

Исследование рынка российского кинопроката

Описание проекта

Нужно изучить рынок российского кинопроката и выявить текущие тренды. Уделить внимание фильмам, которые получили государственную поддержку. Будем работать с данными, опубликованными на портале открытых данных Министерства культуры. Набор данных содержит информацию о прокатных удостоверениях, сборах и государственной поддержке фильмов, а также информацию с сайта КиноПоиск.

Заказчик: Министерство культуры РФ

Цель: Выявить тренды российского кинопроката

Гипотеза: Фильмы с государственной поддержкой интересны российскому зрителю

О качестве данных ничего не известно, поэтому перед подтверждением гипотезы потребуется обзор данных. Мы проверим данные на ошибки и оценим их влияние на исследование. Затем, на этапе предобработки мы поищем возможность исправить самые критичные ошибки данных. Затем проведем исследовательский анализ и закончим все выводами, полученными в ходе исследования.

Таким образом исследование пройдет в три этапа:

1. Изучение данных
2. Предобработка данных
3. Исследовательский анализ

1. Изучение данных

1.1. Прочитаем файлы, произведем первичную оценку полученных данных

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
In [2]: import warnings
warnings.filterwarnings('ignore')
```

```
In [3]: pd.set_option('display.max_columns', None)
```

```
In [4]: try:
data_m = pd.read_csv('/datasets/mkrf_movies.csv')
data_s = pd.read_csv('/datasets/mkrf_shows.csv')
except:
data_m = pd.read_csv('mkrf_movies.csv')
data_s = pd.read_csv('mkrf_shows.csv')
```

```
In [5]: data_m.head()
```

```
Out[5]:
```

	title	puNumber	show_start_date	type	film_studio	production_country
0	Открытый простор	221048915	2015-11-27T12:00:00.000Z	Художественный	Тачстоун Пикчерз, Кобальт Пикчерз, Бикон Пикче...	США
1	Особо важное задание	111013716	2016-09-13T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	СССР
2	Особо опасен	221038416	2016-10-10T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	США Бе
3	Особо опасен	221026916	2016-06-10T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	США Бе
4	Особо опасен	221030815	2015-07-29T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	США Бе

```
In [6]: data_s.head()
```

Out[6]:

	puNumber	box_office
0	111000113	2.450000e+03
1	111000115	6.104000e+04
2	111000116	1.530300e+08
3	111000117	1.226096e+07
4	111000118	1.636841e+08

In [7]: data_m.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7486 entries, 0 to 7485
Data columns (total 15 columns):
#   Column                Non-Null Count  Dtype
---  -
0   title                  7486 non-null   object
1   puNumber               7486 non-null   object
2   show_start_date        7486 non-null   object
3   type                   7486 non-null   object
4   film_studio            7468 non-null   object
5   production_country      7484 non-null   object
6   director               7477 non-null   object
7   producer               6918 non-null   object
8   age_restriction        7486 non-null   object
9   refundable_support     332 non-null    float64
10  nonrefundable_support  332 non-null    float64
11  budget                 332 non-null    float64
12  financing_source       332 non-null    object
13  ratings                6519 non-null   object
14  genres                 6510 non-null   object
dtypes: float64(3), object(12)
memory usage: 877.4+ KB
```

Как мы видим в датафрейме часть данных пропущена, в столбцах: producer, refundable_support, nonrefundable_support, budget, financing_source, ratings, genres.

Также часть столбцов необходимо сконвертировать в другие типы данных:
show_start_date -> datetime; refundable_support, nonrefundable_support, budget -> int;
rating -> float.

Посмотрим на данные в следующем датафрейме:

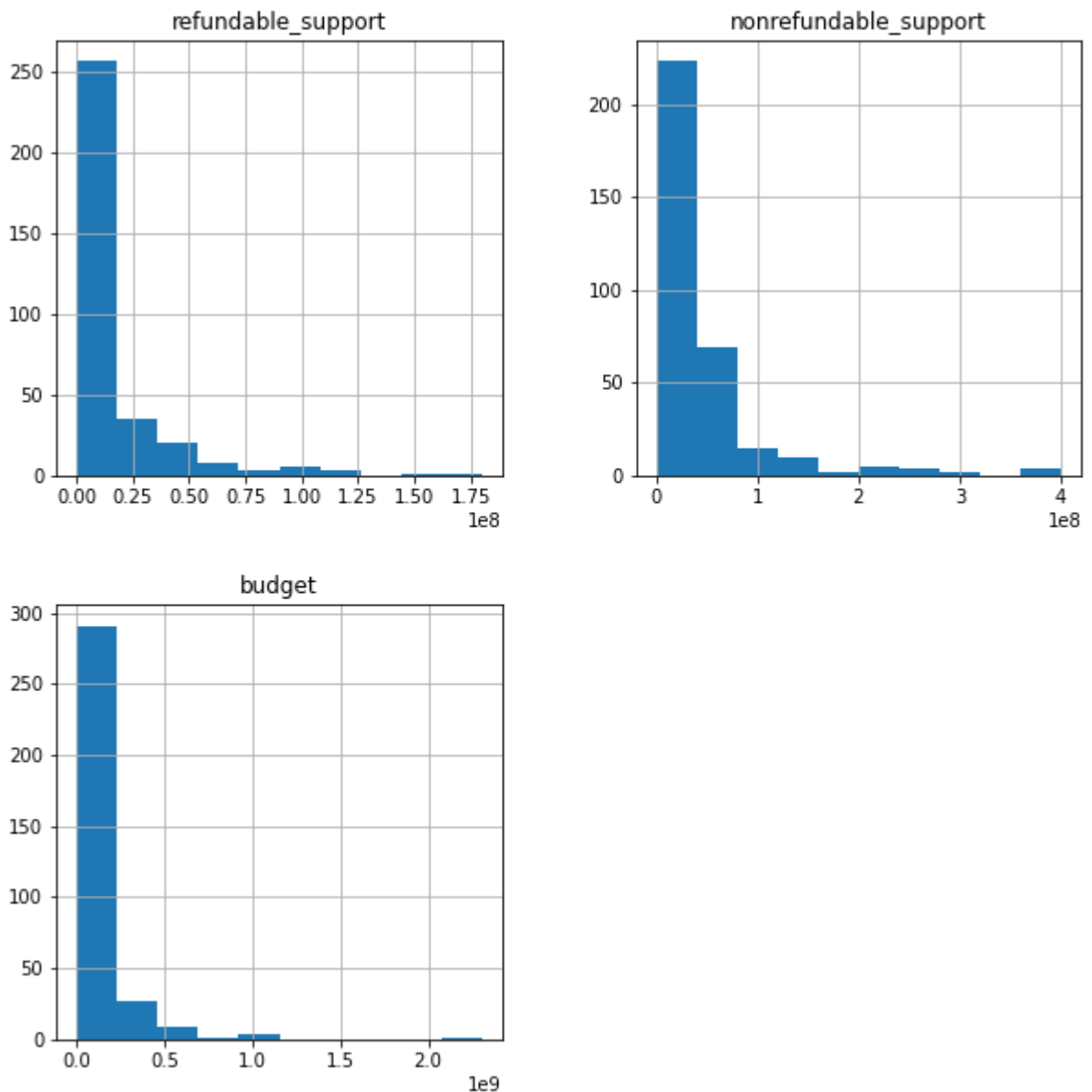
In [8]: data_s.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3158 entries, 0 to 3157
Data columns (total 2 columns):
#   Column      Non-Null Count  Dtype
---  -
0   puNumber    3158 non-null   int64
1   box_office   3158 non-null   float64
dtypes: float64(1), int64(1)
memory usage: 49.5 KB
```

Пропусков данных нет, столбец 'box_office' можно конвертировать в int, т.к. сборы в ру,лях, а копейками не пользуются

1.2. Посмотрим на распределение данных в датафреймах

```
In [9]: data_m.hist(figsize=(10,10));
```



Во всех случаях имеем Пуассоновское распределение данных

1.3. Объединим два датафрейма в один

Для объединения будем использовать столбец 'puNumber', но если мы присмотримся внимательно к этому столбцу в датафреймах, то обнаружим что в одном датафрейме он имеет тип object, а в другом тип int. Поэтому необходимо его сначала привести к единому типу int и затем объединять датафреймы:

Напишем функцию, которая получает строковое значение на входе и возвращает целое число если значение можно конвертировать и NaN, если значение конвертировать нельзя:

```
In [10]: def int_or_str(cell_data):  
         try:  
             return int(cell_data)  
         except:  
             return np.NaN
```

Применим вышеналиченную функцию к каждому значению в столбце 'puNumber'

```
In [11]: data_m['puNumber'] = data_m['puNumber'].apply(int_or_str)
```

Посмотрим сколько строк со значением NaN:

```
In [12]: data_m[data_m['puNumber'].isna()]
```

```
Out[12]:
```

	title	puNumber	show_start_date	type	film_studio	production_countr
1797	Курбан-роман. (История с жертвой)	NaN	2014-05-15T12:00:00.000Z	Художественный	ФОНД "ИННОВАЦИЯ"	Росси

Строку можно удалить так как у фильма нет прокатного удостоверения, других финансовых данных, как нет и рейтинга.

```
In [13]: data_m.drop(1797, axis=0, inplace=True)
```

```
In [14]: data_m[data_m['puNumber'].isna()]
```

```
Out[14]:
```

	title	puNumber	show_start_date	type	film_studio	production_country	director	producer	ag
--	-------	----------	-----------------	------	-------------	--------------------	----------	----------	----

Фильм удален

Теперь конвертируем столбец 'puNumber' в тип int64

```
In [15]: data_m['puNumber'] = data_m['puNumber'].astype('int64')
```

```
In [16]: data_m.dtypes
```

```
Out[16]: title                object
puNumber                    int64
show_start_date            object
type                       object
film_studio                object
production_country          object
director                   object
producer                   object
age_restriction            object
refundable_support         float64
nonrefundable_support      float64
budget                     float64
financing_source           object
ratings                    object
genres                     object
dtype: object
```

Теперь можно произвести объединение двух датафреймов в один:

```
In [17]: data = data_m.merge(data_s, how='left', on='puNumber', validate='many_to_one')
data
```

Out[17]:

	title	puNumber	show_start_date	type	film_studio	production
0	Открытый простор	221048915	2015-11-27T12:00:00.000Z	Художественный	Тачстоун Пикчерз, Кобальт Пикчерз, Бикон Пикче...	
1	Особо важное задание	111013716	2016-09-13T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
2	Особо опасен	221038416	2016-10-10T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	
3	Особо опасен	221026916	2016-06-10T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	
4	Особо опасен	221030815	2015-07-29T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	
...
7480	Сад художника: Американский импрессионизм	126008019	2019-12-23T12:00:00.000Z	Прочие	Севен Артс Продакшнз	Великобритания
7481	Звериная ярость	121037819	2019-12-24T12:00:00.000Z	Художественный	Дэниел Гродник Продакшнз, Вандерфилм Медиа Кор...	
7482	Щенячий патруль: Скорей спешим на помощь!	124010819	2019-12-28T12:00:00.000Z	Художественный	Никелодеон Анимейшн СтудIOS, Спин Мастер	США
7483	Машины песенки. Серия "Испанские мотивы"	124010619	2019-12-30T12:00:00.000Z	Анимационный	Анимаккорд Лтд., ООО Студия "АНИМАККОРД"	Республика Беларусь

	title	puNumber	show_start_date	type	film_studio	production
7484	Машины песенки. Серия "Когда цветут кактусы"	124010719	2019-12- 30T12:00:00.000Z	Анимационный	Анимаккорд Лтд., ООО Студия "АНИМАККОРД"	Республ

7485 rows × 16 columns

```
In [18]: data.describe(include='all')
```

	title	puNumber	show_start_date	type	film_studio	production_cou
count	7485	7.485000e+03	7485	7485	7467	
unique	6771	NaN	1964	8	5490	
top	День святого Валентина_	NaN	2010-12- 17T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
freq	5	NaN	96	4519	341	
mean	NaN	1.351966e+08	NaN	NaN	NaN	
std	NaN	3.835292e+07	NaN	NaN	NaN	
min	NaN	1.811096e+06	NaN	NaN	NaN	
25%	NaN	1.120251e+08	NaN	NaN	NaN	
50%	NaN	1.210155e+08	NaN	NaN	NaN	
75%	NaN	1.240033e+08	NaN	NaN	NaN	
max	NaN	2.310011e+08	NaN	NaN	NaN	

1.4. Посмотрим на наличие дубликатов в датафрейме

```
In [19]: data.duplicated().sum()
```

```
Out[19]: 0
```

в датафрейме data дубликаты отсутствуют, но при этом визуальное ознакомление показывает наличие дубликатов по столбцам. разберемся с этим позже

2. Предобработка данных

2.1. Заполним пропуски в данных

Посмотрим сколько пропущенных значений по столбцам

```
In [20]: pd.DataFrame(round(data.isna().mean()*100)).style.background_gradient('coolwarm')
```


Out[20]:

0

title	0.000000
puNumber	0.000000
show_start_date	0.000000
type	0.000000
film_studio	0.000000
production_country	0.000000
director	0.000000
producer	8.000000
age_restriction	0.000000
refundable_support	96.000000
nonrefundable_support	96.000000
budget	96.000000
financing_source	96.000000
ratings	13.000000
genres	13.000000
box_office	58.000000

2.1.1 Столбец 'producer'

Найдем кол-во пропущенных значений

```
In [21]: data['producer'].isna().sum()
```

Out[21]: 568

Посмотрим на строки датафрейма, содержащие пропуски в столбце 'producer':

```
In [22]: data[data['producer'].isna()]
```

Out[22]:

	title	puNumber	show_start_date	type	film_studio	production_co
1	Особо важное задание	111013716	2016-09- 13T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
5	Остановился поезд	111013816	2016-09- 13T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
6	Любовь и голуби	111007013	2013-10- 18T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
9	Отпуск за свой счет	111019114	2014-12- 01T12:00:00.000Z	Художественный	Киностудия "Мосфильм", Телевидение ВНР	СССР, Венгрия
11	Ответный ход	111019014	2014-12- 01T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
...
7404	Отряд специального назначения	111020419	2019-12- 01T12:00:00.000Z	Художественный	Свердловская киностудия	
7405	Назначаешься внучкой	111020519	2019-12- 01T12:00:00.000Z	Художественный	Свердловская киностудия	
7406	Перед рассветом	111020619	2019-12- 01T12:00:00.000Z	Художественный	Свердловская киностудия	
7407	Ты помнишь	111020719	2019-12- 01T12:00:00.000Z	Художественный	Свердловская киностудия	
7408	Здесь твой фронт	111020819	2019-12- 01T12:00:00.000Z	Художественный	Свердловская киностудия	

568 rows × 16 columns

Информации по продюсерам нет, заменим пропуски значением "нет данных"

```
In [23]: data['producer'] = data['producer'].fillna('нет данных')
```

```
In [24]: data['producer'].isna().sum()
```

Out[24]: 0

2.1.2 Столбцы 'refundable_support', 'nonrefundable_support', 'budget', 'financing_source'

Сколько пропущенных значений в столбце 'financing_source'

```
In [25]: data[data['financing_source'].isna()]
```

Out[25]:

	title	puNumber	show_start_date	type	film_studio	production
0	Открытый простор	221048915	2015-11-27T12:00:00.000Z	Художественный	Тачстоун Пикчерз, Кобальт Пикчерз, Бикон Пикче...	
1	Особо важное задание	111013716	2016-09-13T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
2	Особо опасен	221038416	2016-10-10T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	
3	Особо опасен	221026916	2016-06-10T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	
4	Особо опасен	221030815	2015-07-29T12:00:00.000Z	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	
...
7480	Сад художника: Американский импрессионизм	126008019	2019-12-23T12:00:00.000Z	Прочие	Севен Артс Продакшнз	Великобритания
7481	Звериная ярость	121037819	2019-12-24T12:00:00.000Z	Художественный	Дэниел Гродник Продакшнз, Вандерфилм Медиа Кор...	
7482	Щенячий патруль: Скорей спешим на помощь!	124010819	2019-12-28T12:00:00.000Z	Художественный	Никелодеон Анимейшн СтудIOS, Спин Мастер	США
7483	Машины песенки. Серия "Испанские мотивы"	124010619	2019-12-30T12:00:00.000Z	Анимационный	Анимаккорд Лтд., ООО Студия "АНИМАККОРД"	Республика Беларусь

	title	puNumber	show_start_date	type	film_studio	production
7484	Машины песенки. Серия "Когда цветут кактусы"	124010719	2019-12- 30T12:00:00.000Z	Анимационный	Анимаккорд Лтд., ООО Студия "АНИМАККОРД"	Республ

7153 rows × 16 columns

Для удобства конвертации в int (копейки не используются), заменим все пропуски в столбцах 'refundable_support', 'nonrefundable_support', 'budget' значением -1, оно не повлияет на дальнейший анализ:

```
In [26]: cols = ['refundable_support', 'nonrefundable_support', 'budget']
for col in cols:
    data[col] = data[col].fillna(-1)
```

```
In [27]: cols = ['refundable_support', 'nonrefundable_support', 'budget']
for col in cols:
    print(f'Столбец {col} пропусков {data[col].isna().sum()}')
```

Столбец refundable_support пропусков 0
Столбец nonrefundable_support пропусков 0
Столбец budget пропусков 0

Конвертируем столбцы в int:

```
In [28]: cols = ['refundable_support', 'nonrefundable_support', 'budget']
for col in cols:
    data[col] = data[col].astype(int)
    print(f'Столбец {col} имеет тип {data[col].dtypes}')
```

Столбец refundable_support имеет тип int64
Столбец nonrefundable_support имеет тип int64
Столбец budget имеет тип int64

В столбце 'financing_source' заменим значения NaN на 'нет данных' - столбец категориальный и в нем должны быть категории. 'нет данных' - будет одной из них.

```
In [29]: data['financing_source'] = data['financing_source'].fillna('нет данных')
```

```
In [30]: data['financing_source'].isna().sum()
```

Out[30]: 0

Пропусков в столбце 'financing_source' нет

2.1.3 Столбец 'ratings'

Сколько пропущенных значений в столбце:

```
In [31]: data['ratings'].isna().sum()
```

Out[31]: 966

Рейтинг фильма лежит в пределах от 0 до 10, где 0 минимальное значение, 10 - максимальное. При отсутствии рейтинга его можно заменить на нулевое значение. Затем мы сможем сконвертировать столбец в float тип, что более правильно будет отражать его сущность.

```
In [32]: data['ratings'] = data['ratings'].fillna(0)
```

```
In [33]: data['ratings'].isna().sum()
```

Out[33]: 0

Преобразуем 'ratings' в тип float

```
In [34]: data['ratings'].unique()
```

```
Out[34]: array(['7.2', '6.6', '6.8', '7.7', '8.3', '8.0', '7.8', '8.1', '7.1',  
                '6.0', '7.4', '5.8', '8.7', '6.3', '6.9', '5.0', '4.3', '7.3',  
                '7.0', '6.4', 0, '8.2', '7.5', '6.7', '7.9', '5.9', '6.2', '5.6',  
                '6.5', '2.4', '7.6', '6.1', '8.6', '8.5', '8.8', '5.5', '5.1',  
                '5.7', '5.4', '99%', '4.4', '4.5', '5.3', '4.1', '8.4', '2.6',  
                '3.8', '4.6', '4.8', '4.0', '3.0', '1.6', '4.2', '5.2', '4.7',  
                '4.9', '3.9', '2.7', '3.3', '2.9', '28%', '3.7', '1.4', '3.1',  
                '97%', '3.5', '3.2', '2.8', '1.5', '2.1', '2.5', '9.2', '3.4',  
                '1.1', '3.6', '83%', '64%', '91%', '94%', '62%', '79%', '90%',  
                '19%', '88%', '1.0', '89%', '1.3', '1.9', '1.8', '1.2', '1.7',  
                '9.0', '98%', '8.9', '9.1'], dtype=object)
```

Мы видим, что среди значений столбца присутствуют "странные" значения с %. Посмотрим на них поближе:

```
In [35]: data[data['ratings'].str.contains('%', regex=True, na=False)]
```

Out[35]:

	title	puNumber	show_start_date	type	film_studio	productic
259	Прошлое	221030715	2015-07-29T12:00:00.000Z	Художественный	20 Сенчюри Фокс де Архентина, Чемо, ЭйчБи Филм...	A
811	От винта!	114000212	2012-08-06T12:00:00.000Z	Анимационный	ЗАО "Продюсерский центр "Парадиз"	
1053	Песочный человек	121029212	2012-11-29T12:00:00.000Z	Художественный	Спотлайт Медиа Продакшн	L
1341	Сказка о добре и Вре	111008913	2013-11-15T12:00:00.000Z	Художественный	НП "Студия игрового кино "Лик"	
1804	Спираль.	111001014	2014-01-30T12:00:00.000Z	Художественный	ООО "Компания Питон"	
1873	Короткие истории о любви 2	121002615	2015-02-13T12:00:00.000Z	Художественный	Кавиар, Курт 13, СтритЛайт Филмз, Нэйер Дойче ...	Нидерла - Германи -
1898	Золушка /По сказке Шарля Перро/. Холодное торж...	121003615	2015-02-27T12:00:00.000Z	Художественный	Женр Филмз, Уолт Дисней Пикчерз	
2125	Поездка к матери	111002015	2015-03-02T12:00:00.000Z	Художественный	ЗАО "Киностудия "М"-Фильм"	
2145	Главный	111003415	2015-03-27T12:00:00.000Z	Художественный	ООО "Киностудия "МАСТЕР"	
2437	Самый рыжий Лис	111012715	2015-07-30T12:00:00.000Z	Художественный	ООО "Продюсерский центр "Ленфильм"	
2679	Год Белого Слона	111003915	2015-04-08T12:00:00.000Z	Художественный	ООО "Творческое объединение ЮГ"	
3138	Я умею вязать	111019615	2015-11-27T12:00:00.000Z	Художественный	ООО "Белое Зеркало"	
3430	Наурыз	131000217	2017-03-09T12:00:00.000Z	Художественный	Эс Джи	F

	title	puNumber	show_start_date	type	film_studio	productic
3438	Семь диких историй	126005916	2016-11-28T12:00:00.000Z	Прочие	Макс Бэйкер, Джон Е.Брайан, Пьер-Луи Гарнон, ...	США - Пу
3445	Короткие истории о любви - 4	126006316	2016-12-30T12:00:00.000Z	Прочие	Нетворг Айлэнд Телевижн, Магнетфильм, Шорткатс...	Велико Аргенти Фр
3513	Чужой дом	111015516	2016-09-14T12:00:00.000Z	Художественный	ООО "Лига Продакшн"	Росс Хорвати
3584	Машины Страшили. Серия "Жутчайшая повесть о п...	114003916	2016-12-09T12:00:00.000Z	Анимационный	ООО Студия "АНИМАККОРД", АНИМАККОРД ЛТД.	Россия, F
3785	Охотники	111017816	2016-02-06T12:00:00.000Z	Художественный	Селиверстов Александр Александрович	
3947	Круиз.	211038210	2010-09-03T12:00:00.000Z	Художественный	ООО "ЮНАЙТЕД МУЛЬТИМЕДИА ПРОДЖЕКТС"	
5331	Добыча.	121009411	2011-06-15T12:00:00.000Z	Художественный	Афтер Дарк Филмз	
5454	Вечный холод	111006417	2017-08-28T12:00:00.000Z	Художественный	ЗАО "Производственный комплекс "ГЛАВКИНО", ООО...	
5820	Свинья	121007018	2018-04-11T12:00:00.000Z	Художественный	Дарк Прекёрсор Продакшен, Филмиран, Хедейт Филм	
5909	Мира	112000118	2018-06-04T12:00:00.000Z	Документальный	ООО "Компания "Новые люди"	
6425	Памятные даты России. Партизаны и подпольщики	112013018	2018-11-28T12:00:00.000Z	Документальный	ООО Кинокомпания "Вектор"	
6615	ОТ ВОЙНЫ ДО НАШИХ ДНЕЙ	111026518	2019-01-21T12:00:00.000Z	Художественный	ИП Вайсман Анатолий Александрович	

	title	puNumber	show_start_date	type	film_studio	productic
6736	Колесо времени	111003219	2019-03- 01T12:00:00.000Z	Художественный	ООО "БестМедиа"	
6881	Животные рядом со Святыми	112003719	2019-05- 03T12:00:00.000Z	Документальный	ООО "Кинопрограмма "XXI век"	
7158	Анна Каренина. Интимный дневник	111013919	2019-10- 23T12:00:00.000Z	Художественный	ООО "РТВ"	
7276	Добыча	121028319	2019-10- 03T12:00:00.000Z	Художественный	Блумхаус Продакшнз, Хайд Парк Интертейнмент, Т...	

Никакой связи между цифрами со знаком процента и самими фильмами нет(я проверил рейтинг нескольких фильмов на кинопоиске), можем преобразовать это значение в 0

Напишем функцию, которая принимает строку на вход и возвращает 0, если в строке был обнаружен знак '%', в противном случае возвращает просто саму строку:

```
In [36]: def percent_in_string(string):
         if isinstance(string, int):
             return string
         if string.find('%') > 0:
             return 0
         else:
             return string
```

Применим ее к столбцу

```
In [37]: data['ratings'] = data['ratings'].apply(percent_in_string)
```

Теперь нам ничто не мешает преобразовать столбец в числовой:

```
In [38]: data['ratings'] = pd.to_numeric(data['ratings'])
```

```
In [39]: data['ratings'].dtypes
```

```
Out[39]: dtype('float64')
```

2.1.4 Столбец 'genres'

Посмотрим на пропущенные значения:

```
In [40]: data[data['genres'].isna()]
```

Out[40]:

	title	puNumber	show_start_date	type	film_studio	product
--	-------	----------	-----------------	------	-------------	---------

43	Мемуары гейши (по роману Артура Голдена)	121000606	2012-12-14T12:00:00.000Z	Художественный	Коламбия Пикчерз Корпорейшн, ДримУоркс ЭскейДж...	
150	Жди меня	111013615	2015-08-25T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
154	Дом из песка и тумана (по книге Андре Дюбуи III)	221037813	2013-10-25T12:00:00.000Z	Художественный	Кобальт Пикчерз	
159	Жемчужная девушка	224011014	2014-08-28T12:00:00.000Z	Художественный	Чешское телевидение,Студия "Три брата" агентск...	
227	Полустанок	111026714	2014-12-24T12:00:00.000Z	Художественный	Киностудия "Мосфильм"	
...
7478	Эколав	111023219	2019-12-26T12:00:00.000Z	Художественный	ВГИК	
7479	Акрам Хан: Родом из Карри Хауса	126007919	2019-12-23T12:00:00.000Z	Прочие	Свон Филмз	Вели
7480	Сад художника: Американский импрессионизм	126008019	2019-12-23T12:00:00.000Z	Прочие	Севен Артс Продакшнз	Вели
7482	Щенячий патруль: Скорей спешим на помощь!	124010819	2019-12-28T12:00:00.000Z	Художественный	Никелодеон Анимейшн Студиос, Спин Мастер	С
7483	Машины песенки. Серия "Испанские мотивы"	124010619	2019-12-30T12:00:00.000Z	Анимационный	Анимаккорд Лтд., ООО Студия "АНИМАККОРД"	Респу

975 rows × 16 columns

Наиболее часто появляющееся значение в столбце 'genres':

In [41]: data.mode()['genres']

```
Out[41]: 0    драма
         1     NaN
         Name: genres, dtype: object
```

Заполним пропуски наиболее повторяющимся значением(драма):

```
In [42]: most = data.mode()['genres'][0]
         data.fillna({'genres': most}, inplace=True)
```

```
In [43]: data['genres'].isna().sum()
```

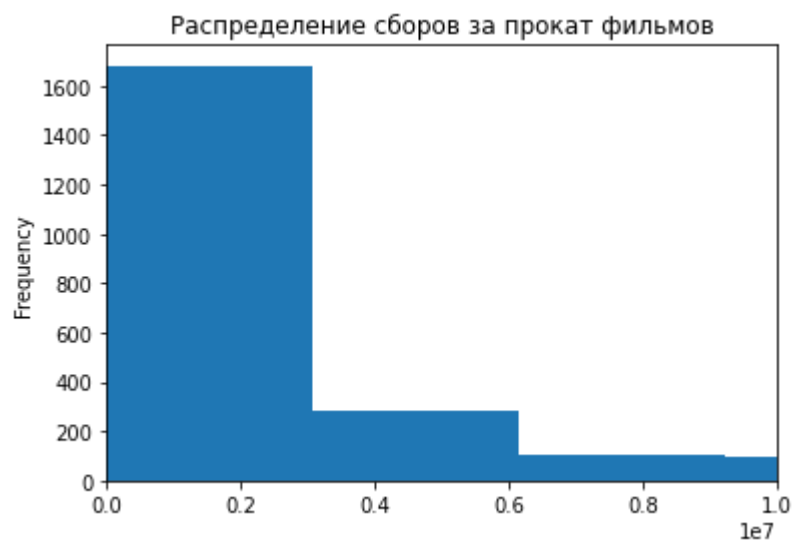
```
Out[43]: 0
```

Пропуски отсутствуют

2.1.5 Столбец 'box_office'

Посмотрим на данные поближе:

```
In [44]: data['box_office'].plot(kind='hist', bins=1000, xlim=(0, 1e7), title='Распределение
```



Узнаем кол-во пропусков:

```
In [45]: data['box_office'].isna().sum()
```

```
Out[45]: 4327
```

Для преобразования столбца 'box_office' в int (копейки не используются), необходимо заменить пропуски значением, которое бы было int и не влияло на дальнейший анализ, примем за это значение -1:

```
In [46]: data['box_office'] = data['box_office'].fillna(-1)
```

```
In [47]: data['box_office'].isna().sum()
```

Out[47]: 0

Конвертируем в int

```
In [48]: data['box_office'] = data['box_office'].astype(int)
```

```
In [49]: data.dtypes
```

```
Out[49]: title                object
puNumber                    int64
show_start_date            object
type                       object
film_studio                object
production_country         object
director                   object
producer                   object
age_restriction            object
refundable_support         int64
nonrefundable_support     int64
budget                     int64
financing_source           object
ratings                    float64
genres                     object
box_office                 int64
dtype: object
```

Посмотрим есть ли еще пропуски

```
In [50]: pd.DataFrame(round(data.isna().mean()*100)).style.background_gradient('coolwarm')
```

Out[50]:

0

title	0.000000
puNumber	0.000000
show_start_date	0.000000
type	0.000000
film_studio	0.000000
production_country	0.000000
director	0.000000
producer	0.000000
age_restriction	0.000000
refundable_support	0.000000
nonrefundable_support	0.000000
budget	0.000000
financing_source	0.000000
ratings	0.000000
genres	0.000000
box_office	0.000000

2.2. Подготовка данных для анализа

2.2.1 Конвертация столбца 'show_start_date'

```
In [51]: data['show_start_date'][0:5]
```

```
Out[51]: 0    2015-11-27T12:00:00.000Z
          1    2016-09-13T12:00:00.000Z
          2    2016-10-10T12:00:00.000Z
          3    2016-06-10T12:00:00.000Z
          4    2015-07-29T12:00:00.000Z
          Name: show_start_date, dtype: object
```

```
In [52]: data['show_start_date'] = pd.to_datetime(data['show_start_date'], format='%Y-%m-%d')
```

```
In [53]: data['show_start_date'].dtypes
```

```
Out[53]: datetime64[ns, UTC]
```

```
In [54]: data.head()
```

Out[54]:

	title	puNumber	show_start_date	type	film_studio	production_country	
0	Открытый простор	221048915	2015-11-27 12:00:00+00:00	Художественный	Тачстоун Пикчерз, Кобальт Пикчерз, Бикон Пикче...	США	
1	Особо важное задание	111013716	2016-09-13 12:00:00+00:00	Художественный	Киностудия "Мосфильм"	СССР	
2	Особо опасен	221038416	2016-10-10 12:00:00+00:00	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	США	Бел
3	Особо опасен	221026916	2016-06-10 12:00:00+00:00	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	США	Бел
4	Особо опасен	221030815	2015-07-29 12:00:00+00:00	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	США	Бел

2.2.2 Столбец 'type'

Посмотрим какие уникальные значения входят в столбец

```
In [55]: data['type'].unique()
```

```
Out[55]: array(['Художественный', 'Анимационный', 'Прочие', 'Документальный',  
                'Научно-популярный', 'Художественный', 'Анимационный',  
                'Музыкально-развлекательный'], dtype=object)
```

Мы видим, что некоторые категории повторяются и разница в повторяющихся - пробелы, уберем их:

```
In [56]: data['type'] = data['type'].str.strip()
```

```
In [57]: data['type'].unique()
```

```
Out[57]: array(['Художественный', 'Анимационный', 'Прочие', 'Документальный',  
              'Научно-популярный', 'Музыкально-развлекательный'], dtype=object)
```

Повторяющихся категорий нет

2.2.3 Столбец 'production_country'

Проверим на пропуски

```
In [58]: data['production_country'].isna().sum()
```

```
Out[58]: 2
```

```
In [59]: data[data['production_country'].isna()]
```

```
Out[59]:
```

	title	puNumber	show_start_date	type	film_studio	production
3185	Детский юмористический киножурнал "Ералаш. Ну ...	111001216	2016-02-09 12:00:00+00:00	Художественный	ООО "Продюсерский центр ЕРАЛАШ"	
4440	Мульт личности. Выпуск 5	214000410	2010-01-25 12:00:00+00:00	Художественный	NaN	

Заменяем пропущенные значения на 'Россия', оба мультфильма произведены в России:

```
In [60]: data['production_country'] = data['production_country'].fillna('Россия')
```

```
In [61]: data['production_country'].isna().sum()
```

```
Out[61]: 0
```

```
In [62]: len(data['production_country'].unique())
```

```
Out[62]: 950
```

```
In [63]: data['production_country'] = data['production_country'].str.strip()
```

