

Комментарий ревьюера 2 Привет еще раз. Спасибо, что доделал работу. Оформление комментариев по работе сохраняется. Только обозначим, что это вторая итерация.

Комментарий ревьюера Привет! Спасибо, что прислал задание:) Меня зовут Слепцов Артем и я буду проверять твой проект) Ты проделал большую работу над проектом, однако его еще можно улучшить. Будет здорово, если ты, надеюсь, не против, если я буду на ты, будешь отвечать на комментарии и участвовать в диалоге. Мои комментарии обозначены пометкой ****Комментарий ревьюера****. Далее в файле ты сможешь найти их в похожих ячейках (если фон комментария **зелёный** - всё сделано правильно. Рекомендации укажу таким же цветом; **оранжевый** - некритичные замечания. Если таких замечаний меньше трех - проект может быть принят без их отработки; **красный** - нужно переделать). Не удаляй эти комментарии и постарайся учесть их в ходе выполнения данного проекта.

Привет!

Приятно, познакомиться, Артем!

Спасибо за поддержку и полезные комментарии :)

Комментарий ревьюера Стоит делать краткое вступление в работу, описание того, что надо делать. В работе необходимо приводить информацию о входных данных: какие столбцы есть в таблице, их названия и какую информацию они несут. Также название работы лучше отображать в отдельной ячейке и делать крупный шрифт (заголовок). Так работа выглядит презентабельно.

Описание проекта

Вы работаете в интернет-магазине «Стримчик», который продаёт по всему миру компьютерные игры. Из открытых источников доступны исторические данные о продажах игр, оценки пользователей и экспертов, жанры и платформы (например, Xbox или PlayStation). Вам нужно выявить определяющие успешность игры закономерности. Это позволит сделать ставку на потенциально популярный продукт и спланировать рекламные кампании.

Перед вами данные до 2016 года. Представим, что сейчас декабрь 2016 г., и вы планируете кампанию на 2017-й. Нужно отработать принцип работы с данными. Не важно, прогнозируете ли вы продажи на 2017 год по данным 2016-го или же 2027-й — по данным 2026 года.

Описание данных games.csv

Name — название игры

Platform — платформа

Year_of_Release — год выпуска

Genre — жанр игры

NA_sales — продажи в Северной Америке (миллионы долларов)

EU_sales — продажи в Европе (миллионы долларов)

JP_sales — продажи в Японии (миллионы долларов)

Other_sales — продажи в других странах (миллионы долларов)

Critic_Score — оценка критиков (от 0 до 100)

User_Score — оценка пользователей (от 0 до 10)

Rating — рейтинг от организации ESRB (англ. Entertainment Software Rating Board). Эта ассоциация определяет рейтинг компьютерных игр и присваивает им подходящую возрастную категорию.

Данные за 2016 год могут быть неполными.

Комментарий ревьюера 2

Вступление в работу очень важно. Так ты поясняешь то, чему она посвящена. Цель работы также хорошо бы вынести в отдельный блок вступления. Еще лучше будет, если ты сделаешь план работы с гиперссылками. Да, есть плагин ToC. Но хорошо бы уметь реализовывать интерактивный план проекта вручную. Как это реализовать - смотри по [ссылке](<https://stackoverflow.com/questions/49535664/how-to-hyperlink-in-a-jupyter-notebook/49717704>).

Шаг 2. Откройте файл с данными и изучите общую информацию

Комментарий ревьюера

Я заметил, что ячейки в твоей тетрадке начинаются не с 1. Перед отправкой работы рекомендую перезапускать ноутбук, чтобы убедиться, что все ячейки выполняются корректно.

Комментарий ревьюера 2 Перезапуск проекта позволяет обнаружить и исправить скрытые в коде ошибки.

