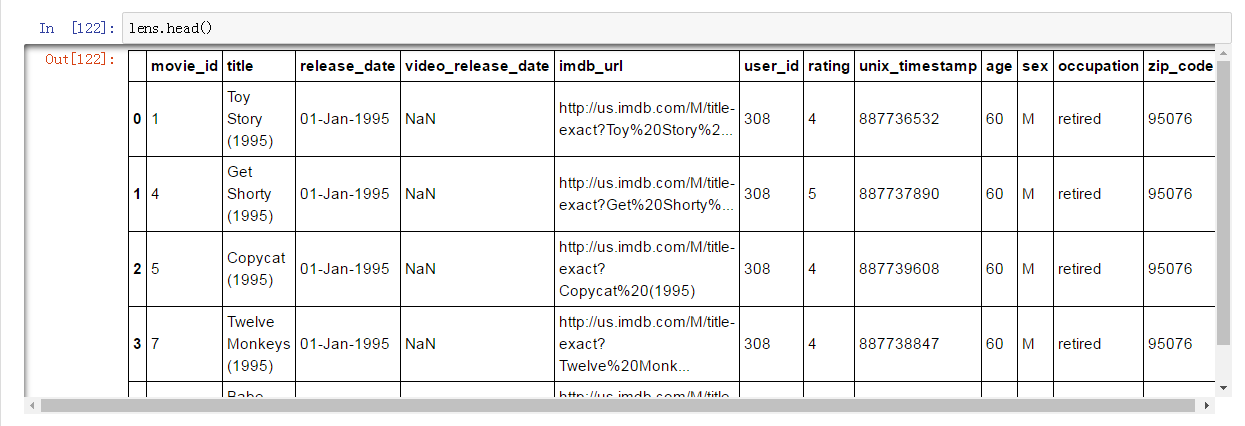
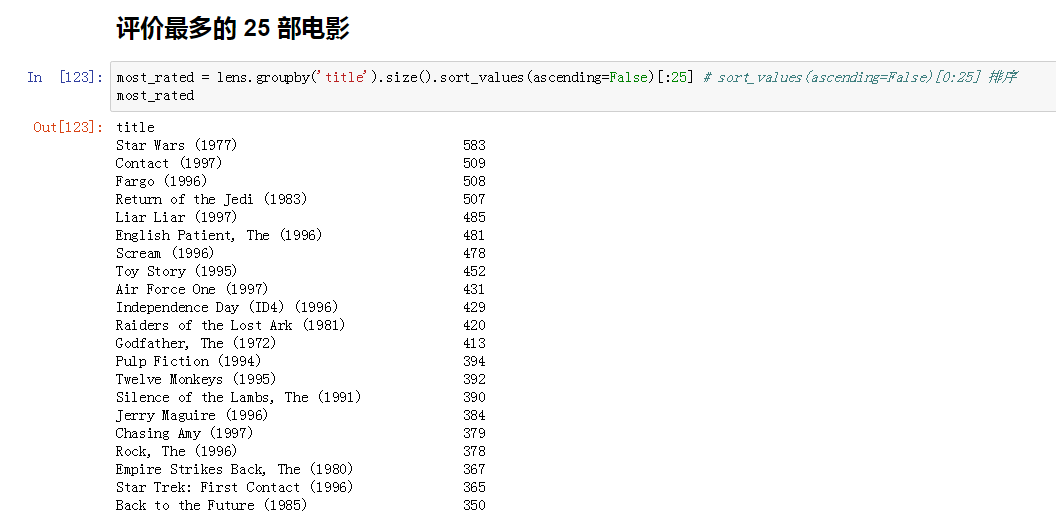
1. 数据读取及导出CSV格式



