电影票房预测数据分析报告

一、数据集

TMDB Box Office Prediction

https://www.kaggle.com/competitions/tmdb-box-office-prediction/data?select=train.csv

该网站未给出测试集的实际票房收入。为方便测试，现将训练集按 5：1 比例拆分为训练集和测试集：

（1）bop_train.csv

（2）bop_test.csv

其中，bop_train.csv 中含有 2500 条电影数据、 bop_test.csv 中含有 500 条电影数据。

接下来以 bop_train.csv 为训练集进行数据分析

二、数据集属性展示

	属性	含义	类型
1	id	标识号	int64
2	belongs_to_collection	属于某一类收藏	object
3	budget	预算（美元）	int64
4	genres	风格列表	object
5	homepage	电影首页的 URL	object
6	imdb_id	IMDB 标识号	object
7	original_language	原始语言	object
8	original_title	原始电影名称	object
9	overview	剧情摘要	object
10	popularity	受欢迎程度	float64
11	poster_path	海报 url 路径	object
12	production_companies	制作电影公司	object
13	production_countries	制作国家	object
14	release_date	首次上映日期	object
15	runtime	电影时长	float64
16	spoken_languages	输出语言	object
17	status	电影状态（已发行等）	object
18	tagline	电影的标语	object
19	title	电影名称	object
20	Keywords	与电影相关的关键字	object
21	cast	演员列表	object
22	crew	剧组	object
23	revenue	收入（美元）	int64

三、探索性数据分析

1、删除对分析无意义的属性（6 种）

imdb_id, original_title, overview, poster_path, status, title

（1） imdb_id 该属性是电影的 IMDB 标识号，对分析票房数据没有意义，故直接删除。

（2） original_title 该属性是电影的原始电影名称，对分析票房数据没有意义，故直接删除

（3） title 该属性是电影的名称，对分析票房数据没有意义，故直接删除。

（4） poster_path 该属性是电影的海报 url 路径。在训练集中少量缺失（1 个），对分析票房数据没有意义，故直接删除。

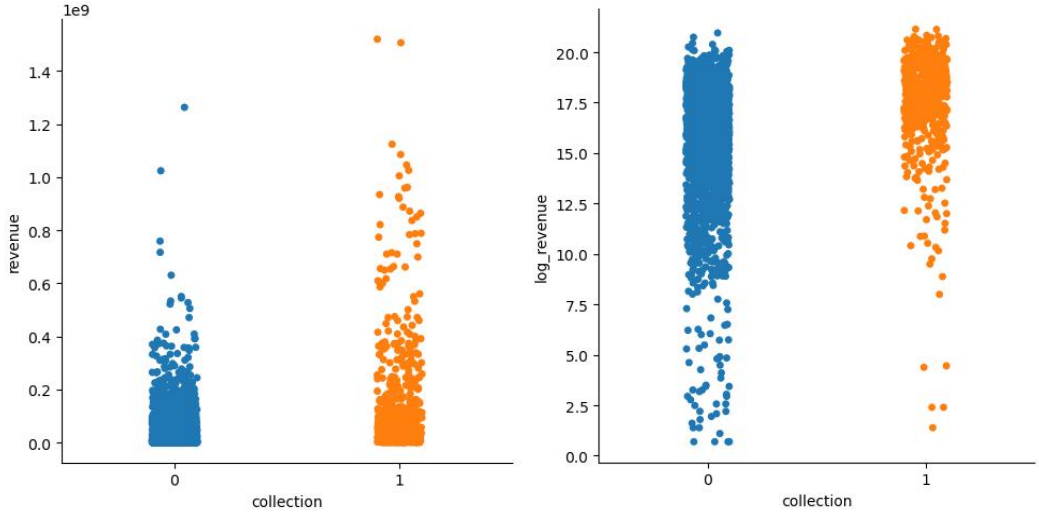
（4） overview 该属性是电影的剧情摘要。在训练集中少量缺失（7 个），对分析票房数据没有意义，故直接删除。

（5） status 该属性是电影的状态。除了极少数（4 个）是‘Rumored’外，均为‘Released’。对分析票房数据没有意义，故直接删除。

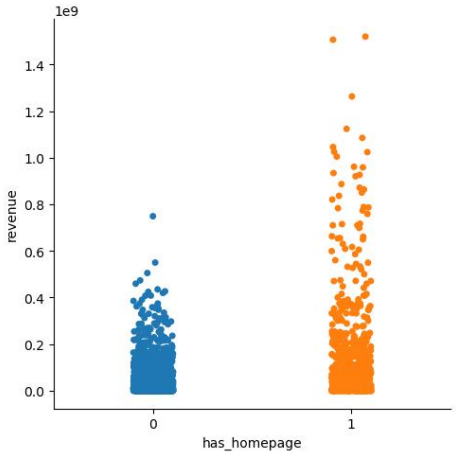
2、转化为“有/无”（1/0）属性后删除原属性（4 种）

belongs_to_collection, homepage, original_language, tagline

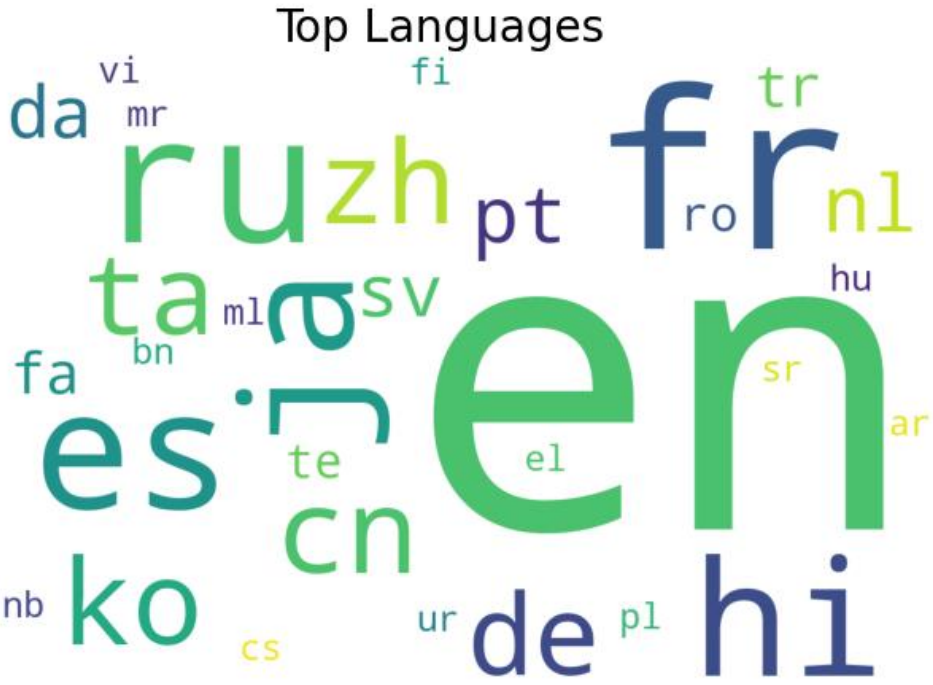
（1） belongs_to_collection 该属性标明电影是否属于某一类收藏。缺失值与非缺失值占比都很大，经过分析后发现：若电影属于某一类收藏，该电影票房一般较高。将其转化为 collection 属性（取值 1/0）表明其是否被收藏。

