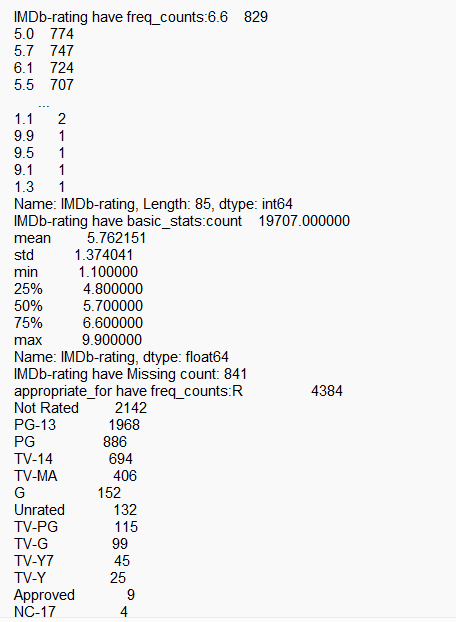
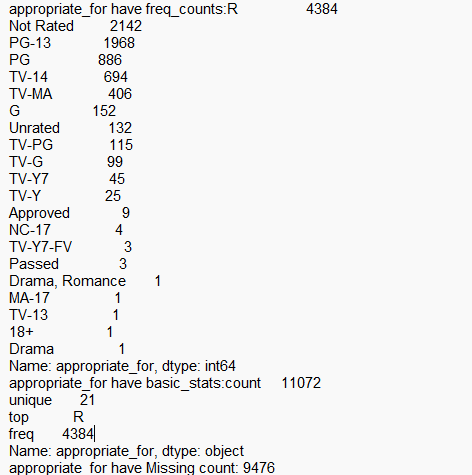
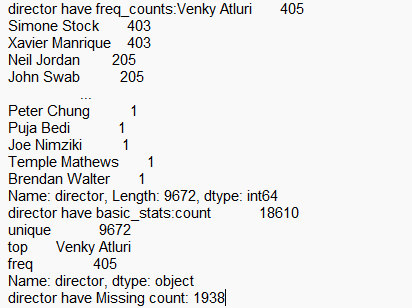
1. **缺失处理前缺失值的分析以及数据分析**

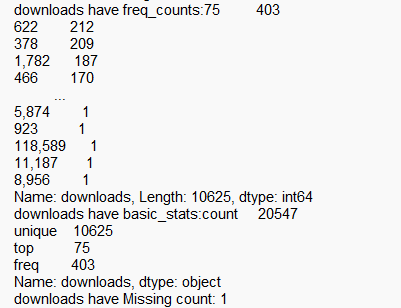
所有列的缺失值都为NaN，即所在值为空。

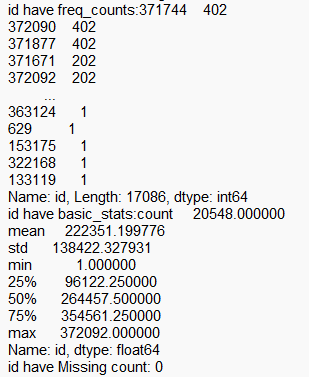
数据分析如以下各图所示，结果在图中可以看到，除了id列无缺失值其他列都存在不同数量的缺失值。其他数据分析指标不单独讨论。