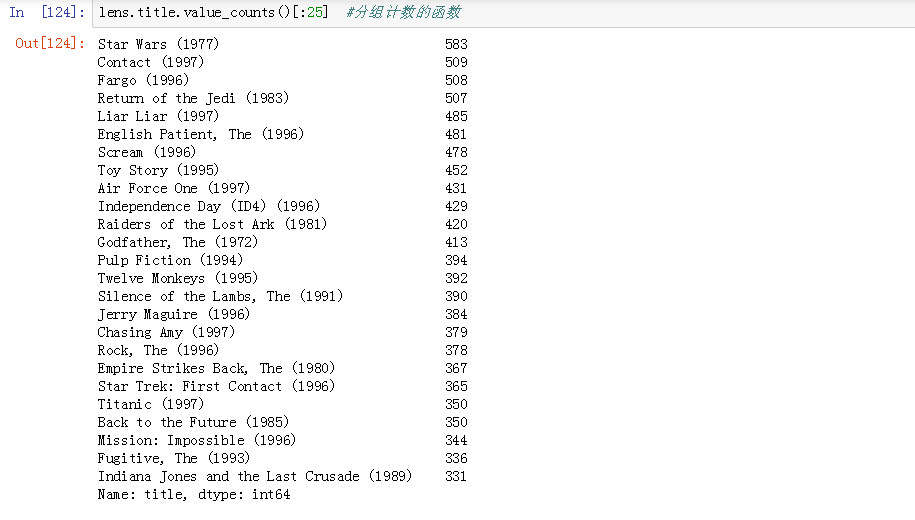


1. 评论最多的25部片子

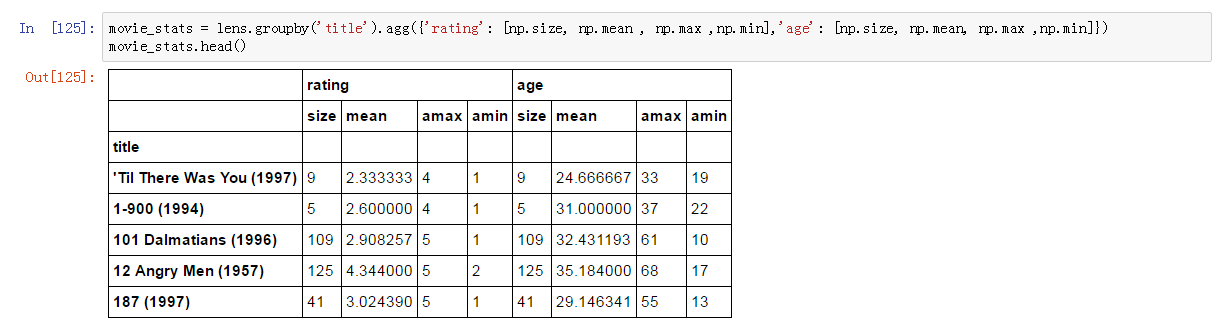


上述代码的含义是先将 DataFrame 按电影标题分组，接下来利用 size 方法计算每组样本的个数，最后按降序方式输出前 25 条观测值。

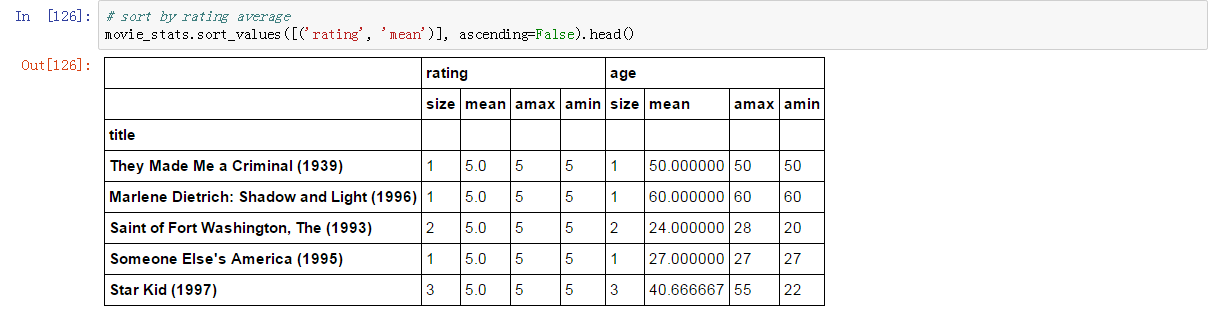
此外，在 Pandas 中有一个非常好用的替代函数—— value\_counts：