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')

from datetime import datetime
from scipy import stats as st

games = pd.read_csv('/datasets/games.csv')
games.head()

games.describe()
games.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16715 entries, 0 to 16714
Data columns (total 11 columns):
Name                16713 non-null object
Platform            16715 non-null object
Year_of_Release     16446 non-null float64
Genre               16713 non-null object
NA_sales            16715 non-null float64
EU_sales            16715 non-null float64
JP_sales            16715 non-null float64
Other_sales         16715 non-null float64
Critic_Score        8137 non-null float64
User_Score          10014 non-null object
Rating              9949 non-null object
dtypes: float64(6), object(5)
memory usage: 1.4+ MB
```

Комментарий ревьюера Правильно, что весь импорт ты проводишь в первой ячейке работы. Так твой коллега, запускающий работу, будет в курсе используемых в ней библиотек и сможет при необходимости быстро настроить окружение.

Шаг 3. Подготовьте данные

Достаточно просто перевести заглавные буквы в строчные

```
In [2]: games.columns = games.columns.str.lower()
```

Комментарий ревьюера Данный метод позволяет нам автоматизировать процесс приведения названий столбцов к нижнему регистру. Так исключается вероятность опечаток.

Остальное тоже, чтобы случайно не путаться

```
In [3]: games['name'] = games['name'].str.lower()
        games['platform'] = games['platform'].str.lower()
        games['genre'] = games['genre'].str.lower()
        games['rating'] = games['rating'].str.lower()
```

Проверка на дубликаты - 0

```
In [4]: games.duplicated().sum()
```

```
Out[4]: 0
```

переводим в формат float, заменив некорректные значения на NaN

```
In [5]: games['user_score'] = pd.to_numeric(games['user_score'], errors='coerce')
```

Комментарий ревьюера Верно, по своей сути tbd и является NaN. Отлично, что определяешь неявные пропущенные значения.

Пропуски в столбце с годом издания заполним нулями и переведем все в int

```
In [6]: games['year_of_release'] = games['year_of_release'].fillna(0)
        games['year_of_release'] = games['year_of_release'].astype('int')
```

В столбце с названиями игр заменим 2 пропуска

```
In [7]: games['name'] = games['name'].fillna('unknown_name')
```

В столбце с жанрами игр заменим 2 пропуска

```
In [8]: games['genre'] = games['genre'].fillna('unknown_genre')
```

Пропуски по возрастному рейтингу

```
In [9]: games['rating'] = games['rating'].fillna('unknown_rating')
```

```
In [10]: games.isnull().sum().sort_values(ascending = False)
```

```
Out[10]: user_score      9125
critic_score    8578
rating           0
other_sales     0
jp_sales        0
eu_sales        0
na_sales        0
genre           0
year_of_release 0
platform        0
name            0
dtype: int64
```

Проверим, какие нечисленные значения встречаются, а то вдруг оценку зрителей - число написали буквами

В основном, пропуски наблюдаются в столбцах 'critic_score', 'user_score' и 'rating'

Скорее всего их подтянули по id игр из другой базы

Возможно, что эти данные отсутствуют по конкретным платформам(устройствам

N64,SNES,SAT,2600,GB,NES,GEN,NG и т.д.) или годам(слишком старые игры)

tbd - означает, что рейтинг был на обсуждении, но, по-видимому, так и не опубликовали в итоге

Комментарий ревьюера Да, восстановить пропущенные значения мы не можем. Данных для этого недостаточно. Лучше работать с меньшим количеством данных хорошего качества. Ошибки в данных устранены. Данные подготовлены к дальнейшему анализу.

Посчитайте суммарные продажи во всех регионах и запишите их в отдельный столбец