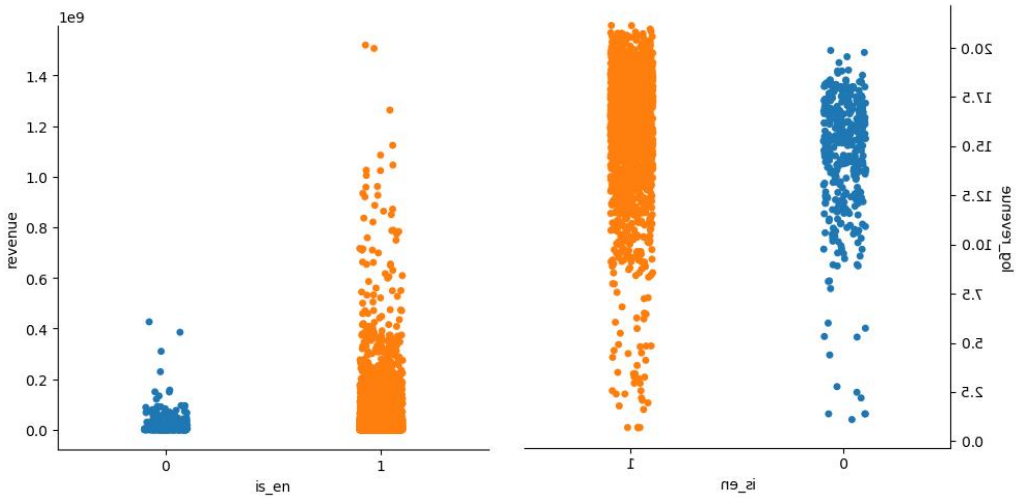


（2） homepage 该属性是电影首页的 URL。缺失值与非缺失值占比都很大，经过分析后发现：若电影首页的 URL 非缺失值时，该电影票房一般较高。将其转化为 has_homepage 属性（取值：1/0）表明该电影是否有首页 URL。

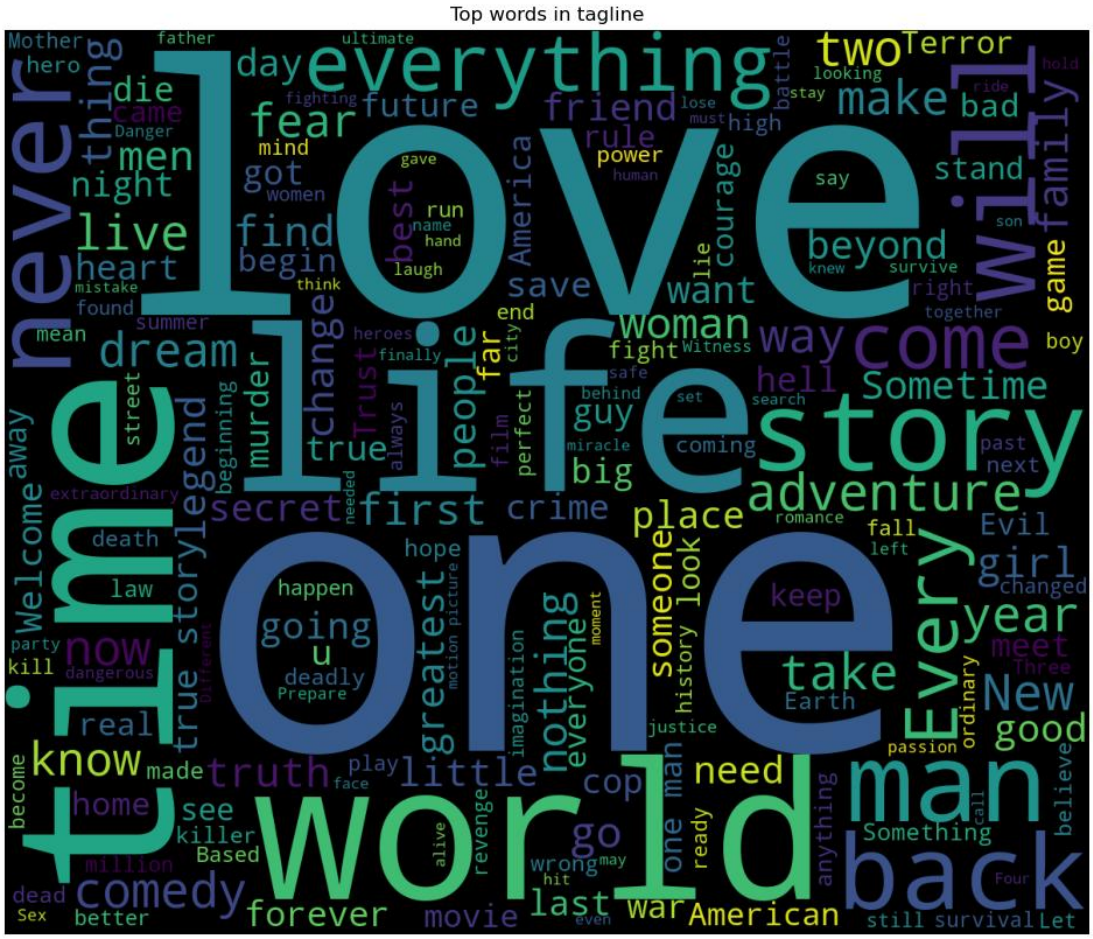


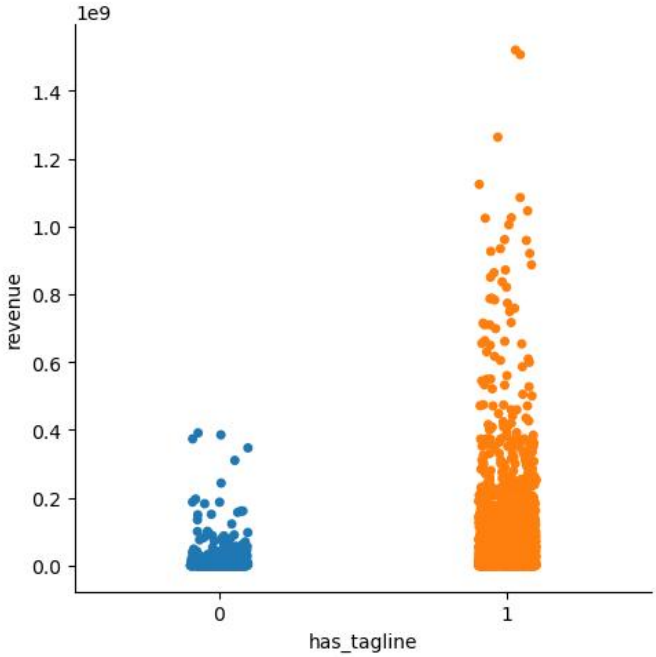
（3）original_language 该属性是电影的原始语言。主要分为‘en’和非‘en’两种，经过分析后发现：若电影的原始语言是‘en’时，该电影票房一般较高。将其转化为 is_en 属性（取值：1/0）表明该电影的原始语言是否是‘en’。





（4） tagline 该属性是电影的标语。下图是可视化效果。经过分析后发现：若电影有标语时，该电影票房一般较高。将其转化为 has_tagline 属性（取值：1/0）表明该电影是否有标语。