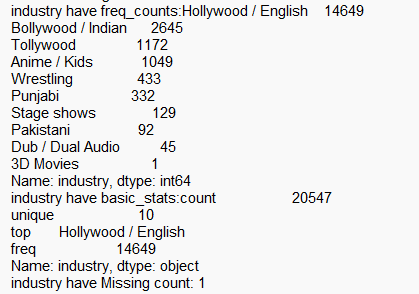


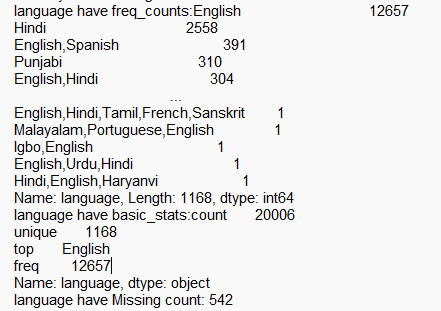


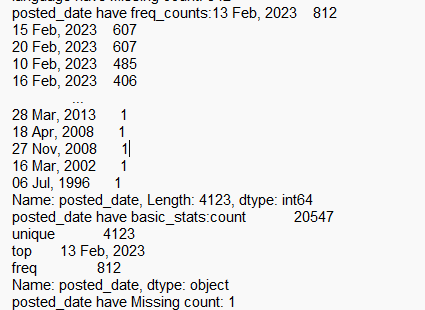


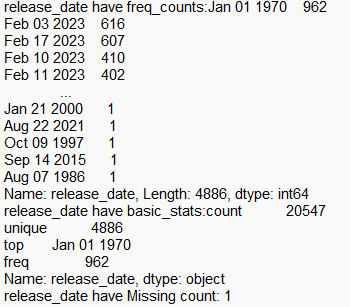


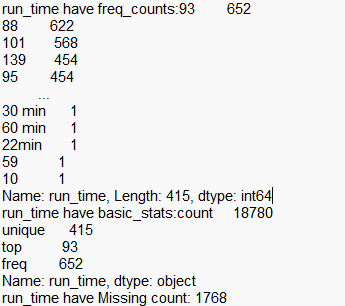


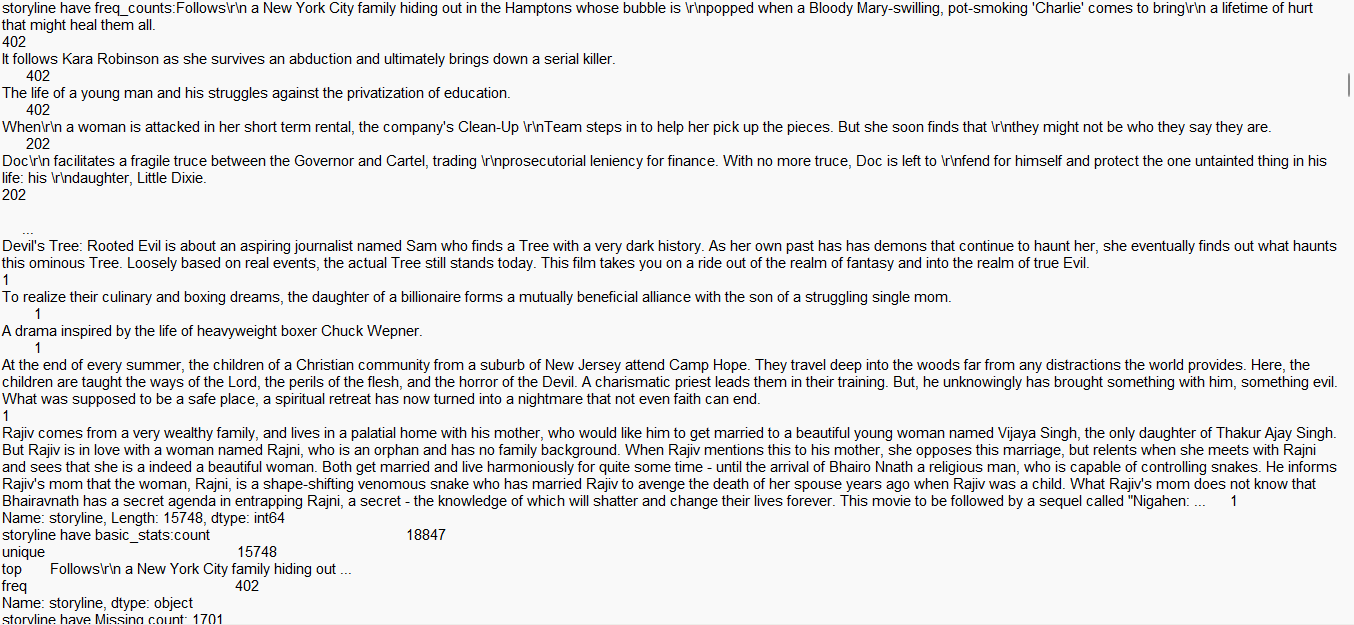


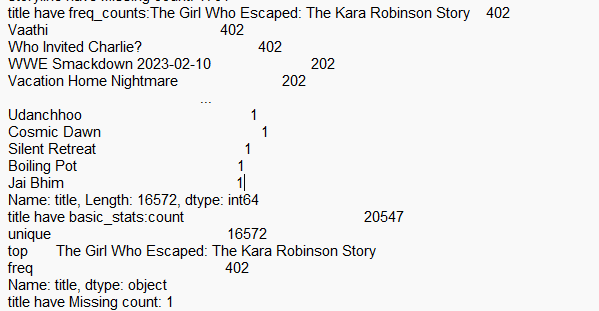




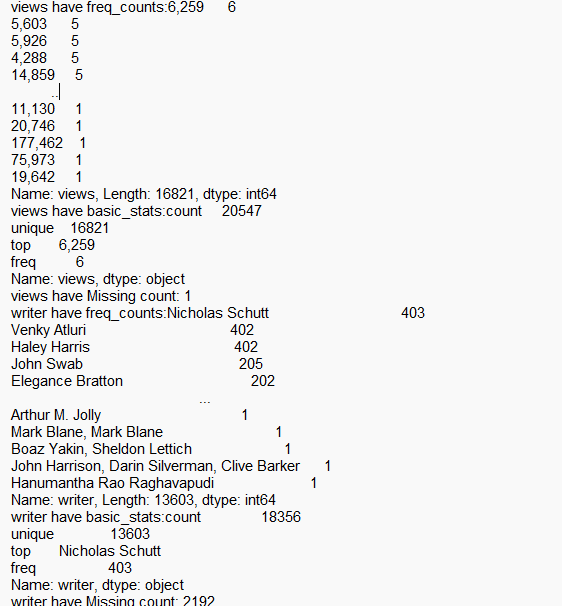