```
In [11]: games['total_sales'] = games['na_sales'] + games['eu_sales'] + games['jp_sales'] + games['other_sales']
games.head()
```

Out[11]:

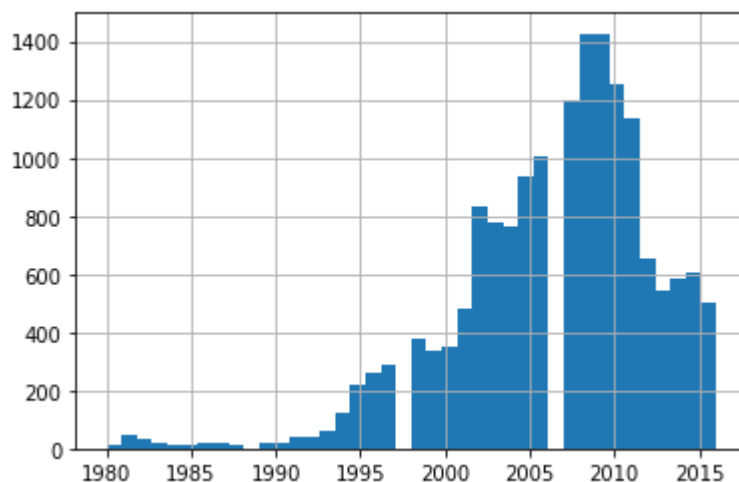
	name	platform	year_of_release	genre	na_sales	eu_sales	jp_sales	other_sales	cr
0	wii sports	wii	2006	sports	41.36	28.96	3.77	8.45	
1	super mario bros.	nes	1985	platform	29.08	3.58	6.81	0.77	
2	mario kart wii	wii	2008	racing	15.68	12.76	3.79	3.29	
3	wii sports resort	wii	2009	sports	15.61	10.93	3.28	2.95	
4	pokemon red/pokemon blue	gb	1996	role-playing	11.27	8.89	10.22	1.00	

Шаг 4. Проведите исследовательский анализ данных

Выход игр по годам

```
In [12]: games[games['year_of_release'] != 0]['year_of_release'].hist(bins=40)
```

Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x7f6363f8b2d0>



Основной пик выхода игр попадает на 2008-2010 годы

Игры начали издаваться еще в начале 1980-х годов

Но понадобилось целых 15-20 лет развития техники, чтобы началось массовое производство игр

На мой взгляд, можно исключить данные до 2000 года, т.к. кол-во выпускаемых в те годы игр незначительное в сравнении с дальнейшими данными

Скорее всего спад связан с развитием мобильных устройств (смартфоны, планшеты и т.п.) -

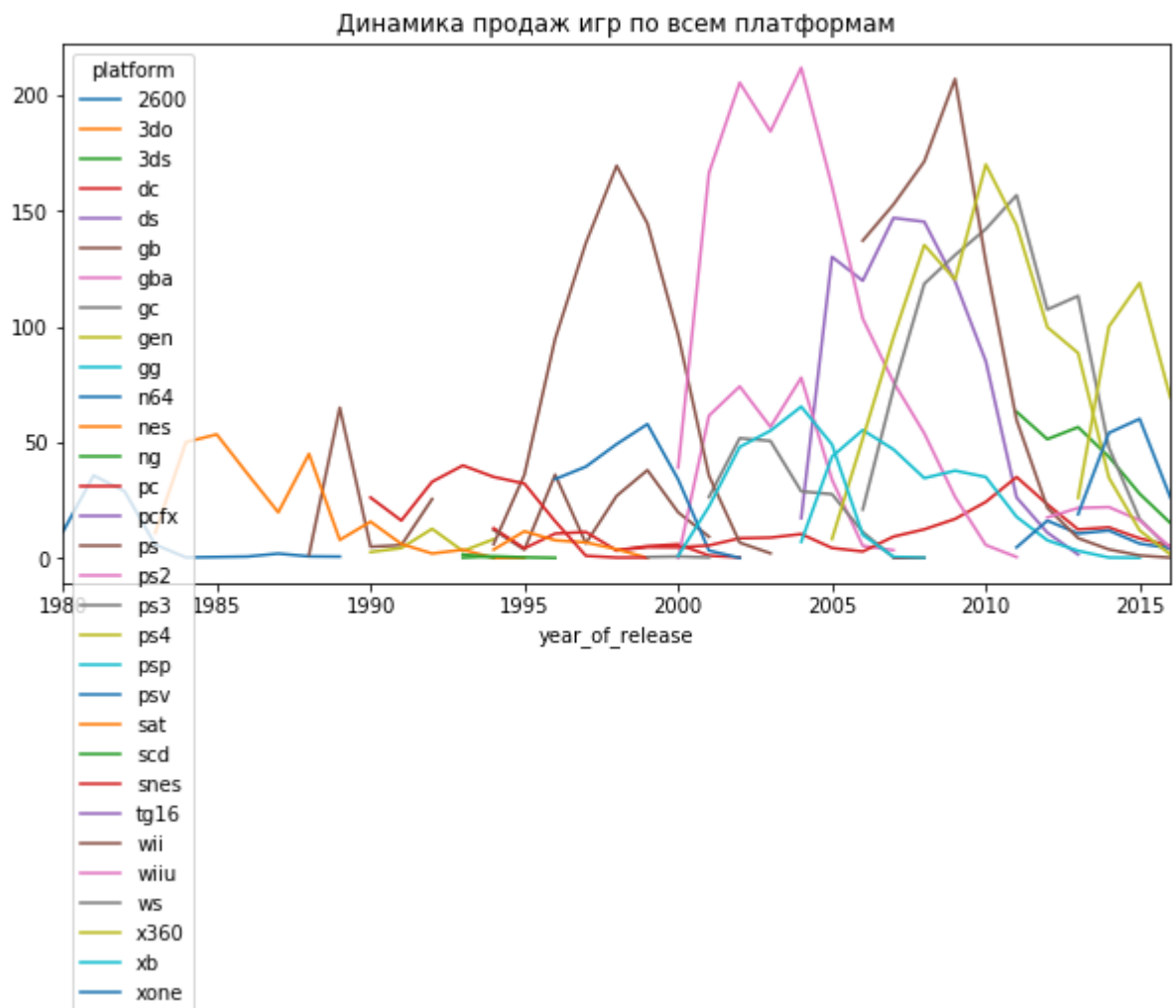
Пользователям проще стало играть в перерывах, в очередях, по пути на работу и т.д.

Комментарий ревьюера Как думаешь, с чем связан спад в индустрии последних лет?

Комментарий ревьюера 2 Соглашусь с озвученной причиной спада.

Посмотрите, как менялись продажи по платформам