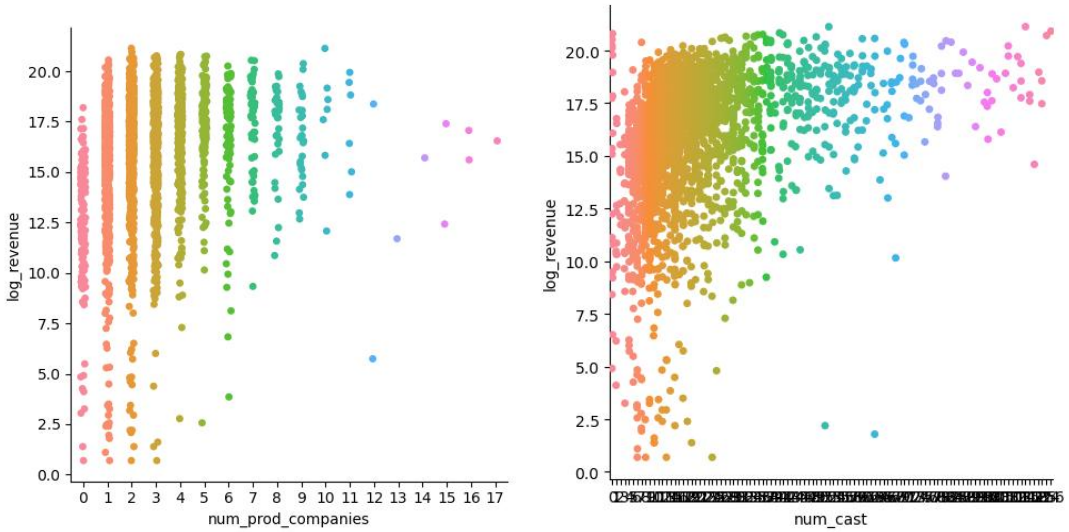


3、转化为计数属性并删除原属性（6 种）

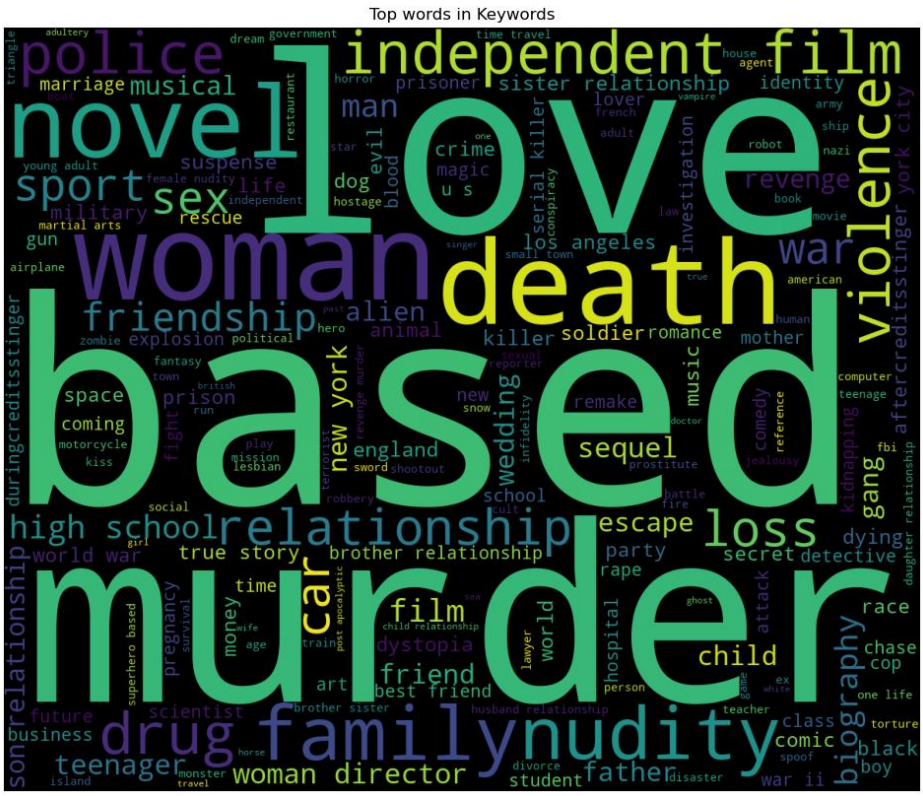
production_companies, production_countries, spoken_languages, Keywords, cast, crew