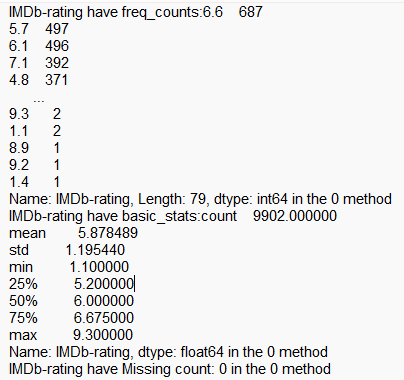


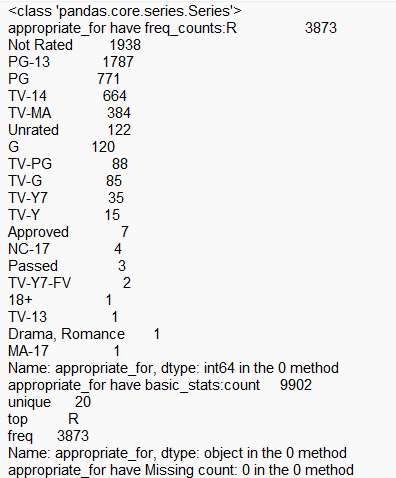
Views和writer

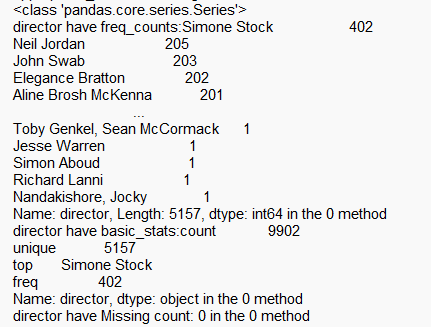


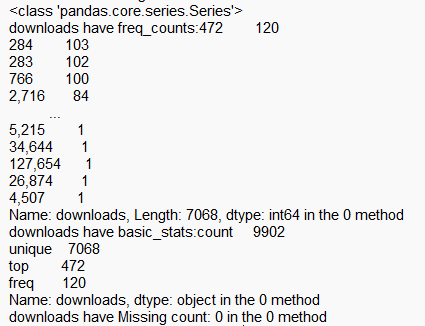
第1种缺失值处理方法：将缺失部分删除

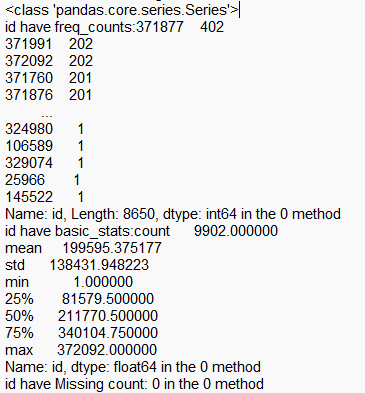
可以看到所有列的缺失值都降为0。但是因为把含有缺失值的行删除了，所以数据量大幅减少。本来有19000多行而现在只剩下9902行数据，所以该方法对数据量的减少有时候很严重。

