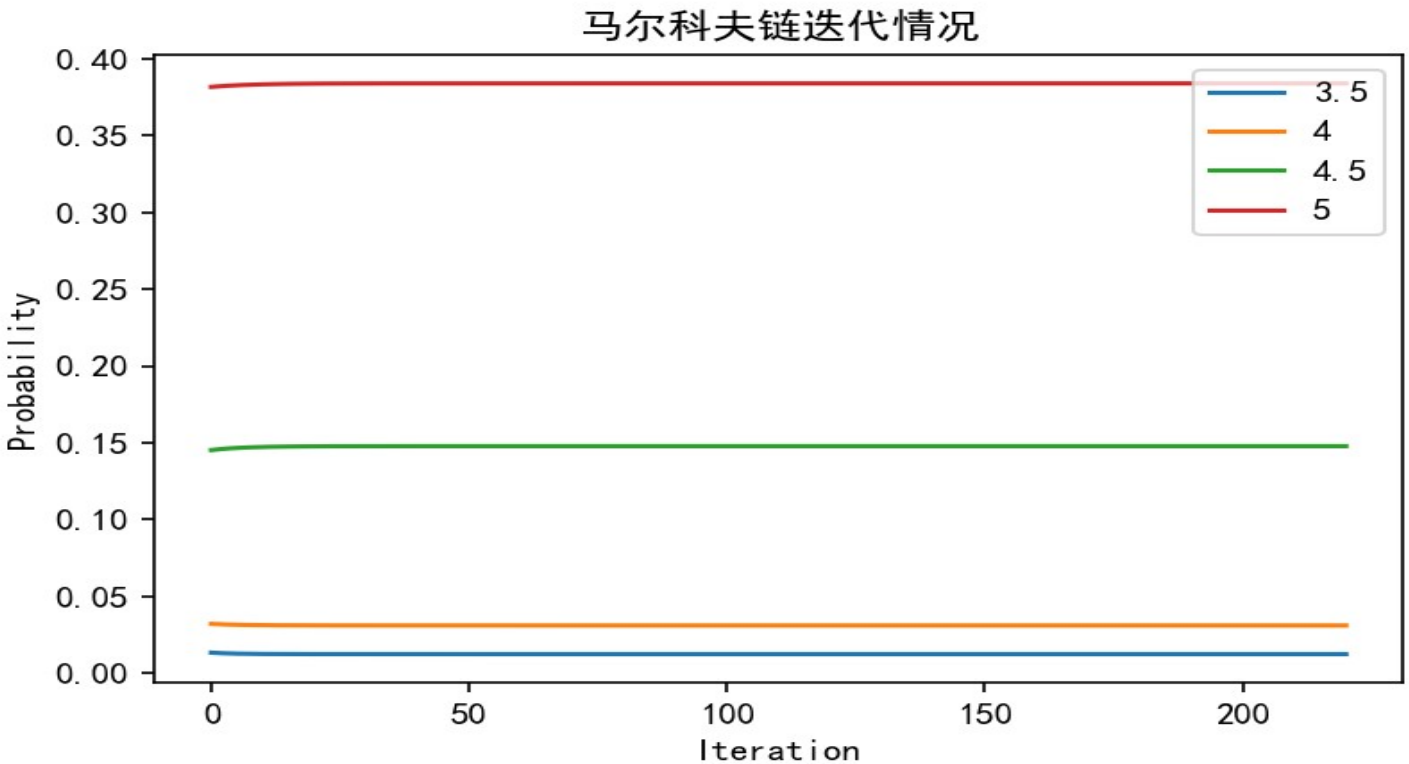

图 3: Airbnb“星级评分”状态转移网络

# Airbnb “星级评分”：信号传递理论与幸存者偏差

- 信号传递理论
  - 该理论假设买卖双方存在利益冲突和信息不对称，而卖方（或者更一般地说，信号提供者）可以通过提供信号来减少信息不对称和相关的不确定性。
  - 例如可以将第三方的评级作为信号。这将有助于相关房源获得消费者的信任，从而获得经济上的成功。
- 幸存者偏差
  - 是一种认知偏差。其逻辑谬误表现为只能看到经过某种筛选而产生的结果，而没有意识到筛选的过程，因此忽略了被筛选掉的关键信息。
  - 考虑到大多数市场并不代表静态结构，而是处于一个稳定的变化过程，因此，当我们考虑市场参与者和参与者类型时，该规律具有普遍性。
- 在“星级评分”中的应用
  - 房东通过“星级评分”像买者提供一个信号：具有较高声誉的 Airbnb 房源能吸引更多的需求，从而更易获得经济上的成功；而那些低声誉的房源，更容易退出。这种“幸存过程”导致大比例的高评级房源。

# Airbnb“星级评分”的平稳分布

- Markov 链的平稳分布
  - 平稳分布提供了使马氏链和初始状态无关的一个办法，并刻画了马氏链在长时间下的极限行为和平均行为。它只与初始状态和转移概率矩阵有关。
- Airbnb“星级评分”的平稳分布
  - 我们以 2019 年 9 月 Airbnb 各星级评分的分布作为初始分布:≤ 3.5:0.014，4.0:0.032，4.5:0.144，5.0:0.384 .
  - 计算得到平稳分布为：≤3.5:0.012，4.0:0.031，4.5:0.147，5.0:0.384.