这些属性大多以 json 格式编码，不方便处理且与票房数据相关性不太密切。但直接丢弃有可能会丢失信息，故将其转化为 num_* 的属性。

转化后与 revenue 的散点图：前四种属性与后两种属性分别大致遵循以下两种，这里不再赘述：

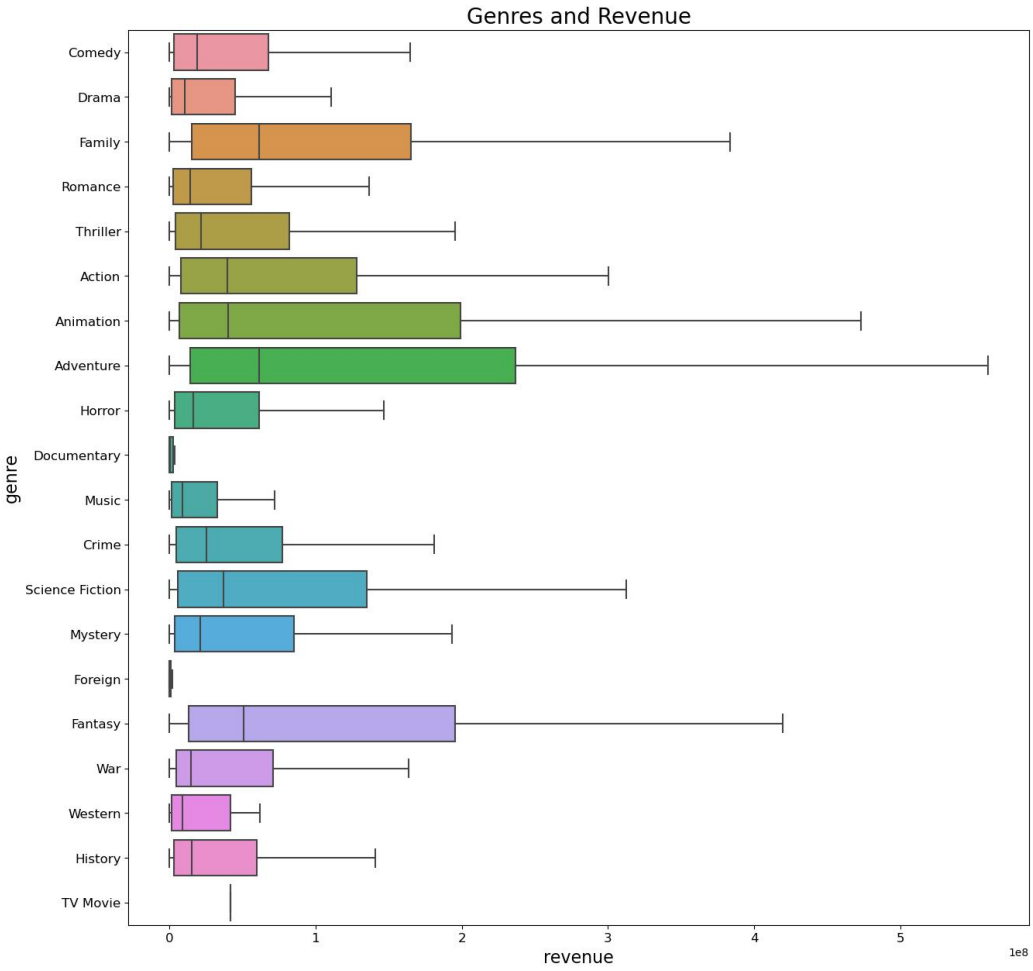


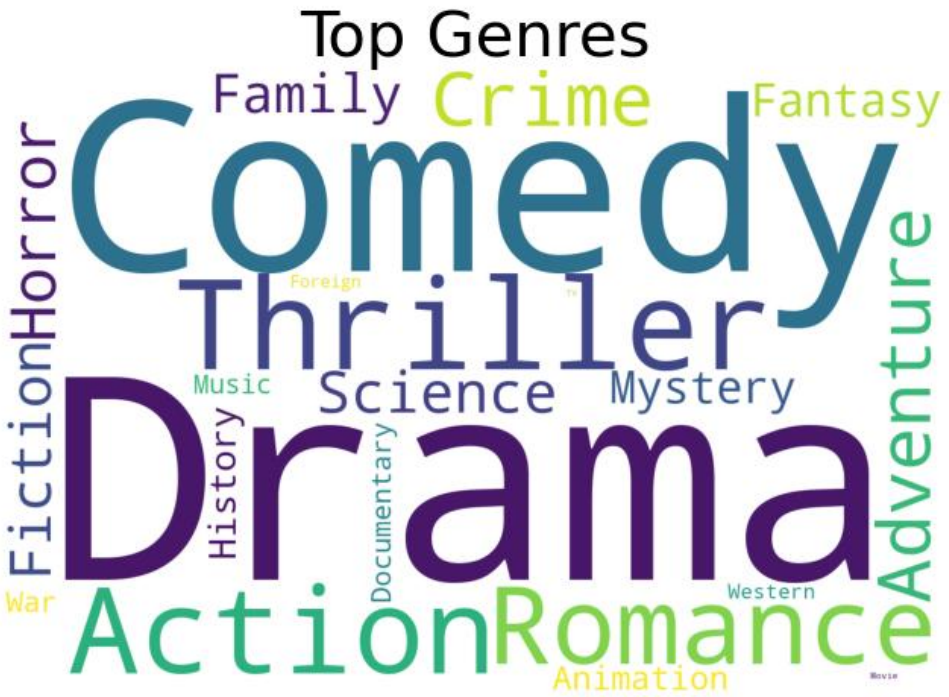
其中， Keywords 属性可视化如下：

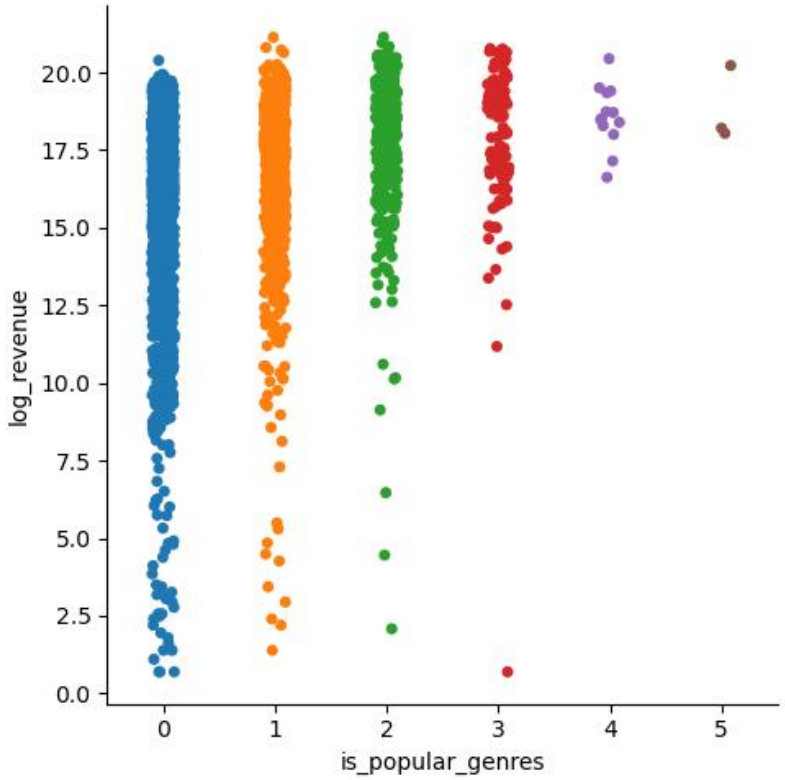


4、genres 转化为计数属性并删除原属性

由于 genres 属性较为特殊，一个电影可能分为不同流派风格，存在一些受欢迎的流派风格。因此将 json 属性拆解后根据票房数量分为流行类型和非流行类型，选取票房中值和均值综合最高的六种类型作为流行类型。分别为[‘Adventure’, ‘Animation’, ‘Fantasy’, ‘Family’, ‘Action’, ‘Science Fiction’]。按该电影流行程度对 is_popular_genres 属性计数，一个电影该属性最高为 6。

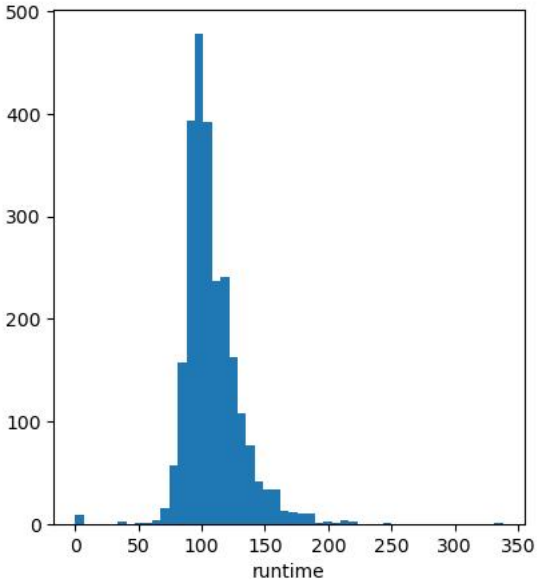

Genres and Revenue
Comedy
Drama
Family
Romance
Thriller
Action
Animation
Adventure
Horror
Documentary
Music
Crime
Science Fiction
Mystery
Foreign
Fantasy
War
Western
History
TV Movie
genre
0
1
2
3
4
5
revenue
1e8


Top Genres
Family
Crime
Fantasy
Comedy
Horror
Fiction
Thriller
Foreign
Music
Science
Mystery
Adventure
Drama
History
Documentary
War
Action
Romance
Western
Animation



5、均值填充缺失值

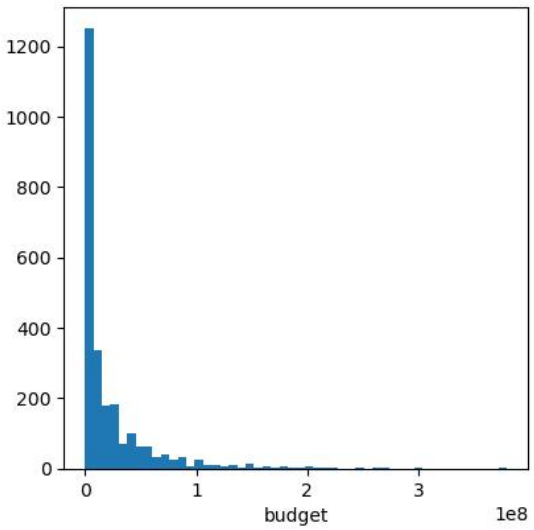
由于 runtime 属性存在少量缺失值（2 个），且其基本满足正态分布。故对其采用均值填充。