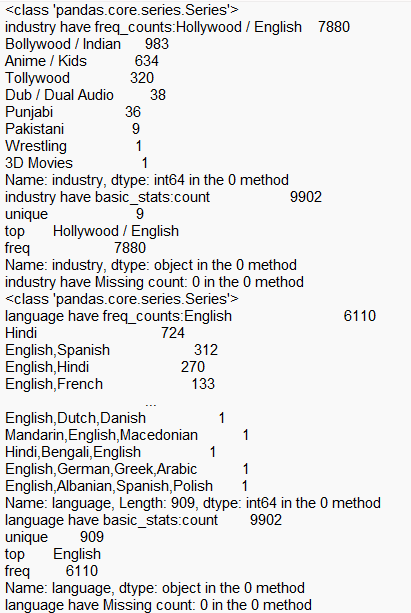


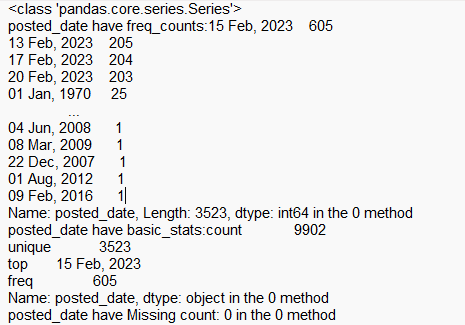


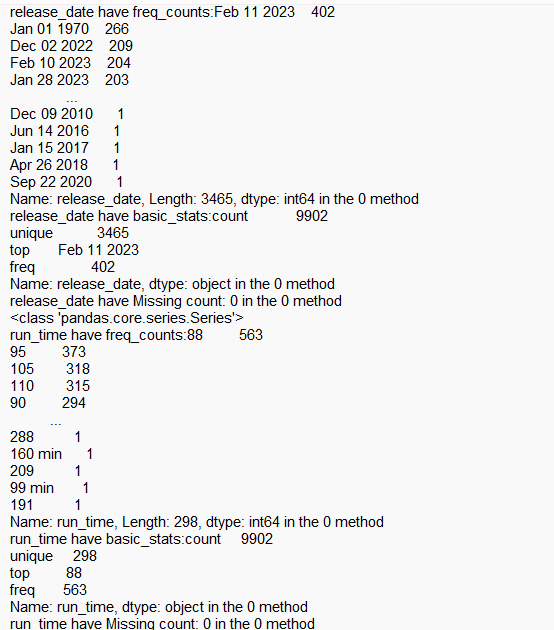


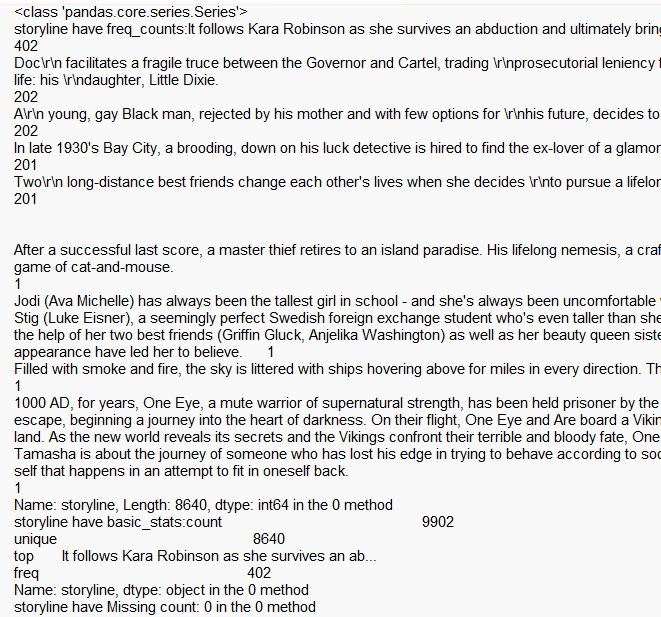


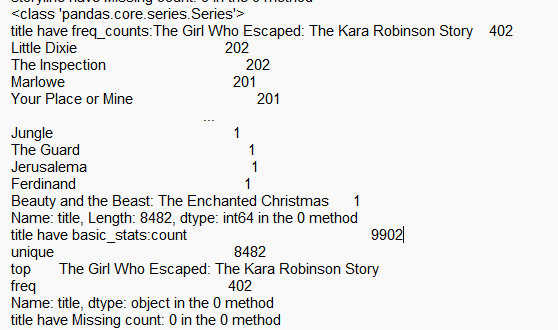


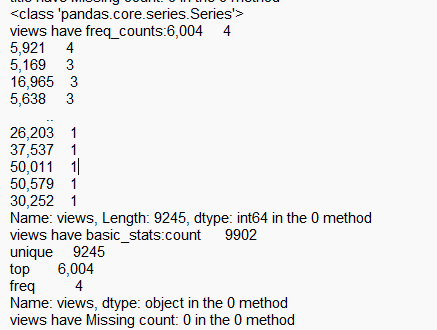