1. 评价最高的电影

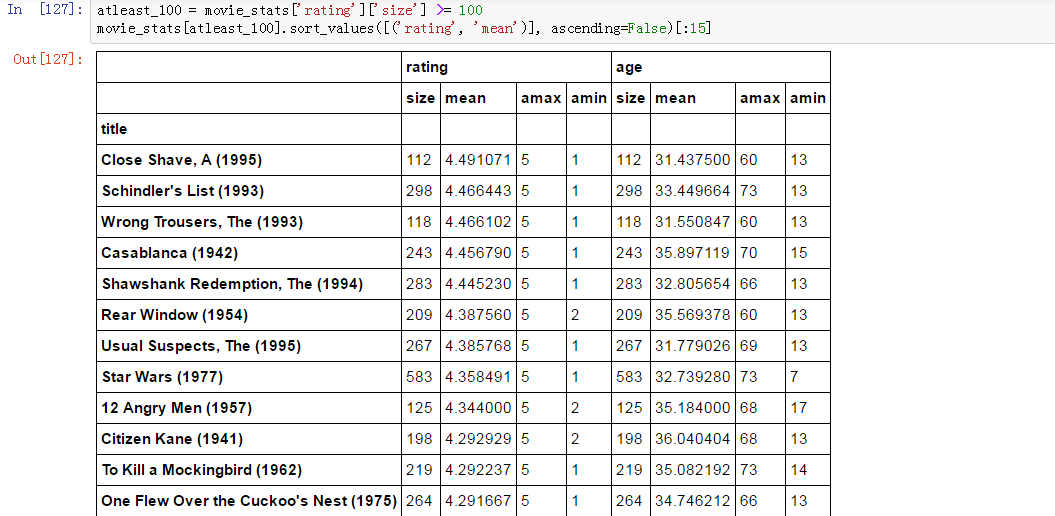


我们可以利用 agg 方法来进行分组汇总计算，其参数包括键值和汇总方法。接下来我们对汇总结果进行排序即可得到评价最高的电影：



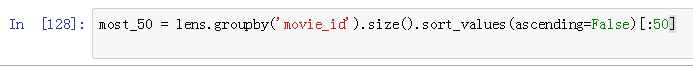
由于 movie\_stats 是一个 DataFrame，因此我们可以利用 sort 方法来排序——Series 对象则使用order 方法。此外，由于该数据集包含多层索引，所以我们需要传递一个元组数据来指定排序变量。