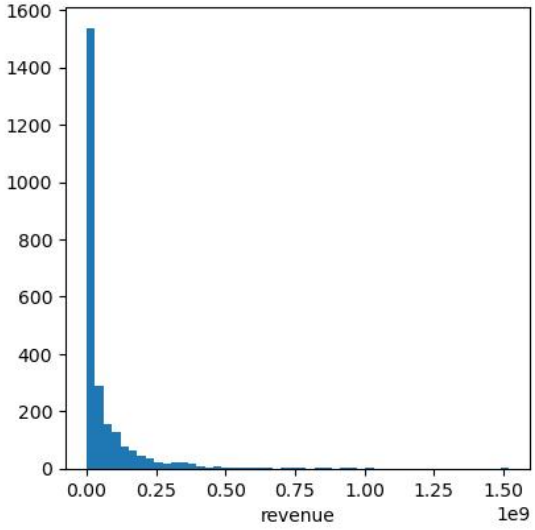


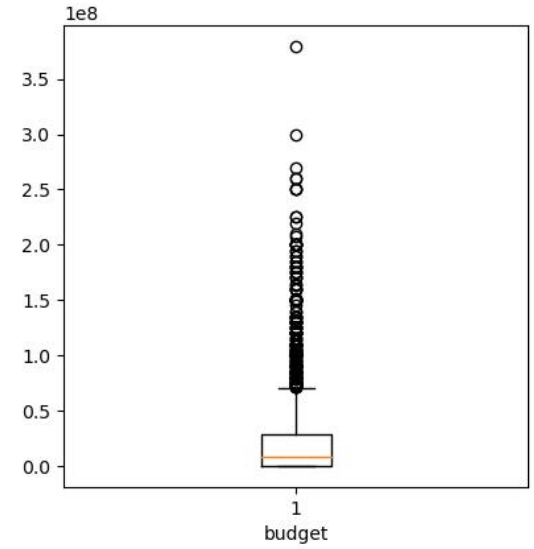
6、取 log 比较

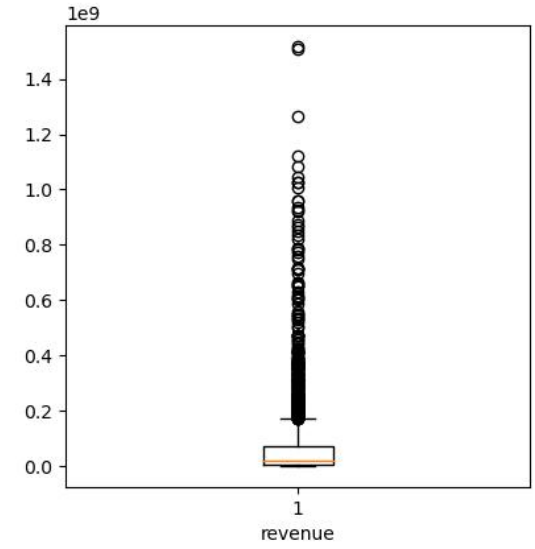
budget, popularity, revenue

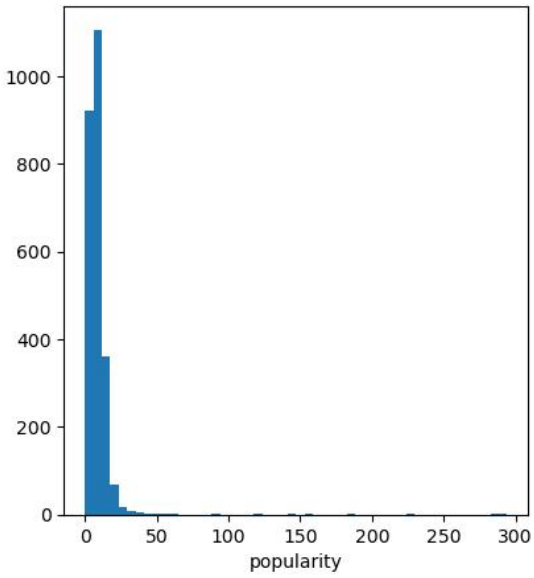
由于这三个数据存在数据分布不均匀的情况，对其取 log 后再进行分析。原始数据分布如下：