```
In [64]: len(data['production_country'].unique())
```

```
Out[64]: 934
```

```
In [65]: data['production_country'].unique()
```

```
Out[65]: array(['США', 'СССР', 'Франция', 'СССР, Венгрия',  
              'Германия-Великобритания', 'Великобритания - Италия',  
              'Чехословакия', 'США - Франция - Турция', 'Новая Зеландия',  
              'Канада - Франция - Испания', 'США-Германия',  
              'США - Великобритания', 'Великобритания', 'США - Германия',  
              'Франция - Мексика - США', 'Россия, Казахстан, США',  
              'СССР, Швеция', 'СССР, Франция, Англия, Куба, ГДР', 'Германия',  
              'Великобритания-США-Германия-КНР',  
              'СССР, ЧССР, Западный Берлин, ПНР', 'СССР, Италия', 'Гонконг, КНР',  
              'США - Франция', 'США - Япония - Франция - Великобритания',  
              'Гонконг - Сингапур - Таиланд - Великобритания', 'США-Канада',  
              'Франция - Италия - Великобритания - США', 'Франция - США',  
              'Ирландия-Великобритания-Германия', 'Чехия', 'США-Австралия',  
              'СССР, Финляндия', 'США-Франция-Великобритания-Австрия',  
              'США - Бельгия', 'США - Ирландия - Великобритания',  
              'Великобритания - США',  
              'Люксембург - Нидерланды - Испания - Великобритания - США - Италия',  
              'Великобритания - Франция - США', 'Новая Зеландия - США',  
              'США - Великобритания - Чехия',  
              'Канада - Франция - Великобритания', 'Ирландия',  
              'Великобритания - Германия - США',  
              'США - Франция - Великобритания', 'Япония', 'СССР, Польша',  
              'Франция - Испания', 'Канада-Франция', 'Германия - Италия - США',  
              'Аргентина - Бразилия', 'Дания - Швейцария - Бельгия - Франция',  
              'США - Испания', 'Германия - США',  
              'Франция - Австрия - Германия - Италия - США', 'Австрия',  
              'США - Канада - Германия', 'США - Канада',  
              'СССР - Швейцария - Франция', 'Россия - Казахстан', 'Гонконг',  
              'Италия', 'США - ОАЭ', 'Россия',  
              'США - Франция - Испания - Великобритания',  
              'Великобритания - Франция', 'США-Италия-Испания-Аргентина',  
              'Франция-Великобритания', 'КНР - Гонконг',  
              'Франция - Италия - Швейцария', 'США - Новая Зеландия',  
              'Франция - Бельгия', 'КНР', 'Германия - Великобритания - Австрия',  
              'Великобритания - Израиль - Франция - Япония - США', 'Швеция',  
              'Испания', 'Дания', 'Франция - Бельгия - Люксембург',  
              'США - Индия - ОАЭ',  
              'Германия-Канада-Великобритания-Швейцария-Франция',  
              'Мексика - Чили', 'Аргентина', 'Бразилия', 'Чили', 'Австралия',  
              'Португалия - Франция', 'США - ЮАР', 'Исландия-Ирландия-Германия',  
              'США-Индонезия', 'Южная Корея', 'Мексика - Испания - Дания - США',  
              'Великобритания-США', 'Франция - Италия', 'США-Франция',  
              'США - Венгрия - Великобритания', 'Бельгия-Франция-Италия',  
              'Чехия - Германия', 'Германия - Франция', 'США-Нидерланды',  
              'Украина', 'Великобритания - Германия', 'Франция-Бельгия',  
              'США-Франция-Испания', 'Италия - Франция', 'Канада',  
              'Швеция-Франция-Дания', 'Германия-Франция-Польша',  
              'Франция-Великобритания - Германия',  
              'США - Бразилия - Франция - Австралия - Великобритания - Германия',  
              'Германия-Австрия', 'Германия-Италия- Шри-Ланка',  
              'Великобритания - Испания',  
              'США - Великобритания - Канада - Швеция',  
              'Босния и Герцеговина - Франция - Великобритания - Германия - Словения - Бе  
льгия - Сербия',  
              'Великобритания - Франция - Италия - США', 'США-ОАЭ',  
              'Испания - Франция', 'США-Великобритания', 'США-ЮАР',
```


'Великобритания-Австралия - США',
'Норвегия - Швеция - Дания - Германия', 'Норвегия', 'Швейцария',
'Греция - Германия - Франция', 'Норвегия', 'Южная Корея - США',
'Чили - Франция - США', 'Австрия - Германия - Франция',
'Швеция - Дания', 'Великобритания - Франция - Бельгия - Италия',
'Франция - Германия', 'Грузия - Франция - Люксембург',
'США - Франция - Великобритания - Бразилия', 'США - КНР',
'Мексика - Франция', 'США - Германия - Гонконг - Сингапур',
'Финляндия - Австрия - Россия', 'Испания - Куба',
'Великобритания - ЮАР', 'Великобритания - Ирландия', 'Израиль',
'Бразилия - Германия - Португалия - Франция',
'Франция - Канада - Бельгия', 'Франция-Испания',
'Испания - Канада - Япония', 'Иран', 'Швеция-Куба',
'США-Великобритания-Ирландия', 'Гонконг - КНР', 'Нидерланды',
'Испания - Канада', 'Япония-Великобритания',
'Великобритания - Норвегия - Дания - Германия',
'Финляндия - Франция - Германия',
'Мексика - Франция - Нидерланды - Германия', 'США - Индия',
'США-Великобритания-Индия', 'США-Канада-Бельгия',
'Республика Узбекистан', 'Турция',
'Сербия - Словения - Хорватия - Черногория - Македония',
'Киргизия', 'Германия - Нидерланды - Беларусь - Россия - Латвия',
'Индия', 'Нидерланды - Россия - Германия',
'Великобритания - Нидерланды - Франция - Хорватия',
'Канада - Южная Корея',
'Великобритания - Польша - КНР - Индия - Норвегия',
'Япония - Франция - США - Южная Корея - Турция',
'Мексика - Аргентина - Великобритания', 'Израиль - Франция',
'Польша - Португалия - Франция - Великобритания',
'Австралия - США', 'Франция - Люксембург - Бельгия', 'Армения',
'Бельгия - Люксембург - Франция - Швейцария',
'США - Великобритания - Канада', 'Норвегия - Швеция',
'Канада - Франция', 'КНР - Гонконг - США',
'США - Сингапур - Малайзия - Индонезия', 'Германия - Россия',
'Япония - Канада', 'Финляндия - Дания - Германия - Ирландия',
'Франция - Люксембург - Германия - Бельгия - Швейцария - Великобритания',
'Канада - США', 'США-Италия-Испания', 'США-Испания',
'США - Швейцария - Франция', 'Испания - США - Колумбия',
'Канада-ЮАР', 'Франция - Россия',
'Германия - Финляндия - Австралия', 'Дания - Швеция', 'Бельгия',
'Германия - Франция - Австрия', 'Франция-Индия', 'ЮАР',
'Франция-Перу', 'Франция - Канада', 'Республика Беларусь',
'Великобритания - Австрия - Франция - США',
'Австрия-Германия-Франция', 'США-Россия', 'Германия-США',
'США - Южная Корея', 'Канада - КНР',
'Дания - Швеция - Италия - Франция - Германия',
'Франция-Италия-Испания-Венгрия', 'Франция - Германия - Бельгия',
'Франция-Великобритания-Чехия', 'Румыния - Франция - Бельгия',
'Болгария - Франция - Израиль',
'Израиль - Германия - Польша - Бельгия - Франция - Люксембург',
'Россия, Германия, Франция, Бельгия', 'Индонезия',
'Франция - Португалия', 'Великобритания - Пуэрто-Рико',
'СССР, ФРГ, Западный Берлин', 'Россия, Республика Беларусь',
'США, Великобритания', 'Франция - Австрия', 'Бельгия-Нидерланды',
'Великобритания - США - Индия', 'США - Великобритания - Австралия',
'Канада - Франция - США - ОАЭ - Великобритания',

'США - Австралия - Франция',
 'Дания-Франция-Бельгия-Германия-Великобритания',
 'Испания - Аргентина', 'Исландия - Финляндия',
 'Россия-Польша-Голландия-Словакия', 'Дания - Германия - Швеция',
 'Южная Корея - США - Канада', 'Таиланд', 'США-Мальта',
 'США -Германия',
 'Великобритания - Германия - Франция - Кипр - США', 'Мексика',
 'США - Великобритания - Ирландия', 'США - Украина',
 'США - Финляндия - Испания - Великобритания - Франция',
 'Австралия-Великобритания',
 'США - Канада - Афганистан - Бельгия - Франция',
 'США - Великобритания - Германия - Новая Зеландия - Бельгия - Франция',
 'Франция-Канада',
 'Ирландия - Финляндия - Бельгия - Великобритания - США - Швейцария',
 'Малайзия - США', 'Германия - Италия',
 'США - Франция - Бельгия - Италия', 'Франция - Ирландия - Швеция',
 'Азербайджан-Россия', 'Великобритания - Италия - Швейцария',
 'США - Германия - Великобритания',
 'Германия - Бельгия - США -Франция', 'США-Китай', 'Чили - Испания',
 'Франция - Швейцария', 'Канада - Испания',
 'США - Румыния - Великобритания', 'США - Япония',
 'США - Россия - Франция', 'Франция-США', 'Парагвай',
 'Германия - Канада', 'США - Канада - Великобритания',
 'Финляндия - Канада', 'Уругвай-Мексика-Германия', 'США - Чили',
 'Германия - Аргентина - Нидерланды - Чили',
 'США - Канада - Индонезия', 'Армения - Казахстан', 'Финляндия',
 'Испания - Италия - Франция',
 'Бельгия - Испания - Канада - Нидерланды',
 'Великобритания - Франция - Швеция - Бельгия',
 'Мексика - Эквадор - Канада - США - Франция - Малайзия - Италия - Аргентина
 - Германия - Индия',
 'Франция - Бельгия', 'Индия - Великобритания',
 'Япония - Великобритания - Франция',
 'Испания - Уругвай - Аргентина',
 'Франция - Италия - Бельгия - Люксембург',
 'Франция - Украина - Грузия - Армения',
 'Сербия - Германия - Венгрия', 'Франция-Польша', 'Румыния-США',
 'Россия-Азербайджан',
 'США - Нидерланды - Бельгия - Венгрия - Греция - Канада',
 'Германия - Франция - Люксембург', 'США - Россия',
 'Великобритания - США - Франция', 'США - Австралия - Дания',
 'Бельгия-Франция', 'США-Мексика',
 'Норвегия - Азербайджан - Россия - Колумбия - Великобритания - Венгрия - Ру
 мыния - Франция - Грузия',
 'США', 'Бразилия - Испания', 'Россия, Франция, Латвия',
 'Нидерланды - США - Германия - Канада - Франция - Ирландия - Великобритани
 я',
 'Уругвай - Колумбия', 'США - Испания - Болгария',
 'Армения - Германия', 'США - Великобритания- Франция - Швеция',
 'США - Мексика', 'Австралия-Турция-США', 'Швеция - Германия',
 'Польша - Италия - Россия', 'Франция - Бразилия',
 'США - Канада - КНР', 'СССР, Венгрия, ЧССР, ГДР',
 'Бельгия - Германия - Люксембург', 'США - Австралия',
 'Бразилия - США', 'США - Великобритания - Канада - КНР',
 'Франция - Чили',
 'Франция - США - Великобритания - Колумбия - Бельгия - Россия',

'Дания - Великобритания - ЮАР', 'Россия-США-Канада-Люксембург',
 'Россия - Республика Кипр', 'Германия - США - Швеция - Франция',
 'Великобритания-Германия-Канада-ЮАР', 'Ирландия, Канада',
 'Япония - Великобритания - Швейцария - Ирландия - Дания - Франция - Польша
 - Австралия - Канада',
 'Великобритания, Канада', 'Бельгия - США',
 'Великобритания - США - Россия', 'Россия-Украина',
 'Германия - США - Великобритания - Израиль', 'Канада - Германия',
 'США- Ю.Корея', 'Индия - Мексика',
 'Великобритания - Франция - Республика Кипр - Швейцария - США - Сербия',
 'Испания - Аргентина - Индия - США',
 'Норвегия - Исландия - США - Великобритания',
 'Турция - Германия - Франция', 'Германия - Великобритания',
 'Австралия - Германия', 'Мексика - Испания',
 'Великобритания - США - Германия - Бельгия',
 'США - Чехия - Франция', 'Россия - Италия', 'Франция-Россия',
 'Норвегия - Дания - Швеция', 'Франция - Швейцария - Германия',
 'Грузия - Украина', 'Россия - Эстония - Финляндия - Беларусь',
 'США - Франция - ЮАР', 'Великобритания - США - Швейцария',
 'Канада-США', 'Мексика - Нидерланды - Германия - Франция',
 'США - Испания - Франция', 'США - Великобритания - КНР',
 'США-Франция-Канада',
 'Япония, Великобритания, Австрия, Германия, Ю.Корея',
 'Великобритания - Ирландия - США', 'Бельгия - Италия - Франция',
 'Швейцария-Германия-ЮАР', 'Великобритания - США - Германия',
 'Германия - Исландия', 'Испания - США', 'Хорватия',
 'Финляндия - Исландия - Швеция',
 'Канада - США - Германия - Франция',
 'США - Великобритания - Испания', 'Ирландия-Великобритания',
 'Аргентина - Испания', 'Австрия - Германия',
 'Германия - Швейцария - Франция - Южная Корея - США',
 'Украина - Нидерланды', 'Венгрия - Германия - Швеция',
 'Германия - Франция - Польша - Турция - Канада - Италия - Россия',
 'СССР, Афганистан', 'СССР, Монголия', 'Россия-Кипр',
 'Россия, Грузия, Испания',
 'Великобритания - Нидерланды - Германия - Франция - Бельгия - Австрия',
 'Испания - Великобритания - Франция',
 'Дания - Норвегия - Великобритания',
 'Бельгия - Франция - Люксембург',
 'Италия - Франция - Швейцария - Великобритания',
 'Франция - Бельгия - Великобритания - Испания - Германия - США',
 'Швейцария - Израиль - Франция - Великобритания',
 'Великобритания - США - Канада - Нидерланды - Франция',
 'Россия, Италия', 'СССР, Албания',
 'Россия, Украина, Республика Беларусь, Литва',
 'Финляндия - Великобритания - Германия', 'Россия, Украина, Польша',
 'Франция - Италия - Бельгия - КНР',
 'Германия - Бельгия - Люксембург - Ирландия', 'КНР - Сингапур',
 'Франция - Бельгия - Великобритания - США - Нидерланды - Канада',
 'США - Россия - Польша - Германия - Пуэрто Рико',
 'Франция - Бельгия - Канада', 'Мексика - США',
 'США - Гонконг - КНР', 'Греция - Россия',
 'Ирландия - Дания - Бельгия - Люксембург - Франция',
 'Эстония - Россия',
 'Великобритания - Исландия - Испания - Швейцария - США',
 'Австралия - Великобритания - Бельгия - Индия',

'Швеция - Германия - Франция - Норвегия',
 'Нидерланды - Бельгия - Германия - Ирландия',
 'Нидерланды - Великобритания - Бельгия',
 'США - Колумбия - Испания', 'Франция-Испания-Бельгия-Панама',
 'США-Великобритания-Чехия-Румыния',
 'Аргентина - Уругвай - Россия - Германия - Франция - Нидерланды',
 'Швеция - Франция - Норвегия - Дания',
 'Италия-Франция-Великобритания', 'Великобритания, Франция',
 'Австралия, США', 'Венесуэла',
 'Великобритания - США - Австралия - Ирландия - Германия - Куба - Канада',
 'Южная Корея - КНР',
 'Аргентина - Мексика - Бразилия - Франция - США', 'Швеция - США',
 'Франция - Германия - Турция - Катар', 'Франция-Китай',
 'Россия, Германия, Великобритания',
 'Великобритания - Италия - Испания', 'Италия - Россия',
 'Япония - КНР - Южная Корея', 'Россия, Германия',
 'Россия-Ирландия', 'Македония - Франция - Великобритания',
 'Франция - Бразилия - Италия', 'Россия-США', 'Армения-Россия',
 'Россия, Румыния', 'Франция - Бельгия', 'Грузия-Франция',
 'ЮАР - США', 'Великобритания-США-Франция',
 'Великобритания - США - Германия - Канада - Австралия',
 'США-Ирландия', 'Россия, Республика Кипр',
 'Германия - Люксембург - Франция', 'Россия-Франция',
 'США - Нидерланды - Финляндия - Великобритания - Италия',
 'Республика Кипр', 'Венгрия', 'Франция - США - Норвегия - Дания',
 'Германия - Австрия - Ирландия', 'США - Финляндия',
 'Италия-Великобритания', 'Россия, Испания',
 'США - Великобритания - Болгария', 'Великобритания - Люксембург',
 'Индия - США', 'США - Канада - Россия - Франция - Чили - Ирландия',
 'Грузия-Россия',
 'Дания - Австрия - Ирландия - Финляндия - Норвегия - Швеция - Нидерланды',
 'США-КНР', 'Грузия', 'Австралия - Ирландия',
 'Великобритания - Канада', 'Италия - Аргентина - Словения',
 'Испания - Франция - Нидерланды - Германия - Бельгия - Великобритания - Канада',
 'Франция - Чехия - Бельгия',
 'Великобритания - Франция - Германия - США',
 'Ирландия - Великобритания - Франция - США - Германия - Нидерланды',
 'США - Канада - Япония - КНР', 'Италия-Франция',
 'США - Великобритания - Исландия', 'Великобритания-Франция',
 'Франция - Германия - Нидерланды',
 'Франция - Германия - Швеция - США - Чехия - Словакия - Великобритания - Нидерланды',
 'Франция - Канада - Н.Зеландия - США - Нидерланды - Германия - Швеция - Россия',
 'Турция - США',
 'Великобритания - США - Канада - Бельгия - Нидерланды - Австрия - Германия',
 'Израиль - Франция - Германия - Палестина - США - Австрия - Великобритания',
 'Аргентина - Франция - Испания', 'СССР, Россия',
 'Великобритания - Франция - Италия - Индия - Дания - КНР - Бангладеш - Камбоджа - Гаити - Уганда',
 'Италия - Франция - Швейцария',
 'Ирландия - Великобритания - Канада', 'Перу', 'США - Норвегия',
 'Испания-Канада', 'Китай-Гонконг', 'США - Германия - КНР',

'Великобритания - Новая Зеландия',
 'Франция - Дания - Швеция - КНР',
 'США - Великобритания - Германия - Бельгия - Дания',
 'США - Франция - Канада - Германия - Австралия - Индия',
 'СССР, Румыния, Франция', 'СССР, ГДР', 'Австралия-Ирландия',
 'США - Китай', 'Великобритания - Франция - Австрия - Швеция',
 'Россия, Таджикистан', 'Нидерланды - Бельгия - Болгария',
 'Великобритания - Канада - США', 'Франция - Бельгия - Испания',
 'Китай-Гонконг-США', 'Франция - Великобритания', 'США - Вьетнам',
 'США - Германия - Франция', 'Германия - Франция - Бельгия',
 'США - Италия',
 'Германия - Дания - Испания - Швеция - Канада - Эстония - Франция',
 'Княжество Андорра - Украина', 'Германия - Люксембург - Бельгия',
 'Великобритания - Россия - Украина - США',
 'Германия - Бельгия - Люксембург - Норвегия',
 'США - Австралия - Новая Зеландия - Великобритания',
 'США - Франция - Канада - Великобритания', 'Испания-Колумбия',
 'Великобритания - Чехия - Франция', 'Россия, Украина', 'Исландия',
 'Великобритания - Франция - Венгрия',
 'Россия, Франция, Германия, Бельгия', 'Франция - Македония',
 'Нидерланды - Франция - Германия - Бельгия - Швеция - Великобритания',
 'Испания - США - Великобритания - Канада', 'Германия - Австрия',
 'Республика Казахстан',
 'Франция - Испания - Дания - Венгрия - Швейцария', 'Франция-КНР',
 'США - Пуэрто-Рико - Франция', 'Мексика - Тайвань - США',
 'Великобритания - Испания - Непал - Индия - Венгрия - Гонконг - Германия -
 Дания - Бахрейн',
 'Великобритания - Аргентина - США - Франция - Россия - Испания',
 'КНР - США', 'Италия - США', 'Финляндия - Швеция - Германия',
 'Канада-Испания', 'Франция - Бельгия - Чехия',
 'Франция-Бельгия-Канада', 'Франция - Германия - США',
 'США - Дания', 'Индия - КНР', 'Чехия-Великобритания-США',
 'США-ФРГ-Россия', 'Ирландия-Нидерланды-Франция-США-Великобритания',
 'Россия, Грузия, Хорватия, Испания',
 'Финляндия - Швеция - Норвегия', 'Сербия - Великобритания - США',
 'США - Великобритания - Франция - Гонконг', 'Швейцария - Франция',
 'Швейцария - Великобритания', 'ОАЭ - США',
 'США - Франция - Германия - Канада - Австралия',
 'Германия - Австралия', 'Иран, Франция',
 'США - Великобритания - Новая Зеландия', 'США-Колумбия',
 'США-Франция-Ирландия', 'Великобритания - Франция - Бельгия',
 'США - Великобритания - Япония', 'США - Франция - Германия',
 'Польша - Ирландия', 'Ирландия - Великобритания - Франция - США',
 'США - Бельгия - Великобритания', 'СССР, ВНР',
 'Испания - Франция - США - Великобритания - Бельгия',
 'Франция - Дания - США', 'США - Гонконг', 'Бразилия - Канада',
 'Франция-Германия', 'США - Колумбия',
 'США - Ирландия - Великобритания - Франция',
 'Великобритания - США - Канада', 'Польша - Франция',
 'Канада - США - Нидерланды - Великобритания - Аргентина',
 'Великобритания - Дания', 'Франция - Бельгия - Япония',
 'Россия, Испания', 'Франция - Бельгия',
 'США - Канада - Германия - Франция', 'США-Канада-Франция',
 'Россия - Германия', 'Россия-Италия', 'Китай - Гонконг',
 'Германия-Австрия-Франция-Италия', 'США-Япония',
 'США - Германия - Канада', 'Испания-Мексика',

'США - Япония - Канада - Великобритания - Германия - Франция',
'США -Великобритания - Австралия', 'Россия-Украина-Германия',
'Израиль - Франция - Великобритания - Германия', 'Корея',
'Россия - Болгария', 'США-Индия',
'Германия-Канада-Франция-Бельгия', 'Германия-Кения',
'Франция - Монако', 'Франция-Испания-Швейцария',
'Австрия-Франция-Германия', 'Румыния',
'США-Великобритания-Франция', 'Канада-Великобритания',
'Канада-Бразилия-Япония', 'Германия - Дания - Норвегия',
'Дания - Германия', 'СССР - Италия',
'СССР - ГДР - Польша - Италия',
'США - Испания - Франция - Великобритания', 'Китай',
'Мексика-Аргентина',
'Германия - Великобритания - Италия - Испания',
'Франция-Германия-Великобритания', 'Россия - Германия - Швейцария',
'Франция-Австралия', 'Голландия', 'Швейцария-Франция',
'Россия-Белоруссия', 'Франция - Испания - Тайвань',
'США-Южная Корея-Новая Зеландия', 'Великобритания-Венгрия',
'Россия-Германия', 'Великобритания-Франция-Германия',
'Франция-Германия-Италия', 'Швеция-Финляндия-Франция-Норвегия',
'Франция-Италия', 'Германия-Израиль',
'Швеция - Германия - Дания - Норвегия', 'США-Бельгия',
'Франция-Гонконг-Ирландия', 'Эстония', 'Литва-Россия-Украина',
'Германия - Италия - Франция',
'Великобритания - Австрия - Германия', 'Бельгия - Франция',
'Россия-Казахстан', 'Испания - Мальта',
'Франция-Россия-Румыния-Италия-Бельгия', 'Россия - Польша',
'Испания - Мексика', 'США-Аргентина', 'Израиль-США-Франция',
'США - Новая Зеландия - Япония',
'Франция - Чехия - Великобритания', 'Россия-Польша',
'Германия-Франция-Великобритания', 'США-Германия-Канада',
'США-Гонконг-Китай', 'США - Великобритания - Германия',
'США-Канада-Австралия', 'Япония-США-Франция',
'Франция-Испания-Германия', 'США - Хорватия - Босния-Герцеговина',
'Великобритания - Чехия - Франция - Италия',
'США - Германия - Нидерланды', 'Германия - Франция - Канада',
'Франция-Грузия-Германия-Россия-Украина-Бельгия',
'Великобритания - Франция - Италия', 'Россия - США',
'Франция - Италия - Иран', 'Россия - Азербайджан', 'Греция',
'Дания - США', 'США-Великобритания-Новая Зеландия',
'США-Германия-Австралия', 'США-Канада-Италия',
'США - Япония - Германия', 'Австралия-США',
'Франция - Великобритания - Багамские острова - США',
'Швейцария - Люксембург', 'Швейцария - Австрия',
'Германия - Швейцария', 'Дания-Швеция-Франция-Германия', 'Польша',
'Швейцария-Австрия', 'Бразилия - Канада - США',
'Нидерланды - Россия', 'Португалия', 'Франция-Аргентина',
'Швеция-Франция', 'Франция-Украина', 'Германия-Украина-Нидерланды',
'Италия-Швейцария', 'США-Германия-Великобритания',
'Латвия - Россия', 'Германия-Франция-Великобритания-США',
'Канада - Италия', 'США - Объединенные Арабские Эмираты',
'США-Венгрия', 'Малайзия', 'Россия - Германия - Украина',
'США - Австралия - Мексика', 'Литва - Венгрия', 'Германия - Китай',
'Казахстан', 'Республика Корея',
'США - Германия - Франция - Великобритания - Канада',
'Бельгия - Германия - Люксембург - Франция', 'Франция - Греция',

'Италия - Канада', 'США - Великобритания - Франция',
 'США-Италия-Греция', 'Германия - США - Великобритания - Ирландия',
 'Великобритания - Франция - Германия',
 'Германия - Нидерланды - ЮАР',
 'США-Япония-Германия-Великобритания-Нидерланды',
 'США-Япония-Германия-Великобритания',
 'Германия - Франция - Польша',
 'Таиланд-Великобритания-Франция-Германия-Испания-Нидерланды',
 'Германия-Италия-Швейцария',
 'Дания - Швеция - Великобритания - Франция - Германия - Нидерланды - Норвег
 ия - Финляндия',
 'Франция - Великобритания - Нидерланды - Люксембург',
 'Нидерланды-Великобритания-Франция-Италия-Япония',
 'Финляндия-Дания-Германия-Ирландия', 'Испания-Аргентина',
 'Великобритания - Нидерланды - Франция - США', 'Канада - Мексика',
 'США, Франция', 'Канада-Южная Корея-США',
 'Великобритания - Франция - Бельгия - США', 'КНР - Канада - США',
 'Италия - Франция - Германия',
 'Швеция - Германия - Франция - Дания', 'Франция - США - Бельгия',
 'Украина - Германия - Латвия - Эстония', 'Австрия - Люксембург',
 'Великобритания - КНР', 'Монголия', 'Франция - Дания',
 'Швеция - Дания - Финляндия',
 'Сингапур - Великобритания - Индонезия - Канада - США',
 'США - Германия - Япония', 'Таиланд - США', 'Франция - Нидерланды',
 'Великобритания - Румыния', 'США - Мальта',
 'Великобритания - Германия - Нидерланды - Дания',
 'Франция - Германия - Литва - Нидерланды - Россия',
 'США - ЮАР - Гонконг', 'Австралия - Колумбия',
 'Великобритания - Польша', 'США - Великобритания - Швеция',
 'Франция - Швеция - Дания - Норвегия', 'Болгария - США',
 'СССР, ФРГ', 'Бельгия - Нидерланды - Франция',
 'Финляндия - Германия', 'Болгария',
 'Грузия - Россия - Украина - Хорватия - Германия',
 'Франция - Люксембург', 'Франция - Германия - Австрия',
 'Норвегия - США', 'Германия-Швеция', 'Франция - Германия - Италия',
 'Франция-Германия-Австрия', 'КНР-США',
 'Дания - Латвия - Россия - США', 'Финляндия - Польша',
 'Германия - Бельгия', 'Германия - США - Франция - Испания',
 'Бельгия-Германия-Канада-Франция-США-Великобритания',
 'Нидерланды - Бельгия - Люксембург', 'Канада-Германия',
 'Люксембург - Бельгия - Франция', 'США-Великобритания-Австралия',
 'Великобритания, Новая Зеландия', 'Бельгия, Великобритания, США',
 'Великобритания, Австралия', 'Исландия, Бельгия',
 'Испания - Швейцария - Великобритания - Германия - Новая Зеландия - Нидерла
 нды - Канада',
 'Франция, Бельгия', 'Великобритания, США', 'Киргизия-Россия',
 'Германия, США', 'Великобритания, США, Испания',
 'Италия - Швейцария - Германия', 'Россия, Армения',
 'Франция, Германия', 'Китай, Канада, США', 'Германия, Австралия',
 'КНР, США', 'США, Великобритания, Люксембург',
 'Испания, Ирландия, Бельгия, Франция',
 'Великобритания - Гонконг - Венгрия - США - Ирландия',
 'США - Пуэрто Рико', 'Аргентина - Испания - Уругвай',
 'Израиль - Германия - Франция - Швейцария',
 'Великобритания - Мексика - США', 'Испания - Болгария - США',
 'Россия-Словакия-Чехия',

'Швейцария - КНР - США - Россия - Республика Корея - Великобритания',
'Россия, Нидерланды, Финляндия',
'Испания - Бельгия - Франция - Португалия - Великобритания',
'Бразилия - Португалия - Франция', 'Швеция - Норвегия',
'Франция - Польша - Бельгия', 'США', 'Китай, США',
'Италия, Франция, Германия', 'Германия - Австрия - Франция',
'Чехия - Словакия - Хорватия', 'Россия, Польша, Финляндия',
'Россия, Латвия, Чешская Республика',
'Франция - Великобритания - Камбоджа - США - КНР',
'Чехия - Испания - США', 'Германия - Люксембург - Дания',
'Великобритания - Испания - Германия', 'Дания - Исландия',
'Германия, Франция, Великобритания',
'Великобритания - Испания - Италия - Латвия - Франция - Эстония',
'Дания, Швеция, Великобритания, Франция, Германия, Норвегия, Финляндия, Нидерланды, Италия',
'Польша - Франция - Великобритания',
'Дания, Норвегия, Венгрия, Чехия',
'Дания - Франция - Германия - Швеция',
'Великобритания, Швеция, США',
'Канада, Франция, Италия, Швейцария, США', 'США, Индия',
'США - Великобритания - Германия - Швеция - Канада',
'Италия, Франция', 'Республика Беларусь, Германия, США, Россия',
'Италия-Швейцария-Франция-Германия',
'Россия - Франция - Великобритания - Латвия', 'Россия-Беларусь',
'Франция - Испания - Румыния - США - Бельгия', 'Италия-США',
'Россия, Австрия', 'Ирландия - Великобритания - США',
'Швейцария, Великобритания, Франция, США, Ирландия',
'Мексика, Франция, Германия, Дания, Швеция', 'Нидерланды, Бельгия',
'Швеция, Норвегия, Финляндия, Франция', 'Испания, Франция, Италия',
'Испания-Франция-Греция-Индия-США-Россия-Великобритания',
'Великобритания - Япония - США',
'Дания, Канада, Норвегия, Австралия, США', 'Аргентина, Испания',
'Россия, Германия, Казахстан, Польша, Китай',
'Португалия, Франция, Польша, США', 'США - Канада - Аргентина',
'Ливан-США', 'Япония - США', 'США, Гонконг',
'Франция - Россия - Швейцария - Румыния - Венгрия',
'Аргентина-Испания', 'Россия, Сербия', 'США - Болгария - Мексика',
'США - Великобритания - Швейцария - Франция',
'Таиланд - КНР - США',
'Великобритания, Нидерланды, Польша, Швейцария, Франция',
'Ирландия - США',
'Германия - Франция - Великобритания - Польша - США',
'Великобритания-Франция-Италия', 'США - Австралия - Индия',
'Франция - КНР', 'Великобритания - Израиль',
'Великобритания - Норвегия - Кения - Хорватия',
'США - Бразилия - Великобритания - Канада', 'Германия, Бельгия',
'Австралия, Канада', 'США, Великобритания, Франция',
'Кения, Германия', 'Нидерланды, Бельгия, Люксембург', 'Колумбия',
'Латвия - Франция', 'Ирландия, Великобритания',
'Франция - Израиль - Германия', 'Швеция, Франция, Великобритания',
'Бельгия - Люксембург', 'Германия - Бельгия - Великобритания',
'Испания, Франция, Великобритания, Дания, Бельгия, Германия',
'Румыния, ЮАР, Иран, Франция, Канада, Великобритания',
'Германия, Франция', 'США, Япония',
'США, Великобритания, Ирландия, Люксембург', 'Республика Армения',
'США, Франция, Япония', 'Япония, США', 'Канада, США',


```

'Республика Кипр - Россия',
'Дания - Франция - Италия - Бельгия - Нидерланды',
'Россия - Грузия', 'Италия, Германия', 'Норвегия, Швеция, Дания',
'Республика Кипр, Россия', 'КНР, Индия, Гонконг, США',
'Израиль, Украина', 'США, Россия', 'США - КНР - Индия - Ю.Корея',
'США - Швеция', 'Франция, Канада', 'Россия-Китай',
'Канада, Великобритания', 'США, Германия',
'Канада, США, Каймановы острова',
'Бразилия, Уругвай, Дания, Норвегия, Чили, Швеция',
'Пуэрто-Рико, Великобритания, США',
'Уругвай - Аргентина - Испания',
'США - Великобритания - Италия - Израиль - Сербия - Индия',
'США - Канада - Франция - Индия', 'Великобритания - Россия - КНР',
'Франция, Новая Зеландия', 'Франция, Люксембург, Бельгия',
'Канада, США, Норвегия', 'Италия, Испания',
'Испания - Германия - Нидерланды', 'Бразилия - США - КНР',
'Великобритания, Германия, Дания, США', '2019',
'Италия - Франция - Бразилия - Германия',
'Норвегия - Швеция - Россия', 'Великобритания, Япония, США',
'Сша, Канада', 'Бельгия, Канада',
'Великобритания - Австралия - США', 'США, Канада',
'Франция - Финляндия', 'Колумбия - США',
'Великобритания, Франция, Бельгия', 'Норвегия - Нидерланды',
'Великобритания, Канада, США', 'Япония, Великобритания',
'Италия - Бельгия - Франция', 'Россия, Бельгия, Финляндия',
'США, Украина', 'Франция, Норвегия',
'Великобритания, Южная Корея, Канада, США, Индия, Китай, Япония',
'Канада - Норвегия', 'Республика Кипр-Россия', 'Германия, Россия',
'Польша - Франция - Испания - Бразилия - Швеция',
'Австрия, Великобритания, Германия', 'Франция - Сенегал - Бельгия',
'Франция, Бельгия, Люксембург', 'Финляндия, Латвия', 'Пакистан',
'Германия, Франция, США', 'Германия - Великобритания - США',
'КНР - Франция', 'Дания, Норвегия, Швеция, Исландия',
'Республика Кипр, США, Россия', 'Новая Зеландия - КНР',
'Дания - Канада - Швеция - Франция - Германия - Великобритания - США'],
dtype=object)

```

Мы видим, что есть повторения: разное написание стран, разные разделители, наличие или отсутствие пробелов. Произведем замены всех двойников и получим единое написание:

```

In [66]: repl_list = [(' -', ','), ('-', ','), (' ', ''), ('Чехословакия', 'ЧССР'), ('ПНР',
for rep in repl_list:
    data['production_country'] = data['production_country'].str.replace(rep[0], rep

```