# Airbnb 评价数与评分的动态关系

- 用 Seaborn 中的 Violinplot 函数绘制的“琴形图”

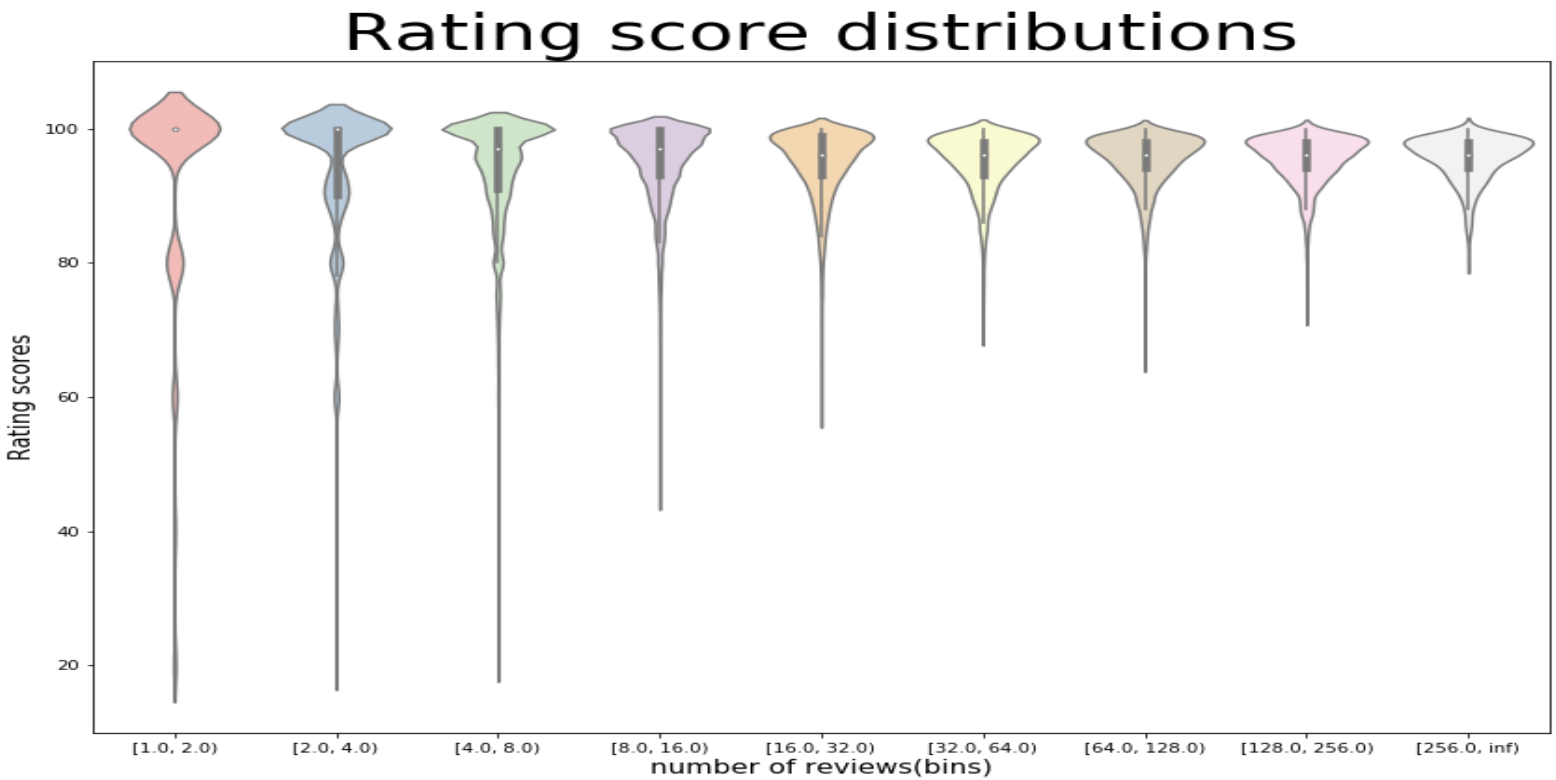



图 4: 不同评价数组中获得的评价等级的分布图

# Airbnb 评价数与评分的动态关系

- 用 Matplotlib 中的 scatter 函数作出散点图

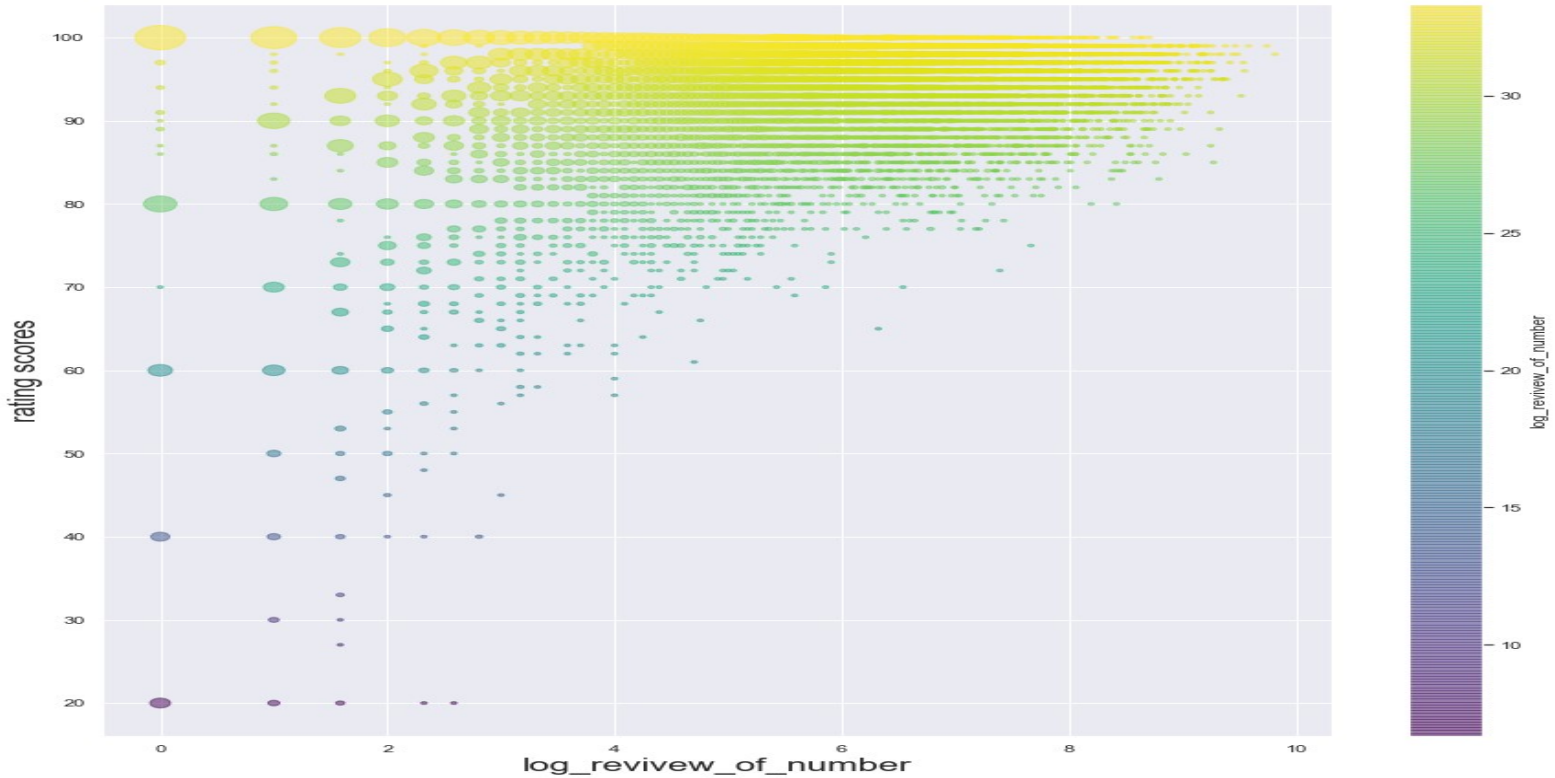



图 5: 不同评价数组中获得的评价等级的分布图

# Airbnb 评价数与评分的动态关系解读

- 大数定律
  - 大数定律表明，当重复做同一实验的次数足够多时 (即)，这些结果的平均值收敛于期望值。
  - 应用到 Airbnb 房源的评级上，这表明，评论数量不断增加的房源获得最高评级的可能性会降低。
- 均值回归
  - 均值回归指的是：如果一个变量在第一次测量时是极端的，那么它在第二次测量时就会趋向于接近平均值
  - 较高的信誉评级可能会引起客户对一个特别好的体验的期望，这可能会有效地增加失望的机会，反过来，促进较低的客户评级。同样，如果预订低评级的房源可能会带来意想不到的积极体验。