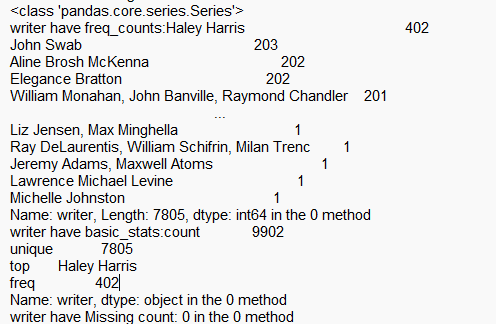




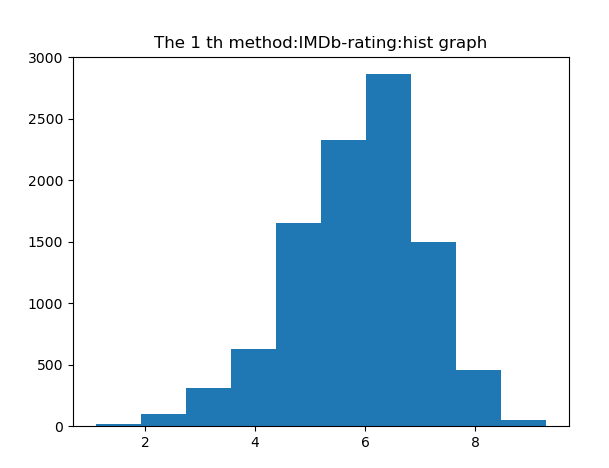


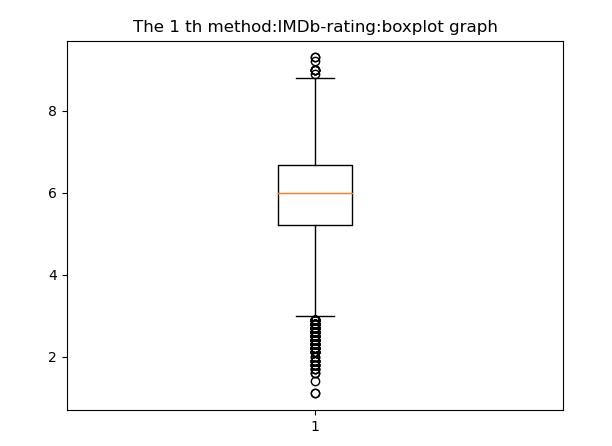


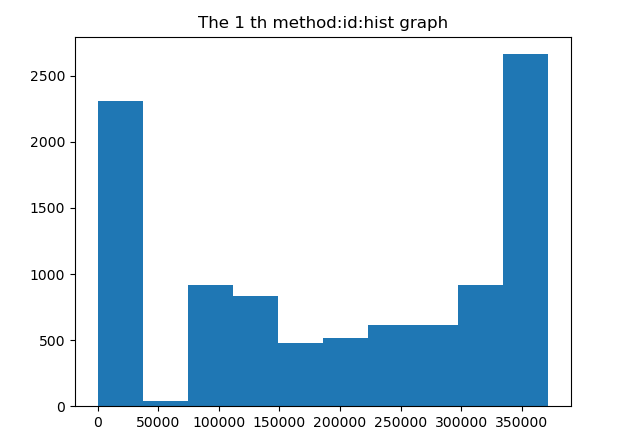


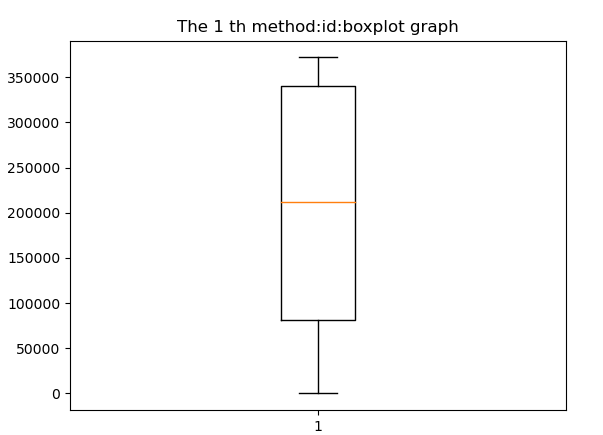


第一种方法的数据分布图，分别是直方图和盒图。

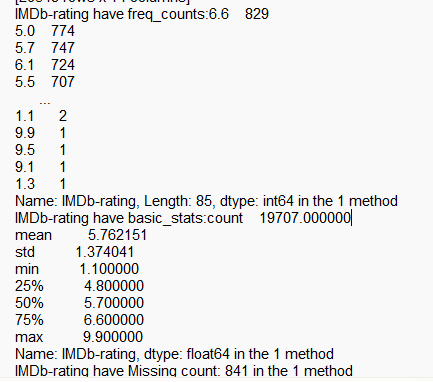