```
In [13]: games.pivot_table(index='year_of_release', columns='platform', values='total_sales', aggfunc = 'sum')
games[games['year_of_release'] != 0].pivot_table(index='year_of_release', columns='platform',
                                                values='total_sales',aggfunc = 'sum').plot(figsize=(10, 5))
plt.title('Динамика продаж игр по всем платформам')
plt.show()
```



Выберите платформы с наибольшими суммарными продажами и постройте распределение по годам

```
In [14]: platform_sales = games.pivot_table(index='platform',
                                             values='total_sales',
                                             aggfunc = 'sum').sort_values(by='total_sales',ascending=False)
#print(platform_sales)
platform_sales_top = platform_sales.query('total_sales > 259')
#print(platform_sales_top)

platform_list = ['ps2', 'x360', 'ps3', 'wii', 'ds', 'ps', 'ps4', 'psp', 'pc']

games_top_platform = games.query('platform in @platform_list')

games_top_platform[games_top_platform['year_of_release'] != 0].pivot_table(index='year_of_release',
                                                                            columns='platform',
                                                                            values='total_sales',
                                                                            aggfunc = 'sum').plot(figsize=(10, 5))
plt.title('Динамика продаж игр по топ-платформам')
plt.show()
```



Найдите популярные в прошлом платформы, у которых сейчас продажи на нуле

ps, ps2, ds, wii, psp

За какой характерный период появляются новые и исчезают старые платформы?

на примере Xbox и PS видно, что в среднем цикл актуальности платформ 8-10 лет

Создадим сводную таблицу в разрезе платформ и выведем максимальное и минимальное значение выпускавшихся игр на них

Далее исключим пороговые значения, таким образом отпадут все "крайние" платформы, по которым нет данных до 1980 года, либо их цикл еще не завершился в 2016 году

Кстати платформа PC сразу отпадет