上表列出来的电影中评价数量都非常少，以致于我们无法从中得到一些有价值的信息。因此我们考虑对数据集进行筛选处理，只分析评价数量大于 100 的电影：



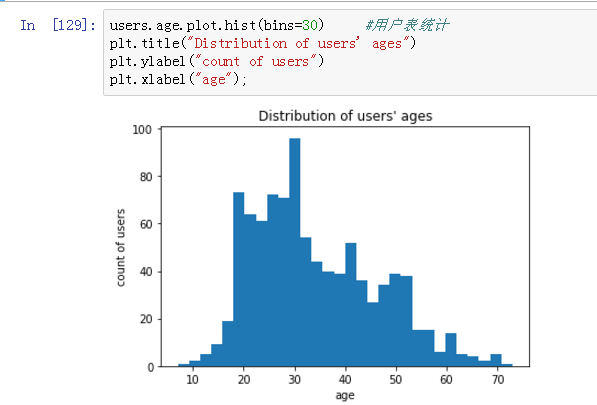
筛选部分数据

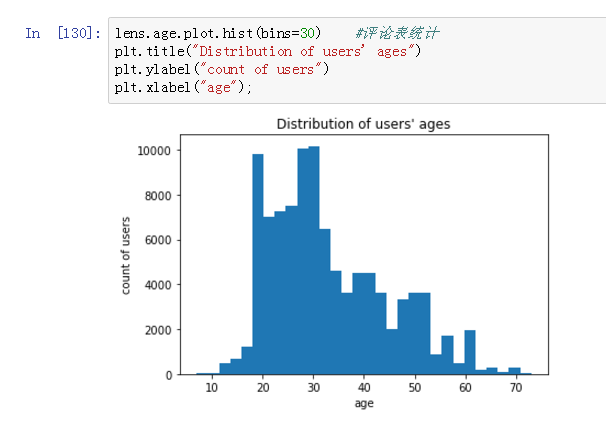
为了便于进一步分析，我们从数据集中筛选出评价数最高的 50 部电影：



**不同年龄段观众之间争议最大的电影**[**¶**](http://localhost:8888/notebooks/6-movie.ipynb#不同年龄段观众之间争议最大的电影)

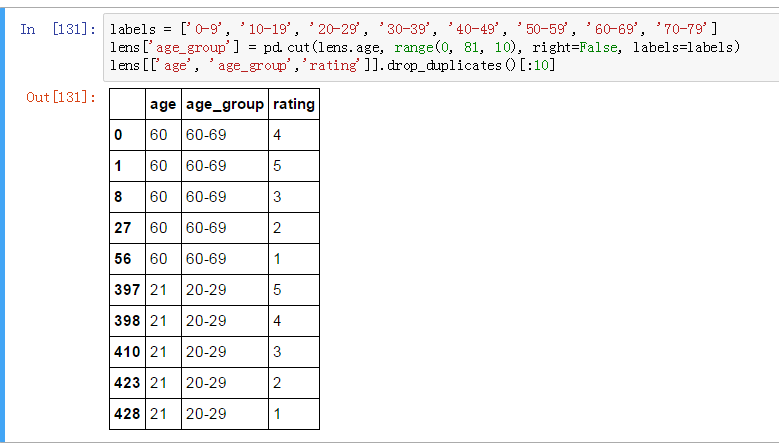
首先，我们来看下数据集中用户的年龄分布情况：





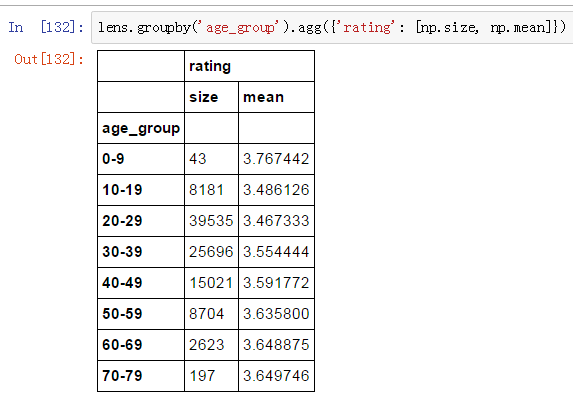
**对用户进行分箱处理**

我认为直接对比不同年龄用户的行为无法得到有价值的信息，所以我们应该根据用户的年龄情况利用pandas.cut 将所有用户进行分箱处理。

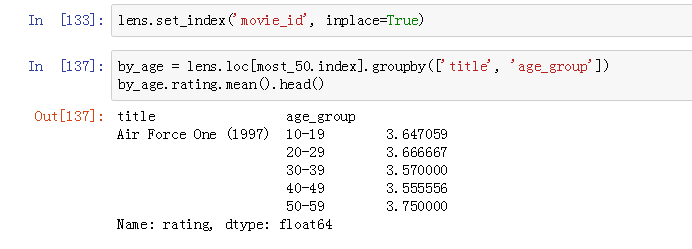


上述代码中，我们首先创建分组标签名，然后根据年龄变量将用户分成八组（0-9, 10-19, 20-29,...）,其中参数 right=False 用于剔除掉区间上界数据，即30岁的用户对应的标签为 30-39。

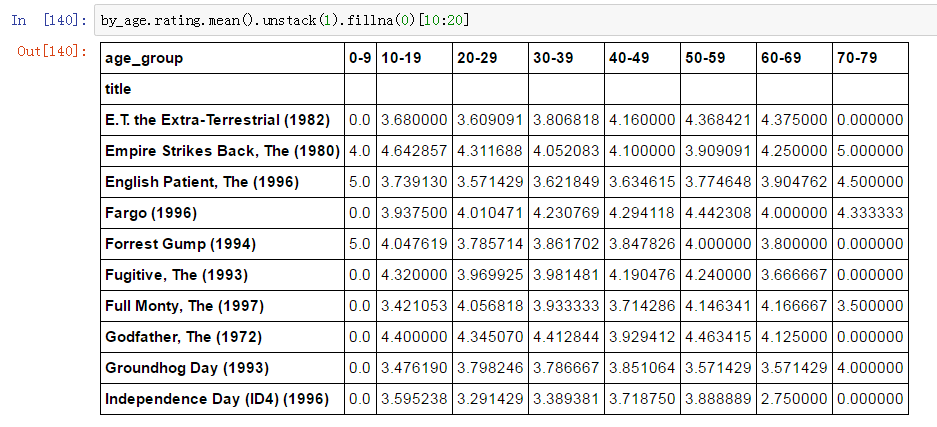
现在我们可以比较不同年龄组之间的评分情况：



从上表中我们可以看出，年轻用户比其他年龄段的用户更加挑剔。接下来让我们看下这 50 部热评电影中不同年龄组用户的评价情况。



需要注意的是，此处的电影标题和年龄组都是索引值，平均评分为 Series 对象。如果你觉得这个展示结果不直观的话，我们可以利用 unstack 方法将其转换成表格形式。



男士与女士分歧最大的电影