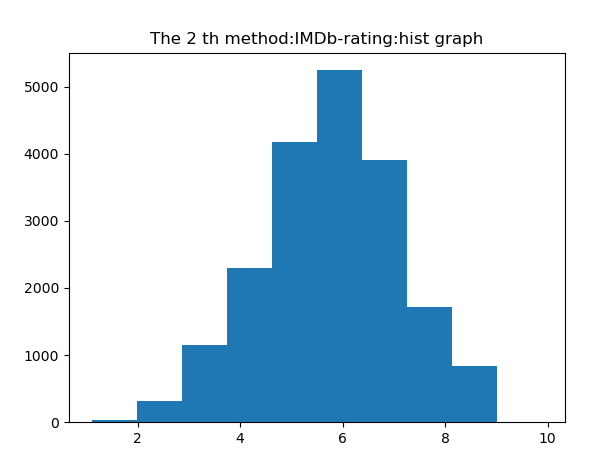


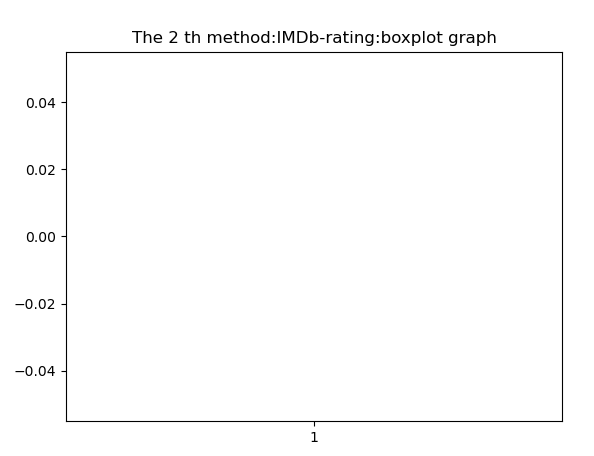


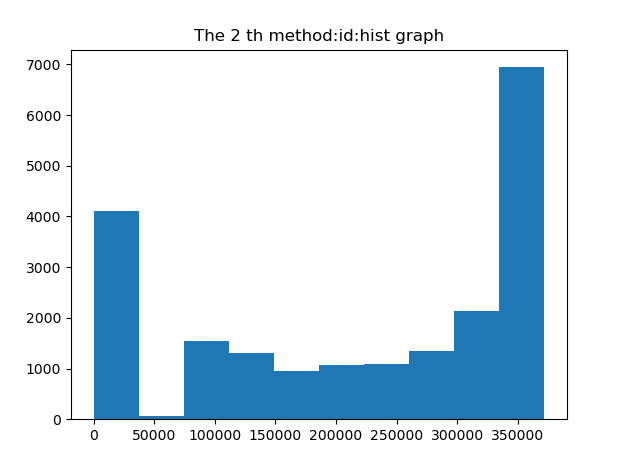


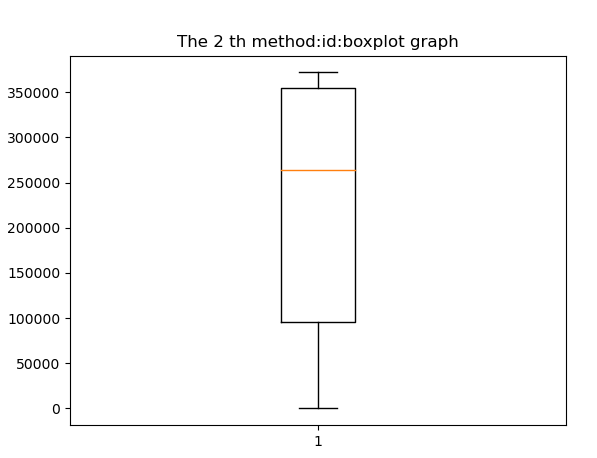
第2种缺失值处理方法：找到最高频率的值并用其填充，在大部分数据分布中，这种方法还算比较准确，但若最高频率值属于极值的范围则该方法效果很差。