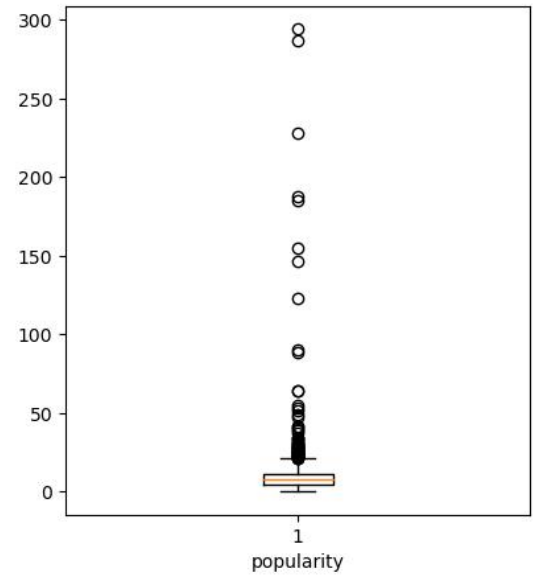




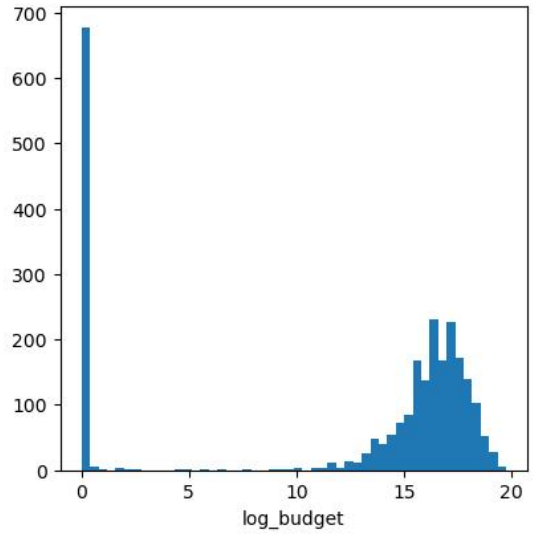


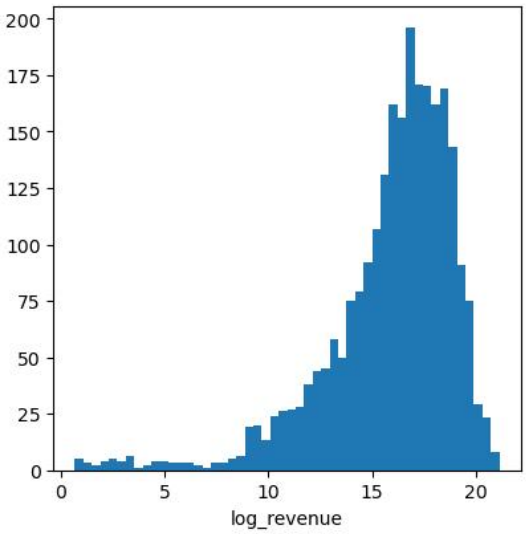


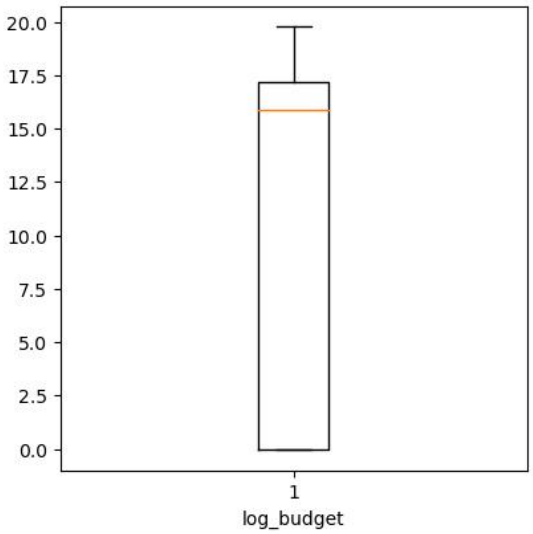


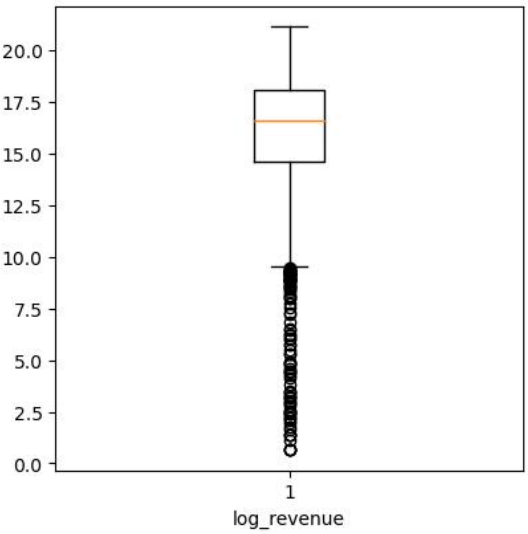


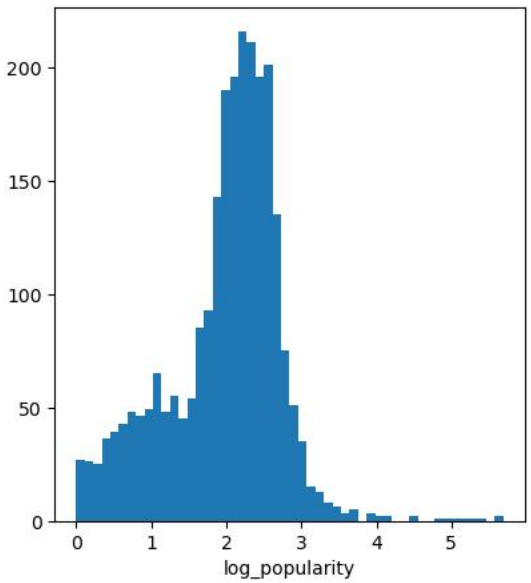
用 log 转换后分布如下：

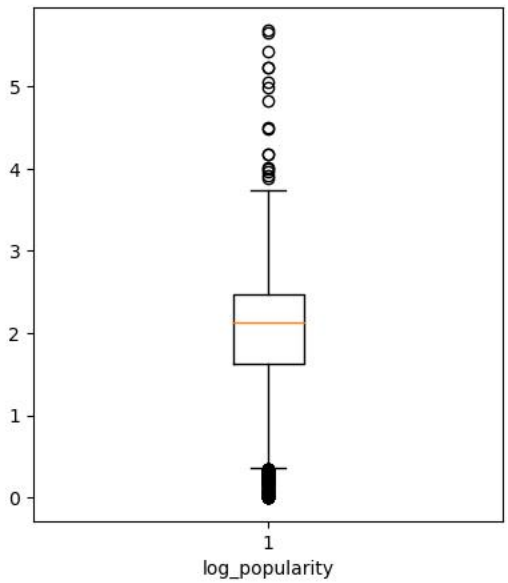




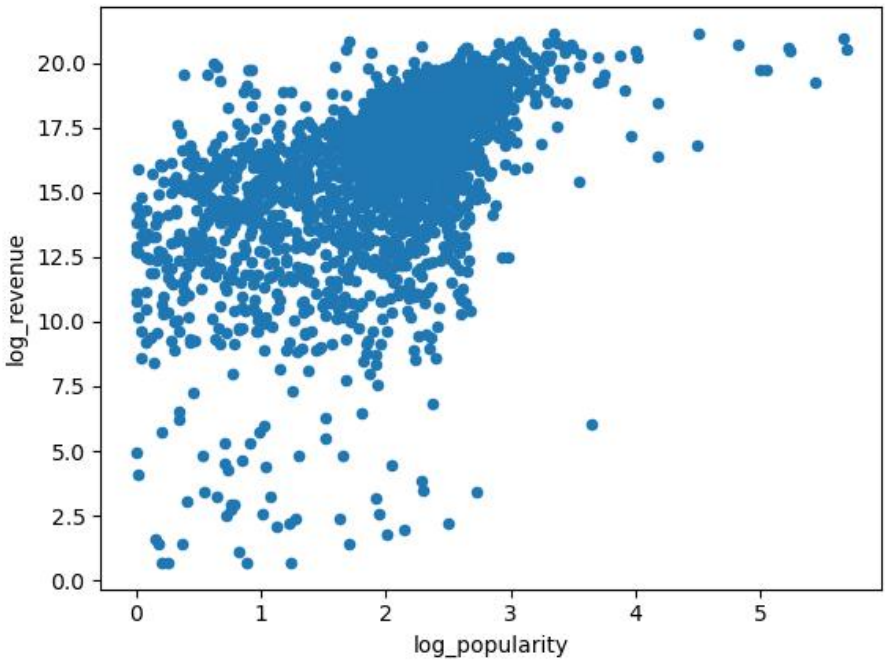








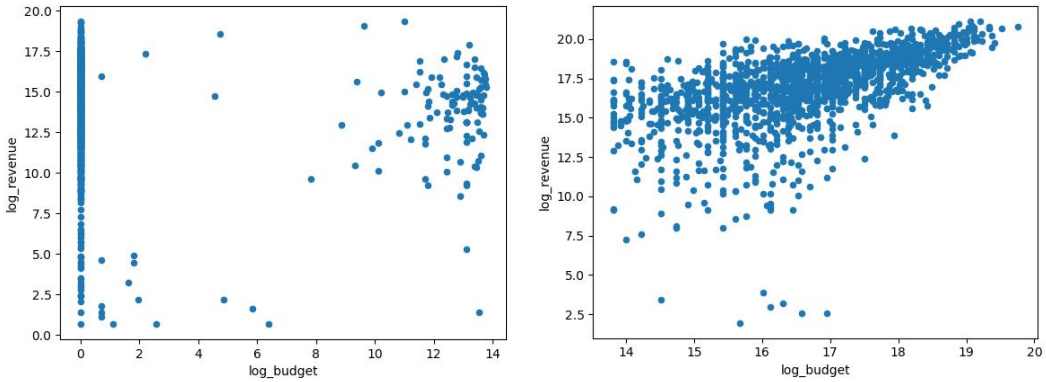
属性 log_popularity 与 log_revenue 的散点图分布如下，两者大致呈现线性关系。



budget 存在大量零值，需要再处理，这里不展示其与票房收入的关系。

7、budget 分数据集分析

budget 存在大量零值，这些零值严重干扰散点图分布。现将 budget < 1000000 的数据作为低预算数据，将其余数据当作高预算数据。分别绘制 log_budget 与 log_revenue 之间的散点图。