```
In [15]: plat_year_pivot = games[games['year_of_release'] != 0].pivot_table(index='platform', values='year_of_release', aggfunc = ('max', 'min'))

plat_year_pivot_act = plat_year_pivot.query('(min > 1980) & (max < 2016) & (min != max)')
plat_year_pivot_act['platform_lifetime'] = plat_year_pivot_act['max'] - plat_year_pivot_act['min']
print(plat_year_pivot_act)
print(plat_year_pivot_act['platform_lifetime'].mean())
```

	max	min	platform_lifetime
platform			
3do	1995	1994	1
dc	2008	1998	10
ds	2013	1985	28
gb	2001	1988	13
gba	2007	2000	7
gc	2007	2001	6
gen	1994	1990	4
n64	2002	1996	6
nes	1994	1983	11
ng	1996	1993	3
ps	2003	1994	9
ps2	2011	2000	11
psp	2015	2004	11
sat	1999	1994	5
scd	1994	1993	1
snes	1999	1990	9
ws	2001	1999	2
xb	2008	2000	8
	8.055555555555555		

Есть одна особо "долгоиграющая" приставка - Nintendo DS (28 лет)

Комментарий ревьюера Значение средней продолжительности существования платформы приведено. Однако выводы ты делаешь по графикам. Стоит также привести расчет данного значения. Подумай, стоит ли включать в расчет все платформы. Будут ли выбросы по продолжительности существования платформ?

Комментарий ревьюера 2 Расчет приведен. Верно, что ты не включил в анализ актуальные на настоящий момент платформы. Период их жизни еще продолжается. По DS мы имеем выброс в 1985 году. На самом деле данная платформа вышла в 2004 году.

Определите, данные за какой период нужно взять, чтобы исключить значимое искажение распределения по платформам в 2016 году

если 1995-2015 - это весь актуальный период,

для построения прогноза проанализируем данные только цикла спада игрового рынка начиная с пика в 2009 году,

возьмем период 2009-2015

```
In [16]: years_list = []
for element in range(2009,2016):
    years_list.append(element)
#print(years_list)

games_data = games.query('year_of_release in @years_list')
#print(games_data.head())
```

Комментарий ревьюера Актуальный период назван. Стоит его значительно уменьшить. Сейчас у тебя в периоде содержится сразу несколько этапов развития индустрии: становление рынка, рост, пик 2008 и 2009 годов, а также сокращение рынка последних годов. Также большей части платформ уже нет в 2016 году, в построении прогноза на 2017 год они нам не помогут. При уменьшении периода в рассмотрение попадут только последние поколения платформ, а также будем рассматривать только конечный на данный момент интервал развития игровой индустрии.

Комментарий ревьюера 2 Такой выбор актуальных данных позволит нам увеличить качество и точность прогноза на 2017 год.

Далее работайте только с данными, которые вы определили. Не учитывайте данные за предыдущие годы

Какие платформы лидируют по продажам, растут или падают? Выберите несколько потенциально прибыльных платформ