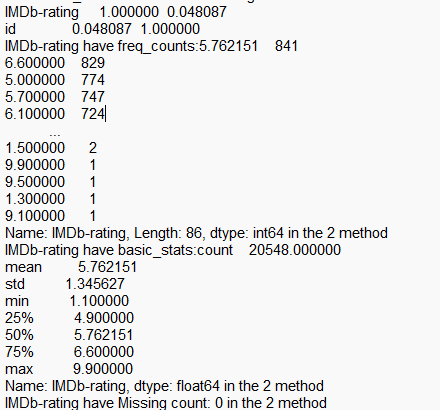


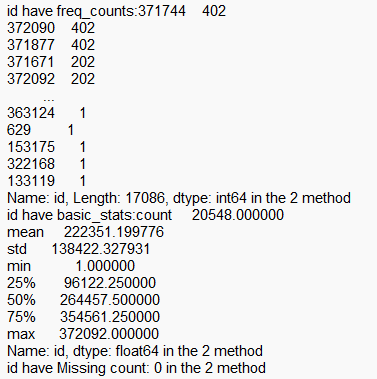


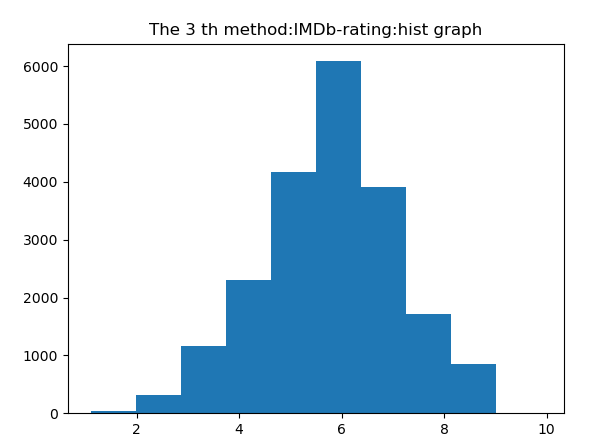


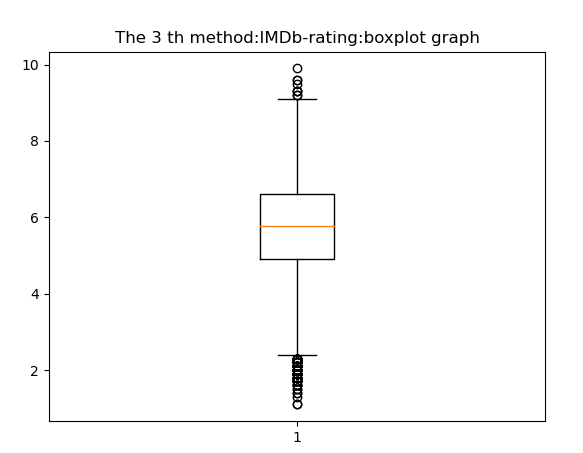
第3种方法：通过属性的相关关系来填补缺失值

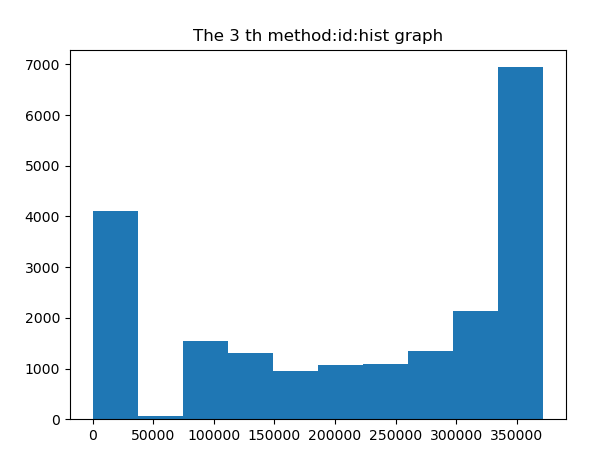
这种方法只能求值为整数或者浮点数的列，所以说在这个数据集中只有IMDb-rating和id适用。在这里与原数据集差异很大的指标是四分位数。可以看到这种方法的指标基本与原数据的各个指标类似，所以效果比较好。