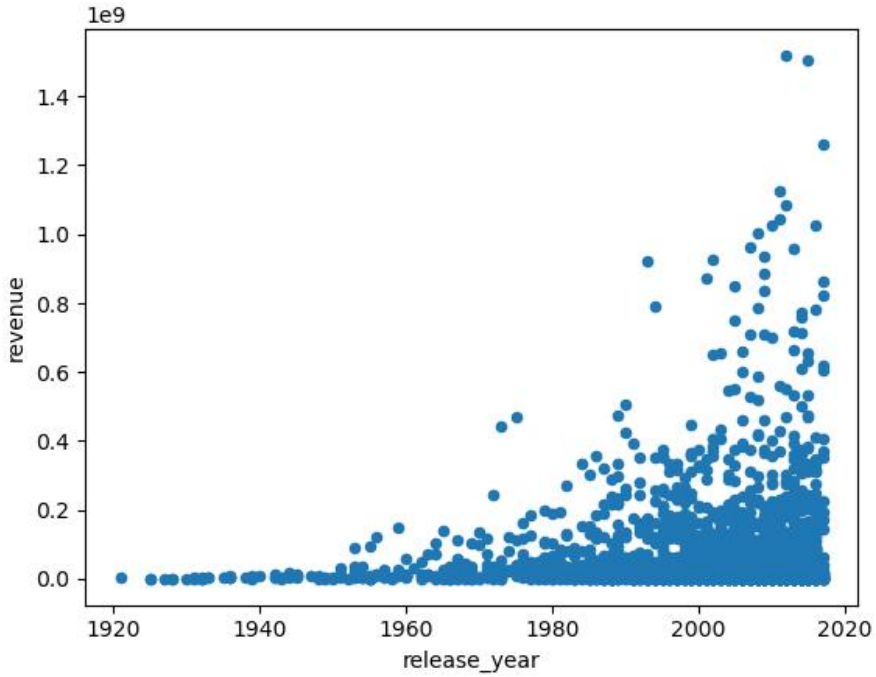


从可视化可以看出，低成本电影票房收入与预算之间没有明显的相关关系。而高成本电影票房收入则与预算之间有着较为明显的相关关系。

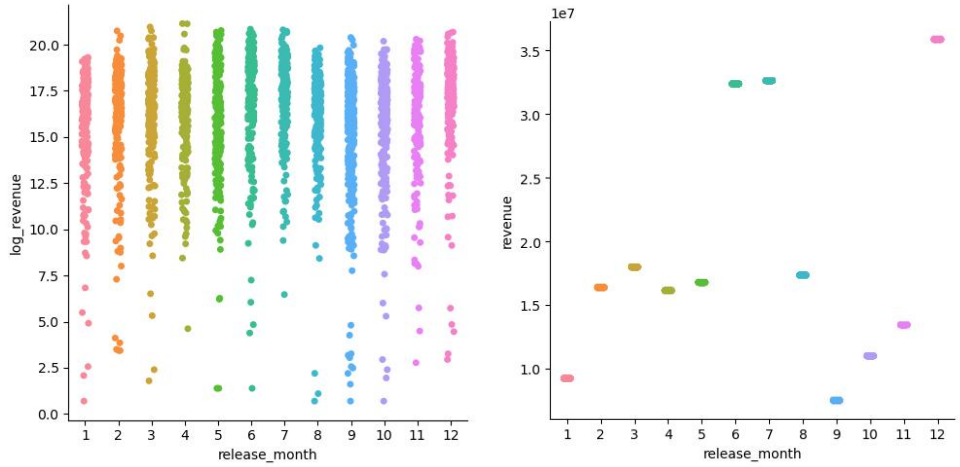
8、release_date 分为年、月、日（星期）三个属性

将 release_date 分为三个属性进行分析。

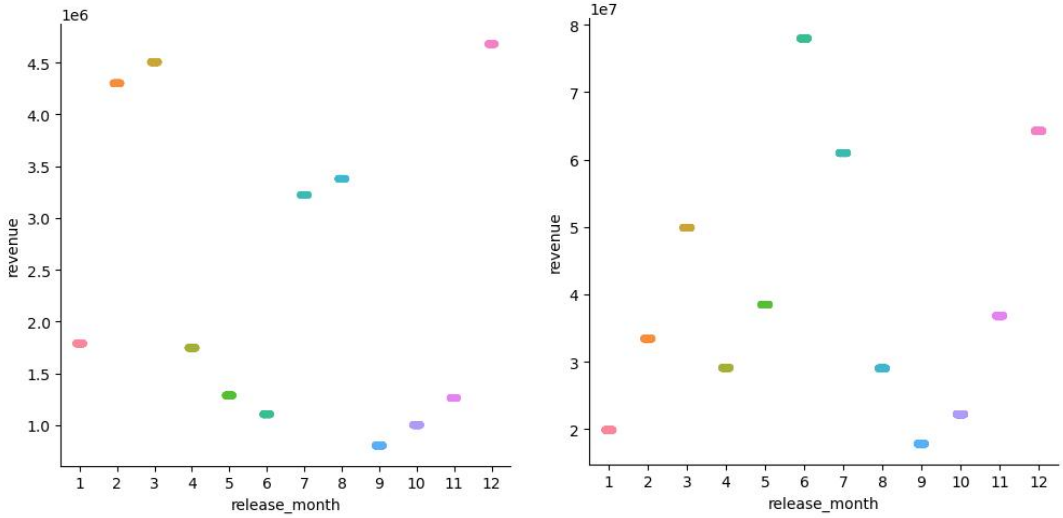
（1）随着年份（release_year）增大，票房收入也在增加。



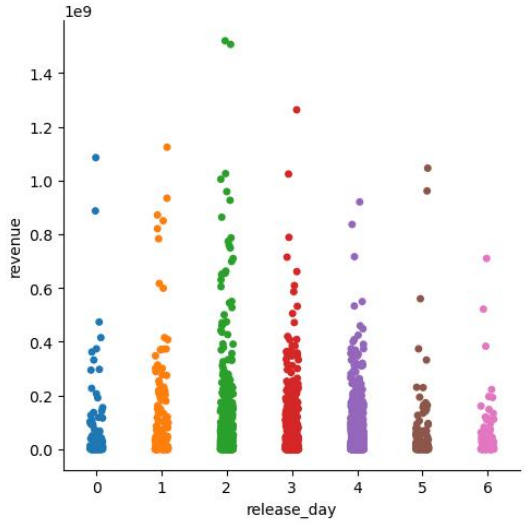
（2）不同月份票房收入呈现一些规律性（右图为票房中值）：

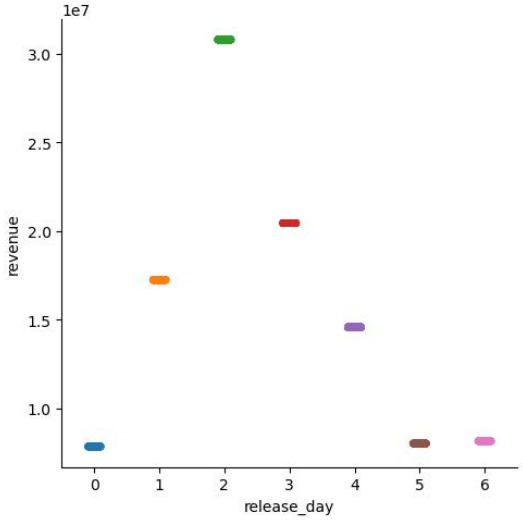


对低成本电影和高成本电影分别可视化（左低右高，均为票房中值）：

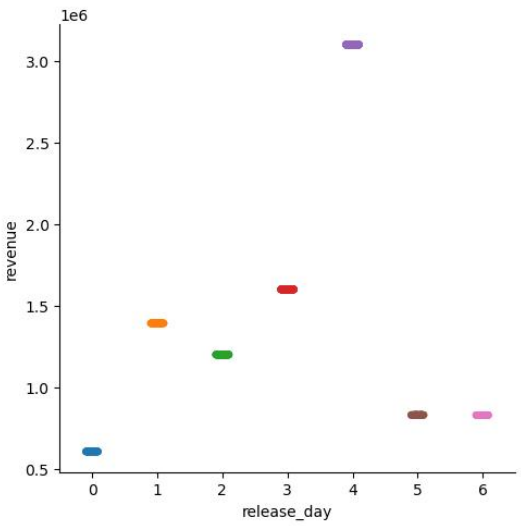


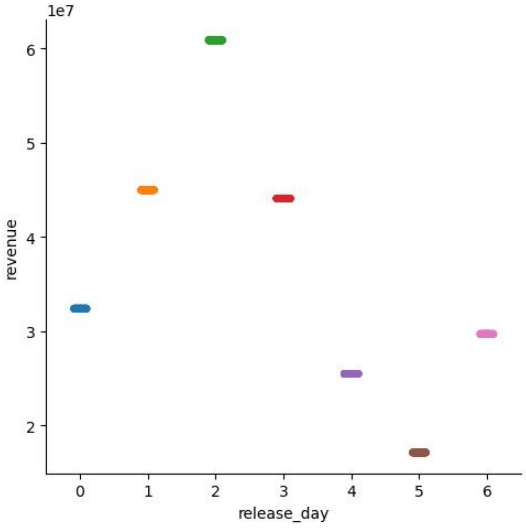
（3）一周内的不同天数也存在着一定的规律性（右图为票房中值）：





对低成本电影和高成本电影分别可视化（左低右高，均为票房中值）：





四、数据初步处理结果

RangeIndex: 2500 entries, 0 to 2499

Data columns (total 19 columns):