```

In [67]: data['production_country'].unique()

```

```
Out[67]: array(['США', 'СССР', 'Франция', 'СССР,Венгрия', 'ФРГ,Великобритания',  
               'Великобритания,Италия', 'ЧССР', 'США,Франция,Турция',  
               'НоваяЗеландия', 'Канада,Франция,Испания', 'США,ФРГ',  
               'США,Великобритания', 'Великобритания', 'Франция,Мексика,США',  
               'Россия,Казахстан,США', 'СССР,Швеция',  
               'СССР,Франция,Англия,Куба,ГДР', 'ФРГ',  
               'Великобритания,США,ФРГ,Китай', 'СССР,ЧССР,ЗападныйБерлин,Польша',  
               'СССР,Италия', 'Гонконг,Китай', 'США,Франция',  
               'США,Япония,Франция,Великобритания',  
               'Гонконг,Сингапур,Таиланд,Великобритания', 'США,Канада',  
               'Франция,Италия,Великобритания,США', 'Франция,США',  
               'Ирландия,Великобритания,ФРГ', 'Чехия', 'США,Австралия',  
               'СССР,Финляндия', 'США,Франция,Великобритания,Австрия',  
               'США,Бельгия', 'США,Ирландия,Великобритания', 'Великобритания,США',  
               'Люксембург,Нидерланды,Испания,Великобритания,США,Италия',  
               'Великобритания,Франция,США', 'НоваяЗеландия,США',  
               'США,Великобритания,Чехия', 'Канада,Франция,Великобритания',  
               'Ирландия', 'Великобритания,ФРГ,США', 'США,Франция,Великобритания',  
               'Япония', 'СССР,Польша', 'Франция,Испания', 'Канада,Франция',  
               'ФРГ,Италия,США', 'Аргентина,Бразилия',  
               'Дания,Швейцария,Бельгия,Франция', 'США,Испания', 'ФРГ,США',  
               'Франция,Австрия,ФРГ,Италия,США', 'Австрия', 'США,Канада,ФРГ',  
               'СССР,Швейцария,Франция', 'Россия,Казахстан', 'Гонконг', 'Италия',  
               'США,ОАЭ', 'Россия', 'США,Франция,Испания,Великобритания',  
               'Великобритания,Франция', 'США,Италия,Испания,Аргентина',  
               'Франция,Великобритания', 'Китай,Гонконг',  
               'Франция,Италия,Швейцария', 'США,НоваяЗеландия', 'Франция,Бельгия',  
               'Китай', 'ФРГ,Великобритания,Австрия',  
               'Великобритания,Израиль,Франция,Япония,США', 'Швеция', 'Испания',  
               'Дания', 'Франция,Бельгия,Люксембург', 'США,Индия,ОАЭ',  
               'ФРГ,Канада,Великобритания,Швейцария,Франция', 'Мексика,Чили',  
               'Аргентина', 'Бразилия', 'Чили', 'Австралия', 'Португалия,Франция',  
               'США,ЮАР', 'Исландия,Ирландия,ФРГ', 'США,Индонезия', 'ЮжнаяКорея',  
               'Мексика,Испания,Дания,США', 'Франция,Италия',  
               'США,Венгрия,Великобритания', 'Бельгия,Франция,Италия',  
               'Чехия,ФРГ', 'ФРГ,Франция', 'США,Нидерланды', 'Украина',  
               'Великобритания,ФРГ', 'США,Франция,Испания', 'Италия,Франция',  
               'Канада', 'Швеция,Франция,Дания', 'ФРГ,Франция,Польша',  
               'Франция,Великобритания,ФРГ',  
               'США,Бразилия,Франция,Австралия,Великобритания,ФРГ', 'ФРГ,Австрия',  
               'ФРГ,Италия,Шри,Ланка', 'Великобритания,Испания',  
               'США,Великобритания,Канада,Швеция',  
               'БоснияиГерцеговина,Франция,Великобритания,ФРГ,Словения,Бельгия,Сербия',  
               'Великобритания,Франция,Италия,США', 'Испания,Франция',  
               'Великобритания,Австралия,США', 'Норвегия,Швеция,Дания,ФРГ',  
               'Норвегия', 'Швейцария', 'Греция,ФРГ,Франция', 'Норвегия',  
               'ЮжнаяКорея,США', 'Чили,Франция,США', 'Австрия,ФРГ,Франция',  
               'Швеция,Дания', 'Великобритания,Франция,Бельгия,Италия',  
               'Франция,ФРГ', 'Грузия,Франция,Люксембург',  
               'США,Франция,Великобритания,Бразилия', 'США,Китай',  
               'Мексика,Франция', 'США,ФРГ,Гонконг,Сингапур',  
               'Финляндия,Австрия,Россия', 'Испания,Куба', 'Великобритания,ЮАР',  
               'Великобритания,Ирландия', 'Израиль',  
               'Бразилия,ФРГ,Португалия,Франция', 'Франция,Канада,Бельгия',  
               'Испания,Канада,Япония', 'Иран', 'Швеция,Куба',  
               'США,Великобритания,Ирландия', 'Нидерланды', 'Испания,Канада',
```

'Япония, Великобритания', ' Великобритания, Норвегия, Дания, ФРГ',
'Финляндия, Франция, ФРГ', 'Мексика, Франция, Нидерланды, ФРГ',
'США, Индия', 'США, Великобритания, Индия', 'США, Канада, Бельгия',
'Республика Узбекистан', 'Турция',
'Сербия, Словения, Хорватия, Черногория, Македония', 'Киргизия',
'ФРГ, Нидерланды, Беларусь, Россия, Латвия', 'Индия',
'Нидерланды, Россия, ФРГ',
'Великобритания, Нидерланды, Франция, Хорватия', 'Канада, Южная Корея',
'Великобритания, Польша, Китай, Индия, Норвегия',
'Япония, Франция, США, Южная Корея, Турция',
'Мексика, Аргентина, Великобритания', 'Израиль, Франция',
'Польша, Португалия, Франция, Великобритания', 'Австралия, США',
'Франция, Люксембург, Бельгия', 'Армения',
'Бельгия, Люксембург, Франция, Швейцария',
'США, Великобритания, Канада', 'Норвегия, Швеция',
'Китай, Гонконг, США', 'США, Сингапур, Малайзия, Индонезия',
'ФРГ, Россия', 'Япония, Канада', 'Финляндия, Дания, ФРГ, Ирландия',
'Франция, Люксембург, ФРГ, Бельгия, Швейцария, Великобритания',
'Канада, США', 'США, Италия, Испания', 'США, Швейцария, Франция',
'Испания, США, Колумбия', 'Канада, ЮАР', 'Франция, Россия',
'ФРГ, Финляндия, Австралия', 'Дания, Швеция', 'Бельгия',
'ФРГ, Франция, Австрия', 'Франция, Индия', 'ЮАР', 'Франция, Перу',
'Франция, Канада', 'Республика Беларусь',
'Великобритания, Австрия, Франция, США', 'США, Россия',
'США, Южная Корея', 'Канада, Китай',
'Дания, Швеция, Италия, Франция, ФРГ',
'Франция, Италия, Испания, Венгрия', 'Франция, ФРГ, Бельгия',
'Франция, Великобритания, Чехия', 'Румыния, Франция, Бельгия',
'Болгария, Франция, Израиль',
'Израиль, ФРГ, Польша, Бельгия, Франция, Люксембург',
'Россия, ФРГ, Франция, Бельгия', 'Индонезия', 'Франция, Португалия',
'Великобритания, Пуэрто-Рико', 'СССР, ФРГ, Западный Берлин',
'Россия, Республика Беларусь', 'Франция, Австрия',
'Бельгия, Нидерланды', 'Великобритания, США, Индия',
'США, Великобритания, Австралия',
'Канада, Франция, США, ОАЭ, Великобритания', 'США, Австралия, Франция',
'Дания, Франция, Бельгия, ФРГ, Великобритания', 'Испания, Аргентина',
'Исландия, Финляндия', 'Россия, Польша, Голландия, Словакия',
'Дания, ФРГ, Швеция', 'Южная Корея, США, Канада', 'Таиланд',
'США, Мальта', 'Великобритания, ФРГ, Франция, Кипр, США', 'Мексика',
'США, Украина', 'США, Финляндия, Испания, Великобритания, Франция',
'Австралия, Великобритания',
'США, Канада, Афганистан, Бельгия, Франция',
'США, Великобритания, ФРГ, Новая Зеландия, Бельгия, Франция',
'Ирландия, Финляндия, Бельгия, Великобритания, США, Швейцария',
'Малайзия, США', 'ФРГ, Италия', 'США, Франция, Бельгия, Италия',
'Франция, Ирландия, Швеция', 'Азербайджан, Россия',
'Великобритания, Италия, Швейцария', 'США, ФРГ, Великобритания',
'ФРГ, Бельгия, США, Франция', 'Чили, Испания', 'Франция, Швейцария',
'Канада, Испания', 'США, Румыния, Великобритания', 'США, Япония',
'США, Россия, Франция', 'Парагвай', 'ФРГ, Канада',
'США, Канада, Великобритания', 'Финляндия, Канада',
'Уругвай, Мексика, ФРГ', 'США, Чили', 'ФРГ, Аргентина, Нидерланды, Чили',
'США, Канада, Индонезия', 'Армения, Казахстан', 'Финляндия',
'Испания, Италия, Франция', 'Бельгия, Испания, Канада, Нидерланды',
'Великобритания, Франция, Швеция, Бельгия',

'Мексика, Эквадор, Канада, США, Франция, Малайзия, Италия, Аргентина, ФРГ, Индия',
 'Франция, Бельгия', 'Индия, Великобритания',
 'Япония, Великобритания, Франция', 'Испания, Уругвай, Аргентина',
 'Франция, Италия, Бельгия, Люксембург',
 'Франция, Украина, Грузия, Армения', 'Сербия, ФРГ, Венгрия',
 'Франция, Польша', 'Румыния, США', 'Россия, Азербайджан',
 'США, Нидерланды, Бельгия, Венгрия, Греция, Канада',
 'ФРГ, Франция, Люксембург', 'Великобритания, США, Франция',
 'США, Австралия, Дания', 'Бельгия, Франция', 'США, Мексика',
 'Норвегия, Азербайджан, Россия, Колумбия, Великобритания, Венгрия, Румыния, Франци
 я, Грузия',
 'США', 'Бразилия, Испания', 'Россия, Франция, Латвия',
 'Нидерланды, США, ФРГ, Канада, Франция, Ирландия, Великобритания',
 'Уругвай, Колумбия', 'США, Испания, Болгария', 'Армения, ФРГ',
 'США, Великобритания, Франция, Швеция', 'Австралия, Турция, США',
 'Швеция, ФРГ', 'Польша, Италия, Россия', 'Франция, Бразилия',
 'США, Канада, Китай', 'СССР, Венгрия, ЧССР, ГДР',
 'Бельгия, ФРГ, Люксембург', 'Бразилия, США',
 'США, Великобритания, Канада, Китай', 'Франция, Чили',
 'Франция, США, Великобритания, Колумбия, Бельгия, Россия',
 'Дания, Великобритания, ЮАР', 'Россия, США, Канада, Люксембург',
 'Россия, Кипр', 'ФРГ, США, Швеция, Франция',
 'Великобритания, ФРГ, Канада, ЮАР', 'Ирландия, Канада',
 'Япония, Великобритания, Швейцария, Ирландия, Дания, Франция, Польша, Австралия, Ка
 нада',
 'Великобритания, Канада', 'Бельгия, США',
 'Великобритания, США, Россия', 'Россия, Украина',
 'ФРГ, США, Великобритания, Израиль', 'Канада, ФРГ', 'США, Ю. Корея',
 'Индия, Мексика',
 'Великобритания, Франция, Кипр, Швейцария, США, Сербия',
 'Испания, Аргентина, Индия, США',
 'Норвегия, Исландия, США, Великобритания', 'Турция, Германия, Франция',
 'Австралия, ФРГ', 'Мексика, Испания',
 'Великобритания, США, ФРГ, Бельгия', 'США, Чехия, Франция',
 'Россия, Италия', 'Норвегия, Дания, Швеция', 'Франция, Швейцария, ФРГ',
 'Грузия, Украина', 'Россия, Эстония, Финляндия, Беларусь',
 'США, Франция, ЮАР', 'Великобритания, США, Швейцария',
 'Мексика, Нидерланды, ФРГ, Франция', 'США, Испания, Франция',
 'США, Великобритания, Китай', 'США, Франция, Канада',
 'Япония, Великобритания, Австрия, ФРГ, Ю. Корея',
 'Великобритания, Ирландия, США', 'Бельгия, Италия, Франция',
 'Швейцария, ФРГ, ЮАР', 'Великобритания, США, ФРГ', 'ФРГ, Исландия',
 'Испания, США', 'Хорватия', 'Финляндия, Исландия, Швеция',
 'Канада, США, ФРГ, Франция', 'США, Великобритания, Испания',
 'Ирландия, Великобритания', 'Аргентина, Испания', 'Австрия, ФРГ',
 'ФРГ, Швейцария, Франция, Южная Корея, США', 'Украина, Нидерланды',
 'Венгрия, ФРГ, Швеция',
 'ФРГ, Франция, Польша, Турция, Канада, Италия, Россия',
 'СССР, Афганистан', 'СССР, Монголия', 'Россия, Грузия, Испания',
 'Великобритания, Нидерланды, ФРГ, Франция, Бельгия, Австрия',
 'Испания, Великобритания, Франция', 'Дания, Норвегия, Великобритания',
 'Бельгия, Франция, Люксембург',
 'Италия, Франция, Швейцария, Великобритания',
 'Франция, Бельгия, Великобритания, Испания, ФРГ, США',
 'Швейцария, Израиль, Франция, Великобритания',
 'Великобритания, США, Канада, Нидерланды, Франция', 'СССР, Албания',

'Россия, Украина, Республика Беларусь, Литва',
 'Финляндия, Великобритания, ФРГ', 'Россия, Украина, Польша',
 'Франция, Италия, Бельгия, Китай', 'ФРГ, Бельгия, Люксембург, Ирландия',
 'Китай, Сингапур',
 'Франция, Бельгия, Великобритания, США, Нидерланды, Канада',
 'США, Россия, Польша, ФРГ, Пуэрто Рико', 'Франция, Бельгия, Канада',
 'Мексика, США', 'США, Гонконг, Китай', 'Греция, Россия',
 'Ирландия, Дания, Бельгия, Люксембург, Франция', 'Эстония, Россия',
 'Великобритания, Исландия, Испания, Швейцария, США',
 'Австралия, Великобритания, Бельгия, Индия',
 'Швеция, ФРГ, Франция, Норвегия', 'Нидерланды, Бельгия, ФРГ, Ирландия',
 'Нидерланды, Великобритания, Бельгия', 'США, Колумбия, Испания',
 'Франция, Испания, Бельгия, Панама',
 'США, Великобритания, Чехия, Румыния',
 'Аргентина, Уругвай, Россия, ФРГ, Франция, Нидерланды',
 'Швеция, Франция, Норвегия, Дания', 'Италия, Франция, Великобритания',
 'Венесуэла',
 'Великобритания, США, Австралия, Ирландия, ФРГ, Куба, Канада',
 'Южная Корея, Китай', 'Аргентина, Мексика, Бразилия, Франция, США',
 'Швеция, США', 'Франция, ФРГ, Турция, Катар', 'Франция, Китай',
 'Россия, ФРГ, Великобритания', 'Великобритания, Италия, Испания',
 'Италия, Россия', 'Япония, Китай, Южная Корея', 'Россия, ФРГ',
 'Россия, Ирландия', 'Македония, Франция, Великобритания',
 'Франция, Бразилия, Италия', 'Россия, США', 'Армения, Россия',
 'Россия, Румыния', 'Грузия, Франция', 'ЮАР, США',
 'Великобритания, США, ФРГ, Канада, Австралия', 'США, Ирландия',
 'ФРГ, Люксембург, Франция', 'Россия, Франция',
 'США, Нидерланды, Финляндия, Великобритания, Италия', 'Кипр',
 'Венгрия', 'Франция, США, Норвегия, Дания', 'ФРГ, Австрия, Ирландия',
 'США, Финляндия', 'Италия, Великобритания', 'Россия, Испания',
 'США, Великобритания, Болгария', 'Великобритания, Люксембург',
 'Индия, США', 'США, Канада, Россия, Франция, Чили, Ирландия',
 'Грузия, Россия',
 'Дания, Австрия, Ирландия, Финляндия, Норвегия, Швеция, Нидерланды',
 'Грузия', 'Австралия, Ирландия', 'Италия, Аргентина, Словения',
 'Испания, Франция, Нидерланды, ФРГ, Бельгия, Великобритания, Канада',
 'Франция, Чехия, Бельгия', 'Великобритания, Франция, ФРГ, США',
 'Ирландия, Великобритания, Франция, США, ФРГ, Нидерланды',
 'США, Канада, Япония, Китай', 'США, Великобритания, Исландия',
 'Франция, ФРГ, Нидерланды',
 'Франция, ФРГ, Швеция, США, Чехия, Словакия, Великобритания, Нидерланды',
 'Франция, Канада, Н. Зеландия, США, Нидерланды, ФРГ, Швеция, Россия',
 'Турция, США',
 'Великобритания, США, Канада, Бельгия, Нидерланды, Австрия, ФРГ',
 'Израиль, Франция, ФРГ, Палестина, США, Австрия, Великобритания',
 'Аргентина, Франция, Испания', 'СССР, Россия',
 'Великобритания, Франция, Италия, Индия, Дания, Китай, Бангладеш, Камбоджа, Гаити, Уганда',
 'Италия, Франция, Швейцария', 'Ирландия, Великобритания, Канада',
 'Перу', 'США, Норвегия', 'США, ФРГ, Китай',
 'Великобритания, Новая Зеландия', 'Франция, Дания, Швеция, Китай',
 'США, Великобритания, ФРГ, Бельгия, Дания',
 'США, Франция, Канада, ФРГ, Австралия, Индия', 'СССР, Румыния, Франция',
 'СССР, ГДР', 'Великобритания, Франция, Австрия, Швеция',
 'Россия, Таджикистан', 'Нидерланды, Бельгия, Болгария',
 'Великобритания, Канада, США', 'Франция, Бельгия, Испания',

'США,Вьетнам', 'США,ФРГ,Франция', 'ФРГ,Франция,Бельгия',
'США,Италия', 'ФРГ,Дания,Испания,Швеция,Канада,Эстония,Франция',
'КняжествоАндорра,Украина', 'ФРГ,Люксембург,Бельгия',
'Великобритания,Россия,Украина,США',
'ФРГ,Бельгия,Люксембург,Норвегия',
'США,Австралия,НоваяЗеландия,Великобритания',
'США,Франция,Канада,Великобритания', 'Испания,Колумбия',
'Великобритания,Чехия,Франция', 'Исландия',
'Великобритания,Франция,Венгрия', 'Россия,Франция,ФРГ,Бельгия',
'Франция,Македония',
'Нидерланды,Франция,ФРГ,Бельгия,Швеция,Великобритания',
'Испания,США,Великобритания,Канада', 'РеспубликаКазахстан',
'Франция,Испания,Дания,Венгрия,Швейцария',
'США,Пуэрто,Рико,Франция', 'Мексика,Тайвань,США',
'Великобритания,Испания,Непал,Индия,Венгрия,Гонконг,ФРГ,Дания,Бахрейн',
'Великобритания,Аргентина,США,Франция,Россия,Испания', 'Китай,США',
'Италия,США', 'Финляндия,Швеция,ФРГ', 'Франция,Бельгия,Чехия',
'Франция,ФРГ,США', 'США,Дания', 'Индия,Китай',
'Чехия,Великобритания,США', 'США,ФРГ,Россия',
'Ирландия,Нидерланды,Франция,США,Великобритания',
'Россия,Грузия,Хорватия,Испания', 'Финляндия,Швеция,Норвегия',
'Сербия,Великобритания,США', 'США,Великобритания,Франция,Гонконг',
'Швейцария,Франция', 'Швейцария,Великобритания', 'ОАЭ,США',
'США,Франция,ФРГ,Канада,Австралия', 'ФРГ,Австралия',
'Иран,Франция', 'США,Великобритания,НоваяЗеландия', 'США,Колумбия',
'США,Франция,Ирландия', 'Великобритания,Франция,Бельгия',
'США,Великобритания,Япония', 'США,Франция,ФРГ', 'Польша,Ирландия',
'Ирландия,Великобритания,Франция,США',
'США,Бельгия,Великобритания', 'СССР,ВНР',
'Испания,Франция,США,Великобритания,Бельгия', 'Франция,Дания,США',
'США,Гонконг', 'Бразилия,Канада',
'США,Ирландия,Великобритания,Франция', 'Великобритания,США,Канада',
'Польша,Франция', 'Канада,США,Нидерланды,Великобритания,Аргентина',
'Великобритания,Дания', 'Франция,Бельгия,Япония',
'США,Канада,ФРГ,Франция', 'США,Канада,Франция',
'ФРГ,Австрия,Франция,Италия', 'США,ФРГ,Канада', 'Испания,Мексика',
'США,Япония,Канада,Великобритания,ФРГ,Франция',
'Россия,Украина,ФРГ', 'Израиль,Франция,Великобритания,ФРГ',
'Корея', 'Россия,Болгария', 'ФРГ,Канада,Франция,Бельгия',
'ФРГ,Кения', 'Франция,Монако', 'Франция,Испания,Швейцария',
'Австрия,Франция,ФРГ', 'Румыния', 'США,Великобритания,Франция',
'Канада,Великобритания', 'Канада,Бразилия,Япония',
'ФРГ,Дания,Норвегия', 'Дания,ФРГ', 'СССР,ГДР,Польша,Италия',
'США,Испания,Франция,Великобритания', 'Мексика,Аргентина',
'ФРГ,Великобритания,Италия,Испания', 'Франция,ФРГ,Великобритания',
'Россия,ФРГ,Швейцария', 'Франция,Австралия', 'Голландия',
'Россия,Белоруссия', 'Франция,Испания,Тайвань',
'США,ЮжнаяКорея,НоваяЗеландия', 'Великобритания,Венгрия',
'Великобритания,Франция,ФРГ', 'Франция,ФРГ,Италия',
'Швеция,Финляндия,Франция,Норвегия', 'ФРГ,Израиль',
'Швеция,ФРГ,Дания,Норвегия', 'Франция,Гонконг,Ирландия', 'Эстония',
'Литва,Россия,Украина', 'ФРГ,Италия,Франция',
'Великобритания,Австрия,ФРГ', 'Испания,Мальта',
'Франция,Россия,Румыния,Италия,Бельгия', 'Россия,Польша',
'США,Аргентина', 'Израиль,США,Франция', 'США,НоваяЗеландия,Япония',
'Франция,Чехия,Великобритания', 'ФРГ,Франция,Великобритания',

'США, Великобритания, ФРГ', 'США, Канада, Австралия',
'Япония, США, Франция', 'Франция, Испания, ФРГ',
'США, Хорватия, Босния, Герцеговина',
'Великобритания, Чехия, Франция, Италия', 'США, ФРГ, Нидерланды',
'ФРГ, Франция, Канада', 'Франция, Грузия, ФРГ, Россия, Украина, Бельгия',
'Великобритания, Франция, Италия', 'Франция, Италия, Иран', 'Греция',
'Дания, США', 'США, ФРГ, Австралия', 'США, Канада, Италия',
'США, Япония, ФРГ', 'Франция, Великобритания, Багамские острова, США',
'Швейцария, Люксембург', 'Швейцария, Австрия', 'ФРГ, Швейцария',
'Дания, Швеция, Франция, ФРГ', 'Польша', 'Бразилия, Канада, США',
'Нидерланды, Россия', 'Португалия', 'Франция, Аргентина',
'Швеция, Франция', 'Франция, Украина', 'ФРГ, Украина, Нидерланды',
'Италия, Швейцария', 'Латвия, Россия',
'ФРГ, Франция, Великобритания, США', 'Канада, Италия',
'США, Объединенные Арабские Эмираты', 'США, Венгрия', 'Малайзия',
'Россия, ФРГ, Украина', 'США, Австралия, Мексика', 'Литва, Венгрия',
'ФРГ, Китай', 'Казахстан', 'Республика Корея',
'США, ФРГ, Франция, Великобритания, Канада',
'Бельгия, ФРГ, Люксембург, Франция', 'Франция, Греция',
'Италия, Канада', 'США, Италия, Греция',
'ФРГ, США, Великобритания, Ирландия', 'ФРГ, Нидерланды, ЮАР',
'США, Япония, ФРГ, Великобритания, Нидерланды',
'США, Япония, ФРГ, Великобритания',
'Таиланд, Великобритания, Франция, ФРГ, Испания, Нидерланды',
'ФРГ, Италия, Швейцария',
'Дания, Швеция, Великобритания, Франция, ФРГ, Нидерланды, Норвегия, Финляндия',
'Франция, Великобритания, Нидерланды, Люксембург',
'Нидерланды, Великобритания, Франция, Италия, Япония',
'Великобритания, Нидерланды, Франция, США', 'Канада, Мексика',
'Канада, Южная Корея, США', 'Великобритания, Франция, Бельгия, США',
'Китай, Канада, США', 'Италия, Франция, ФРГ',
'Швеция, ФРГ, Франция, Дания', 'Франция, США, Бельгия',
'Украина, ФРГ, Латвия, Эстония', 'Австрия, Люксембург',
'Великобритания, Китай', 'Монголия', 'Франция, Дания',
'Швеция, Дания, Финляндия',
'Сингапур, Великобритания, Индонезия, Канада, США', 'США, ФРГ, Япония',
'Таиланд, США', 'Франция, Нидерланды', 'Великобритания, Румыния',
'Великобритания, ФРГ, Нидерланды, Дания',
'Франция, ФРГ, Литва, Нидерланды, Россия', 'США, ЮАР, Гонконг',
'Австралия, Колумбия', 'Великобритания, Польша',
'США, Великобритания, Швеция', 'Франция, Швеция, Дания, Норвегия',
'Болгария, США', 'СССР, ФРГ', 'Бельгия, Нидерланды, Франция',
'Финляндия, ФРГ', 'Болгария', 'Грузия, Россия, Украина, Хорватия, ФРГ',
'Франция, Люксембург', 'Франция, ФРГ, Австрия', 'Норвегия, США',
'ФРГ, Швеция', 'Дания, Латвия, Россия, США', 'Финляндия, Польша',
'ФРГ, Бельгия', 'ФРГ, США, Франция, Испания',
'Бельгия, ФРГ, Канада, Франция, США, Великобритания',
'Нидерланды, Бельгия, Люксембург', 'Люксембург, Бельгия, Франция',
'Бельгия, Великобритания, США', 'Великобритания, Австралия',
'Исландия, Бельгия',
'Испания, Швейцария, Великобритания, ФРГ, Новая Зеландия, Нидерланды, Канада',
'Киргизия, Россия', 'Великобритания, США, Испания',
'Италия, Швейцария, ФРГ', 'Россия, Армения',
'США, Великобритания, Люксембург',
'Испания, Ирландия, Бельгия, Франция',
'Великобритания, Гонконг, Венгрия, США, Ирландия', 'США, Пуэрто-Рико',

'Аргентина, Испания, Уругвай', 'Израиль, ФРГ, Франция, Швейцария',
'Великобритания, Мексика, США', 'Испания, Болгария, США',
'Россия, Словакия, Чехия',
'Швейцария, Китай, США, Россия, Республика Корея, Великобритания',
'Россия, Нидерланды, Финляндия',
'Испания, Бельгия, Франция, Португалия, Великобритания',
'Бразилия, Португалия, Франция', 'Швеция, Норвегия',
'Франция, Польша, Бельгия', 'США', 'ФРГ, Австрия, Франция',
'Чехия, Словакия, Хорватия', 'Россия, Польша, Финляндия',
'Россия, Латвия, Чехия', 'Франция, Великобритания, Камбоджа, США, Китай',
'Чехия, Испания, США', 'ФРГ, Люксембург, Дания',
'Великобритания, Испания, ФРГ', 'Дания, Исландия',
'Великобритания, Испания, Италия, Латвия, Франция, Эстония',
'Дания, Швеция, Великобритания, Франция, ФРГ, Норвегия, Финляндия, Нидерланды, Итали
ия',
'Польша, Франция, Великобритания', 'Дания, Норвегия, Венгрия, Чехия',
'Дания, Франция, ФРГ, Швеция', 'Великобритания, Швеция, США',
'Канада, Франция, Италия, Швейцария, США',
'США, Великобритания, ФРГ, Швеция, Канада',
'Республика Беларусь, ФРГ, США, Россия',
'Италия, Швейцария, Франция, ФРГ',
'Россия, Франция, Великобритания, Латвия', 'Россия, Беларусь',
'Франция, Испания, Румыния, США, Бельгия', 'Россия, Австрия',
'Ирландия, Великобритания, США',
'Швейцария, Великобритания, Франция, США, Ирландия',
'Мексика, Франция, ФРГ, Дания, Швеция', 'Нидерланды, Бельгия',
'Швеция, Норвегия, Финляндия, Франция', 'Испания, Франция, Италия',
'Испания, Франция, Греция, Индия, США, Россия, Великобритания',
'Великобритания, Япония, США', 'Дания, Канада, Норвегия, Австралия, США',
'Россия, ФРГ, Казахстан, Польша, Китай',
'Португалия, Франция, Польша, США', 'США, Канада, Аргентина',
'Ливан, США', 'Япония, США',
'Франция, Россия, Швейцария, Румыния, Венгрия', 'Россия, Сербия',
'США, Болгария, Мексика', 'США, Великобритания, Швейцария, Франция',
'Таиланд, Китай, США',
'Великобритания, Нидерланды, Польша, Швейцария, Франция',
'Ирландия, США', 'ФРГ, Франция, Великобритания, Польша, США',
'США, Австралия, Индия', 'Великобритания, Израиль',
'Великобритания, Норвегия, Кения, Хорватия',
'США, Бразилия, Великобритания, Канада', 'Австралия, Канада',
'Кения, ФРГ', 'Колумбия', 'Латвия, Франция', 'Франция, Израиль, ФРГ',
'Швеция, Франция, Великобритания', 'Бельгия, Люксембург',
'ФРГ, Бельгия, Великобритания',
'Испания, Франция, Великобритания, Дания, Бельгия, ФРГ',
'Румыния, ЮАР, Иран, Франция, Канада, Великобритания',
'США, Великобритания, Ирландия, Люксембург', 'Республика Армения',
'США, Франция, Япония', 'Кипр, Россия',
'Дания, Франция, Италия, Бельгия, Нидерланды', 'Россия, Грузия',
'Италия, ФРГ', 'Норвегия, Швеция, Дания', 'Китай, Индия, Гонконг, США',
'Израиль, Украина', 'США, Китай, Индия, Ю. Корея', 'США, Швеция',
'Россия, Китай', 'Канада, США, Каймановы острова',
'Бразилия, Уругвай, Дания, Норвегия, Чили, Швеция',
'Пуэрто-Рико, Великобритания, США', 'Уругвай, Аргентина, Испания',
'США, Великобритания, Италия, Израиль, Сербия, Индия',
'США, Канада, Франция, Индия', 'Великобритания, Россия, Китай',
'Франция, Новая Зеландия', 'Канада, США, Норвегия', 'Италия, Испания',


```
'Испания,ФРГ,Нидерланды', 'Бразилия,США,Китай',
'Великобритания,ФРГ,Дания,США', '2019',
'Италия,Фрация,Бразилия,ФРГ', 'Норвегия,Швеция,Россия',
'Сша,Канада', 'Бельгия,Канада', 'Франция,Финляндия',
'Колумбия,США', 'Норвегия,Нидерланды', 'Италия,Бельгия,Франция',
'Россия,Бельгия,Финляндия', 'Франция,Норвегия',
'Великобритания,ЮжнаяКорея,Канада,США,Индия,Китай,Япония',
'Канада,Норвегия', 'Польша,Франция,Испания,Бразилия,Швеция',
'Австрия,Великобритания,ФРГ', 'Франция,Сенегал,Бельгия',
'Финляндия,Латвия', 'Пакистан', 'ФРГ,Франция,США',
'ФРГ,Великобритания,США', 'Китай,Франция',
'Дания,Норвегия,Швеция,Исландия', 'Кипр,США,Россия',
'НоваяЗеландия,Китай',
'Дания,Канада,Швеция,Франция,ФРГ,Великобритания,США'], dtype=object)
```

Отсортируем страны в каждом элементе по алфавиту, возможно таким образом мы уберем повторяющиеся элементы:

```
In [68]: def sort_countries(strg):
         if isinstance(strg, str):
             return ', '.join(sorted(strg.split(',')))
         else:
             print(strg)
             return strg
```

```
In [69]: data['production_country'] = data['production_country'].apply(sort_countries)
```

```
In [70]: len(data['production_country'].unique())
```

```
Out[70]: 668
```

```
In [71]: data['production_country'].isna().sum()
```

```
Out[71]: 0
```

Таким образом мы убрали 275 повторяющихся значений. Из 950 осталось 675.

2.2.4 Столбец 'director'

```
In [72]: len(data['director'].unique())
```

```
Out[72]: 4812
```

```
In [73]: data['director'].isna().sum()
```

```
Out[73]: 9
```

```
In [74]: data[data['director'].isna()]
```

Out[74]:

	title	puNumber	show_start_date	type	film_studio	production_cour
764	Три звезды. Концерт в Waldbuhne Theater, Берли...	126003512	2012-12-04 12:00:00+00:00	Прочие	Райзинг Алтернатив	Великобрита
1735	Помпеи. Британский музей	126003413	2013-09-11 12:00:00+00:00	Прочие	МортуСкрин, Британский музей	Великобрита
3839	Зур концерт (Большой концерт).	217027010	2010-06-24 12:00:00+00:00	Художественный	ООО "Барс Рекордс"	Рос
4051	Ганс Сэйфуллин "Авылым чишмэлэре. Ганс Сэйфулл...	217017310	2010-04-22 12:00:00+00:00	Художественный	ООО "Барс- Рекордс"	Рос
4297	Изге Вэли (Святой Валентин)	217020610	2010-05-20 12:00:00+00:00	Художественный	ООО "Барс Рекордс"	Рос
4414	Bon Jovi. The Circle Tour	126001710	2010-11-11 12:00:00+00:00	Прочие	Эншутц Интертейнмент Груп	С
4440	Мульт личности. Выпуск 5	214000410	2010-01-25 12:00:00+00:00	Художественный	NaN	Рос
4922	Георг Фридрих Гендель. Мессия. Оратория. Корол...	126001111	2011-04-25 12:00:00+00:00	Прочие	Опус Арт	Великобрита
4941	Питер Гэбриэл и New Blood Orchestra 3D	126002711	2011-11-11 12:00:00+00:00	Прочие	Омниверс Визион	Великобрита

Так как 'director' это категориальный столбец и мы не знаем, кто может быть директором этих фильмов, заменим пропуск в столбце значением 'нет данных'

In [75]: `data['director'] = data['director'].fillna('нет данных')`

In [76]: `data.director.isna().sum()`

Out[76]: 0

Сколько всего уникальных элементов в столбце director?

```
In [77]: len(data.director.unique())
```

Out[77]: 4812

Функция сортировки директоров по алфавиту внутри одной строки:

```
In [78]: def sort_directors(strg):  
         if isinstance(strg, str):  
             return ', '.join(sorted(strg.split(',')))  
         else:  
             print(strg)  
             return strg
```

Применим ее к столбцу 'director'

```
In [79]: data['director'] = data['director'].apply(sort_directors)
```

посмотрим, изменилось ли что-нибудь?