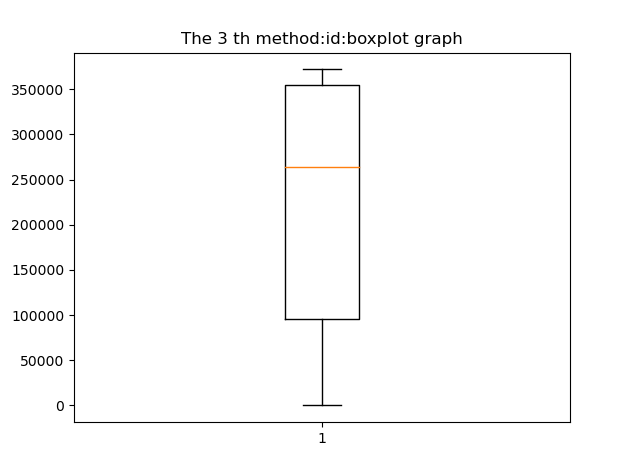






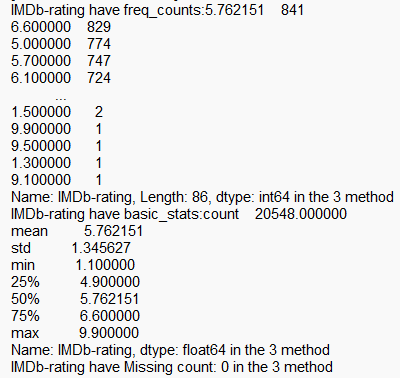


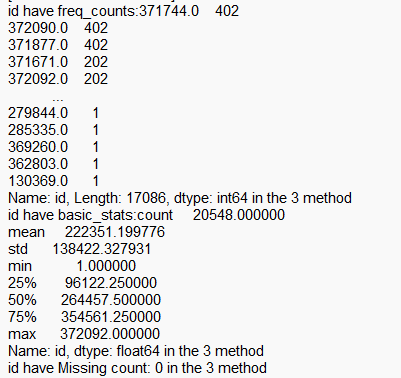


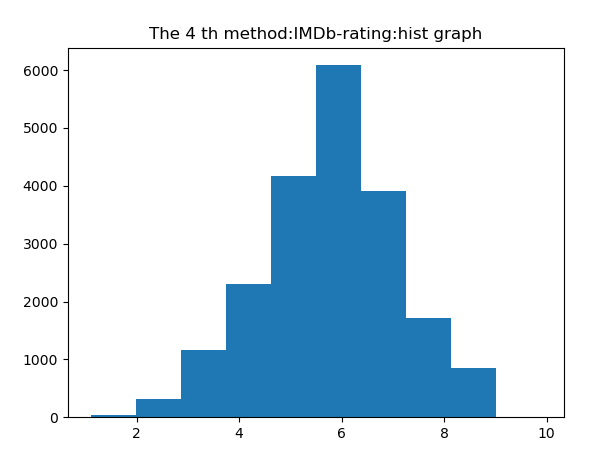


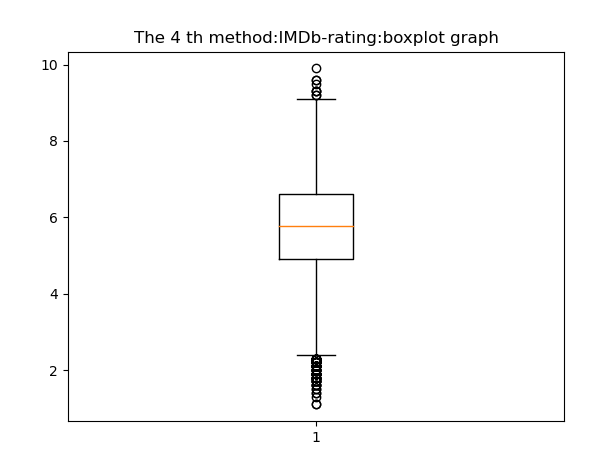
第四种方法(通过数据对象之间的相似性来填补缺失值)：

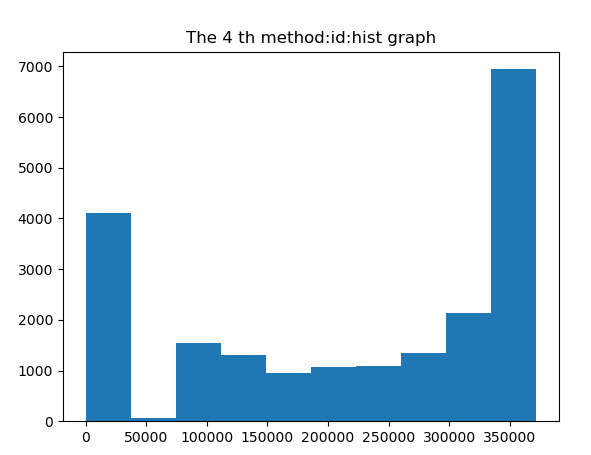
这种方法只能求值为整数或者浮点数的列，所以说在这个数据集中只有IMDb-rating和id适用。仔细观察发现它的各项指标和原数据集是最像的，比如标准差都是i.34多，均值也相差无几，所以这种基于KNN算法的缺失值插补方法：KNNImputer的深度学习方法效果比较好。

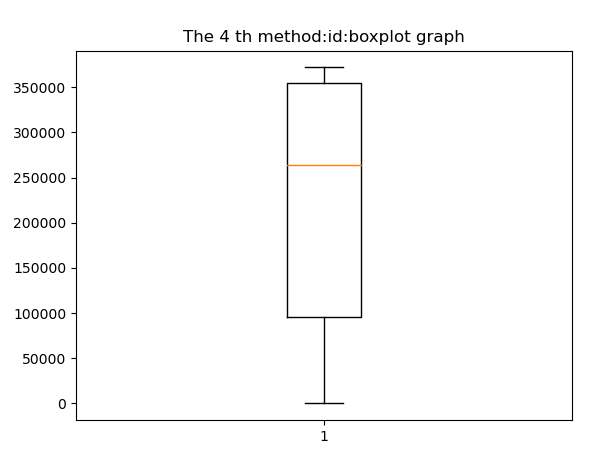












1. **数据集说明**

上述实验数据集采用movies\_dataset.csv。该代码可以经过微小的修改来处理github\_dataset.csv，这两个数据集的选取来自作业要求。由于github\_dataset.csv和实验数据集比较相似，这个数据集的实验略过。