```
In [17]: games_data.pivot_table(index = 'platform', values = 'total_sales', aggfunc = 'sum').sort_values(by='total_sales', ascending=False)
```

Out[17]:

	total_sales
platform	
ps3	715.07
x360	669.18
wii	429.76
ps4	244.89
ds	243.29
3ds	242.67
pc	133.62
xone	133.17
psp	101.83
wiiu	77.59
psv	49.56
ps2	32.49

Лидеры по продажам за рассматриваемый период - PS3, Xbox360 и Wii

Однако все эти платформы уже завершают свой цикл популярности

Потенциально прибыльными можно считать PS4, пришедший на замену популярного некогда PS3 и Xbox One, заменивший Xbox 360

по PC продажи не такие высокие, как по игровым приставкам, однако игры на PC актуальны во все времена

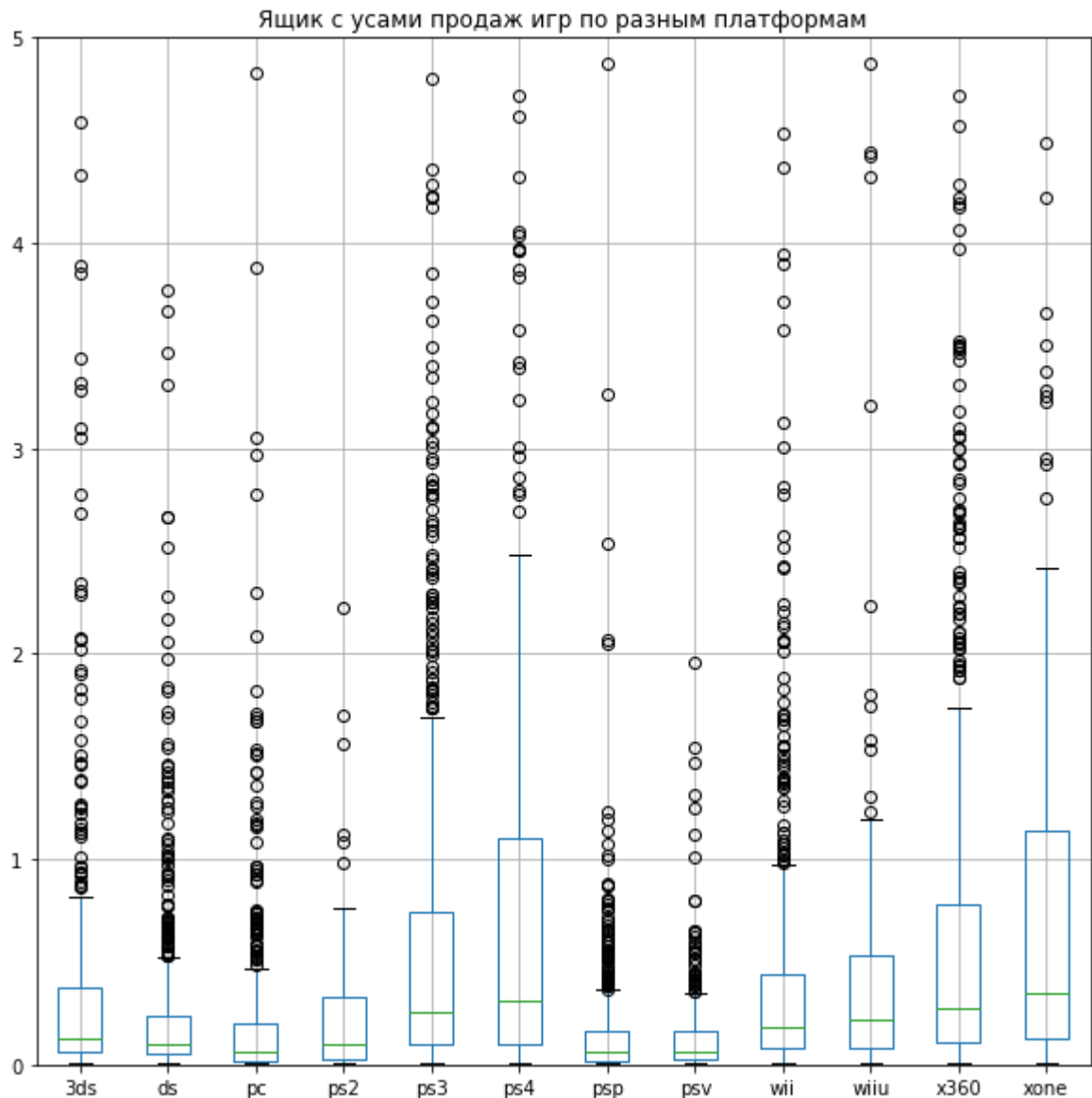
Комментарий ревьюера Да, именно эти платформы мы и можем назвать перспективными на 2017 год.

Постройте график «ящик с усами» по глобальным продажам каждой игры и разбивкой по платформам