```
In [80]: len(data.director.unique())
```

Out[80]: 4812

Ничего не изменилось, считаем, что все значения уникальны

```
In [81]: print(data.director.unique())
```

```
['Кевин Костнер' 'Е.Матвеев' 'Тимур Бекмамбетов' ... 'Ник Пауэлл'  
 'А.Беляев' 'И.Трусов']
```

```
In [82]: data[data['director'].str.contains('Бекмамбетов', regex=False)]
```

Out[82]:

	title	puNumber	show_start_date	type	film_studio	production_count
2	Особо опасен	221038416	2016-10-10 12:00:00+00:00	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	CL
3	Особо опасен	221026916	2016-06-10 12:00:00+00:00	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	CL
4	Особо опасен	221030815	2015-07-29 12:00:00+00:00	Художественный	Юниверсал Пикчерз, Кикстарт Продакшнз, Марк Пл...	CL
417	Президент Линкольн: Охотник на вампиров	121013812	2012-06-07 12:00:00+00:00	Художественный	Абрахам Продакшнз, Базелевс Продакшн, Тим Берт...	CL
1985	Ёлки 1914	111023014	2014-12-10 12:00:00+00:00	Художественный	ООО "ТаББаК"	Росс
3609	Ёлки 5	111018016	2016-12-15 12:00:00+00:00	Художественный	ООО "ТаББаК"	Росс
3712	Бен-Гур (2016)	221042816	2016-11-25 12:00:00+00:00	Художественный	Дезерт Вулф Продакшн, Майкл Грандаж Компмни, Р...	CL
3713	Бен-Гур (2016)	121024416	2016-09-05 12:00:00+00:00	Художественный	Дезерт Вулф Продакшн, Майкл Грандаж Компмни, Р...	CL
4528	Ёлки	111010710	2010-12-08 12:00:00+00:00	Художественный	ООО "ТаББаК"	Росс
4957	Ирония судьбы. Продолжение.	111005911	2011-09-05 12:00:00+00:00	Художественный	ООО "Базелевс"	Росс

	title	puNumber	show_start_date	type	film_studio	production_count
6081	Чик-Чирик	114003818	2018-08-14 12:00:00+00:00	Анимационный	ООО "Си-Джи-Фэктори"	Росс
6302	Учитель	111019518	2018-10-17 12:00:00+00:00	Художественный	ООО "ТаББаК"	Росс
6508	Ёлки Последние	111023918	2018-12-10 12:00:00+00:00	Художественный	ООО "ТаББаК"	Росс

Тем не менее, при ближайшем рассмотрении мы видим, что фио одного и того же режисера могут быть написаны по разному, надо это привести к одному виду.

Напишем функцию , которая получает на вход строку, анализирует ее и возвращает нужный формат - И.Фамилия:

```
In [83]: def cut_name(el_col_strng):
            el_col_str = el_col_strng.strip().rstrip('.')
            s = ''
            if el_col_str.find(',') > 0:
                lst = sorted(el_col_str.split(','))
                for i, val in enumerate(lst):
                    val = val.strip()
                    if val.find(' ') > 0:
                        names = val.split()
                        s += names[0][0] + '.' + names[1]
                    else:
                        s += val
                if i < len(lst) - 1:
                    s += ", "
            return s
            elif el_col_str.find(' ') >= 0:
                el_col_str = el_col_str.strip().rstrip('.')
                names = el_col_str.split()
                s += names[0][0] + '.' + names[1]
            return s
            else:
                return el_col_str
```

Применим функцию к столбцу 'director' и отформатируем его в едином стиле:

```
In [84]: data.director = data.director.apply(cut_name)
```

```
In [85]: len(data.director.unique())
```

```
Out[85]: 4613
```

Таким образом мы обрали почти 200 повторений и привели все значения столбца к

единому формату

2.2.5 Столбец 'age_restriction'

```
In [86]: data['age_restriction'].unique()
```

```
Out[86]: array(['«18+» - запрещено для детей', '«6+» - для детей старше 6 лет',  
               '«12+» - для детей старше 12 лет',  
               '«16+» - для детей старше 16 лет',  
               '«0+» - для любой зрительской аудитории'], dtype=object)
```

2.2.6 Столбец 'producer'

Посмотрим на кол-во уникальных значений в столбце 'producer':

```
In [87]: len(data.producer.unique())
```

```
Out[87]: 5350
```

Применим функцию, которую мы применяли для форматирования столбца 'director' к столбцу 'producer':

```
In [88]: data.producer = data.producer.apply(cut_name)
```

```
In [89]: len(data.producer.unique())
```

```
Out[89]: 5281
```

Так как функция cut_name() форматировала строку, значит форматированию подверглись и значения 'нет данных', вернем их обратно

```
In [90]: data.producer = data.producer.str.replace('н.данных', 'нет данных')
```

Мы видим, что работа функции позволила удалить часть повторяющихся значений - 69

```
In [91]: data.producer.isna().sum()
```

```
Out[91]: 0
```

2.2.7 Столбец 'film_studio'

Проверим наличие пропущенных значений:

```
In [92]: data['film_studio'].isna().sum()
```

```
Out[92]: 18
```

```
In [93]: data[data['film_studio'].isna()]
```

Out[93]:

	title	puNumber	show_start_date	type	film_studio	production_coun
1293	Stars in Shorts	121011114	2014-06-03 12:00:00+00:00	Художественный	NaN	Великобритан Cl
4440	Мульт личности. Выпуск 5	214000410	2010-01-25 12:00:00+00:00	Художественный	NaN	Рос
6104	Значит любит	111013818	2018-08-10 12:00:00+00:00	Художественный	NaN	Рос
6180	Несейка. Младшая дочь	112007018	2018-09-01 12:00:00+00:00	Документальный	NaN	Рос
6378	Ян Антонышев	112011818	2018-11-15 12:00:00+00:00	Документальный	NaN	Рос
6379	Ян Антонышев	112011918	2018-11-15 12:00:00+00:00	Документальный	NaN	Рос
6398	Доживем до каникул?..	111021918	2018-11-15 12:00:00+00:00	Художественный	NaN	Рос
6484	До и После	112014418	2018-12-05 12:00:00+00:00	Документальный	NaN	Рос
6493	Калмыцкий геше Нгаванг Вангьял	113002018	2018-12-05 12:00:00+00:00	Научно- популярный	NaN	Рос
6496	Мой папа Чингисхан	112014918	2018-12-06 12:00:00+00:00	Документальный	NaN	Рос
6497	В плену города грёз	112014818	2018-12-06 12:00:00+00:00	Документальный	NaN	Рос
6498	Последнее фото	112014518	2018-12-06 12:00:00+00:00	Документальный	NaN	Рос
6506	Мой селфхарм	113001918	2018-12-06 12:00:00+00:00	Научно- популярный	NaN	Рос
6510	Обстоятельства места и времени	112015018	2018-12-12 12:00:00+00:00	Документальный	NaN	Рос

	title	puNumber	show_start_date	type	film_studio	production_coun
6637	Эрик Булатов. Моя Третьяковка, серии 1-2	113000319	2019-01-28 12:00:00+00:00	Научно- популярный	NaN	Рос
6638	Брови	111000719	2019-03-01 12:00:00+00:00	Художественный	NaN	Рос
6661	Топливо	111001419	2019-02-12 12:00:00+00:00	Художественный	NaN	Рос
6721	Путешествие в Париж	111002919	2019-03-01 12:00:00+00:00	Художественный	NaN	Рос

Заменяем пропуски значением "нет данных" - самое подходящее значение

```
In [94]: data['film_studio'] = data['film_studio'].fillna('нет данных')
```

```
In [95]: data['film_studio'].isna().sum()
```

```
Out[95]: 0
```

```
In [96]: len(data['film_studio'].unique())
```

```
Out[96]: 5491
```

Напишем функцию сортировки каждого значения столбца 'film_studio', уберем ненужные пробелы, после этого проверим изменилось ли кол-во уникальных значений:

```
In [97]: def sort_studio(s):
s = ''
if s.find(',') > 0:
    new_list = sorted(s.split(','))
    for i, el in enumerate(new_list):
        el = el.strip()
        s += el
        if i < len(new_list) - 1:
            s += ', '
    return s
else:
    return s.strip()
```

```
In [98]: data['film_studio'] = data['film_studio'].apply(sort_studio)
```

```
In [99]: len(data['film_studio'].unique())
```

```
Out[99]: 5462
```


Убрали 30 дубликатов

2.2.8 Столбец 'genres'

Сколько уникальных значений в столбце 'genres'

```
In [100...] len(data.genres.unique())
```

Out[100]: 742

Напишем функцию форматирования и сортировки значений столбца 'genres':

```
In [101...] def sort_genres(strng):  
    s = ''  
    if strng.find(',') > 0:  
        new_list = sorted(strng.split(','))  
        for i, el in enumerate(new_list):  
            el = el.strip().lower()  
            s += el  
            if i < len(new_list) - 1:  
                s += ', '  
        return s  
    else:  
        return strng.strip()
```

```
In [102...] data.genres = data.genres.apply(sort_genres)
```

```
In [103...] len(data.genres.unique())
```

Out[103]: 473

Убрали 269 дубликатов

2.2.9 Столбец 'financing_source'

Сколько уникальных элементов в столбце 'financing_source'

```
In [104...] len(data['financing_source'].unique())
```

Out[104]: 4

```
In [105...] data['financing_source'].unique()
```

```
Out[105]: array(['нет данных', 'Министерство культуры', 'Фонд кино',  
                'Министерство культуры, Фонд кино'], dtype=object)
```

дубликатов нет

2.2.10 Столбец 'title'

Сколько уникальных элементов?

```
In [106... len(data.title.unique())
```

Out[106]: 6771

Проверим на пропуски:

```
In [107... data.title.isna().sum()
```

Out[107]: 0

Отформатируем значения столбца 'title':

```
In [108... data.title = data.title.apply(lambda x: x.strip().rstrip('.'))
```

```
In [109... len(data.title.unique())
```

Out[109]: 6722

В итоге удалили 49 дубликатов

2.2.11 Столбцы 'refundable_support', 'nonrefundable_support', 'budget', 'box_office'

Для упрощения работы выделим количественные столбцы в отдельный датафрейм:

```
In [110... cols = ['refundable_support', 'nonrefundable_support', 'budget', 'box_office']  
data_q = data[cols]
```

```
In [111... data_q.describe()
```

Out[111]:

	refundable_support	nonrefundable_support	budget	box_office
count	7.485000e+03	7.485000e+03	7.485000e+03	7.485000e+03
mean	5.262515e+05	2.172569e+06	5.643321e+06	3.226716e+07
std	5.781491e+06	1.614999e+07	4.753131e+07	1.606111e+08
min	-1.000000e+00	-1.000000e+00	-1.000000e+00	-1.000000e+00
25%	-1.000000e+00	-1.000000e+00	-1.000000e+00	-1.000000e+00
50%	-1.000000e+00	-1.000000e+00	-1.000000e+00	-1.000000e+00
75%	-1.000000e+00	-1.000000e+00	-1.000000e+00	1.081445e+06
max	1.800000e+08	4.000000e+08	2.305074e+09	3.073569e+09

Выберем только значения не совпадающие с -1(отсутствие данных):

```
In [112... data_qm = data_q[data_q['budget'] != -1]
```

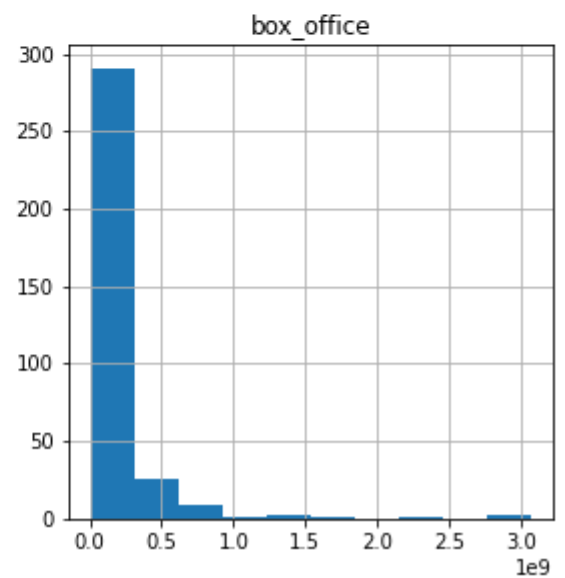
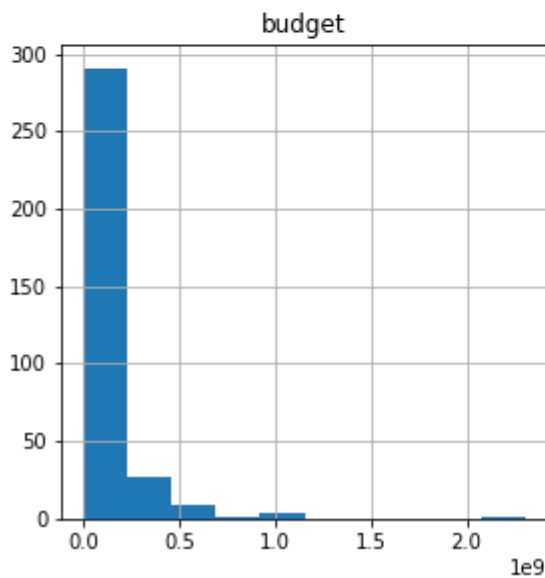
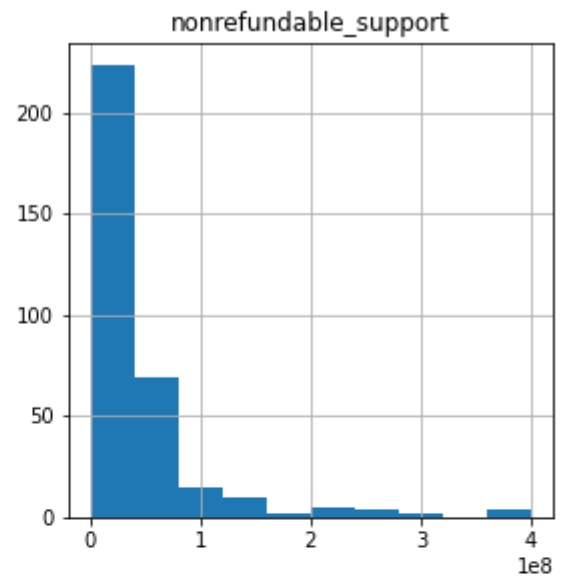
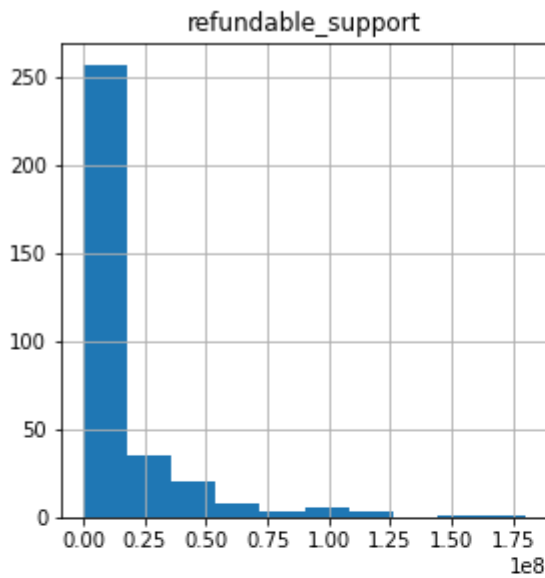
Посмотрим на распределения значений в столбцах:

```
In [113... data_qm.describe()
```

```
Out[113]:
```

	refundable_support	nonrefundable_support	budget	box_office
count	3.320000e+02	3.320000e+02	3.320000e+02	3.320000e+02
mean	1.186446e+07	4.898099e+07	1.272297e+08	1.268479e+08
std	2.491656e+07	5.998012e+07	1.885883e+08	3.287626e+08
min	0.000000e+00	0.000000e+00	0.000000e+00	-1.000000e+00
25%	0.000000e+00	2.500000e+07	4.200000e+07	9.094312e+05
50%	0.000000e+00	3.000000e+07	6.864992e+07	1.214419e+07
75%	1.500000e+07	4.037500e+07	1.419853e+08	1.031074e+08
max	1.800000e+08	4.000000e+08	2.305074e+09	3.073569e+09

```
In [114... data_qm.hist(figsize=(10,10));
```



Никаких особенностей в данных нет

2.2.12 Добавим столбцы: 'show_year', 'movie_director', 'movie_genre', 'state_support_share'

2.2.10.1 Столбец 'show_year' - год проката

```
In [115...] data['show_year'] = data['show_start_date'].dt.year
```

2.2.10.2 Столбец 'movie_director' - год проката

```
In [116...] data['movie_director'] = data['director'].apply(lambda x: x.split(',')[0])
```

2.2.10.3 Столбец 'movie_genre' - жанр фильма

```
In [117...] data['movie_genre'] = data['genres'].apply(lambda x: x.split(',')[0])
```

refundable_support nonrefundable_support budget

2.2.10.4 Столбец 'state_support_share' - доля гос. поддержки

```
In [118... data['state_support_share'] = (  
data.apply(lambda row: round((row['refundable_support'] + row['nonrefundable_support']  
)
```

3. Исследовательский анализ данных

3.1. Сколько фильмов выходило в прокат каждый год

```
In [119... d_pvt = pd.pivot_table(data, index='show_year', values='title', aggfunc='count')  
d_pvt.columns = ['movie_qty']  
d_pvt = d_pvt.reset_index()  
d_pvt
```

Out[119]:

	show_year	movie_qty
0	2010	985
1	2011	622
2	2012	593
3	2013	630
4	2014	806
5	2015	705
6	2016	818
7	2017	505
8	2018	891
9	2019	930

	show_year	movie_qty
0	2010	985
1	2011	622
2	2012	593
3	2013	630
4	2014	806
5	2015	705
6	2016	818
7	2017	505
8	2018	891
9	2019	930

Построим график:

```
In [120... d_pvt.plot(kind='bar', x='show_year', y='movie_qty', xlabel='годы выпуска фильмов',
```



С 2013 года наблюдается стабильный рост выхода фильмов в прокат, с просадкой в 2017. Фильмы вышедшие в 2010 году(максимум за период 2010-2019) скорее всего фильмы, производство которых началось еще до кризиса 2008 года и их готовность на момент кризиса была очень высокой. фильмы 2011-2013 годов фильмы, которые снимались в пост кризисный период и их кол-во значительно меньше фильмов 2010 года.

3.2. Доля фильмов с указанием прокатных сборов от всех вышедших фильмов

```
In [121]: t_qty = data['title'].count()
          bo_qty = data.query('box_office != -1')['title'].count()
          total_share = round(bo_qty / t_qty, 4) * 100
          total_share
```

Out[121]: 42.19

Доля фильмов с указанием прокатных сборов по всем годам составляет 42.19%

3.3. Доля фильмов с указанием прокатных сборов от всех вышедших фильмов по годам

Получим общее число фильмов вышедших в прокат по годам

```
In [122]: data_all = pd.pivot_table(data, index='show_year', values='title', aggfunc='count')
```

Получим число фильмов с указанием прокатных сборов

```
In [123]: data_bo = data[data['box_office'] != -1].pivot_table(values='title', index='show_year')
```

Объединим два датафрейма и рассчитаем какую долю составляют фильмы с указанием

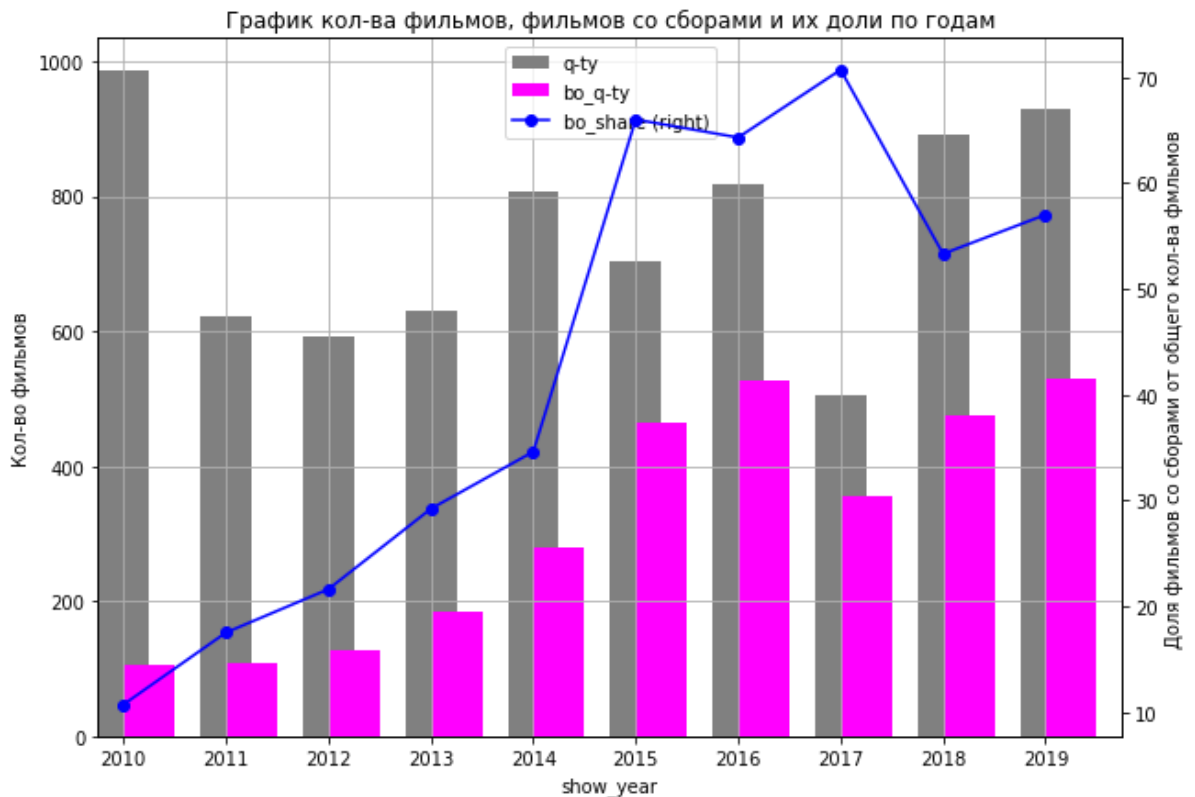
прокатных сборов от общего числа фильмов, по годам выпуска

```
In [124]: data_merged = data_all.merge(data_bo, on='show_year', how='left')
data_merged['bo_share'] = round(data_merged['bo_q-ty'] / data_merged['q-ty'],3) * 100
data_merged
```

```
Out[124]:
```

	show_year	q-ty	bo_q-ty	bo_share
0	2010	985	105	10.7
1	2011	622	109	17.5
2	2012	593	128	21.6
3	2013	630	184	29.2
4	2014	806	279	34.6
5	2015	705	465	66.0
6	2016	818	526	64.3
7	2017	505	357	70.7
8	2018	891	475	53.3
9	2019	930	530	57.0

```
In [125]: ax = data_merged.plot(kind='bar', x='show_year', y='q-ty', title='График кол-ва фил')
data_merged.plot(kind='bar', x='show_year', y='bo_q-ty', ax=ax, color='magenta', pos=1)
data_merged.plot(c='blue', x='show_year', y='bo_share', ax=ax, secondary_y=True, xl=1)
ax.right_ax.set_ylabel('Доля фильмов со сборами от общего кол-ва фмльмов');
```



Из графика мы видим, что с 2010 по 2014 доля фильмов со сборами не высокая - информация по сборам фильмов в данный период минимальна. В то время как с 2015 по 2017 можно наблюдать значительный рост доли фильмов со сборами. После него идет снижение в 2018-2019 годах, при этом даже сниженное значение остается выше значений 2010-2014 годов. Таким образом можно сказать, что весь период 2010-2019 поделится на две части с низкой долей фильмов со сборами и высокой долей фильмов со сборами. При этом среднее значение доли в период 2010-2014 равно:

```
In [126...] data_merged.query('show_year < 2015')['bo_share'].mean()
```

Out[126]: 22.72

В то время как среднее за период с 2015 по 2019 равно:

```
In [127...] round(data_merged.query('show_year > 2014')['bo_share'].mean(),2)
```

Out[127]: 62.26

Видно, что 2015-2019 годы более полно представлены с точки зрения информации о сборах фильмов в прокате, чем 2010-2014 годы. Естественно, что минимальное значение находится в периоде 2010-2014 годов и соответствует 2010 году, в то время как максимальное значение находится в периоде 2015-2019 и соответствует 2017 году.

3.4. Сборы от показа фильмов - среднее, медиана, изменения по годам

Почистим значения в столбце 'box_office': заменим большие минусовые значения (чего быть не должно) на -1(нет инфо)

```
In [128...] data.loc[data['box_office']== -2147483648, 'box_office'] = data.query('box_office =
```

Выберем данные по сборам в отдельный датафрейм:

```
In [129...] data_box_off = data.query('box_office != -1').pivot_table(index='show_year', values
```

```
In [130...] data_box_off
```



```
Out[130]:
```

	show_year	min	max	mean	median
		box_office	box_office	box_office	box_office
0	2010	0	1446400	2.313004e+04	1700.0
1	2011	125	11618085	1.293832e+05	3000.0
2	2012	0	2311220	5.433924e+04	5660.0
3	2013	0	19957031	1.619549e+05	3522.5
4	2014	0	1016410635	2.668442e+07	18160.0
5	2015	0	1658861425	8.494057e+07	4920933.0
6	2016	0	2048126108	9.100057e+07	3846679.0
7	2017	7900	2779686144	1.360328e+08	9968340.0
8	2018	41225	2271754004	1.045651e+08	8891102.0
9	2019	4398	3073568690	9.136926e+07	4627798.0

Переименуем колонки для удобства пользования:

```
In [131...] data_box_off.columns = ['show_year', 'box_office_min', 'box_office_max', 'box_office
```

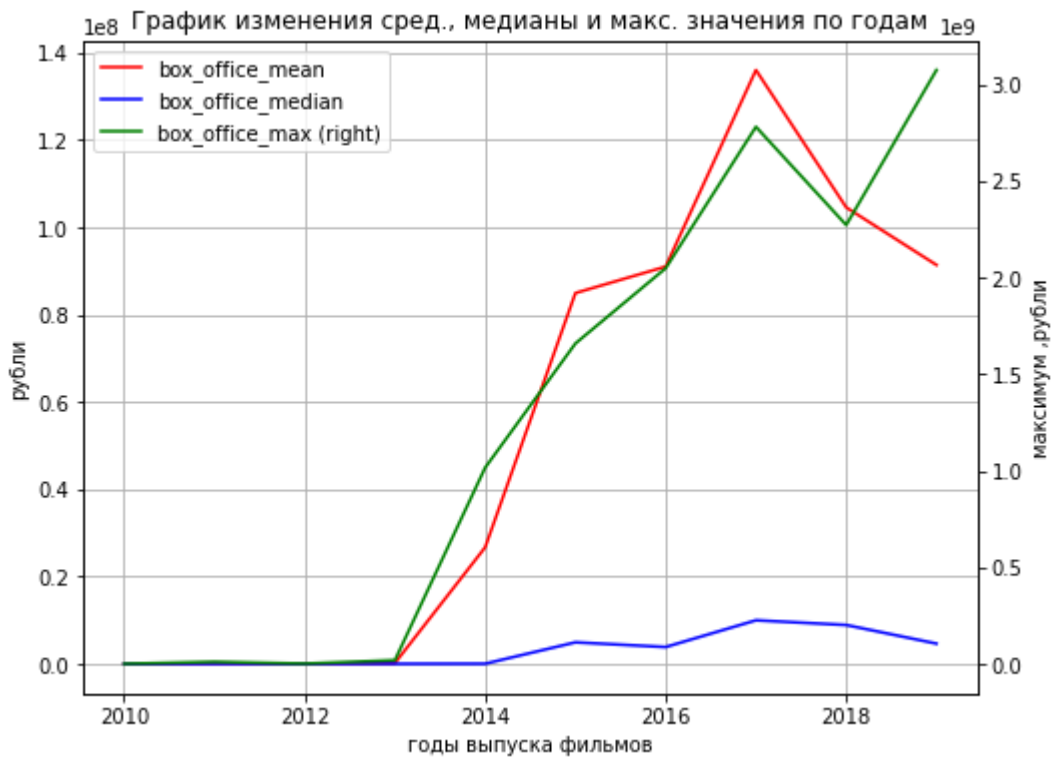
```
In [132...] data_box_off
```

```
Out[132]:
```

	show_year	box_office_min	box_office_max	box_office_mean	box_office_median
0	2010	0	1446400	2.313004e+04	1700.0
1	2011	125	11618085	1.293832e+05	3000.0
2	2012	0	2311220	5.433924e+04	5660.0
3	2013	0	19957031	1.619549e+05	3522.5
4	2014	0	1016410635	2.668442e+07	18160.0
5	2015	0	1658861425	8.494057e+07	4920933.0
6	2016	0	2048126108	9.100057e+07	3846679.0
7	2017	7900	2779686144	1.360328e+08	9968340.0
8	2018	41225	2271754004	1.045651e+08	8891102.0
9	2019	4398	3073568690	9.136926e+07	4627798.0

Подготовим график изменения среднего, медианы и максимального значения сборов от показа фильмов, по годам:

```
In [133...] ax = data_box_off.plot(kind='line', c='r', x='show_year', y='box_office_mean', figs
data_box_off.plot(kind='line', ax=ax, c='b', x='show_year', y='box_office_median',
data_box_off.plot(kind='line', c='g', ax=ax, secondary_y=True, x='show_year', y='bo
ax.right_ax.set_ylabel('максимум ,рубли');
```



Мы видим, что колебания среднего значения сборов по годам очень сильно отличаются от колебаний медианы. Это наглядно демонстрирует сильную зависимость среднего от значений в наборе данных. На графике видно, что среднее повторяет движение максимальных значений в наборе данных(доказательство выше сказанного)

3.5. Влияние возрастных ограничений на сборы от показа фильмов

```
In [134... pvt = data.query('box_office != -1').pivot_table(index='show_year', columns='age_re
pvt
```

Out[134]:

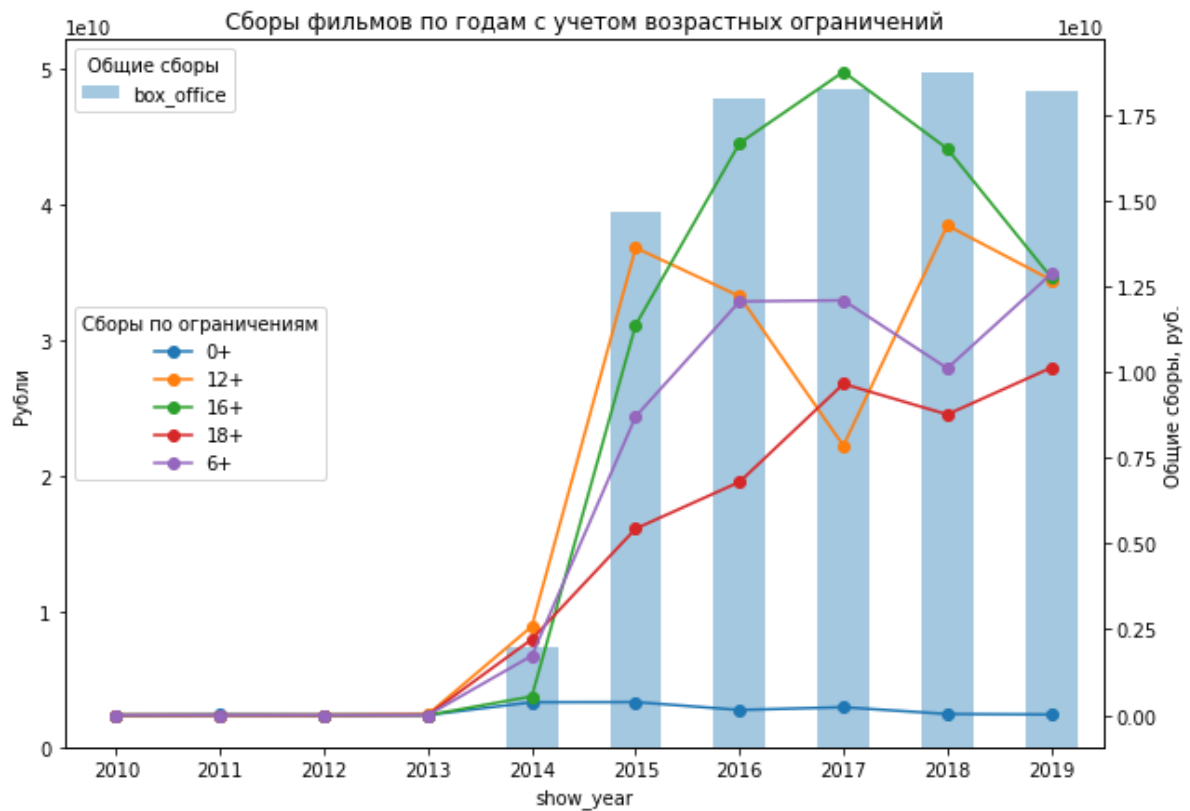
age_restriction	show_year	«0+» - для любой зрительской аудитории	«12+» - для детей старше 12 лет	«16+» - для детей старше 16 лет	«18+» - запрещено для детей	«6+» - для детей старше 6 лет
0	2010	1887611.0	1.201460e+05	3.665120e+05	5.438500e+04	0.000000e+00
1	2011	13499871.0	2.198910e+05	3.815130e+05	1.040000e+03	4.500000e+02
2	2012	2266483.0	1.087340e+05	3.254216e+06	1.825800e+05	1.143410e+06
3	2013	2710302.0	2.091034e+07	6.580690e+05	5.161245e+06	3.597520e+05
4	2014	375562257.0	2.584782e+09	5.442801e+08	2.209713e+09	1.730614e+09
5	2015	379054567.0	1.361608e+10	1.136812e+10	5.432308e+09	8.701799e+09
6	2016	150228839.0	1.220445e+10	1.666455e+10	6.793930e+09	1.205315e+10
7	2017	229598930.0	7.851428e+09	1.874504e+10	9.651496e+09	1.208614e+10
8	2018	32449001.0	1.426729e+10	1.650360e+10	8.760085e+09	1.010497e+10
9	2019	17746066.0	1.268020e+10	1.275342e+10	1.012180e+10	1.285255e+10

Переименуем столбцы для комфортного пользования:

```
In [135... pvt.columns=['show_year', '0+', '12+', '16+', '18+', '6+']
```

Построим график изменения суммы сборов в зависимости от возрастных ограничений

```
In [136... #ax = plt.subplots()
ax = data.query('box_office != -1').pivot_table(index='show_year', values='box_offi
pvt.plot(x='show_year', ax=ax, y=['0+', '12+', '16+', '18+', '6+'], figsize=(10, 7), ti
ax.legend(title='Общие сборы', loc='upper left');
ax.right_ax.legend(title='Сборы по ограничениям', loc='center left')
ax.right_ax.set_ylabel('Общие сборы, руб.');
```



На графике видно, что наибольшие сборы получали фильмы с ограничением возраста "16+", при этом фильмы с ограничением "18+" одни из самых не популярных по сборам, ниже сборы только у фильмов с граничением "0+". Это наталкивает на мысль посмотреть удельные сборы(сборы на 1 фильм) по ограничениям и годам:

```
In [137... pvt_count = data.query('box_office != -1').pivot_table(index='show_year', columns='pvt_count
```

Out[137]:

age_restriction	show_year	sum				
		«0+» - для любой зрительской аудитории	«12+» - для детей старше 12 лет	«16+» - для детей старше 16 лет	«18+» - запрещено для детей	«6+» - для детей старше 6 лет
0	2010	1887611.0	1.201460e+05	3.665120e+05	5.438500e+04	0.000000e+00
1	2011	13499871.0	2.198910e+05	3.815130e+05	1.040000e+03	4.500000e+02
2	2012	2266483.0	1.087340e+05	3.254216e+06	1.825800e+05	1.143410e+06
3	2013	2710302.0	2.091034e+07	6.580690e+05	5.161245e+06	3.597520e+05
4	2014	375562257.0	2.584782e+09	5.442801e+08	2.209713e+09	1.730614e+09
5	2015	379054567.0	1.361608e+10	1.136812e+10	5.432308e+09	8.701799e+09
6	2016	150228839.0	1.220445e+10	1.666455e+10	6.793930e+09	1.205315e+10
7	2017	229598930.0	7.851428e+09	1.874504e+10	9.651496e+09	1.208614e+10
8	2018	32449001.0	1.426729e+10	1.650360e+10	8.760085e+09	1.010497e+10
9	2019	17746066.0	1.268020e+10	1.275342e+10	1.012180e+10	1.285255e+10

Установим индекс в виде года выпуска фильма, для удобства дальнейших вычислений:

```
In [138... pvt_count.set_index('show_year', inplace=True)
```

Переименуем столбцы, также для удобства:

```
In [139... pvt_count.columns = ['sum_0+', 'sum_12+', 'sum_16+', 'sum_18+', 'sum_6+', 'count_0+', 'co
```

Посмотрим, что получилось:

```
In [140... pvt_count
```

Out[140]:

	sum_0+	sum_12+	sum_16+	sum_18+	sum_6+	count_0+	count_12+
show_year							
2010	1887611.0	1.201460e+05	3.665120e+05	5.438500e+04	0.000000e+00	22.0	
2011	13499871.0	2.198910e+05	3.815130e+05	1.040000e+03	4.500000e+02	30.0	
2012	2266483.0	1.087340e+05	3.254216e+06	1.825800e+05	1.143410e+06	30.0	
2013	2710302.0	2.091034e+07	6.580690e+05	5.161245e+06	3.597520e+05	37.0	
2014	375562257.0	2.584782e+09	5.442801e+08	2.209713e+09	1.730614e+09	38.0	
2015	379054567.0	1.361608e+10	1.136812e+10	5.432308e+09	8.701799e+09	39.0	
2016	150228839.0	1.220445e+10	1.666455e+10	6.793930e+09	1.205315e+10	41.0	
2017	229598930.0	7.851428e+09	1.874504e+10	9.651496e+09	1.208614e+10	3.0	
2018	32449001.0	1.426729e+10	1.650360e+10	8.760085e+09	1.010497e+10	5.0	
2019	17746066.0	1.268020e+10	1.275342e+10	1.012180e+10	1.285255e+10	3.0	

Добавим в таблицу новые столбцы со значением удельных сборов(сборы на 1 фильм) по возрастным ограничениям и годам:

```
In [141]: arl = ['spec_0+', 'spec_12+', 'spec_16+', 'spec_18+', 'spec_6+']
for i in range(5):
    pvt_count[arl[i]] = round(pvt_count.iloc[:,i] / pvt_count.iloc[:,i+5],1)
```

Если появились пропуски заполним их 0:

```
In [142]: pvt_count = pvt_count.reset_index().fillna(0)
```

```
In [143]: pvt_count
```

Out[143]:

	show_year	sum_0+	sum_12+	sum_16+	sum_18+	sum_6+	count_0+	count_12+
0	2010	1887611.0	1.201460e+05	3.665120e+05	5.438500e+04	0.000000e+00	22.0	
1	2011	13499871.0	2.198910e+05	3.815130e+05	1.040000e+03	4.500000e+02	30.0	
2	2012	2266483.0	1.087340e+05	3.254216e+06	1.825800e+05	1.143410e+06	30.0	
3	2013	2710302.0	2.091034e+07	6.580690e+05	5.161245e+06	3.597520e+05	37.0	
4	2014	375562257.0	2.584782e+09	5.442801e+08	2.209713e+09	1.730614e+09	38.0	
5	2015	379054567.0	1.361608e+10	1.136812e+10	5.432308e+09	8.701799e+09	39.0	
6	2016	150228839.0	1.220445e+10	1.666455e+10	6.793930e+09	1.205315e+10	41.0	
7	2017	229598930.0	7.851428e+09	1.874504e+10	9.651496e+09	1.208614e+10	3.0	
8	2018	32449001.0	1.426729e+10	1.650360e+10	8.760085e+09	1.010497e+10	5.0	
9	2019	17746066.0	1.268020e+10	1.275342e+10	1.012180e+10	1.285255e+10	3.0	

Построим график изменений удельных сборов по возрастным категориям и годам:

```
In [144... pvt_count.plot(x='show_year', y=['spec_0+', 'spec_12+', 'spec_16+', 'spec_18+', 'spec_6+'],
```



По графику мы видим, что картина изменилась и теперь наиболее популярные по удельным сборам фильмы с возрастными ограничениями "6+" и "12+", но по графику сложно понять кто из них лидер нашего исследования, посчитаем общие удельные сборы за все годы и узнаем победителя:

```
In [145... for i in range(11,16):
    su = round(sum(pvt_count.iloc[:,i]),0)
    print(f'Удельная сумма сборов по возрастным ограничениям {pvt_count.columns[i]}')
```

Удельная сумма сборов по возрастным ограничениям spec_0+ за все годы равна 1128894
12.0
Удельная сумма сборов по возрастным ограничениям spec_12+ за все годы равна 920121
662.0
Удельная сумма сборов по возрастным ограничениям spec_16+ за все годы равна 498321
275.0
Удельная сумма сборов по возрастным ограничениям spec_18+ за все годы равна 305497
946.0
Удельная сумма сборов по возрастным ограничениям spec_6+ за все годы равна 9700025
35.0

Победителем выходят фильмы с возрастным ограничением "6+", с небольшим отставанием идут фильмы "12+", как было видно и из графика, затем фильмы "16+" и с неюольши отставанием от них фильмы "18+" и замыкают группу фильмы с ограничением "0+"

Колебания сборов происходят скорее всего из-за успешности (сценарий, актеры,

уровень и качество съемок, реклама) вышедших фильмов, чем фильм более успешен, тем больше сборов он приносит. Почему побеждают фильмы с ограничением "6+" и "12+" тоже понятно - это фильмы для детей, а дети ходят в кино с родителями, т.е. количество проданных билетов сразу возрастает минимум в два раза (с одним родителем), а то и в три (с двумя родителями). Что также видно по суммам сборов выше.

4. Анализ фильмов с государственной поддержкой

Вынесем фильмы с гос поддержкой в отдельный дата фрейм:

```
In [146...] data_st_sup = data.query('(refundable_support + nonrefundable_support) > 0')
```

Заменяем значение в столбце бюджет в строке 7113 на 0 (был большой минус, чего быть не может):

```
In [147...] data_st_sup.loc[7113, 'budget'] = 0
```

```
In [148...] data_st_sup[data_st_sup['state_support_share'] < 0]
```


Out[148]:

	title	puNumber	show_start_date	type	film_studio	productic
2052	14+	111023614	2014-12-19 12:00:00+00:00	Художественный	ООО Киностудия "Сентябрь"	
2057	Духless 2	111000415	2015-01-26 12:00:00+00:00	Художественный	ООО "Арт Пикчерс Студия", ООО "Киностудия "Слово"	
2471	Воин (2015)	111017315	2015-09-28 12:00:00+00:00	Художественный	ООО "Арт Пикчерс Студия", ООО "Форпост Продакшн"	
2531	Бармен	111009615	2015-05-26 12:00:00+00:00	Художественный	ООО "Арт Пикчерс Студия", АО "ВайТ Медиа"	
2614	Савва. Сердце воина	114001715	2015-07-21 12:00:00+00:00	Анимационный	ООО "Арт Пикчерс Студия", ООО "Глюкоза Продакшн"	
2683	Тряпичный союз	111010215	2015-06-08 12:00:00+00:00	Художественный	ООО "Кинокомпания "КОКТЕБЕЛЬ"	
2787	Срочно выйду замуж	111017115	2015-09-30 12:00:00+00:00	Художественный	ООО "Арт Пикчерс Студия", АО "ВайТ Медиа"	
2873	Помню - не помню!	111004916	2016-03-01 12:00:00+00:00	Художественный	ООО "КиноФирма"	
3046	Любовь с ограничениями	111008216	2016-04-29 12:00:00+00:00	Художественный	ООО "Арт Пикчерс Студия", АО "ВайТ Медиа"	
3564	Притяжение (2016)	111018116	2016-12-16 12:00:00+00:00	Художественный	ООО "Арт Пикчерс Студия", ООО "Водород 2011"	
3674	Чудо-юдо	114001817	2017-06-27 12:00:00+00:00	Анимационный	ООО "КиноФирма"	
3738	Мифы	111004317	2017-06-07 12:00:00+00:00	Художественный	ООО "Арт Пикчерс Студия", ООО "ХАЙПФИЛМ"	
5672	Селфи	111012117	2018-02-01 12:00:00+00:00	Художественный	ООО "Арт Пикчерс Студия", ООО "Киностудия "Слово"	

	title	puNumber	show_start_date	type	film_studio	productic
5805	Два хвоста	114001018	2018-05-31 12:00:00+00:00	Анимационный	ООО "ЛИЦЕНЗИОННЫЕ БРЕНДЫ"	
6007	Принцесса и дракон	114003018	2018-08-23 12:00:00+00:00	Анимационный	ООО "ЛИЦЕНЗИОННЫЕ БРЕНДЫ"	
6322	Баба Яга	114006518	2018-11-08 12:00:00+00:00	Анимационный	ООО "Арт Пикчерс Студия"	
6822	Большое путешествие	114002419	2019-04-20 12:00:00+00:00	Анимационный	ООО "ЛИЦЕНЗИОННЫЕ БРЕНДЫ"	

По фильмам у которых 'state_support_share' равно 0 или -1 мы ничего сказать не сможем, потому уберем их из анализа:

```
In [149... data_st_sup = data_st_sup[data_st_sup['state_support_share'] > 0]
```

Попробуем найти какие-то взаимосвязи между столбцами дата фрейма и господдержкой 'state_support_share'. Будем рассуждать логически, может ли быть какая-то связь между:

1. названием фильма и 'state_support_share' ? Скорее всего они никак не связаны. Выбор название фильма это случайный процесс.
2. тоже самое относится и к номеру прокатного удостоверения (выдается случайным образом, ну или по порядку, что в нашем случае тоже будет случайным)
3. дата начала проката также никак не связана с финансированием со стороны государства (если бы мы жили в СССР, то такая связь могла бы существовать, многие выходы фильмов обычно приурочивались к каким-либо датам и праздникам, хотя в то время все фильмы выходили только благодаря финансированию государства)
4. тип фильма может быть связан с поддержкой государства - нужно рассмотреть
5. студия, которая занималась производством фильма - можно поискать зависимости
6. страна - все фильмы поддерживаемые государством производятся в этом государстве, так что поддержка в этом случае 100%
7. режисер - связь вполне может быть, так как господдержка скорее всего будет предоставлена известному режисеру, чем неизвестному. Посмотрим есть ли связь в этом случае?
8. продюсер - связь также может быть, как и в случае с режисером. Посмотрим.
9. возрастные ограничения, связь не очень очевидна, но попробуем поискать
10. рейтинг - посмотрим на наличие связи
11. жанр - посмотрим на наличие связи
12. сборы - посмотрим наличие связи

Итак план мы набрасали, в бой:

4.1. Связь типа фильма и господдержки

Тип фильма является категориальным столбцом и самое главное строковым.

Невозможно искать связь между строковым значением и числовым. Для этого нужно перевести строковое значение в числовое, или другими словами закодировать столбец 'type'

Закодируем тип фильма кол-вом появлений в дата фрейме:

```
In [150]: data_st_sup['type'].value_counts()
```

```
Out[150]: Художественный    289
Анимационный             25
Документальный           1
Name: type, dtype: int64
```

```
In [151]: data_st_sup.type.unique()[::-1]
```

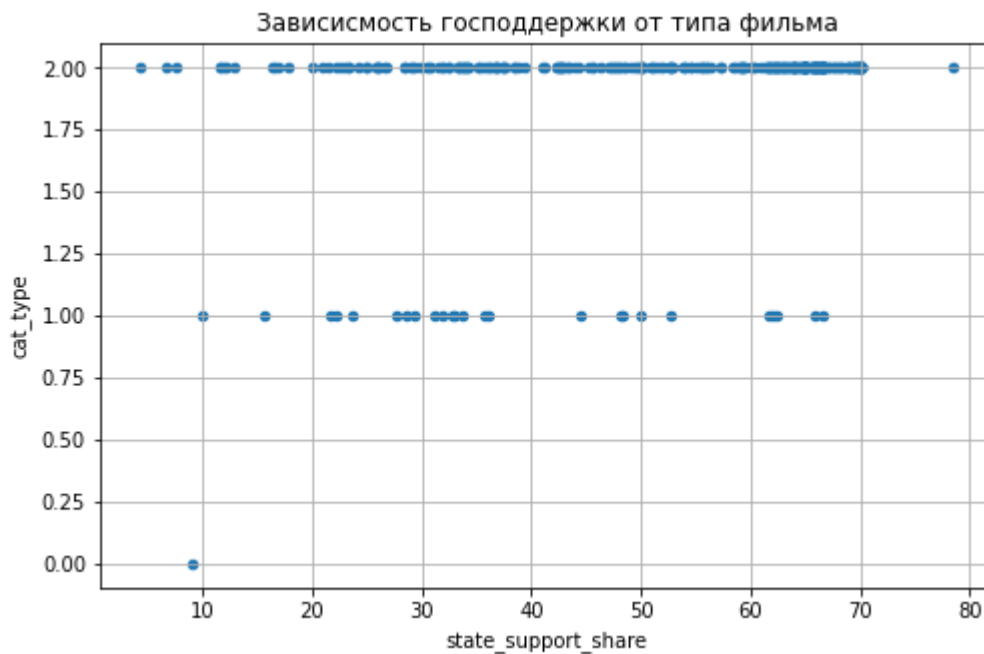
```
Out[151]: array(['Документальный', 'Анимационный', 'Художественный'], dtype=object)
```

Создадим категориальные столбец 'cat_type', который будет состоять из кодов типов фильмов, причем они будут отражать кол-во появления каждого типа в дата фрейме:

```
In [152]: data_st_sup['cat_type'] = pd.Categorical(data_st_sup['type'], data_st_sup.type.unique())
data_st_sup['cat_type'] = data_st_sup['cat_type'].astype('category')
data_st_sup['cat_type'] = data_st_sup['cat_type'].cat.as_ordered()
```

Посмотрим на наличие зависимость на графике:

```
In [153]: data_st_sup.plot(kind='scatter', x='state_support_share', y='cat_type', grid=True,
```



Расчитаем коэффициент корреляции:

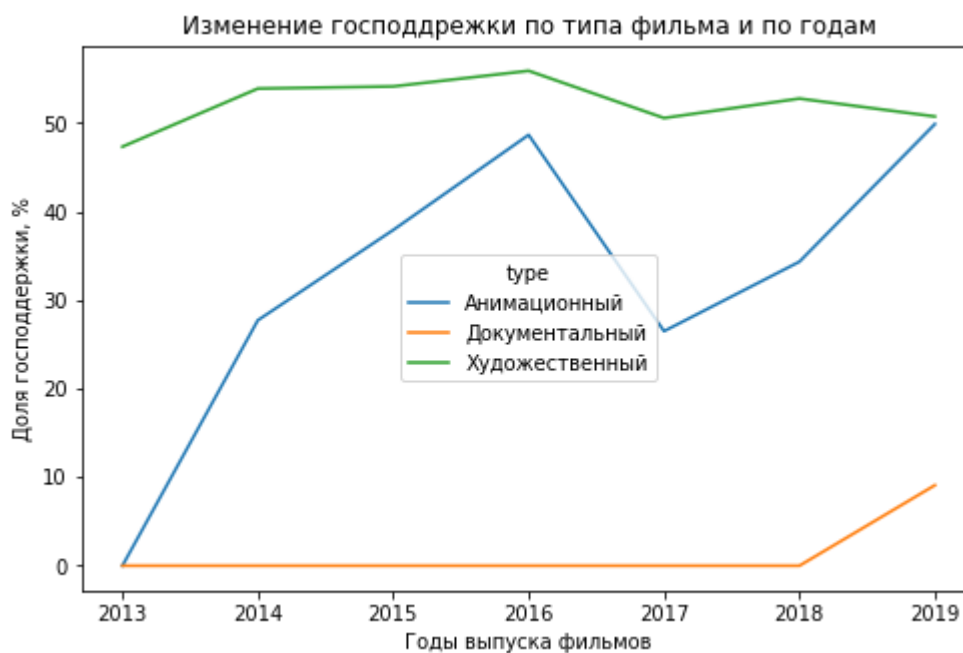
```
In [154...] data_st_sup['cat_type'].corr(data_st_sup['state_support_share'])
```

Out[154]: 0.25897290613068386

Связь очень слабая

Посмотрим зависимость от года выпуска в прокат:

```
In [155...] pd.pivot_table(data_st_sup, columns='type', values='state_support_share', index='sh
y=['Анимационный', 'Документальный', 'Художественный'], title='Изменение господдержки
```



Несмотря на низкую корреляцию - 0.26, мы видим, что и по кол-ву и по доле

господдержки лидирует "художественный фильм", затем идет "анимационный" и с минимальной поддержкой идет "документальный". Таким образом прослеживается явная зависимость господдержки от того какой фильм собираются снимать. Это наверно достаточно логично, ведь художественные фильмы гораздо популярнее, чем анимаионные и тем более документальные.

4.2. Связь студии, создавшей фильм, и господдержки

Посмотрим сколько раз та или иная студия выпускала фильмы в прокат, с господдержкой:

```
In [156... data_st_sup['film_studio'].value_counts()
```

```
Out[156]: 000 "Студия анимационного кино "Мельница"
7
000 "ТаББаК"
7
000 "Кинокомпания "СТВ"
6
000 "ВИЗАРТ ФИЛЬМ"
5
000 "Каро Продакшн"
4

..
000 "Муха", 000 "Анимационная студия "Сказка"
1
000 "ЦЕНТРОКИНО", 000 "КИНОДАНЦ"
1
000 "Муравей Продакшн", 000 "Продюсерский центр "Новое Время"
1
ФГУП "Киноконцерн "Мосфильм", 000 "Продюсерская компания Александра Литвинова"
1
000 "Студия Третий Рим", 000 "ТаББаК"
1
Name: film_studio, Length: 247, dtype: int64
```

Здесь уже наблюдается небольшая зависимость, какие-то студии выпускали фильмы с господдержкой несколько раз, а какие-то только единожды
продолжим наш анализ дальше:

Найдем уникальные значения столбца студия фильма:

```
In [157... data_st_sup['film_studio'].unique()
```

```

Out[157]: array(['000 "Кинокомпания Отражение", АНО содействия развитию кинематографии "Ангел-Фильм"',
                  '000 "Компания "АТК-Студио"', '000 "Компания "РЕАЛ-ДАКОТА"',
                  '000 "Студия "Пассажир"', '000 "Студия "Мастер-Фильм"',
                  '000 "Компания "Реал-Дакота"',
                  'Jura Podnieka Studija, Rezo Productions, 000 "Продюсерский Центр "Хорошо Продакшн"',
                  '000 "Водород 2011"', '000 "Кинокомпания "Ракурс"', '000 "ТаББаК"',
                  '000 "Продюсерская кинокомпания "Киномир"',
                  '000 "Пиманов и партнеры"', '000 "ВИЗАРТ ФИЛЬМ"', '000 "Пендулум"',
                  'ФГУП "Киноконцерн "Мосфильм", 000 "Продюсерская компания Александра Литвинова"',
                  '000 "Арт Пикчерс Студия", 000 "Корнер Ворк"',
                  '000 "АМАЛЬГАМА ПРОДАКШН"', '000 "Амкарт"', '000 "Каро Продакшн"',
                  '000 "Студия "Кинодел", 000 "Кинокомпания "Новые люди"',
                  '000 "Группа компаний "Андреевский флаг", 000 "Творческое объединение "Кинокомпания "Андреевский флаг"',
                  '000 "Ультра стори", 000 "Энджой мувиз"',
                  '000 "Киностудия "МАСТЕР"', '000 "Артлайт"',
                  '000 "Кинокомпания "Фигаро"', '000 "Валдай"',
                  '000 "Смарт ТВ Продакшн"',
                  '000 "Центр Юпитер ФМ", 000 "Кинокомпания "Фортуна Фильм"',
                  '000 "Магнум-фильм" по заказу 000 "ММЕ"',
                  '000 "Инвестмент Бизнес Консалтинг", 000 "Пролайн-медиа", 000 "Киностудия "Никола-фильм"',
                  '000 "Нон-Стоп Продакшн"', '000 "29 февраля"',
                  '000 "Продюсерский центр "Всё хорошо"',
                  'АНО "Творческая студия "Стелла"', '000 "Кинокомпания "СТВ"',
                  '000 "Магнум-фильм"',
                  'НП "Интерфест", 000 "Компания "РЕАЛ-ДАКОТА"',
                  '000 "Кинокомпания "Ладья"', '000 "ТАН Фильм"',
                  '000 "Росмедиа Продакшн" по заказу 000 "МАРС МЕДИА ЭНТЕРТЕЙНМЕНТ"',
                  '000 "Первое творческое объединение"',
                  '000 "Студия анимационного кино "Мельница"', '000 "АРТЛАЙТ"',
                  '000 "Кинокомпания "ТАЛАН"', '000 "Киностудия "ВЕРТИКАЛЬ"',
                  '000 "Продюсерская компания Николая Расторгуева"',
                  '000 "Кинопрограмма XXI век"', '000 "ЛИГА ПРОДАКШН"',
                  '000 "Студия "РОЙ", Культурный фонд "Кино-телекомпания "Православная энциклопедия"',
                  '000 "Аристократ", 000 "Революция Фильм", 000 "Пропеллер продакшн"',
                  '000 "СТН-фильм"', '000 "Продюсерский центр "Ленфильм"',
                  '000 "ЭН-ДЖИ ПРОДАКШН"',
                  '000 "Кинокомпания МАРС-фильм" по заказу 000 "МАРС МЕДИА ЭНТЕРТЕЙНМЕНТ"',
                  '000 "Смешарики"',
                  '000 "ЛИНКЕД ФИЛМС", 000 "Эппл Фильм Продакшн", 000 "Метрафильмс"',
                  '000 "Студия "ТРЕТИЙ РИМ"', '000 "Продюсерский центр "ВГИК-Дебют"',
                  'Departures Films, РСТ, 000 "Петрушка Продакшн", 000 "Студия ТРЕТИЙ РИМ"',
                  '000 "Весёлая Компания", 000 "ТаББаК"', '000 "Кинокомпания "Небо"',
                  '000 "Студия ГРИНСИТИ"', '000 "КОМПАНИЯ "РЕАЛ-ДАКОТА"',
                  '000 "Анимационная студия РИМ"',
                  'ОАО "ТПО "Киностудия им.М.Горького", 000 "Валдай", 000 "ЭЙРФИЛЬМ", 000 "Продюсерский центр Киностудии им. М.Горького"',
                  '000 "Ол Медиа Компани"', '000 "Телесто-фильм"',
                  'Ma.ja.de Film production GmbH, Al Film Productions, 000 "Студия "Пассажиры"',
                  '000 "Продюсерская компания "Среда", 000 "Продакшн Сервис"',

```

'ИКС-Фильме Креатив Пул ГмбХ (X-Filme Creative Pool GmbH), ЗАО "Студия "Ф. А.Ф."',

'000 "Кинокомпания Дебют"', '000 "КИНОБАЙТ", 000 "КИНОДАНЦ"',

'000 "Студия Ортодокс"',

'Индивидуальный Предприниматель Шальопа А.Г.',

'000 "КЕЙСТОУН ПРОДАКШН ГРУП" по заказу 000 "Фильмы навсегда" при участии

АО "ВБД Груп"',

'000 "Каргофильм"', '000 "АБС"',

'000 "Студия "ТРИТЭ" Никиты Михалкова"',

'000 "Продюсерский Центр "Хорошо Продакшн"',

'000 "Продюсерский центр "Парадиз"', '000 "Энджой мувиз"',

'000 "ПРОДАКШН ВЭЛЮ ВОРЛД ВАЙД"',

'000 "Киностудия "ВЕРТИКАЛЬ", ФГУП "Киноконцерн "Мосфильм"',

'000 "Русские Золотые Серии"', '000 "Пан-Атлантик Студио"',

'000 "Студия Продюсерского Кино"', '000 "Инспирейшн Филмс"',

'000 "Корнер Ворк"', '000 "Медиа 77"', '000 "КиноПроГрупп"',

'000 "Кинокомпания "Небо" по заказу АО "ВБД Груп" и 000 "Кинодом" при участии 000 "Фильмы навсегда" и ЗАО "СТУДИЯ ПАЛАМЕДА"',

'000 "Кинодом"', '000 "Белое Зеркало"',

'000 "Продюсерский центр "ЗОЛОТОЙ ВЕК"', '000 "Кинофабрика"',

'000 "Продюсерский центр "Синема Продакшн"',

'НП Киностудия детских и юношеских фильмов "Илья Муромец"',

'000 "Дип Коммьюникейшн"',

'000 "Медиа Арт Студио", 000 "Продюсерский центр "Синема Продакшн", 000 "Студия 25"',

'000 "Киностудия "КУРЬЕР"',

'МАНУФАКТУР ПРОДАКШН, ТРЕМОРА, 000 "Пролайн-Медиа"',

'000 Кинокомпания "Аврора продакшнс" по заказу 000 "Фильмы навсегда медиа"',

'000 "Компания "Новые люди"',

'АО "ВБД Груп", 000 Кинокомпания "Инвада Фильм"',

'000 "Форс Медиа"',

'000 "Мастерская Павла Лунгина". 000 "Арт Пикчерс Студия"',

'000 "Студия Третий Рим", 000 "ТаББак"', '000 "ВГИК-ДЕБЮТ"',

'000 "ТПО "РОК"', '000 "Студия АСДС"', '000 "ДжемСтудия"',

'Drife Filmproduktion GmbH & Co.KG, 000 "Продюсерский центр А.С.Кончаловского"',

'000 "СТУДИЯ "2-Б-2" ИНТЕРТЭЙНМЕНТ"',

'000 "Кинопрограмма "XXI век"',

'000 "Продюсерский центр "ВГИК - Дебют"',

'000 Кинокомпания "Лунапарк"', '000 "Студия "СЛОН"',

'000 "ВВП Альянс"', '000 "Бармалей - Фильм"',

'000 "Продюсерская фирма Игоря Толстунова"',

'000 "РЕНОВАЦИО интертейнмент", 000 "Энджой мувиз"',

'000 "Кинокомпания МАРС-фильм" по заказу 000 "ММЕ"',

'000 "Стар Медиа Дистрибьюшн"',

'000 "РЕНОВАЦИО интертейнмент", 000 "Энджой Мувиз", 000 "Ангел"',

'000 "Кинобюро" по заказу 000 "Среда и Ко"',

'Фонд поддержки развития общества "Наши дети"', '000 "ЛЕНКИНО"',

'000 "ЛЕДА-ФИЛЬМ"', '000 "Энджой мувиз", 000 "Фреш-Фильм"',

'АО "ТПО "Киностудия им. М.Горького", 000 "Слоу ТВ", 000 "РТВ"',

'000 "Крылья"', 'ЗАО "ПРОДЮСЕРСКАЯ КОМПАНИЯ ВАЛЕРИЯ ТОДОРОВСКОГО"',

'000 "Киностудия Арго"', '000 "КИНОБЮРО"',

'ЧУК "КИНОКОМПАНИЯ СЕРГЕЯ БЕЗРУКОВА", Компания Production Value WorldWide (000 "ПРОДАКШН ВЭЛЮ ВОРЛД ВАЙД)',

'000 "Колобанга"',

'000 "М-Продакшн Медиа", ФГУП "Киноконцерн "Мосфильм"',
 '000 "ЦЕНТРОКИНО", 000 "КИНОДАНЦ"',
 '000 "Арт Пикчерс Студия", 000 "2Д Целлулоид"',
 '000 "Кинокомпания КИТ"', '000 "ХАЙПФИЛМ"',
 '000 "РЕНОВАЦИО интертейнмент", 000 "Пятый элемент"',
 '000 "Матильда", 000 "ТПО "РОК"',
 'АО "ТПО "Киностудия им.М.Горького"', '000 "КАРГО"',
 '000 "Лемон Филмз Студิโอ"',
 'АО "ВБД Груп", 000 "Киностудия "Слово" по заказу 000 "Уолт Дисней Компании СНГ"',
 '000 Продюсерский центр "Фёдор Добронравов"', 'ЗАО "Аэроплан"',
 '000 "РБ ПРОДАКШН"', '000 "Продюсерский центр ИВАН"',
 'Некоммерческий фонд поддержки кинематографа "Пример интонации" (Фонд Александра Сокурова)',
 'ОЧУ ДО "Киношкола "Лестница", ИП Петрухин А.А., 000 "Магnum Групп", 000 "Корпорация Русская Фильм Группа"',
 '000 "Энджой Мувиз"', '000 "КИНОДАНЦ"',
 '000 "КМФ" по заказу 000 "ММЕ"', '000 Кинокомпания "100фильм"',
 '000 "Продюсерский центр ЕРАЛАШ"',
 '000 "Продюсерский центр "АЛЬТАИР-Ц"', '000 "Киностудия КИТ"',
 'Государственная телевизионная компания "Телеканал "Россия" - филиал федерального государственного унитарного предприятия "Всероссийская государственная телевизионная и радиовещательная компания", 000 "Арт Пикчерс Студия", 000 "Водород 2011"',
 '000 "Фреш-Фильм"', '000 "КЕДР ПРО"',
 '000 "Ди Пи Жи", 000 "Друг Друга"',
 '000 "Муха", 000 "Анимационная студия "Сказка"',
 'ЗАО "МЕДИА-ТРЕСТ"', 'АНО "КиноДело продакшн"',
 '000 "Киностудия "Слово", 000 "Студия "ТРИТЭ" Никиты Михалкова", 000 "ДК интертейнмент"',
 '000 "Студия "МОСТ", 000 "РФГ Корп"', '000 "САГа"',
 'ФОНД РАЗВИТИЯ НАЦИОНАЛЬНОГО КИНО, 000 "Телесто-фильм"',
 '000 "Фонд национального кино "Патриот"',
 '000 "Киностудия "Медведь"',
 '000 Кинокомпания "Инвада Фильм", 000 "Артлайт"',
 '000 "Арк Пикчерс", 000 "Русская Фильм Группа", 000 "Корпорация Русская Фильм Группа"',
 'Грид Анимэйшн, Роллинг Пикчерс, 000 "КиноАтис"',
 '000 "СТГ МЕДИА" по заказу 000 "Киностудия КИТ"',
 'НФ "Пример интонации (Фонд Александра Сокурова)"',
 'ЗАО "Мотор фильм студия"', '5 ПРОДАКШН", 000 "58',
 '8Heads Productions, Furia Film, 000 "Кинокомпания "Гейт"',
 'АО "ТПО "СДФ", 000 "Кино-Арт"',
 '000 МИП "Продюсерский центр СПбГИКИТ-Дебют"',
 '000 "Студия ТРЕТИЙ РИМ"', '000 "Киностудия "АРК-ФИЛЬМ"',
 '000 "Кинокомпания "Спутник"', '000 "Кинокомпания Большое кино"',
 'Arizona Productions, Homeless Bob, 000 "Пан-Атлантик Студио"',
 'Компания "SIRENA film", Компания "TASSE film", 000 "Студия МЕТРАФИЛЬМС"',
 '000 "Продюсерская компания "Слово"',
 'ФГУП "ТПО "Киностудия "Союзмультфильм"',
 '000 "Кинокомпания Флагман" по заказу 000 "ММЕ"',
 '000 "Продюсерский центр "Парадиз", 000 "Белое Зеркало',
 '000 "Студия "8 рядов"',
 '000 "Кейстоун продакшн груп" по заказу АО "ВБД Груп"',
 '000 "КиноТрест"',
 '000 "Амедиа Продакшн" и АО "Телекомпания", 000 "Кинокомпания МАРС-фильм" п


```

о заказу 000 "ММЕ"',
    '000 "Продюсерская компания Александра Литвинова"',
    '000 "РЕНОВАЦИО интертейнмент"', '000 "Чайка Синема"',
    '000 "КИНОКОМПАНИЯ "ЗАДОРНОЕ КИНО"', '000 "Вайс Филмс"',
    '000 "Кинокомпания Сергея Дворцевого"',
    '000 Кинокомпания "Соливс" по заказу АО "ВБД Груп"',
    'АО "Ленфильм"', '000 "РБ Продакшн"',
    '000 "ARCHANGEL STUDIOS"', 000 "Кинокомпания Блесс-Фильм"',
    '000 "Кинокомпания "Небо" по заказу АО "ВБД Груп"',
    '000 "ТриоФильм"', '000 "Мастерская Павла Лунгина"',
    'Автономная некоммерческая организация "Творческая студия "Стелла"',
    '000 "Медиа Универсал Ивент"', '000 "БЛИЦ ФИЛЬМ"',
    '000 "Компания Парк Синема"', '000 "Киностудия Соль"',
    '000 "САГА"', '000 "Компания "АТК-СТУДИО"',
    '000 "Пропеллер Продакшн", АО "Интерфест"', '000 "ВЕГА ФИЛЬМ"',
    '000 "АДРЕС ФИЛЬМ", 000 "Мастерская Павла Лунгина"',
    '000 "1-2-3 Продакшн", 000 "Друг Друга"',
    '000 "Лаборатория НТС", 000 "РФГ Корп"', '000 "Мармот-фильм"',
    '000 Кинокомпания "Глобус-фильм"',
    '000 Кинокомпания "Инвада Фильм"',
    '20 стэпс продакшнс, 000 "Кинокомпания Атлантик"',
    '000 "Студия "Рой"',
    '000 "Кинокомпания "СТВ", 5 Продакшн", 000 "58',
    'Лтд., Лтд., Лтд.; Чайна Интернэшнл Филм & Медиа Корп., Спаркл Ролл Медиа К
орпорейшн, 000 "Кинокомпания "СТВ"; 000 "РФГ Корп"; 000 "РФГ"; АО "Маринс Групп Ин
тертейнмент"; 000 "Фетисов Иллюзион"; ИП Петрухин; Чайна Филм Ко.',
    '000 "КиТ фильм", 000 "2Д Целлулоид"',
    '000 "Продюсерский центр "Молодежные инициативы"',
    '000 "Кей Ди Студиос"',
    '000 "Жан Виго Италия" (Jean Vigo Italia S.R.L.), НО Фонд Андрея Кончаловск
ого по поддержке кино и сценических искусств',
    '000 "Продюсерский центр "Новое Время"',
    '000 "Муравей Продакшн", 000 "Продюсерский центр "Новое Время"',
    '000 "Медиа Универсал"', '000 "Карго"', '000 "Лаборатория Кино"',
    '000 "ВитЭл"', '000 "Кремлин Филмз"',
    '"Артеми́с", "Гринлит", 000 "Зебра"', '000 "СОК"', 'АО "Аэроплан"',
    '000 "МЕМ-МЕДИА" по заказу АО "ВБД Груп"', 'ЗАО "Дирекция кино"',
    '000 "Водород 2011", 000 "Арт Пикчерс Студия"',
    'АО "ТПО "Киностудия им. М.Горького"', dtype=object)

```

```
In [158... len(data_st_sup['film_studio'].unique())
```

Out[158]: 247

Создадим категориальный столбец для студии фильма:

```
In [159... data_st_sup['cat_film_studio'] = pd.Categorical(data_st_sup['film_studio'], data_st
data_st_sup['cat_film_studio'] = data_st_sup['cat_film_studio'].astype('category')
```

```
In [160... data_st_sup.plot(kind='scatter', y='state_support_share', x='cat_film_studio', grid
```



```
In [161...] data_st_sup['cat_film_studio'].corr(data_st_sup['state_support_share'])
```

```
Out[161]: -0.10310434375720631
```

Значения корреляции говорят, что связи практически никакой нет.