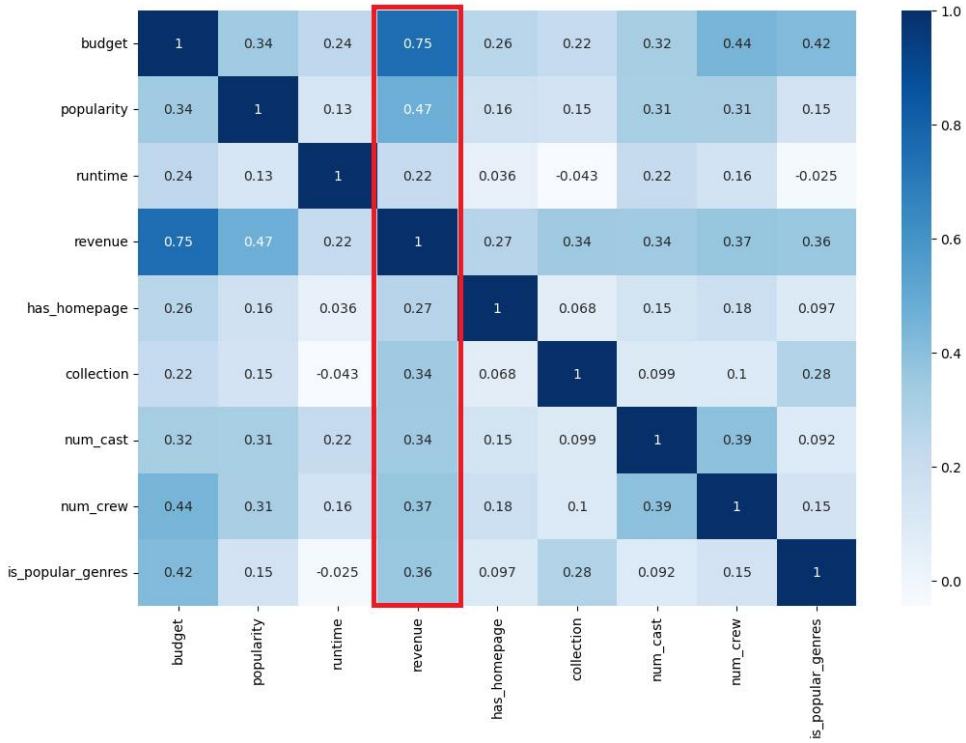
#	Column	Non-Null Count	Dtype
0	id	2500 non-null	int64
1	budget	2500 non-null	int64
2	popularity	2500 non-null	float64
3	runtime	2500 non-null	float64
4	revenue	2500 non-null	int64
5	has_homepage	2500 non-null	int64
6	release_year	2500 non-null	int64
7	release_day	2500 non-null	int32

8	release_month	2500 non-null	int32
9	is_en	2500 non-null	int64
10	collection	2500 non-null	int64
11	num_prod_countries	2500 non-null	int64
12	num_prod_companies	2500 non-null	int64
13	num_spoken_languages	2500 non-null	int64
14	num_cast	2500 non-null	int64
15	num_crew	2500 non-null	int64
16	has_tagline	2500 non-null	int64
17	num_keywords	2500 non-null	int64
18	is_popular_genres	2500 non-null	int64

五、分析数据相关性

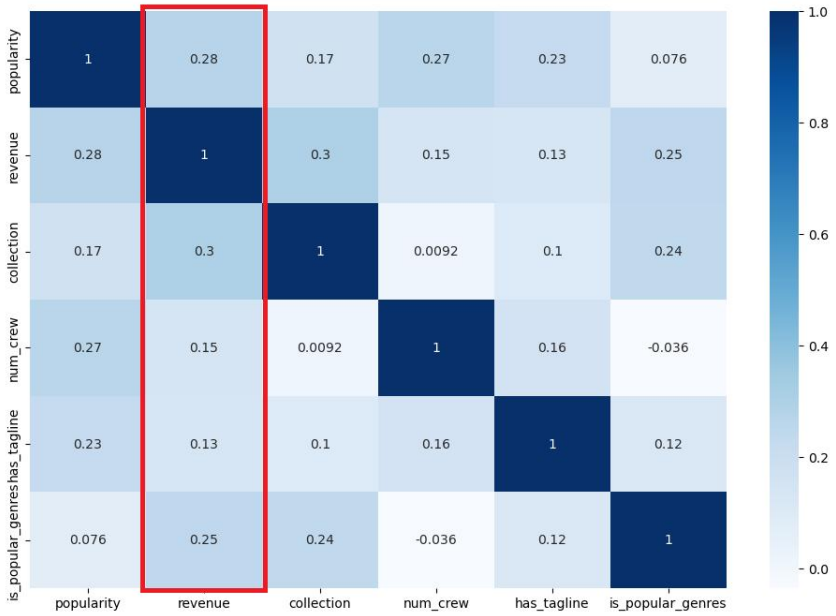
1、完整数据相关性

下图展示与票房收入相关性大于 0.2 的属性间的热点图：



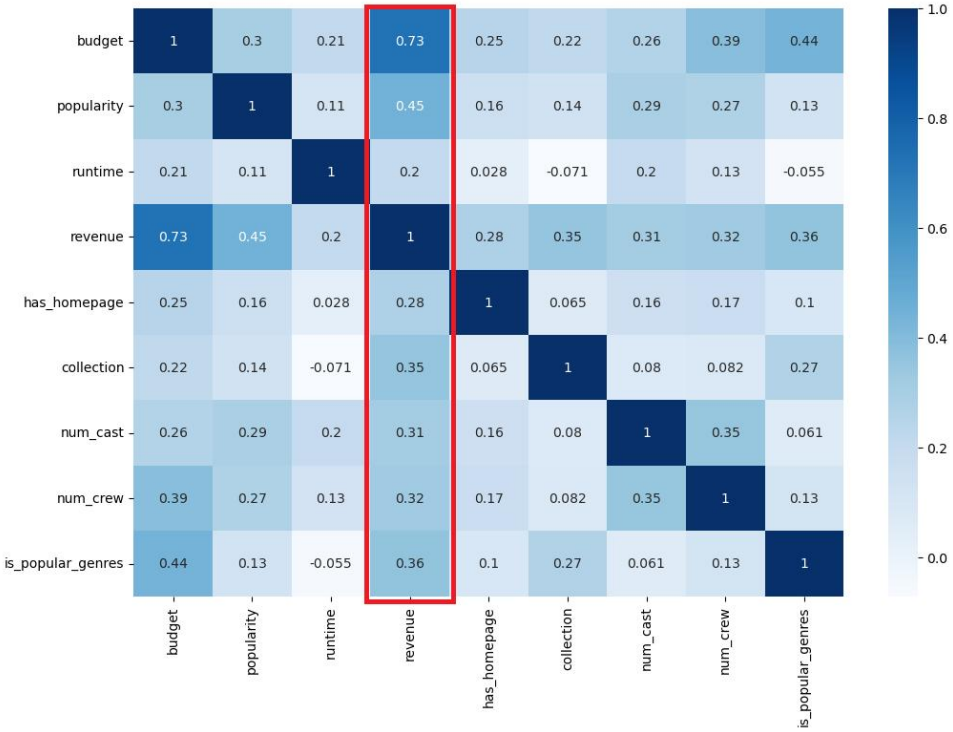
2、低成本数据相关性

下图展示低成本电影数据中与票房收入相关性大于 0.1 的属性间的热点图：



3、高成本数据相关性

下图展示高成本电影数据中与票房收入相关性大于 0.2 的属性间的热点图：



分析：从三幅相关关系热点图中可以看出，完整数据和高成本电影数据中票房收入与预算、受欢迎指数、是否被收藏等属性间都有较好的相关关系。而在低成本电影数据中，票房关系与各属性间的相关关系都很差。因此，尽管在 2500 条电影数据中有 805 条低成本电影数据，依然建议将整个数据集作为一个整体进行预测。