```
In [18]: games_data.describe()

pivot_game_sale = games_data.pivot_table(index='name', columns='platform', values='total_sales', aggfunc='sum')
#print(pivot_game_sale)

plot = pivot_game_sale.boxplot(figsize=(10, 10)).set_ylim(0, 5)
plt.title('Ящик с усами продаж игр по разным платформам')
plt.show()
#plot = pivot_game_sale.boxplot(figsize=(10, 10))
#plt.show()
```



Комментарий ревьюера С предупреждениями также стоит разобраться. Иногда их бывает слишком много, поэтому важно уметь их скрывать. В этом тебе поможет библиотека warnings. Попробуй найти подходящий метод и убрать предупреждения.

Велика ли разница в продажах? А в средних продажах на разных платформах? Опишите результат

Есть отдельные "успешные" игры, продажи по которым превышают 20, 25 и даже 30 млн. долл, что отличается от среднего и выходит далеко за пределы "усов"

В целом, по платформам ситуация схожая: нижний ус упирается в 0, а верхний - находится в пределах 1 млн. долл. по большей части платформ. Есть более успешные платформы - PS3, Xbox360 и Wiiu с верхним усом более 1 млн долл.; и платформы PS4 и Xbox One - следующие поколения вышеназванных платформ, нормальное распределение продаж игры по которым может достигать 2.5 млн. долл.