```
In [162...] uniq = data_st_sup['film_studio'].unique()
```

Сгруппируем данные по студии и рассчитаем кол-во фильмов выпущенное студией с гос поддержкой и среднюю господдержку для этой студии

Отсортируем результат по убыванию кол-ва выпущенных фильмов и ограничим вывод кол-вом более 1 выпущенного фильма:

```
In [163...] pt = pd.pivot_table(data_st_sup, index='film_studio', values=['state_support_share']
pt.columns = ['film_studio', 'ratings_ss', 'count_ss', 'mean_ss']
pt.sort_values('count_ss', ascending=False)
pt[pt['count_ss'] > 1].sort_values('count_ss', ascending=False)
```

Out[163]:

	film_studio	ratings_ss	count_ss	mean_ss
215	ООО "ТаББаК"	5.000000	7	62.827143
212	ООО "Студия анимационного кино "Мельница"	6.314286	7	41.451429
96	ООО "Кинокомпания "СТВ"	6.183333	6	40.595000
58	ООО "ВИЗАРТ ФИЛЬМ"	6.300000	5	39.200000
153	ООО "Нон-Стоп Продакшн"	5.950000	4	58.040000
170	ООО "Продюсерский центр "Ленфильм"	5.800000	4	58.970000
83	ООО "Каро Продакшн"	5.375000	4	59.702500
112	ООО "Киностудия "ВЕРТИКАЛЬ"	5.825000	4	54.367500
203	ООО "Студия "СЛОН"	5.833333	3	56.640000
141	ООО "Магnum-фильм"	6.466667	3	45.123333
125	ООО "Компания "Новые люди"	5.666667	3	63.993333
159	ООО "Пиманов и партнеры"	5.000000	3	30.376667
55	ООО "ВВП Альянс"	3.866667	3	66.056667
165	ООО "Продюсерский Центр "Хорошо Продакшн"	6.000000	2	65.725000
200	ООО "Студия "Пассажир"	5.300000	2	65.745000
186	ООО "Русские Золотые Серии"	2.900000	2	56.800000
205	ООО "Студия "ТРИТЭ" Никиты Михалкова"	7.550000	2	68.680000
173	ООО "Продюсерский центр "Парадиз"	5.000000	2	56.645000
214	ООО "ТПО "РОК"	5.200000	2	66.460000
218	ООО "Ультра стори", ООО "Энджой мувиз"	4.450000	2	37.525000
220	ООО "Форс Медиа"	6.000000	2	55.870000
168	ООО "Продюсерский центр "ВГИК-Дебют"	6.450000	2	67.470000
17	АО "ТПО "Киностудия им.М.Горького"	6.100000	2	60.435000
149	ООО "Медиа Универсал Ивент"	6.700000	2	61.330000
164	ООО "Продюсерская фирма Игоря Толстунова"	6.150000	2	68.095000
154	ООО "Ол Медиа Компани"	6.100000	2	57.655000
25	ЗАО "Мотор фильм студия"	6.300000	2	63.130000
124	ООО "Компания "АТК-Студио"	2.350000	2	70.000000
108	ООО "Кинокомпания Флагман" по заказу ООО "ММЕ"	6.350000	2	64.745000
104	ООО "Кинокомпания МАРС-фильм" по заказу ООО "М..."	5.850000	2	63.610000
92	ООО "Кинокомпания "Небо"	6.400000	2	55.395000
90	ООО "Кинодом"	6.800000	2	42.080000
82	ООО "Каргофильм"	5.250000	2	66.075000

	film_studio	ratings_ss	count_ss	mean_ss
66	ООО "ДжемСтудия"	6.250000	2	41.090000
63	ООО "Водород 2011"	6.250000	2	50.720000
46	ООО "Анимационная студия РИМ"	4.300000	2	23.060000
43	ООО "АРТЛАЙТ"	5.650000	2	65.185000
228	ООО "Энджой мувиз"	6.400000	2	48.490000

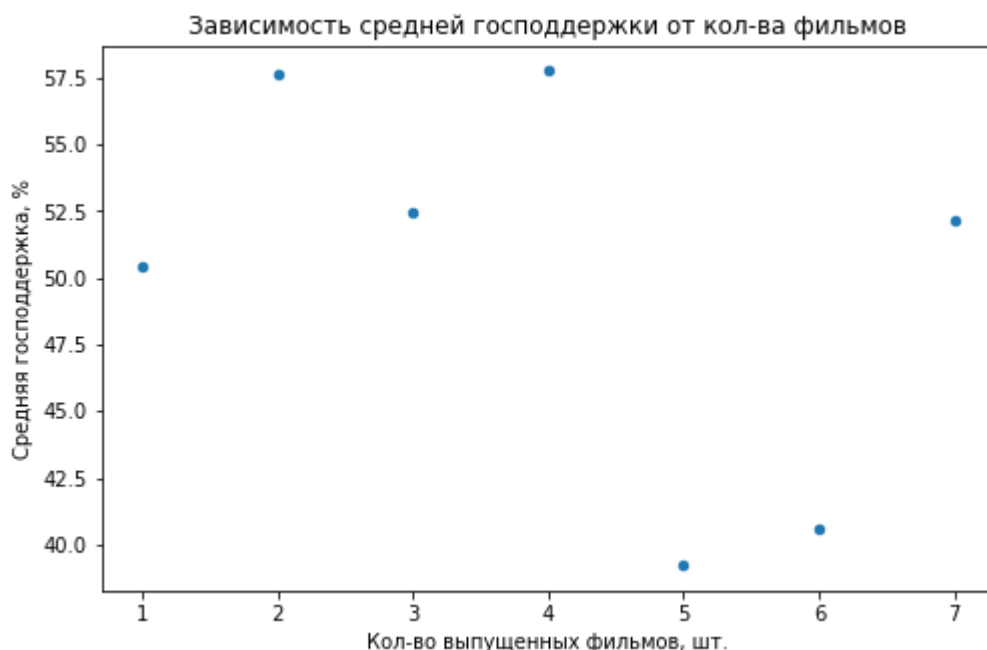
Есть подозрения, что господдержка убывает с ростом выпущенных выпущенных одной студией фильмов. Проверим, гипотезу:

```
In [164...] pt1 = pd.pivot_table(pt, index='count_ss', values='mean_ss').reset_index()
```

```
In [165...] pt1['count_ss'].corr(pt1['mean_ss'])
```

```
Out[165]: -0.4355003246394049
```

```
In [166...] pt1.plot(kind='scatter', x='count_ss', y='mean_ss', figsize=(8,5), xlabel='Кол-во в
```



Есть не очень сильная отрицательная зависимость - снижение господдержки с ростом кол-ва выпущенных фильмов с поддержкой

В целом мы видим, что есть компании, которые снимают фильмов с господдержкой больше чем остальные. Проверим сколько сняли эти же компании без господдержки:

Выберем данные, в которых отсутствует информация по господдержке и посчитаем кол-во выпущенных ими фильмов с 2010 по 2019 (весь период) и средний рейтинг:

```
In [167...] pt1 = data[data['state_support_share'] < 0].pivot_table(index='film_studio', values
```

```
pt1.columns = ['film_studio', 'ratings', 'q-ty']
pt1
```

Out[167]:

	film_studio	ratings	q-ty
2555	Киностудия "Мосфильм"	6.924348	345
3737	ООО "Мульт в кино"	0.572000	75
433	Анимаккорд Лтд	1.578261	23
5299	нет данных	1.788889	18
4945	ФГУП "ТПО "Киностудия "Союзмультфильм"	2.313333	15
...
1885	Дор Филм, Жордан Филмз, Интернэшнл Традерс, Ка...	7.400000	1
1884	Дор Филм Продукционгесселшафт, Бавария Пикчерз	6.600000	1
1883	ДоплоДок	0.000000	1
1882	Доорз Проперти, Игл Рок Интертейнмент, МортусКрин	8.400000	1
2650	Коррино Медиа Корпорейшн, Стаккато Филмз, Тулч...	7.100000	1

5300 rows × 3 columns

объединим два дата фрейма в один и получим информацию по каждой студии - рейтинг фильма с господдержкой, кол-во фильмов с господдержкой, средняя доля господдержки и средний рейтинг фильмов без господдержки, кол-во фильмов без господдержки 2010 по 2019:

In [168... `pt.merge(pt1, how='left', on='film_studio').sort_values('count_ss', ascending=False`

Out[168]:

	film_studio	ratings_ss	count_ss	mean_ss	ratings	q-ty
212	ООО "Студия анимационного кино "Мельница"	6.31	7	41.45	5.93	7.0
215	ООО "ТаББаК"	5.00	7	62.83	4.98	11.0
96	ООО "Киноккомпания "СТВ"	6.18	6	40.60	4.83	7.0
58	ООО "ВИЗАРТ ФИЛЬМ"	6.30	5	39.20	7.90	1.0
153	ООО "Нон-Стоп Продакшн"	5.95	4	58.04	6.48	4.0
83	ООО "Каро Продакшн"	5.38	4	59.70	4.80	1.0
112	ООО "Киностудия "ВЕРТИКАЛЬ"	5.83	4	54.37	6.78	5.0
170	ООО "Продюсерский центр "Ленфильм"	5.80	4	58.97	3.20	4.0
159	ООО "Пиманов и партнеры"	5.00	3	30.38	0.00	0.0
125	ООО "Компания "Новые люди"	5.67	3	63.99	0.00	2.0
203	ООО "Студия "СЛОН"	5.83	3	56.64	5.97	3.0
141	ООО "Магнум-фильм"	6.47	3	45.12	0.00	0.0
55	ООО "ВВП Альянс"	3.87	3	66.06	2.10	2.0
214	ООО "ТПО "РОК"	5.20	2	66.46	6.60	6.0
43	ООО "АРТЛАЙТ"	5.65	2	65.18	5.40	2.0
205	ООО "Студия "ТРИТЭ" Никиты Михалкова"	7.55	2	68.68	0.00	0.0
46	ООО "Анимационная студия РИМ"	4.30	2	23.06	2.40	1.0
92	ООО "Киноккомпания "Небо"	6.40	2	55.40	0.00	0.0
164	ООО "Продюсерская фирма Игоря Толстунова"	6.15	2	68.10	5.42	5.0
165	ООО "Продюсерский Центр "Хорошо Продакшн"	6.00	2	65.72	0.00	0.0
200	ООО "Студия "Пассажир"	5.30	2	65.74	6.70	1.0
218	ООО "Ультра стори", ООО "Энджой мувиз"	4.45	2	37.52	0.00	0.0
90	ООО "Кинодом"	6.80	2	42.08	6.80	1.0
168	ООО "Продюсерский центр "ВГИК-Дебют"	6.45	2	67.47	0.00	0.0
63	ООО "Водород 2011"	6.25	2	50.72	0.00	1.0
173	ООО "Продюсерский центр "Парадиз"	5.00	2	56.64	0.00	0.0
82	ООО "Каргофильм"	5.25	2	66.08	6.10	1.0
186	ООО "Русские Золотые Серии"	2.90	2	56.80	0.00	0.0
104	ООО "Киноккомпания МАРС-фильм" по заказу ООО "М..."	5.85	2	63.61	0.00	0.0
66	ООО "ДжемСтудия"	6.25	2	41.09	0.00	0.0

	film_studio	ratings_ss	count_ss	mean_ss	ratings	q-ty
154	ООО "Ол Медиа Компани"	6.10	2	57.66	0.00	0.0
108	ООО "Кинокомпания Флагман" по заказу ООО "ММЕ"	6.35	2	64.74	0.00	0.0
25	ЗАО "Мотор фильм студия"	6.30	2	63.13	0.00	0.0
220	ООО "Форс Медиа"	6.00	2	55.87	0.00	0.0
124	ООО "Компания "АТК-Студио"	2.35	2	70.00	0.00	0.0
17	АО "ТПО "Киностудия им.М.Горького"	6.10	2	60.44	7.70	1.0
149	ООО "Медиа Универсал Ивент"	6.70	2	61.33	0.00	0.0
228	ООО "Энджой мувиз"	6.40	2	48.49	4.61	8.0

Мы видим, что только не более 20% от всех компаний снимают без господдержки фильмов больше, чем с господдержкой. Это может означать, что многие студии создаются под конкретные проекты, с господдержкой. Мы также можем видеть, что у фильмов с господдержкой рейтинг выше: 5.68(с господдержкой) против 4.59(без господдержки).

4.3. Связь режисера фильма и господдержки

Посмотрим сколько раз тот или иной режисер снимал фильмы, с господдержкой:

```
In [169...] data_st_sup['director'].value_counts()
```

```
Out[169]: Р.Давлетьяров          4
          С.Андреасян          4
          Д.Дьяченко           4
          Е.(псевдоним)         3
          А.Аксененко          3
          ..
          А.Красовский          1
          Р.(псев.С.Васильев)   1
          Т.Цоцория, Я.Поляруш, К.при  1
          А.Пинхасов            1
          Е.Яцкина, А.Рубинштейн    1
          Name: director, Length: 268, dtype: int64
```

Здесь уже наблюдается небольшая зависимость, какие-то режисеры снимали фильмы с господдержкой несколько раз, а какие-то только единожды
продолжим наш анализ дальше:

Найдем уникальные значения столбца режисера фильма и их кол-во:

```
In [170...] data_st_sup['director'].unique()
```

```

Out[170]: array(['А.Атанесян', 'А.Якимчук', 'Р.Давлетьяров', 'С.Комаров',
                'Е.(псевдоним)', 'М.Угаров', 'С.Серегин', 'Р.Давлетьяров',
                'В.Глаголева', 'А.Войтинский', 'Д.Федоров', 'М.Свешников',
                'Т.Аленикова', 'А.Пиманов', 'А.Цицилин', 'А.Стриженов', 'С.Попов',
                'Д.Месхиев', 'В.Щегольков', 'А.Прошкин', 'П.Санаев', 'Ф.Коршунов',
                'С.Мокрицкий', 'А.Мельник', 'А.Аксененко', 'Ю.Кара', 'К.Сухарьков',
                'В.Демент', 'Ф.Абрютин', 'М.Бальчюнас',
                'Э.Архангельская', 'М.Пандурски', 'К.Шипенко', 'Д.Елеонский',
                'А.Звягинцев', 'Е.Татаров', 'А.Федорченко', 'Ю.Фетинг',
                'В.Харченко', 'А.Волков', 'И.Чижиков', 'А.Чижиков', 'А.Меликян',
                'Э.Манжеева', 'А.Калинкин', 'Н.Джорджадзе', 'В.Игнатьев',
                'Ф.Дмитриев', 'Р.Каримов', 'А.Аравин', 'А.Прошкин', 'С.Говорухин',
                'К.Ангелина', 'А.Карпиловский', 'С.Лоренци', 'В.Хотиненко',
                'В.при', 'С.Комаров', 'О.Асадулин', 'И.Северов',
                'А.Ананикян', 'В.Рейнгеверц', 'К.Белевич', 'Д.Чернов', 'А.Герман',
                'А.Темников', 'Р.Ильясов', 'С.Пускепалис', 'Н.Кудряшова',
                'Е.Хазанова', 'Э.Оганесян',
                'Р.Прыгунов', 'Р.Гигинеишвили', 'К.Оганисян', 'И.Джендубаев',
                'А.(псевдоним)', 'Ю.Попович', 'К.Кузин', 'И.Вырыпаев', 'Д.Дьяченко',
                'Э.Бордуков', 'А.Миндадзе', 'А.Нужный', 'Ш.Кромер', 'А.Галибин',
                'И.Кордзаия', 'Е.Баранов', 'Г.(Ю.Куценко)',
                'К.Дружинин', 'А.Шальопа', 'И.Волкова',
                'В.Николаев', 'М.Волков', 'А.Галат', 'Т.Воронецкая', 'Д.Сергин',
                'Н.Лебедев', 'А.Амиров', 'К.Геворкян', 'Т.Мамедов', 'Р.Артемьев',
                'А.Матисон', 'В.Сторожева', 'А.Чернакова', 'А.Котт', 'А.Китайцев',
                'О.Лопато', 'Т.Игуменцева', 'В.Лакисов', 'О.Беляева', 'Ю.Белюсева',
                'И.Угольников', 'А.Коваленко', 'М.Кравченко', 'А.Томашевский',
                'Д.Шмидт', 'А.Михалков', 'П.Руминов', 'Н.при', 'Н.Бондарчук',
                'А.Майовер', 'Д.Крылова', 'А.Соколов', 'М.Михайлова', 'Ю.Васильев',
                'М.мл', 'И.Болотников', 'А.Федотов', 'И.Твердовский',
                'С.Подгаевский', 'П.Лунгин', 'Д.Киселев', 'И.Шахназаров',
                'И.Учитель', 'Д.Светозаров', 'Н.Меркулова', 'А.Сурикова',
                'А.Мизгирев', 'А.Кончаловский', 'В.Бортко', 'А.Хант', 'Р.Шульман',
                'П.Чухрай', 'С.Гинзбург', 'М.Демченко', 'Н.Хомерики', 'В.Барбэ',
                'С.Андреасян',
                'А.Смирнова', 'Н.Кудряшова', 'Н.Назарова', 'О.Бычкова', 'Р.Литвинова', 'А.Пармас', 'А.Г
огленкова',
                'А.Лукьянчиков', 'С.Ушаков',
                'Б.Хлебников', 'В.Котт', 'А.Котт', 'А.Карпиловский', 'Ю.Кулаков',
                'И.Перин', 'Е.Борисова', 'С.Андреасян', 'Л.Марголин', 'М.Жерневский',
                'А.Шавкери', 'В.Перельман', 'И.Джендубаев', 'М.Лихачева', 'Р.Непомнящий', 'Т.Бекмамбе
тов', 'А.Котт',
                'Ю.Грымов', 'Э.Веливис', 'В.Тодоровский', 'В.Шевельков',
                'В.Татарский', 'К.Баскакова', 'Э.Уулу',
                'Ф.Дмитриев', 'Д.Шмидт', 'В.Торопчин', 'А.Касаткин', 'Ф.при',
                'Т.Корнев', 'А.Романец', 'М.Шевчук', 'А.Волгин', 'О.Карас',
                'К.Худяков', 'Р.Волобуев', 'М.Расходников', 'А.Учитель',
                'Т.(псевдоним)',
                'Е.Шелякин', 'Н.Меркулова', 'Н.Сайфуллаева', 'П.Руминов', 'Р.Гигинеишвили', 'А.Меликя
н', 'А.Чупов',
                'А.Андрющенко', 'К.Оганесян', 'Э.Парри',
                'И.Пшонкин', 'А.Колпин', 'В.Бедошвили', 'В.Марков', 'И.Шурховецкий',
                'К.Коваленко', 'Е.Жигаева', 'А.Богуславский', 'О.Зуева',
                'К.Максимов', 'О.Попова',
                'А.(Ж.Крыжовников)', 'А.(А.Нужный)', 'Д.Киселев', 'А.Карпиловский',
                'Д.Суворов', 'А.Мегердичев', 'К.Феоктистов',

```



```
'Г.Курлаев, М.и, С.Морозов, Е.Соколов, В.Панжев',
'T.Цоцория, Я.Поляруш, К.при', 'Д.Тюрин', 'Р.Мосафир', 'О.Трофим',
'Л.Марголин', 'Ф.Фархшатова', 'К.Васильев',
'М.Волков, В.Мухаметзянов', 'К.Дружинин', 'Д.Козловский',
'П.Дроздов', 'К.Хабенский, А.Малюков', 'А.Горчилин', 'А.Луканев',
'T.Жалнин', 'К.Малкин-Фам', 'М.Котик, И.Евланникова, Д.Моллер',
'Н.Мещанинова', 'В.Битоков', 'И.Максимов', 'Е.Шелякин',
'И.Волошин', 'В.Алфёров', 'Н.Михалкова', 'А.Игудин',
'К.(псевдоним, Г.Ильин', 'В.Алеников', 'Н.Меркулова, А.Чупов',
'М.Идов', 'А.Сухарев, Г.Шенгелия', 'С.Соколов', 'К.Плетнёв',
'К.Соколов', 'Т.Кеосаян', 'К.Бронзит', 'С.Черников',
'А.Пармас, А.(псевдоним, Е.Баранов, Т.Бекмамбетов, А.Котт',
'А.Сидоров', 'А.Цицилин, Р.Ленс', 'В.Потапов, А.Карелин',
'А.Велединский', 'В.Николаев, М.Бабенко', 'А.Сахелашвили',
'А.Баршак', 'Д.Панченко', 'П.Сидоров', 'М.(псевдоним',
'С.Дворцовой', 'И.Зайцев, О.Урушев', 'А.Золотухин', 'А.Домогаров',
'В.Фанасютина', 'Р.Гигинеишвили', 'Е.Бедарев', 'Ф.Попов',
'Р.Прыгунов', 'С.Саркисов', 'М.Агранович', 'Т.Тания', 'Н.Назарова',
'В.Гай', 'А.Якимчук, В.Лагунов', 'Е.Яцкина, А.Рубинштейн',
'А.Лунгин', 'Ю.Разыков', 'Н.Сайфуллаева', 'Р.Паушу', 'Б.Акопов',
'А.Пармас', 'Ю.Быков', 'П.Левченко', 'С.Проскурина',
'Р.(псев.С.Васильев)', 'А.Красовский', 'О.Степченко',
'М.Зыков, Ф.Абрютин', 'А.Аббасов', 'А.Попова',
'А.Богатырев, А.Камачо', 'Е.Михалков-Кончаловский', 'М.Морсков',
'К.Оганисян', 'А.Пинхасов', 'В.Чигинский', 'Н.Джорджадзе',
'О.Городецкая', 'И.Пшонкин, О.Ужинов, В.Бедошвили, Д.Цыбулькин',
'А.Кравчук', 'Д.Шмидт, К.Феокистов', 'Ф.Бондарчук'], dtype=object)
```

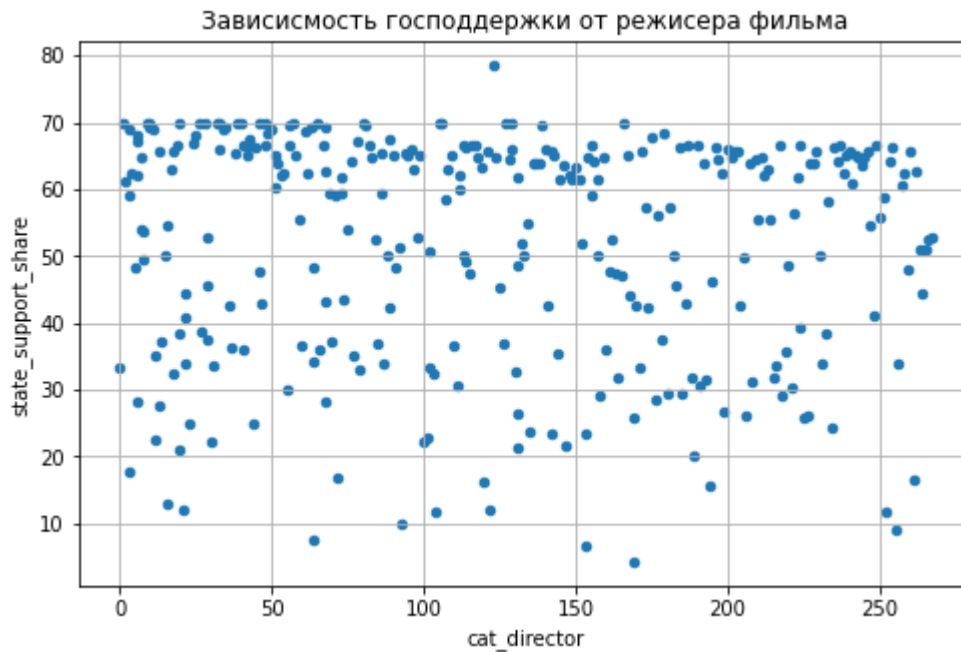
```
In [171]: len(data_st_sup['director'].unique())
```

```
Out[171]: 268
```

Создадим категориальный столбец для режисера фильма:

```
In [172]: data_st_sup['cat_director'] = pd.Categorical(data_st_sup['director'], data_st_sup['director'].unique())
data_st_sup['cat_director'] = data_st_sup['cat_director'].astype('category')
```

```
In [173]: data_st_sup.plot(kind='scatter', y='state_support_share', x='cat_director', grid=True)
```



```
In [174...] data_st_sup['cat_director'].corr(data_st_sup['state_support_share'])
```

```
Out[174]: -0.05583167359759007
```

Значения корреляции говорят, что связи практически никакой нет.

```
In [175...] uniq = data_st_sup['director'].unique()
```

Сгруппируем данные по режисеру и рассчитаем кол-во фильмов снятое режисером с гос поддержкой и среднюю господдержку по этому режисера
Отсортируем результат по убыванию кол-ва снятых фильмов и ограничим вывод кол-вом более 1 снятого фильма:

```
In [176...] pt = pd.pivot_table(data_st_sup, index='director', values=['state_support_share', '
pt.columns = ['director', 'ratings_ss', 'count_ss', 'mean_ss']
pt.sort_values('count_ss', ascending=False)
pt[pt['count_ss'] > 1].sort_values('count_ss', ascending=False)
```

Out[176]:

	director	ratings_ss	count_ss	mean_ss
209	Р.Давлетьяров	7.15	4	56.38
96	Д.Дьяченко	5.65	4	50.94
218	С.Андреасян	6.02	4	39.59
1	А.(псевдоним	6.17	3	30.05
159	К.Шипенко	6.83	3	45.27
109	Е.(псевдоним	5.17	3	48.60
3	А.Аксененко	5.30	3	39.79
225	С.Мокрицкий	6.60	3	43.15
201	П.Руминов	6.80	2	42.08
200	П.Лунгин	6.30	2	57.71
212	Р.Каримов	5.80	2	65.63
173	М.Свешников	3.00	2	69.70
172	М.Расходников	6.20	2	55.82
161	М.(псевдоним	4.90	2	53.01
158	К.Худяков	6.70	2	62.81
132	И.Твердовский	6.05	2	61.00
131	И.Северов	5.60	2	62.62
226	С.Подгаевский	4.65	2	58.34
188	Н.Хомерики	6.15	2	67.94
100	Д.Месхиев	6.80	2	33.75
89	В.Хотиненко	5.75	2	68.31
86	В.Тодоровский	7.45	2	44.48
77	В.Глаголева	6.25	2	59.44
63	А.Цицилин	6.30	2	46.76
60	А.Федорченко	6.65	2	68.03
57	А.Темников	6.75	2	68.05
48	А.Прошкин	5.55	2	49.06
45	А.Пиманов	4.75	2	28.90
35	А.Матисон	5.90	2	62.40
28	А.Котт	6.30	2	55.02
23	А.Карпиловский	6.85	2	58.81
15	А.Галибин	7.00	2	60.64
14	А.Волгин	5.65	2	14.99

	director	ratings_ss	count_ss	mean_ss
13	А.Войтинский	5.75	2	51.56
11	А.Богуславский	5.20	2	15.10
246	Ф.Дмитриев	6.80	2	51.41

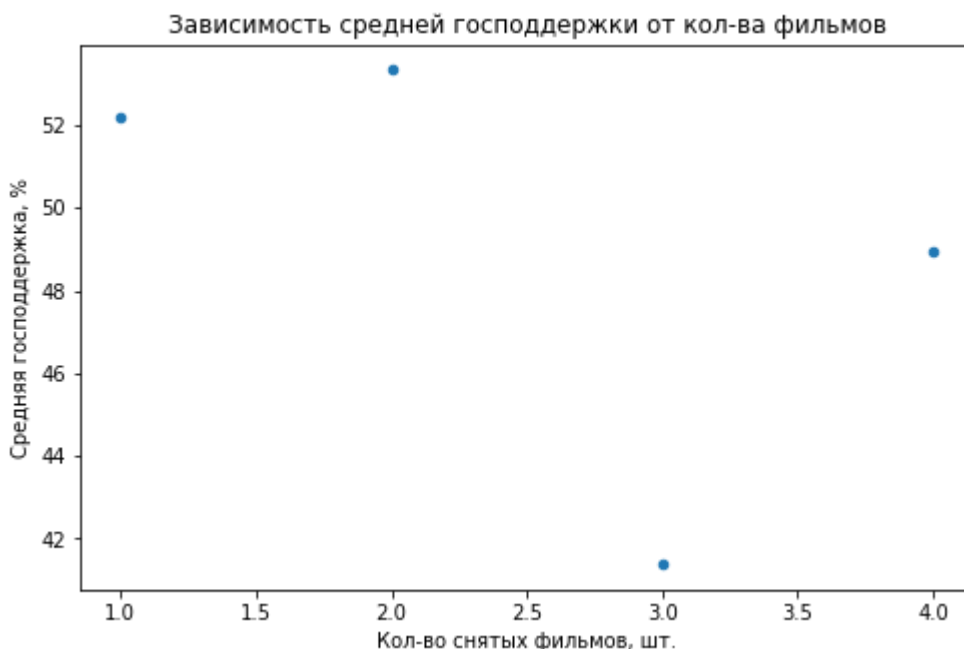
Посмотрим есть ли какая-то связь между кол-вом снятых фильмов и господдержкой:

```
In [177...] pt1 = pd.pivot_table(pt, index='count_ss', values='mean_ss').reset_index()
```

```
In [178...] pt1['count_ss'].corr(pt1['mean_ss'])
```

```
Out[178]: -0.5167052835857882
```

```
In [179...] pt1.plot(kind='scatter', x='count_ss', y='mean_ss', figsize=(8,5), xlabel='Кол-во с
```



Есть не очень сильная отрицательная зависимость - снижение господдержки с ростом кол-ва снятых фильмов с поддержкой

В целом мы видим, что есть компании, которые снимают фильмов с господдержкой больше чем остальные. Проверим сколько сняли эти же компании без господдержки:

Выберем данные, в которых отсутствует информация по господдержке и посчитаем кол-во выпущенных ими фильмов с 2010 по 2019 (весь период) и средний рейтинг:

```
In [180...] pt1 = data[data['state_support_share'] < 0].pivot_table(index='director', values=['
pt1.columns = ['director', 'ratings', 'q-ty']
pt1
```

Out[180]:

	director	ratings	q-ty
3015	О.Семёнова	0.68	63
1665	Д.Червяцов	1.33	18
3649	С.Содерберг	7.11	17
3374	Р.Скотт	7.09	16
3674	С.Уайт	5.70	15
...
1740	Е.Гроховская	6.30	1
1741	Е.Гусева	7.10	1
1742	Е.Данченко	0.00	1
1743	Е.Дербез	7.60	1
2229	К.Статский, Е.Соломина, А.Баршак	1.40	1

4459 rows × 3 columns

объединим два дата фрейма в один и получим информацию по каждому режисеру - рейтинг фильма с господдержкой, кол-во фильмов с господдержкой, средняя доля господдержки и средний рейтинг фильмов без господдержки, кол-во фильмов без господдержки с 2010 по 2019:

In [181...

```
pt.merge(pt1, how='left', on='director').sort_values('count_ss', ascending=False).f
```

Out[181]:

	director	ratings_ss	count_ss	mean_ss	ratings	q-ty
209	Р.Давлетьяров	7.15	4	56.38	4.60	3.0
96	Д.Дьяченко	5.65	4	50.94	6.88	4.0
218	С.Андреасян	6.02	4	39.59	3.50	7.0
225	С.Мокрицкий	6.60	3	43.15	5.10	1.0
3	А.Аксененко	5.30	3	39.79	5.80	4.0
159	К.Шипенко	6.83	3	45.27	7.07	3.0
109	Е.(псевдоним	5.17	3	48.60	4.48	5.0
1	А.(псевдоним	6.17	3	30.05	4.09	9.0
35	А.Матисон	5.90	2	62.40	6.45	2.0
172	М.Расходников	6.20	2	55.82	6.00	1.0
173	М.Свешников	3.00	2	69.70	0.00	0.0
45	А.Пиманов	4.75	2	28.90	0.00	0.0
48	А.Прошкин	5.55	2	49.06	7.00	2.0
226	С.Подгаевский	4.65	2	58.34	4.65	2.0
100	Д.Месхиев	6.80	2	33.75	7.20	1.0
57	А.Темников	6.75	2	68.05	0.00	0.0
28	А.Котт	6.30	2	55.02	5.48	4.0
60	А.Федорченко	6.65	2	68.03	3.08	4.0
63	А.Цицилин	6.30	2	46.76	0.00	0.0
188	Н.Хомерики	6.15	2	67.94	6.27	3.0
212	Р.Каримов	5.80	2	65.63	6.85	4.0
89	В.Хотиненко	5.75	2	68.31	0.00	1.0
77	В.Глаголева	6.25	2	59.44	0.00	0.0
86	В.Тодоровский	7.45	2	44.48	6.70	2.0
201	П.Руминов	6.80	2	42.08	4.10	2.0
246	Ф.Дмитриев	6.80	2	51.41	0.00	0.0
200	П.Лунгин	6.30	2	57.71	6.70	1.0
14	А.Волгин	5.65	2	14.99	0.00	1.0
15	А.Галибин	7.00	2	60.64	0.00	0.0
13	А.Войтинский	5.75	2	51.56	6.80	1.0
158	К.Худяков	6.70	2	62.81	7.20	2.0
23	А.Карпиловский	6.85	2	58.81	7.20	1.0
132	И.Твердовский	6.05	2	61.00	6.80	1.0

	director	ratings_ss	count_ss	mean_ss	ratings	q-ty
131	И.Северов	5.60	2	62.62	0.00	0.0
11	А.Богуславский	5.20	2	15.10	0.00	0.0
161	М.(псевдоним	4.90	2	53.01	0.00	2.0

Мы видим, что в отличии от студий режисеры снимают и фильмы с господдержкой и без поддержки. Мы также можем видеть, что у фильмов с господдержкой рейтинг такой же, как и фильмов без господдержки: 5.62(с господдержкой) против - 5.21(без господдержки). Это может говорить, что наличие и отсутствие господдержки не оказывает влияния на съемки фильма режисером. Что достаточно логично - режисер заботиться о своем имидже.

4.4. Связь продюсера фильма и господдержки

Посмотрим сколько раз тот или иной продюсер снимал фильмы, с господдержкой:

```
In [182... data_st_sup['producer'].value_counts()
```

```
Out[182]: С.Сельянов 8
А.Боярский, С.Сельянов 7
А.Рязанцев, Ю.Обухов 4
Е.Маскина, С.Говорухин 4
Е.Мелентьев, В.Денисюк 3
..
А.Рязанцев, В.Поляков, Г.Андреасян, С.Андреасян 1
О."ВИЗАРТ" 1
В.Поляков, Г.Малков 1
Д.Литвинов, В.Хотиненко 1
А.Войтинский, В.Поляков, Л.Карасев, Г.Малков 1
Name: producer, Length: 256, dtype: int64
```

Здесь уже наблюдается небольшая зависимость, какие-то продюсеры снимали фильмы с господдержкой несколько раз, а какие-то только единожды продолжим наш анализ дальше:

Найдем уникальные значения столбца продюсер фильма и их кол-во:

```
In [183... data_st_sup['producer'].unique()
```

```

Out[183]: array(['М.Бабаханов, Р.Бутко, А.Атанесян', 'А.Тютрюмов',
                  'А.Котелевский, А.Олейников, Р.Давлетьяров', 'Л.Антонова',
                  'Л.Антонова, А.Миндадзе', 'А.Герасимов', 'нет данных', 'Н.Иванова',
                  'С.Сельянов', 'В.Сидоренко', 'Е.Аронова, Т.Бекмамбетов',
                  'Т.Яковенко', 'О.Погодина, А.Пиманов', 'О."ТаББаК", О."ВИЗАРТ',
                  'А.Стриженов, Л.Лищинская, Л.Якубович', 'К.Шахназаров, А.Литвинов',
                  'И.Угольников', 'С.Новиков, А.Феофанов', 'Д.Пиркулов',
                  'А.Рязанцев, Ю.Обухов', 'Е.Олесов, Н.Мокрицкая', 'А.Мельник',
                  'В.Муругов, В.Поляков, И.Капитонов, Г.Малков', 'Ю.Кара',
                  'А.Новиков', 'Н.Будкина', 'С.Зернов, С.Кучмаева', 'О."ПЦ, О."Грин',
                  'А.Пискунова, Ю.Арабов, Э.Архангельская', 'О."ММЕ", Р.Дишдишян',
                  'И.Каленов', 'С.Мелькумов, А.Роднянский',
                  'А.Федорченко, Л.Лебедев, Д.Воробьев', 'И.Плиско', 'Ф.Попов',
                  'А.Меликян', 'А.Котелевский, Р.Давлетьяров', 'Я.Живов, Е.Гликман',
                  'В.Пономаренко, Д.Ефремов', 'И.Букреев, П.Поляков',
                  'А.Боярский, С.Сельянов', 'А.Котелевский, А.Новиков',
                  'Е.Маскина, С.Говорухин', 'Н.Расторгуев, И.Николенко', 'В.Есинов',
                  'К.Гечмен-Вальдек, Н.Горшкова, Ж.Пиччоли', 'В.Хотиненко, С.Кравец',
                  'А.Алькама, А.Бельская, А.Виткин, В.Малый, Г.Подземельный, К.Виткин, Р.Давл
етьяров',
                  'Н.Гостюшина, И.Мишин', 'Э.Пичугин', 'А.Ананикян',
                  'О."Беркана", О."ММЕ", Р.Дишдишян',
                  'Д.Рудовский, Ф.Бондарчук, И.Попов', 'А.Васильев',
                  'А.Новин, А.Темникова, Е.Бренькова, Е.Миронов',
                  'Ф.Попов, В.Мальшев',
                  'А.Качко, А.Новин, А.Темникова, Д.Аронин, Е.Бренькова, И.Гаврилов, Е.Мироно
в',
                  'М.Затуловская, Т.Бекмамбетов',
                  'Д.Жалинский, Р.Гигинеишвили, Э.Родоная, Э.Илоян', 'Т.Бекмамбетов',
                  'А.Казаков, А.(псевдоним, С.Корнихин, Т.Бекмамбетов',
                  'В.Щербин, О.Степанцова, А.Сафронов', 'Р.Давлетьяров',
                  'В.Зорин, Р.Атоян', 'С.Кучмаева, С.Зернов',
                  'Р.Татаринцев, Э.Илоян, Г.Шабанов',
                  'М.Дегтярь, Я.Живов, Е.Гликман',
                  'А.Анненский, В.Харьков, Л.Антонова, Л.Блаватник, Х.Декерт, А.Миндадзе',
                  'Р.Сорокин, А.Цекало', 'А.Шютте, Ш.Арндт, А.Гуськов', 'Б.Токарев',
                  'Е.Мелентьев', 'Г.Палагаев, И.Охлобыстин, Л.Калинский',
                  'А.Шальопа',
                  'А.Рубцов, А.Троцюк, В.Шляппо, Д.Дьяченко, Д.Жалинский, М.Ткаченко, Э.Илоя
н',
                  'А.Готлиб', 'О."Кинокомпания, О."ВИЗАРТ', 'Е.Белова',
                  'А.Златопольский, Н.Михалков, О."Студия, Л.Верещагин',
                  'М.Ксинопуло, Н.Иванова',
                  'А.Гюлумян, А.Манасарян, Д.Кулаков, Г.Нерсисян',
                  'В.Лихтина, В.Поляков, Г.Андреасян, Э.Акопян, С.Андреасян',
                  'М.Карнеева, А.Кублицкий', 'С.Говорухин, К.Шахназаров',
                  'Ю.Сапронов, А.Чернакова', 'Е.Филиппова', 'М.Линьков, В.Зверев',
                  'А.Гюлумян, А.Манасарян, В.Саргсян, Г.Нерсисян',
                  'А.Агеев, А.Гавриш, Д.Осмеркин, Д.Литвинов',
                  'В.Зорин, О.Бровченко, Р.Атоян', 'Р.Мусаев', 'С.Иванникова',
                  'А.Троцюк, В.Шляппо, Д.Д.Жалинский, И.Родоная, М.Ткаченко, Р.Гигинеишвили,
Э.Илоян',
                  'В.Поляков, Г.Малков', 'Д.Козловский, С.Бобза, С.Ливнев',
                  'Д.Улюкаев, С.Кикнавелидзе', 'С.Кабанова',
                  'А.Львович, А.Тельнов, В.Рубан',
                  'А.Соколов, М.Журомская, Э.Айнулова', 'А.Ковтунец',

```


'В.Кузнецов, Е.Яцура', 'М.Журомская, Н.Дорошкевич, Э.Айнулова',
 'М.мл, Г.Шадур', 'А.Сигле',
 'А.Кушаев, А.Троцюк, В.Шляппо, Д.Жалинский, Э.Илоян',
 'Н.Мокрицкая',
 'А.Котелевский, А.Троцюк, В.Шляппо, Д.Жалинский, Э.Илоян, А.Новиков',
 'Д.Литвинов, Э.Болотаев, В.Северцев',
 'Д.Рудовский, Е.Панфилов, Ф.Бондарчук, П.Лунгин',
 'Е.Миронов, Т.Бекмамбетов', 'Ф.Пастухов', 'А.Пиманов',
 'А.Сигле, Д.Светозаров', 'А.Семенов', 'А.Голутва',
 'Ф.Дайле, А.Кончаловский', 'Н.Бортко', 'Ю.Рязанов, В.Есинов',
 'Ф.Попов, В.Мадышев',
 'А.Кушаев, А.Троцюк, В.Шляппо, Д.Жалинский, И.Бурец, Э.Илоян, Д.Нелидов',
 'С.Еремеева', 'Е.Гордецкая, А.Радько', 'Н.Горина, С.Снежкин',
 'С.Козлов, И.Толстунов',
 'А.Рязанцев, В.Поляков, Г.Андреасян, С.Андреасян',
 'Н.Дрозд, Н.Смирнова, С.Сельянов, Э.Пичугин',
 'О."КИНОКОМПАНИЯ, О."ММЕ", Р.Дишдишян', 'О."Стар, О."Централ',
 'А.Войтинский, В.Поляков, Л.Карасев, Г.Малков', 'А.Котт, А.Цекало',
 'Ф.поддержки, Ю.Рязанов', 'А.Остапко, Е.Борисова', 'О."ВИЗАРТ',
 'А.Ананикян, А.Рязанцев, В.Колесник, В.Поляков, Г.Андреасян, М.Галустян, С.
 Андреасян',
 'И.Стромилова, М.Затуловская, Т.Бекмамбетов', 'С.Зернов, Ю.Грымов',
 'В.Горяинов, Л.Ярмольник, Р.Дишдишян',
 'А.Златопольский, В.Тодоровский', 'М.Бибенина, В.Шевельков',
 'Ю.Бахшиев', 'С.Безруков', 'Ю.Сапронов', 'А.Ревва',
 'Е.Вильшанская, М.Кржижевская', 'Е.Мелентьев, В.Денисюк',
 'А.Пескова, Д.Рудовский, Ф.Бондарчук, Ю.Храпов, В.Соловьев',
 'В.Ремизов, Е.Бабенко', 'М.Османн, Ю.Козырев, И.Стюарт',
 'А.Кузьминов, В.Поляков, Г.Шабанов, Е.замалиев, Ж.Бадмацыренов, Э.Илоян, Г.
 Малков',
 'А.Достман, В.Винокур, А.Учитель', 'С.Зернов',
 'А.Андрющенко, В.Муругов, М.Врубель, С.Сельянов', 'П.Иванова',
 'А.Златопольский, Б.Бакурадзе, С.Сельянов',
 'А.Троцюк, В.Шляппо, Д.Жалинский, М.Озкан, Э.Илоян',
 'М.Мамулина, Ф.Добронравов, Ю.Воронова, Е.Сарычева', 'Г.Васильев',
 'К.Буслов', 'А."Телекомпания", О."Централ', 'Н.Янкин',
 'И.Петрухин, О."Магнум, О.ДО, О."Корпорация',
 'В.Поляков, М.Кучмент, Г.Малков',
 'И.Козловский, О."ДК, О."ММЕ", Р.Дишдишян',
 'А.Пиманов, Д.Анисимов, О.Погодина', 'В.Сторожева, В.Брумберг',
 'Г.Стрижевская, Т.Бекмамбетов', 'О."Продюсерский',
 'А.Царик, К.Кутуев, Р.Витрянюк, Т.Цоцория', 'О."Централ',
 'А.Андрющенко, А.Златопольский, Д.Рудовский, Ф.Бондарчук, М.Врубель',
 'Г.Андреасян, С.Андреасян, А.Ананикян',
 'Р.Хаит, С.Петрейков, С.Сельянов, Л.Барац',
 'А.Рыданов, П.Попов, И.Толстунов',
 'А.Кремер, А.Черномазов, Е.Кононенко, И.Найшуллер, Н.Куликов, С.Корнихин',
 'О.Антипов, Д.Щербанов', 'Д.Месхиев, С.Гармаш, А.Златопольский',
 'А.Златопольский, Л.Верещагин, Н.Михалков, П.Ануров, Д.Козловский',
 'А.Петрухин', 'А.Васильев, М.Ложевский, А.Савельев',
 'Т.Варжапетян, Е.Гликман', 'М.Журомская, Э.Айнулова',
 'Д.Коцюбинский, М.Волобуев', 'А.Новиков, А.Котелевский',
 'И.Малкин-Фам, И.Петрухин, О."Арк, О."Русская, О."Корпорация',
 'А.Политовски, В.Сотсков, Ж.Ватеркейн, М.Мертенс, С.Зернов, Я.Гуссен, М.Кот
 ик',
 'О."Киностудия', 'Г.Подземельный, Д.Фикс, М.Курбатов',

```

'Я.Драбкин, А.Гуськов',
'А.Воронкова, А.Гутовская, Д.Сичел, Ж.Тедеева-Калинина, И.Волошин, Л.Филосо
ва',
'В.Харьков, Е.Григорьев, А.Тельнов',
'А.Михалкова, К.Комолова, М.Королев, Н.Михалкова',
'А.Смирнов, А.Евменов', 'Д.Потемкин',
'О."Киноспутник", О."Кинокомпания',
'А.Ананикян, Г.Андреасян, С.Андреасян', 'В.Глухов',
'Б.Машковцев, Н.Маковский, С.Струсовский, Ю.Слащева',
'А.Епифанов, О.Филипук, Т.Петрик, Р.Дишдишян', 'С.Кикнавелидзе',
'Д.Кеосаян', 'А.Троцюк, В.Шляппо, Д.Жалинский, Э.Илоян',
'А.Агеев, А.Горский, А.Плотников, Н.Агеев, П.Гудков, З.Болотаев',
'Л.Блаватник, Н.Михалкова, Н.при, Р.Дишдишян, А.Златопольский',
'О."Централ, О."ВИЗАРТ', 'А.Литвинов',
'В.Игнатъев, Р.Дишдишян, В.Поляков',
'В.Верещагин, П.Степанов, Г.Шабанов',
'Е.Север, Т.Стацман, М.Гуцериев', 'О."КИНОКОМПАНИЯ',
'Д.Литвинов, В.Северцев', 'М.(псевдоним, А.Ревва', 'С.Дворцовой',
'А.Троцюк, В.Шляппо, Д.Жалинский, О.Урушев, Э.Илоян',
'В.Шевц, Г.Куценко, Д.Земсков, Т.Куранова, В.Быркин',
'А.Кушаев, А.Троцюк, В.Шляппо, Д.Жалинский, Р.Гигинеишвили, Э.Илоян',
'А.Шисканов, В.Горяинов, Д.Щербанов, К.Елкин, М.Беспалов, О.Антипов, Р.Бори
севич, С.Куликов, С.Торчилин',
'Е.Панфилов, П.Лунгин', 'Б.Анисимов, В.Верещагин, В.Маслов',
'А.Жижневский, В.Колесник, Р.Юсипов, С.Саркисов',
'А.Роднянский, С.Мелькумов', 'С.Сельянов, Д.Добужинский',
'В.Бебия, С.Давдиев, Т.Тания', 'Д.Ковалевский',
'А.Васильев, В.Гай, А.Савельев', 'Г.Поземельный, Р.Давлетъяров',
'К.Фам, К.Михайлова',
'Д.Горелик, Е.Панфилов, П.Лунгин, С.Осипьян, С.Штерн, В.Кречетова',
'О."МАРС, Р.Дишдишян',
'В.Федорович, Е.Никишов, О."1-2-3, П.Лиленфельд, С.Корнихин',
'В.Тодоровский, Л.Ярмольник', 'В.Малышев, Ф.Попов',
'К.Эрнст, Н.Дрозд-Макан, С.Сельянов',
'А.Новиков, И.Есин, А.Котелевский', 'И.Лопатёнок, С.Зернов',
'Д.Литвинов, В.Хотиненко', 'Д.Фикс, М.Курбатов, Т.Каримов',
'С.Сельянов, А.Гуськов', 'А.Петрухин, С.Сельянов',
'А.Пескова, Д.Тульчинский, С.Квашилава, Ю.Храпов, В.Соловьев',
'Д.Якунин, О.Лахно, Ф.Абрютин', 'К.Саксаганская, А.Учитель',
'Н.Фонд', 'А.Разуваева, Ю.Перкуль, И.Добровольский',
'А.Леонтьев, И.Добровольский, Ю.Перкуль, О.Ледяева',
'В.Маслов, И.Магомедов, Б.Анисимов', 'К.Оганисян',
'А.Камынин, А.Седых, В.Галыгин, В.Колесник, Г.Атаев, Е.Михайлова',
'В.Поляков, Г.Малков, М.Кучмент, С.Бобза', 'О.Капанец',
'Ю.Соболевская',
'В.Ряшин, И.Бурец, М.Абрамов, Н.Трынкин, О.Филипук, Ф.Ламшин, П.Зарукин',
'И.Попов, С.Саркисов, Г.Васильев',
'А.Кушаев, А.Троцюк, В.Верещагин, В.Дусмухаметов, В.Шляппо, Д.Жалинский, Р.
Минасбекян, Т.Бадзиев, Э.Илоян',
'К.Эрнст, А.Максимов',
'А.Андрющенко, Д.Рудовский, М.Врубель, Ф.Бондарчук',
'А.Роднянский, М.Вайсберг, Р.Минасбекян, С.Мелькумов'],
dtype=object)

```

```
In [184... len(data_st_sup['producer'].unique())
```

Out[184]: 256

Создадим категориальный столбец для режисера фильма:

```
In [185...] data_st_sup['cat_producer'] = pd.Categorical(data_st_sup['producer'], data_st_sup['  
data_st_sup['cat_producer'] = data_st_sup['cat_producer'].astype('category')
```

```
In [186...] data_st_sup.plot(kind='scatter', y='state_support_share', x='cat_producer', grid=Tr
```



```
In [187...] data_st_sup['cat_producer'].corr(data_st_sup['state_support_share'])
```

Out[187]: -0.0780924105619028

Значения корреляции говорят, что связи практически никакой нет.

```
In [188...] uniq = data_st_sup['producer'].unique()
```

Сгруппируем данные по продюсеру и рассчитаем кол-во фильмов снятое продюсером с гос поддержкой и среднюю господдержку по этому продюсеру
Отсортируем результат по убыванию кол-ва снятых фильмов и ограничим вывод кол-вом более 1 снятого фильма:

```
In [189...] pt = pd.pivot_table(data_st_sup, index='producer', values=['state_support_share', '  
pt.columns = ['producer', 'ratings_ss', 'count_ss', 'mean_ss']  
pt.sort_values('count_ss', ascending=False)  
pt[pt['count_ss'] > 1].sort_values('count_ss', ascending=False)
```

Out[189]:

	producer	ratings_ss	count_ss	mean_ss
234	С.Сельянов	6.15	8	40.83
11	А.Боярский, С.Сельянов	6.31	7	41.45
70	А.Рязанцев, Ю.Обухов	5.38	4	59.70
144	Е.Маскина, С.Говорухин	5.83	4	54.37
246	Э.Пичугин	5.67	3	61.35
223	С.Еремеева	5.83	3	56.64
204	О."ММЕ", Р.Дишдишян	5.87	3	45.16
192	Н.Мокрицкая	5.67	3	63.99
146	Е.Мелентьев, В.Денисюк	4.80	3	12.29
81	А.Тютрюмов	1.57	3	68.33
151	Е.Филиппова	6.00	3	52.85
57	А.Петрухин	6.60	2	47.72
102	В.Муругов, В.Поляков, И.Капитонов, Г.Малков	4.45	2	37.52
18	А.Голутва	6.25	2	41.09
244	Ф.Попов, В.Малышев	7.05	2	68.37
242	Ф.Попов	3.10	2	53.88
237	Т.Бекмамбетов	3.90	2	69.42
19	А.Готлиб	5.25	2	66.08
232	С.Мелькумов, А.Роднянский	6.75	2	49.72
224	С.Зернов	5.30	2	64.97
28	А.Златопольский, Н.Михалков, О."Студия, Л.Вере...	7.55	2	68.68
218	Р.Татаринцев, Э.Илюян, Г.Шабанов	6.10	2	57.66
47	А.Меликян	6.60	2	36.08
200	О."Кинокомпания, О."ВИЗАРТ	6.25	2	34.40
195	Н.Янкин	6.60	2	65.60
50	А.Новиков	3.10	2	65.50
191	Н.Иванова	6.25	2	59.44
187	Н.Будкина	6.15	2	67.76
162	К.Буслов	5.35	2	55.46
161	И.Угольников	3.55	2	39.42
12	А.Васильев	6.25	2	27.96
53	А.Новин, А.Темникова, Е.Бренькова, Е.Миронов	6.75	2	68.05
54	А.Остапко, Е.Борисова	5.45	2	64.00

	producer	ratings_ss	count_ss	mean_ss
143	Е.Гордецкая, А.Радько	3.80	2	66.62
122	Г.Стрижевская, Т.Бекмамбетов	6.35	2	56.50
117	Г.Андреасян, С.Андреасян, А.Ананикян	5.75	2	35.71
255	нет данных	7.30	2	65.96

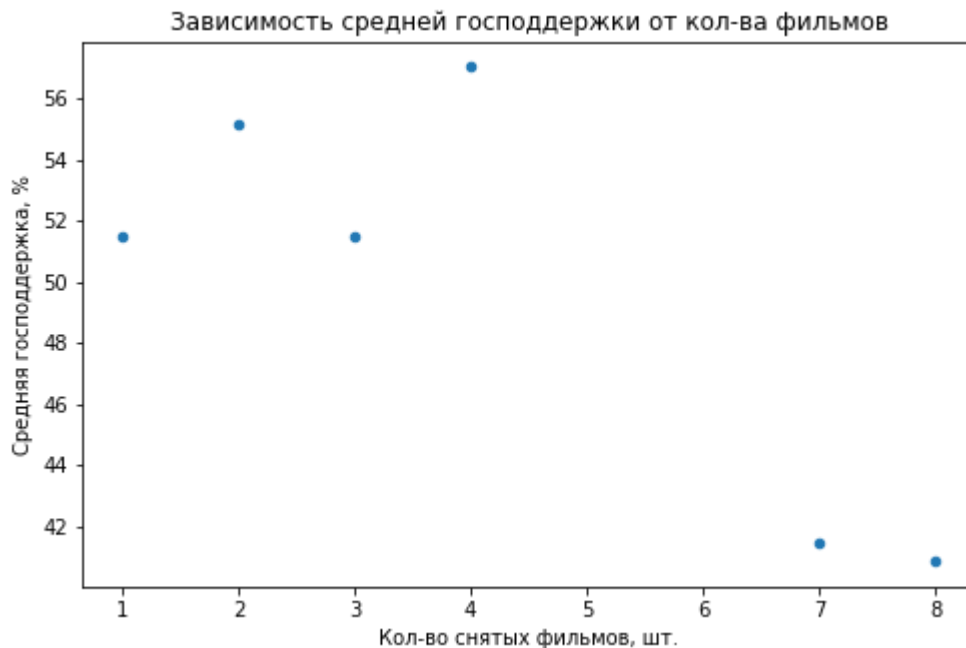
Посмотрим есть ли какая-то связь между кол-вом снятых фильмов и господдержкой:

```
In [190...] pt1 = pd.pivot_table(pt, index='count_ss', values='mean_ss').reset_index()
```

```
In [191...] pt1['count_ss'].corr(pt1['mean_ss'])
```

```
Out[191]: -0.8155463738020683
```

```
In [192...] pt1.plot(kind='scatter', x='count_ss', y='mean_ss', figsize=(8,5), xlabel='Кол-во с
```



Есть отрицательная зависимость - снижение господдержки с ростом кол-ва снятых фильмов с поддержкой

В целом мы видим, что есть продюсеры, которые снимают фильмов с господдержкой больше чем остальные. Проверим сколько сняли эти же продюсеры без господдержки:

Выберем данные, в которых отсутствует информация по господдержке и посчитаем кол-во снятых ими фильмов с 2010 по 2019 (весь период) и средний рейтинг:

```
In [193...] pt1 = data[data['state_support_share'] < 0].pivot_table(index='producer', values=['
pt1.columns = ['producer', 'ratings', 'q-ty']
pt1
```

Out[193]:

	producer	ratings	q-ty
5084	нет данных	5.89	566
4126	П.Смирнов	0.62	69
1902	Д.Ловейко, М.Ратина, О.Кузовков	1.30	28
3986	О.Кузовков, Д.Ловейко	0.93	17
604	А.Тельнов	3.32	12
...
1810	Д.Клифтон, П.Сафран, Д.Блум	4.30	1
1809	Д.Клифтон, Д.Брукс	5.30	1
1807	Д.Клейкенс	6.40	1
1806	Д.Клацкевич, Б.Брайн	7.80	1
5085	с.Дональд, Т.Ковент-Гарден	0.00	1

5086 rows × 3 columns

объединим два дата фрейма в один и получим информацию по каждому продюсеру - рейтинг фильма с господдержкой, кол-во фильмов с господдержкой, средняя доля господдержки и средний рейтинг фильмов без господдержки, кол-во фильмов без господдержки с 2010 по 2019:

In [194... `pt.merge(pt1, how='left', on='producer').sort_values('count_ss', ascending=False).f`

Out[194]:

	producer	ratings_ss	count_ss	mean_ss	ratings	q-ty
234	С.Сельянов	6.15	8	40.83	5.92	10.0
11	А.Боярский, С.Сельянов	6.31	7	41.45	7.35	2.0
144	Е.Маскина, С.Говорухин	5.83	4	54.37	6.78	5.0
70	А.Рязанцев, Ю.Обухов	5.38	4	59.70	4.80	1.0
151	Е.Филиппова	6.00	3	52.85	0.00	0.0
223	С.Еремеева	5.83	3	56.64	6.47	3.0
246	Э.Пичугин	5.67	3	61.35	4.25	6.0
81	А.Тютрюмов	1.57	3	68.33	4.00	3.0
204	О."ММЕ", Р.Дишдишян	5.87	3	45.16	6.57	3.0
192	Н.Мокрицкая	5.67	3	63.99	4.27	6.0
146	Е.Мелентьев, В.Денисюк	4.80	3	12.29	6.20	3.0
195	Н.Янкин	6.60	2	65.60	6.70	1.0
54	А.Остапко, Е.Борисова	5.45	2	64.00	6.50	3.0
117	Г.Андреасян, С.Андреасян, А.Ананикян	5.75	2	35.71	0.00	0.0
53	А.Новин, А.Темникова, Е.Бренькова, Е.Миронов	6.75	2	68.05	0.00	1.0
50	А.Новиков	3.10	2	65.50	5.75	2.0
102	В.Муругов, В.Поляков, И.Капитонов, Г.Малков	4.45	2	37.52	0.00	0.0
57	А.Петрухин	6.60	2	47.72	5.80	1.0
218	Р.Татаринцев, Э.Илоян, Г.Шабанов	6.10	2	57.66	0.00	0.0
162	К.Буслов	5.35	2	55.46	6.50	1.0
187	Н.Будкина	6.15	2	67.76	0.00	0.0
200	О."Кинокомпания, О."ВИЗАРТ	6.25	2	34.40	0.00	0.0
224	С.Зернов	5.30	2	64.97	3.85	2.0
232	С.Мелькумов, А.Роднянский	6.75	2	49.72	7.10	2.0
191	Н.Иванова	6.25	2	59.44	6.75	2.0
28	А.Златопольский, Н.Михалков, О."Студия, Л.Вере...	7.55	2	68.68	0.00	0.0
47	А.Меликян	6.60	2	36.08	0.00	0.0
255	нет данных	7.30	2	65.96	5.89	566.0
12	А.Васильев	6.25	2	27.96	0.00	0.0
122	Г.Стрижевская, Т.Бекмамбетов	6.35	2	56.50	0.00	0.0
143	Е.Гордецкая, А.Радько	3.80	2	66.62	0.00	0.0
237	Т.Бекмамбетов	3.90	2	69.42	6.28	4.0

	producer	ratings_ss	count_ss	mean_ss	ratings	q-ty
244	Ф.Попов, В.Малышев	7.05	2	68.37	0.00	0.0
242	Ф.Попов	3.10	2	53.88	6.65	4.0
18	А.Голутва	6.25	2	41.09	7.00	1.0
19	А.Готлиб	5.25	2	66.08	4.90	4.0

Мы видим, что в отличии от студий продюсеры снимают и фильмы с господдержкой и без поддержки. Мы также можем видеть, что у фильмов с господдержкой рейтинг такой же, как и фильмов без господдержки: 5.70(с господдержкой) против - 5.40(без господдержки). Это может говорить, что наличие и отсутствие господдержки не оказывает влияния на съемки фильма продюсером. Что достаточно логично - продюсер заботиться о своем имидже.

4.5. Связь возрастных ограничений и господдержки

Посмотрим сколько фильмов с различными возрастными ограничениями было снято с господдержкой:

```
In [195]: data_st_sup['age_restriction'].value_counts()
```

```
Out[195]: «16+» - для детей старше 16 лет      118
«12+» - для детей старше 12 лет      103
«6+» - для детей старше 6 лет        59
«18+» - запрещено для детей         27
«0+» - для любой зрительской аудитории  8
Name: age_restriction, dtype: int64
```

Здесь мы видим что максимальные значения по кол-ву выпущенных в прокат фильмов имеют категории "16+" и "12+"
продолжим наш анализ дальше:

Найдем уникальные значения столбца возрастные ограничения и их кол-во:

```
In [196]: data_st_sup['age_restriction'].unique()
```

```
Out[196]: array(['«16+» - для детей старше 16 лет',
                '«12+» - для детей старше 12 лет', '«6+» - для детей старше 6 лет',
                '«0+» - для любой зрительской аудитории',
                '«18+» - запрещено для детей'], dtype=object)
```

```
In [197]: len(data_st_sup['age_restriction'].unique())
```

```
Out[197]: 5
```

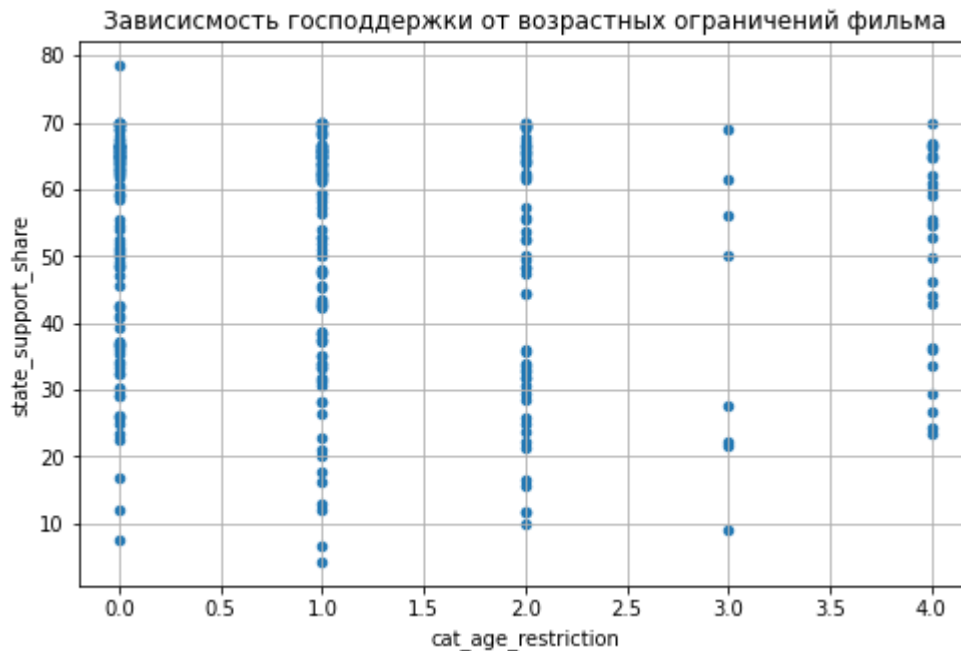
Создадим категориальный столбец для возрастных ограничений фильма:

```
In [198]: data_st_sup['cat_age_restriction'] = pd.Categorical(data_st_sup['age_restriction'],
```



```
data_st_sup['cat_age_restriction'] = data_st_sup['cat_age_restriction'].astype('cat
```

```
In [199... data_st_sup.plot(kind='scatter', y='state_support_share', x='cat_age_restriction',
```



```
In [200... data_st_sup['cat_age_restriction'].corr(data_st_sup['state_support_share'])
```

```
Out[200]: -0.11742178982342626
```

Значения корреляции говорят, что связи практически никакой нет.

```
In [201... uniq = data_st_sup['age_restriction'].unique()
```

Сгруппируем данные по возрастным ограничениям и рассчитаем кол-во фильмов снятое с тем или иным возрастным ограничением с гос поддержкой и среднюю господдержку по этому возрастному ограничению

Отсортируем результат по убыванию кол-ва снятых фильмов и ограничим вывод кол-вом более 1 снятого фильма:

```
In [202... pt = pd.pivot_table(data_st_sup, index='age_restriction', values=['state_support_sh
pt.columns = ['age_restriction', 'ratings_ss', 'count_ss', 'mean_ss']
pt.sort_values('count_ss', ascending=False)
pt[pt['count_ss'] > 1].sort_values('count_ss', ascending=False)
```

```
Out[202]:
```

	age_restriction	ratings_ss	count_ss	mean_ss
2	«16+» - для детей старше 16 лет	5.68	118	54.47
1	«12+» - для детей старше 12 лет	5.64	103	51.54
4	«6+» - для детей старше 6 лет	5.77	59	48.48
3	«18+» - запрещено для детей	5.80	27	51.24
0	«0+» - для любой зрительской аудитории	5.61	8	39.68

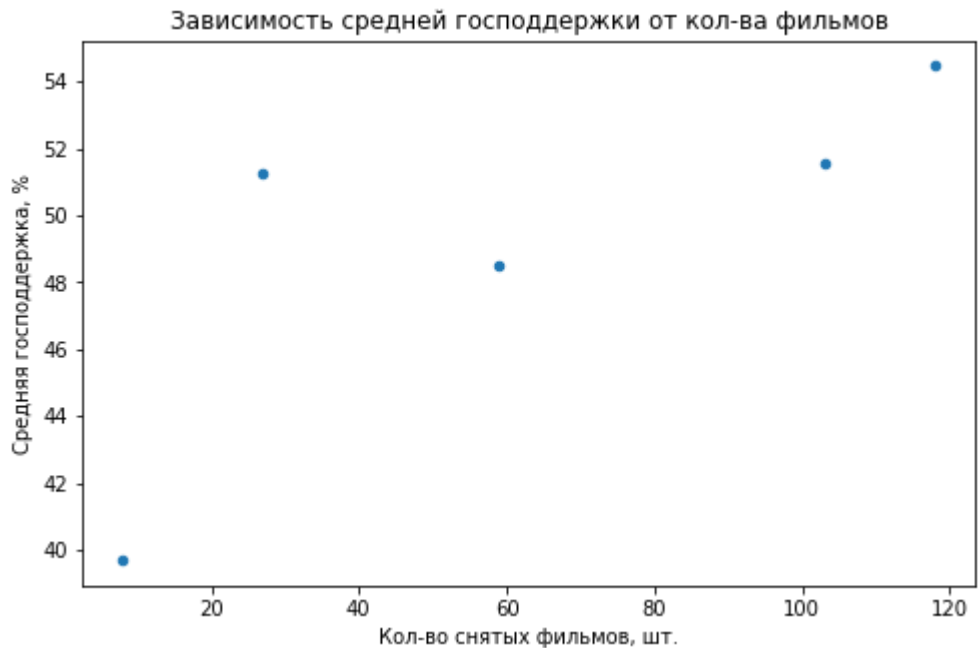
Посмотрим есть ли какая-то связь между кол-вом снятых фильмов с учетом возрастных ограничений и господдержкой:

```
In [203...] pt1 = pd.pivot_table(pt, index='count_ss', values='mean_ss').reset_index()
```

```
In [204...] pt1['count_ss'].corr(pt1['mean_ss'])
```

```
Out[204]: 0.7795156449002477
```

```
In [205...] pt1.plot(kind='scatter', x='count_ss', y='mean_ss', figsize=(8,5), xlabel='Кол-во с
```



Есть сильная прямая зависимость - рост доли господдержки с ростом кол-ва снятых фильмов

Посмотрим, какой рейтинг имеют фильмы без господдержки по возрастным ограничениям:

Выберем данные, в которых отсутствует информация по господдержке и посчитаем кол-во выпущенных ими фильмов с 2010 по 2019 (весь период) и средний рейтинг:

```
In [206...] pt1 = data[data['state_support_share'] < 0].pivot_table(index='age_restriction', va  
pt1.columns = ['age_restriction', 'ratings', 'q-ty']  
pt1
```

Out[206]:

	age_restriction	ratings	q-ty
2	«16+» - для детей старше 16 лет	6.04	2733
3	«18+» - запрещено для детей	5.90	1578
1	«12+» - для детей старше 12 лет	5.62	1489
0	«0+» - для любой зрительской аудитории	3.73	803
4	«6+» - для детей старше 6 лет	5.42	567

объединим два дата фрейма в один и получим информацию по каждому режиссеру - рейтинг фильма с господдержкой, кол-во фильмов с господдержкой, средняя доля господдержки и средний рейтинг фильмов без господдержки, кол-во фильмов без господдержки с 2010 по 2019:

In [207]: `pt.merge(pt1, how='left', on='age_restriction').sort_values('count_ss', ascending=F`

Out[207]:

	age_restriction	ratings_ss	count_ss	mean_ss	ratings	q-ty
2	«16+» - для детей старше 16 лет	5.68	118	54.47	6.04	2733
1	«12+» - для детей старше 12 лет	5.64	103	51.54	5.62	1489
4	«6+» - для детей старше 6 лет	5.77	59	48.48	5.42	567
3	«18+» - запрещено для детей	5.80	27	51.24	5.90	1578
0	«0+» - для любой зрительской аудитории	5.61	8	39.68	3.73	803

Мы видим, что возрастные ограничения фильмов с господдержкой и без поддержки практически одинаковы. Есть пара моментов: первый - фильмы "18+" с гос поддержкой снимаются реже (4 место в списке) чем без господдержки(занимает 2 место по кол-ву фильмов). Это может говорить о том, что господдержка направлена на фильмы с широкой аудиторией(для детей и взрослых). Второй момент - фильмы для "0+" с господдержкой имеют более высокий средний рейтинг, чем фильмы без господдержки. Возникает гипотеза, что есть определенные требования к фильмам с гос поддержкой по качеству материала и всем прочим моментам съемок, ниже которых финансирование не осуществляется. В целом у фильмов с господдержкой рейтинг такой же, как и фильмов без господдержки: 5.7(с господдержкой) против - 5.34(без господдержки).

4.6. Связь рейтинга фильма и господдержки

In [208]: `data_st_sup.plot(kind='scatter', x='state_support_share', y='ratings', grid=True, t`



```
In [209...] data_st_sup['ratings'].corr(data_st_sup['state_support_share'])
```

```
Out[209]: -0.08551210671245976
```

Согласно результату связь отсутствует

Посмотрим зависимость рейтинга от года выпуска в прокат:

```
In [210...] ax = pd.pivot_table(data_st_sup, values='ratings', index='show_year').fillna(0).res
y='ratings', title='Изменение среднего рейтинга фильмов по годам',figsize=(8,5), xl
pd.pivot_table(data_st_sup, index='show_year', values='film_studio', aggfunc='count
x='show_year', y='film_studio',secondary_y=True, ax=ax
)
```

```
Out[210]: <AxesSubplot:label='dc4522c8-0ea3-46ac-9bf8-f70612fd60b1'>
```



Средний рейтинг фильмов в 2013 году был достаточно высоким, затем в 2014 произошло снижение, с падением до нижнего значения в 2016. Скорее всего в 2016 году государственными органами были приняты меры по прекращению финансирования студий выпускающих фильмы с низким рейтингом, после чего кол-во студий временно упало и с 2017 началось восстановление вместе со средним рейтингом фильмов.

Посмотрим на рейтинг фильмов с господдержкой и без господдержки

In [211...

```
s1 = pd.pivot_table(data_st_sup, index='show_year', values='ratings', aggfunc='mean')
s1.columns = ['show_year', 'state_sup_ratings']
ax = s1.plot(x='show_year', y='state_sup_ratings', style='o-', grid=True, figsize=(8, 4))
s2 = data.query('state_support_share < 0').pivot_table(index='show_year', values='ratings')
s2.columns = ['show_year', 'no_state_sup_ratings']
s2.plot(x='show_year', y='no_state_sup_ratings', ax=ax, title='Рейтинги с гос поддержкой')
```

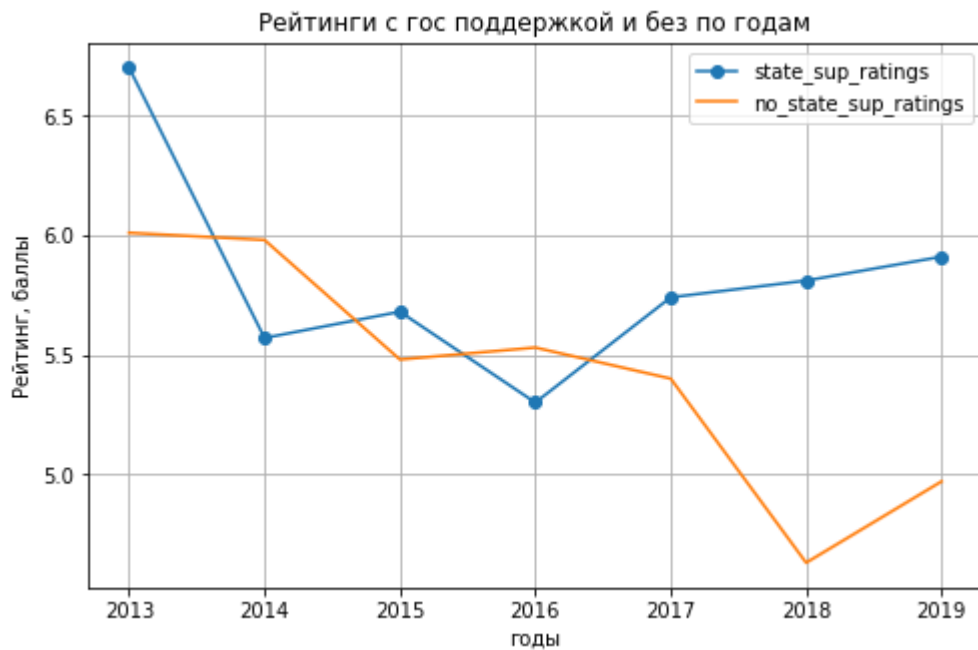


График показывает, что 2016 года рейтинги снижались, но после 2016 рейтинги фильмов с господдержкой поехали вверх, а рейтинги фильмов без господдержки продолжали падать.

4.7. Связь жанра фильма и господдержки

Жанр фильма является категориальным столбцов и самое главное строковым. Невозможно искать связь между строковым значением и числовым. Для этого нужно перевести строковое значение в числовое, или другими словами закодировать столбец 'genres'

Закодирем тип фильма кол-вом появлений в дата фрейме:

```
In [212... data_st_sup['movie_genre'].value_counts()
```

```
Out[212]: драма          99
          комедия        90
          боевик         25
          военный        22
          детектив        14
          биография       11
          мультфильм     9
          детский         7
          триллер         7
          приключения     6
          мелодрама       6
          ужасы           5
          криминал        4
          фантастика      2
          семейный        2
          история         2
          вестерн         1
          мюзикл          1
          реальное ТВ     1
          фэнтези         1
          Name: movie_genre, dtype: int64
```

```
In [213... len(data_st_sup['movie_genre'].unique())
```

```
Out[213]: 20
```

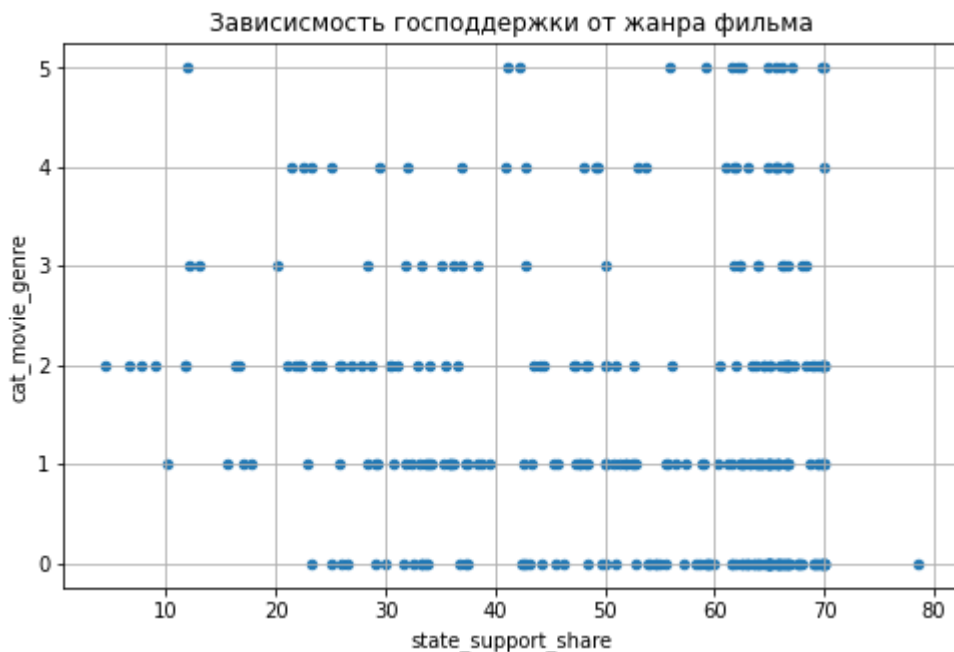
Мы видим, что на 5 первые жанров приходится почти 80% выпуска всех фильмов с господдержкой. Для удобства анализа оставим первые 5 категорий остальные поместим в категорию 'другой'

```
In [214... genres_list = ['драма', 'комедия', 'боевик', 'военный', 'детектив']
data_st_sup['movie_genre_new'] = data_st_sup['movie_genre'].apply(lambda x: x if x
```

Создадим категориальные столбец 'cat_type', который будет состоять из кодов типов фильмов, причем они будут отражать кол-во появления каждого типа в дата фрейме:

```
In [215... data_st_sup['cat_movie_genre'] = pd.Categorical(data_st_sup['movie_genre_new'], dat
data_st_sup['cat_movie_genre'] = data_st_sup['cat_movie_genre'].astype('category')
```

```
In [216... data_st_sup.plot(kind='scatter', x='state_support_share', y='cat_movie_genre', grid
```



```
In [217...] data_st_sup['cat_movie_genre'].corr(data_st_sup['state_support_share'])
```

Out[217]: -0.1338895999866057

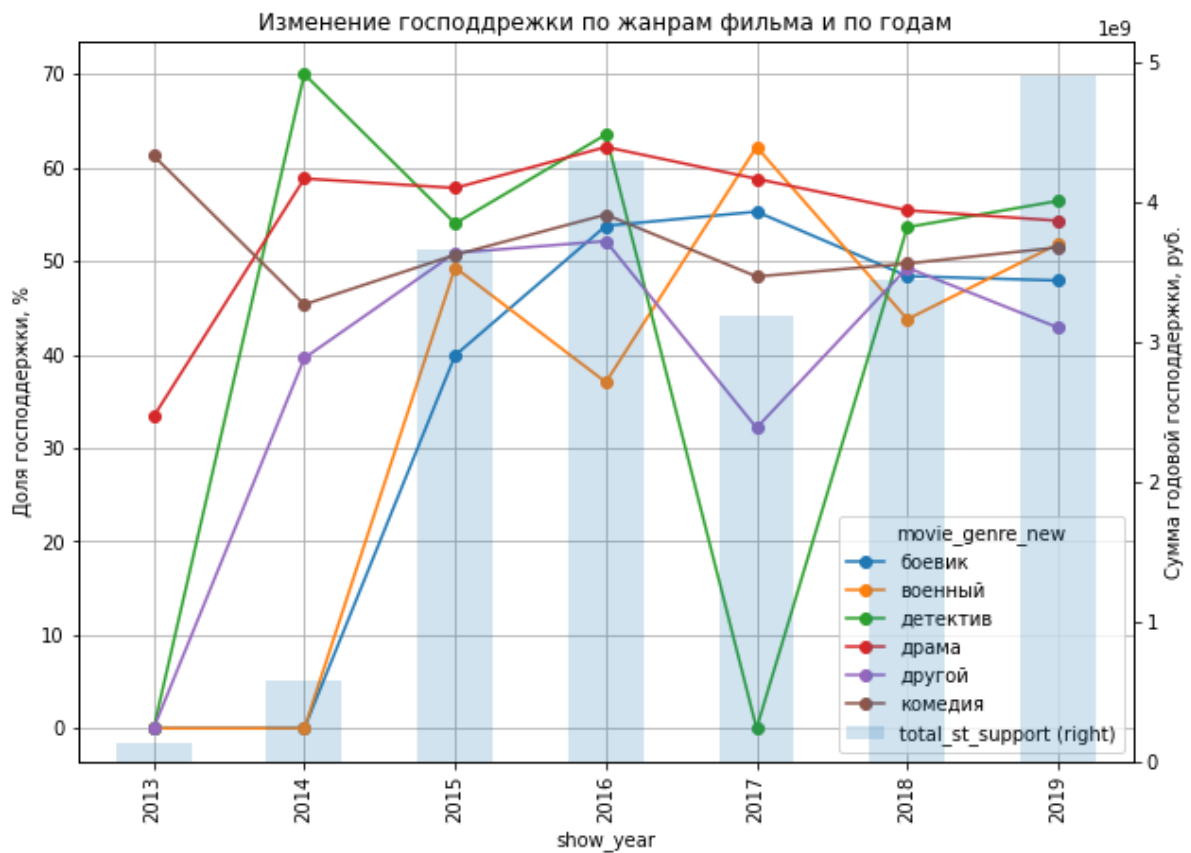
Посмотрим зависимость от года выпуска в прокат:

```
In [218...] pd.pivot_table(data_st_sup, columns='movie_genre_new', values='state_support_share')
```

Out[218]:

	movie_genre_new	show_year	боевик	военный	детектив	драма	другой	комедия
0	2013	0.000000	0.000000	0.000	33.370000	0.000000	61.270000	
1	2014	0.000000	0.000000	70.000	58.834286	39.620000	45.352500	
2	2015	39.906667	49.311667	54.032	57.821429	50.879286	50.664583	
3	2016	53.770000	37.035000	63.540	62.207500	52.136923	54.974118	
4	2017	55.287500	62.220000	0.000	58.805556	32.150000	48.346875	
5	2018	48.390000	43.768333	53.625	55.430000	49.294444	49.734615	
6	2019	47.920000	51.832000	56.440	54.321500	42.881176	51.458667	

```
In [219...] ax = pd.pivot_table(data_st_sup, columns='movie_genre_new', values='state_support_s
y=['боевик', 'военный', 'детектив', 'драма', 'другой', 'комедия'], title='Измен
data['total_st_support'] = data['refundable_support'] + data['nonrefundable_support']
pd.pivot_table(data, index='show_year', values='total_st_support', aggfunc=sum).res
ax.right_ax.set_ylabel('Сумма годовой господдержки, руб.');
```

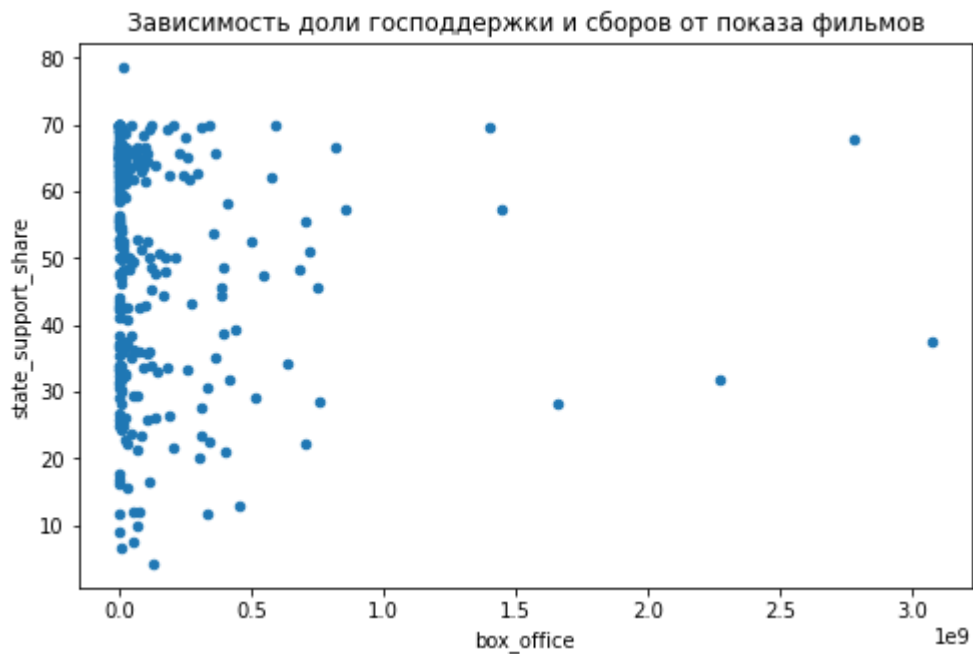



Коэффициент корреляции показывает отсутствие связи между жанром фильма и господдержкой, что подтверждает график изменения доли поддержки по годам - видно, что все жанры имеют господдержку в районе 40% - 60%. Видно, что в 2017 году произошел провал в поддержке всех жанров, за исключением военных фильмов и боевиков - в 2017 сумма господдержки была снижена (на графике столбцы светло-голубого цвета). До 2019 был разброс в поддержке того или иного жанра, но к 2019 все жанры стали поддерживаться примерно в одних пропорциях в районе 50%.

4.8. Связь сборов от показа и господдержки

Посмотрим, какие связи могут быть у сборов от показа фильмов и господдержки:

In [220... `data_st_sup.plot(kind='scatter', x='box_office', y='state_support_share', figsize=(`



```
In [221...] data_st_sup['box_office'].corr(data['state_support_share'])
```

```
Out[221]: -0.11559813665948603
```

Связи нет

Посмотрим, окупаются ли фильмы с господдержкой:

```
In [222...] round(data_st_sup['box_office'].mean() - data_st_sup['budget'].mean(), 2)
```

```
Out[222]: -1888940.22
```

Мы видим, что в среднем сборы фильмов меньше, чем вложенный бюджет, т.е. фильмы с господдержкой не окупаются

Сравним фильмы с господдержкой и без господдержки:

```
In [223...] s1 = pd.pivot_table(data_st_sup, index='show_year', values='box_office', aggfunc='s
s1.columns = ['show_year', 'state_sup_box_office']
ax = s1.plot(x='show_year', y='state_sup_box_office', style='o-', grid=True, figsize
s2 = data.query('state_support_share < 0').pivot_table(index='show_year', values='b
s2.columns = ['show_year', 'no_state_sup_box_office']
s2.plot(x='show_year', y='no_state_sup_box_office', ax=ax, title='Сборы от показа с г
```

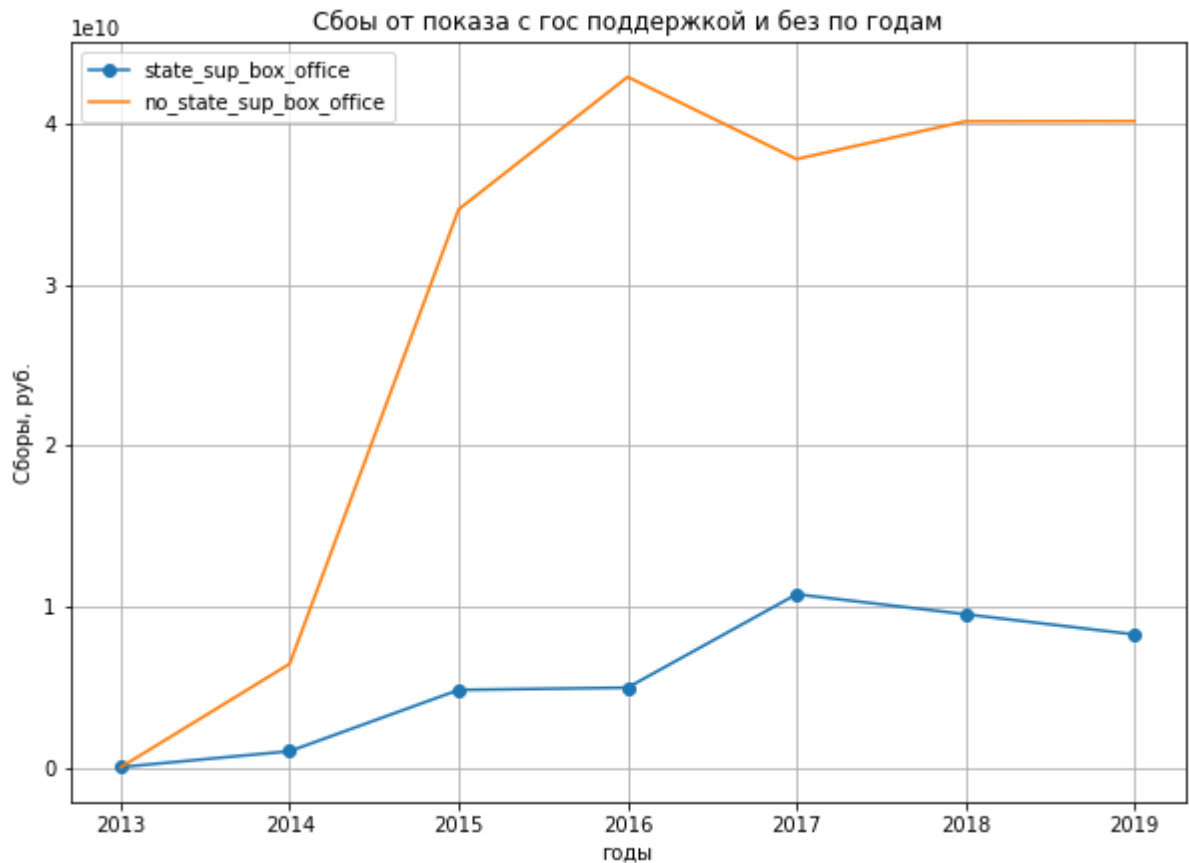


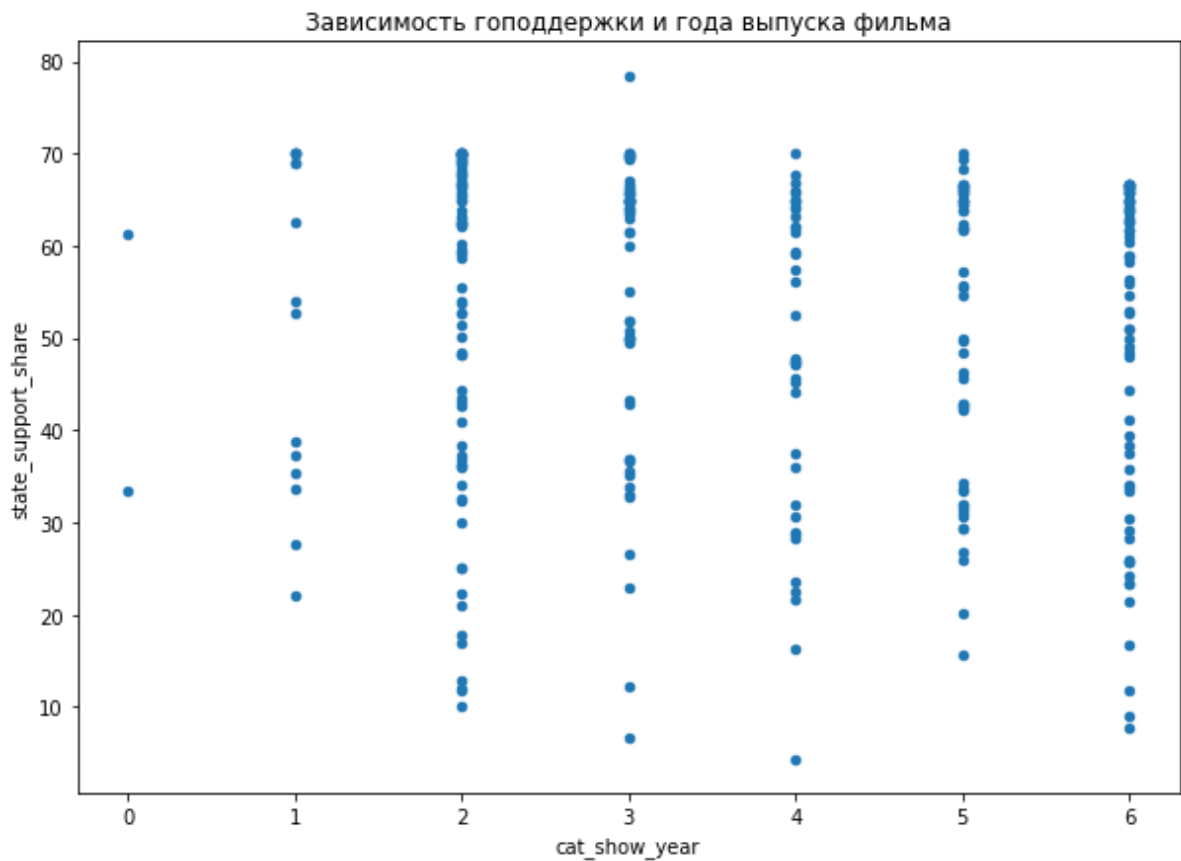
График показывает, что сборы за фильмы с поддержкой государства ниже сборов без поддержки более чем в четыре раза. Это говорит о меньшем интересе к фильмам с господдержкой со стороны зрителей.

4.9. Связь года выпуска в прокат и господдержки

Закодируем год выпуска:

```
In [224...] data_st_sup['cat_show_year'] = pd.Categorical(data_st_sup['show_year'], data_st_sup
data_st_sup['cat_show_year'] = data_st_sup['cat_show_year'].astype('category').cat.
```

```
In [225...] data_st_sup.plot(kind='scatter', x='cat_show_year', y='state_support_share', title=
```



```
In [226...] data_st_sup['cat_show_year'].corr(data_st_sup['state_support_share'])
```

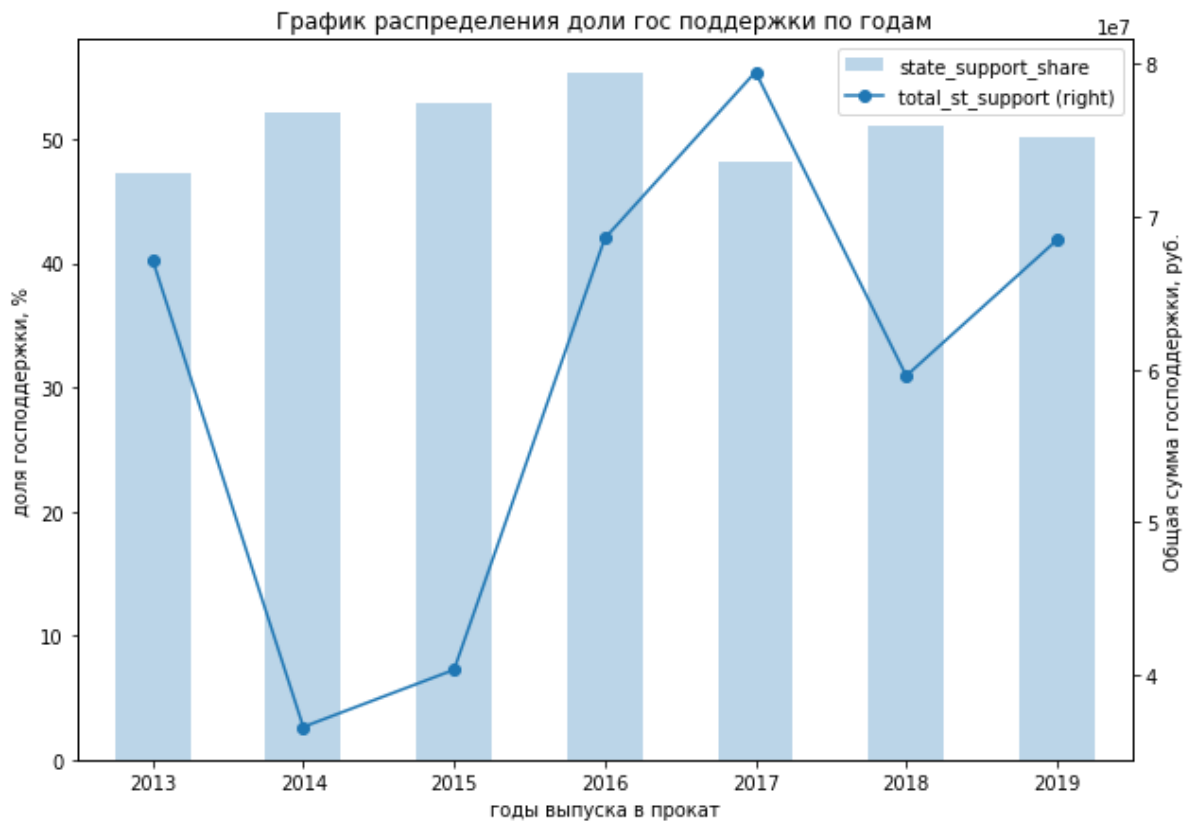
```
Out[226]: -0.07310684682641214
```

Корреляция показывает отсутствие связи

Посмотрим как распределялась господдержка по годам:

```
In [227...] data_st_sup['total_st_support'] = data_st_sup['refundable_support'] + data_st_sup['
```

```
In [228...] ax = data_st_sup.pivot_table(index='show_year', values='state_support_share').reset_
xlabel='годы выпуска в прокат', ylabel='доля господдержки, %', title='График распре
data_st_sup.pivot_table(index='show_year', values='total_st_support').reset_index()
ax.right_ax.set_ylabel('Общая сумма господдержки, руб.');
```



На графике мы видим, что колебания суммы господдержки по годам имеют место, особенно это было в начале периода(2013-2016), когда колебания были значительными. В последующие годы колебания, сократились. Если говорить про уровень поддержки в течение всего периода, то она сильно не изменялась и была примерно на уровне 50%, что и отразилось в коэффициенте корреляции (выше).

4.10. Связь страны производства фильма и господдержки

```
In [229...] data_st_sup['production_country'].unique()
```

```
Out[229]: array(['Россия', 'Латвия, Россия, Франция', 'Россия, Украина',
        'Италия, Россия', 'Польша, Россия, Украина',
        'Великобритания, Россия, ФРГ', 'Россия, ФРГ', 'Армения, Россия',
        'Россия, Словакия, Чехия', 'Латвия, Россия, Чехия',
        'Казахстан, Китай, Польша, Россия, ФРГ', 'Россия, Сербия',
        'Грузия, Россия', 'Китай, Россия', 'Бельгия, Россия, Финляндия'],
        dtype=object)
```

```
In [230...] len(data_st_sup['production_country'].unique())
```

```
Out[230]: 15
```

```
In [231...] data_st_sup['production_country'].value_counts()
```

```
Out[231]: Россия 299
Италия, Россия 2
Россия, ФРГ 2
Россия, Украина 1
Казахстан, Китай, Польша, Россия, ФРГ 1
Китай, Россия 1
Бельгия, Россия, Финляндия 1
Россия, Сербия 1
Латвия, Россия, Франция 1
Латвия, Россия, Чехия 1
Польша, Россия, Украина 1
Армения, Россия 1
Грузия, Россия 1
Великобритания, Россия, ФРГ 1
Россия, Словакия, Чехия 1
Name: production_country, dtype: int64
```

Мы видим, что основное кол-во фильмов с господдержкой - фильмы произведенные только в России. Часть фильмов, получившая господдержку была снята с участием других стран, но это кол-во менее 5%. Итог абсолютно очевиден - государство поддерживает фильмы, которые снимаются в России.

4.11. Связь источника финансирования и господдержки

Посмотрим какой источник финансирования больше участвовал в господдержке:

```
In [232]: data_st_sup['financing_source'].value_counts()
```

```
Out[232]: Министерство культуры 163
Фонд кино 131
Министерство культуры, Фонд кино 21
Name: financing_source, dtype: int64
```

Лидер по кол-ву поддержанных фильмов - Министерство культуры. Посмотрим, что по суммам поддержки.

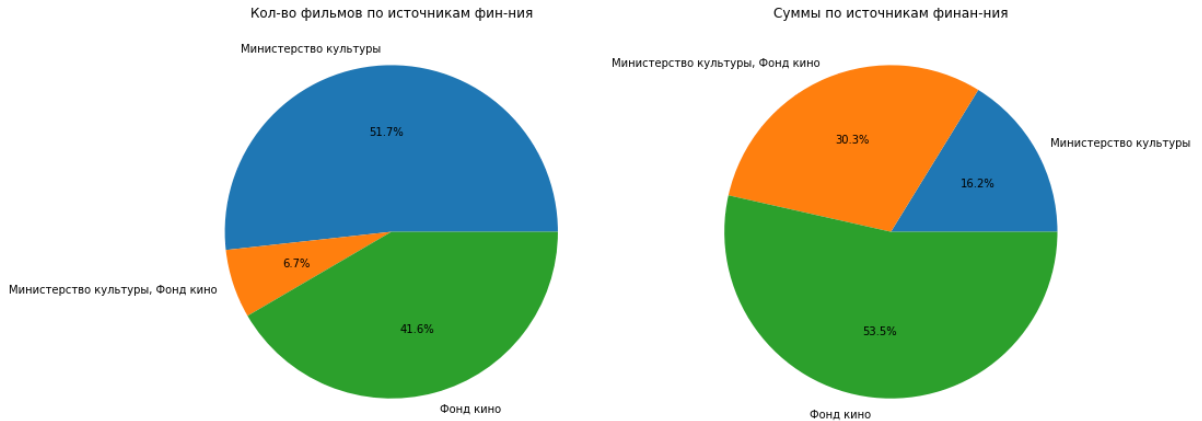
```
In [233]: pt1 = pd.pivot_table(data_st_sup, index='financing_source', values=['state_support_
pt1.columns = ['mean_ratings', 'mean_state_support_share', 'movie_qty', 'mean_total_st
pt1
```

```
Out[233]:
```

	mean_ratings	mean_state_support_share	movie_qty	mean_total_st_support
financing_source				
Министерство культуры	5.636196	52.965276	163	2.963596e+07
Министерство культуры, Фонд кино	6.123810	54.128095	21	5.535387e+07
Фонд кино	5.689313	49.829618	131	9.782921e+07

Построим pie chart для визуализации долей участия каждого источника финансирования в господдержке:

```
In [234... ax = plt.subplot(121)
pt1.plot(kind='pie', y='movie_qty',ax=ax, autopct='%1.1f%%', figsize=(15,10), legend
ax = plt.subplot(122)
pt1.plot(kind='pie', y='mean_total_st_support',ax=ax, autopct='%1.1f%%', legend=Fa1
```



Мы видим, что по кол-ву фильмов, которым была оказана господдержка лидер - Министерство культуры, при этом по общим средствам вложенным в господдержку Министерство культуры на последнем месте. В то время как Фонд кино по кол-ву фильмов не сильно уступает Министерству культуры, но в денежном выражении потратил на поддержку в разы больше.

5. Общие выводы

В процессе работы были проведены следующие мероприятия:

1. Объединены два дата фрейма в один
2. Новый дата фрейм проверен на дубликаты
3. Обработаны столбцы дата фрейма на предмет пропущенных значений, дубликатов в столбцах, форматирование данных в столбцах
4. Конвертирование столбцов в соответствующий сущности столбца формат данных
5. Добавлены необходимы для дальнейшего анализа данных столбцы
6. Проанализированы следующие связи:

- выход фильмов по годам
- доля фильмов со сборами от проката больше нуля к общему кол-ву вышедших фильмов
- доля фильмов со сборами от проката больше нуля к общему кол-ву вышедших фильмов по годам
- сборы от показа фильмов - среднее, медиана, изменения по годам
- влияние возрастных ограничений на сборы от показа фильмов

7. Проанализированы фильмы с господдержкой на предмет наличия связи господдержки и:

- тип фильма
- студия производства фильма
- режисер фильма
- продюсер
- рейтинг фильма
- возрастные ограничения
- жанр фильма
- сборы от показов
- год выпуска фильма
- страны производства
- источника финансирования

В ходе анализа были выявлены следующие тенденции:

1. начиная с 2012 года кол-во фильмов выходящих в прокат постоянно увеличивалось, исключение 2017 год (провал)
2. доля фильмов с указанием данных о сборах за прокат составляет 42%
3. эта доля в периодс 2010 по 2014 была очень низкой - 22%, с 2015 по 2019 доля выросла и стала 62 %
4. Максимальные сборы с учетом возрастных ограничений получали фильмы "16+", но при оценке удельных сборов (сборы на 1 фильм) оказалось, что более популярными оказались две возрастных категории: "6+" и "12+", что вполне логично, т.к. вместе с детьми на фильмы идут их родители, что увеличивает сборы от проката в разы(кол-во родителей)

При анализе фильмов с господдержкой получили следующие результаты:

1. Тип фильмом пользующийся наибольшей господдержкой - художественный
2. Студии снимающие много фильмов с господдержкой получают рейтинги за снятые фильмы выше, чем за фильмы без господдержки. Студии снимающие много фильмов с господдержкой также много снимают и без гос поддержки, что говорит об уверенной, многолетней работе этих студий, в то время как студии снимающие по 1 -2 фильма с господдержкой могут не снять больше ни одного фильма - студии однодневки.
3. В случае с режисером здесь средний рейтинг фильмов с господдержкой практически такой же как и без господдержки, что говорит о том, что режисеры заботятся о своей репутации
4. Такая же ситуация и с продюсерами, уровень их работы с господдержкой также высок , как и без господдержки
5. С точки зрения возрастных ограничений наибольшей господдержкой пользуются фильмы с ограничениями "12+" и "16+", это показывает на желание привлечь к просмотру максимальный объем активной аудитории зрителей и их родителей.

6. Средний рейтинг фильмов с господдержкой с 2017 года значительно выше среднего рейтинга фильмов без господдержки, видно определенное влияние госорганов устанавливающих определенные требования при финансировании фильмов
7. С точки зрения жанра, наиболее популярными при госфинансировании являются 5 категорий: драма, комедия, боевик, военный, детектив, которые имеют в среднем по 50% господдержки от общего бюджета фильма
8. Сборы от показа фильмов с господдержкой в целом низкие и не позволяют фильмам окупиться. В целом сравнивая сборы от проката фильмов с господдержкой и без господдержки мы обнаружили значительное отставание сборов от фильмов с господдержкой более чем в 4 раза в сравнении со сборами от фильмов без господдержки. Фильмы с господдержкой не пользуются большой популярностью у зрителей, хотя те кто их посмотрел оценивают их высоко.
9. В годовом распределении господдержка в период с 2013 по 2019 годы, имела провал в 2014-2015 годах, с последующим восстановлением в 2016 и далее.
10. При анализе стран, которые участвовали в производстве фильмов с господдержкой, было выявлено, что преимущественно поддержку получали фильмы произведенные в России - 95% и с небольшим добавлением фильмов снятых совместно с другими странами - 5%
11. Источниками финансирования являются Министерство культуры и Фонд кино, также есть фильмы с их взаимным участием. Больше всего фильмов было профинансировано Фондом кино (по суммам финансирования), а по кол-ву Министерством культуры.

В ходе исследования наша гипотеза о популярности фильмов снимаемых с участием государства не подтвердилась - фильмы без господдержки более популярны среди зрителей