Отчетливо видно, что с эволюцией конкретной платформы растут и продажи по играм

К примеру, PS4 > PS3 > PS2

А XboxOne > Xbox360

Wiiu > Wii

Хоть мы и взяли данные за 2009-2015 гг, все равно на разнице продаж может сказываться рост цен на игры со временем, как минимум из-за инфляции, а также увеличения расходов на производство игр

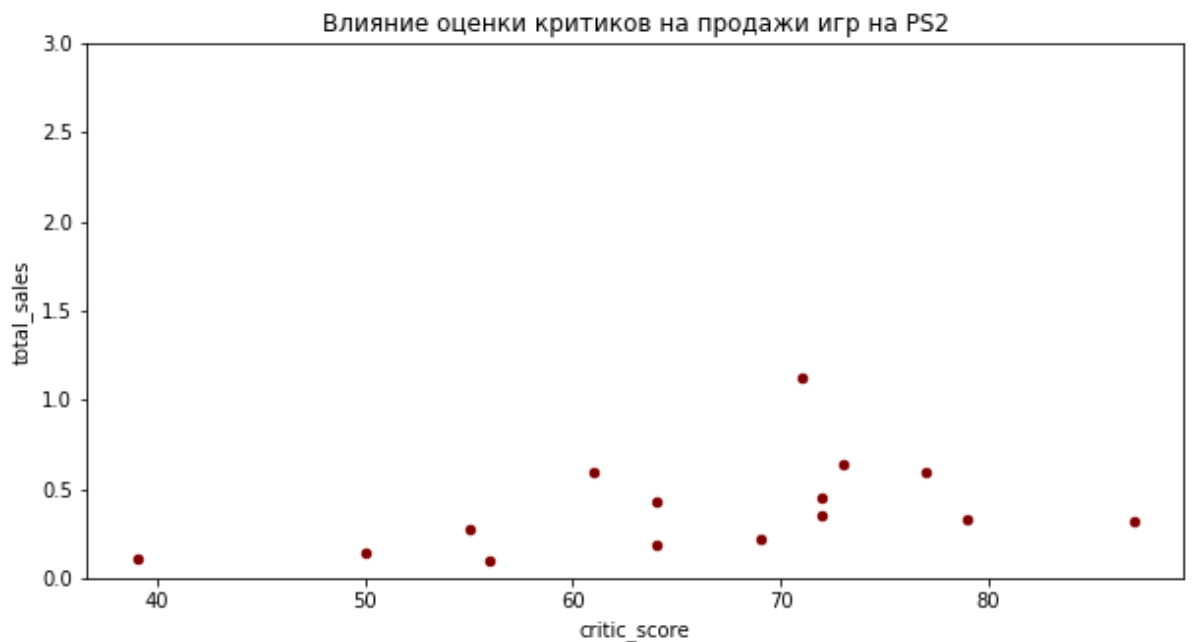
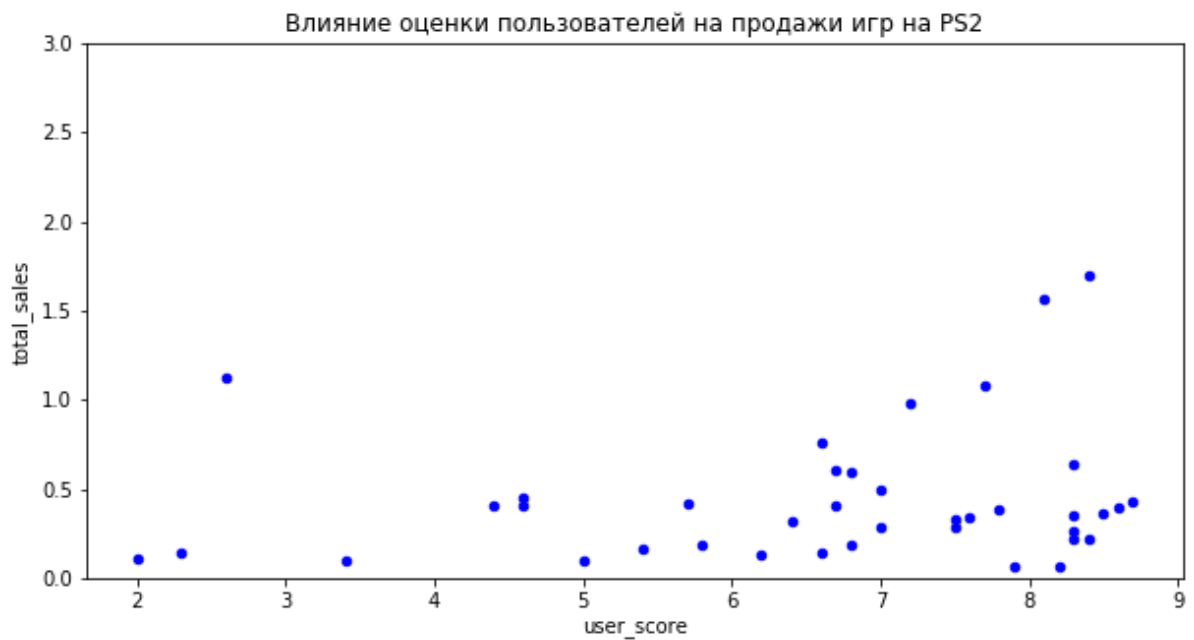
Комментарий ревьюера Подумай, из-за чего формируется разница между платформами. Старайся не только описывать результат, но и трактовать его.

Посмотрите, как влияют на продажи внутри одной популярной платформы отзывы пользователей и критиков

```
In [19]: games_data_ps2 = games_data.query('platform == "ps2"')
# print(games_data_ps2.head())

games_data_ps2.plot(x='user_score', y='total_sales', kind='scatter', color='blue', alpha = 1, figsize=(10, 5)).set_ylim(0, 3)
plt.title('Влияние оценки пользователей на продажи игр на PS2')
plt.show()

games_data_ps2.plot(x='critic_score', y='total_sales', kind='scatter', color='maroon', alpha = 1, figsize=(10, 5)).set_ylim(0, 3)
plt.title('Влияние оценки критиков на продажи игр на PS2')
plt.show()
```

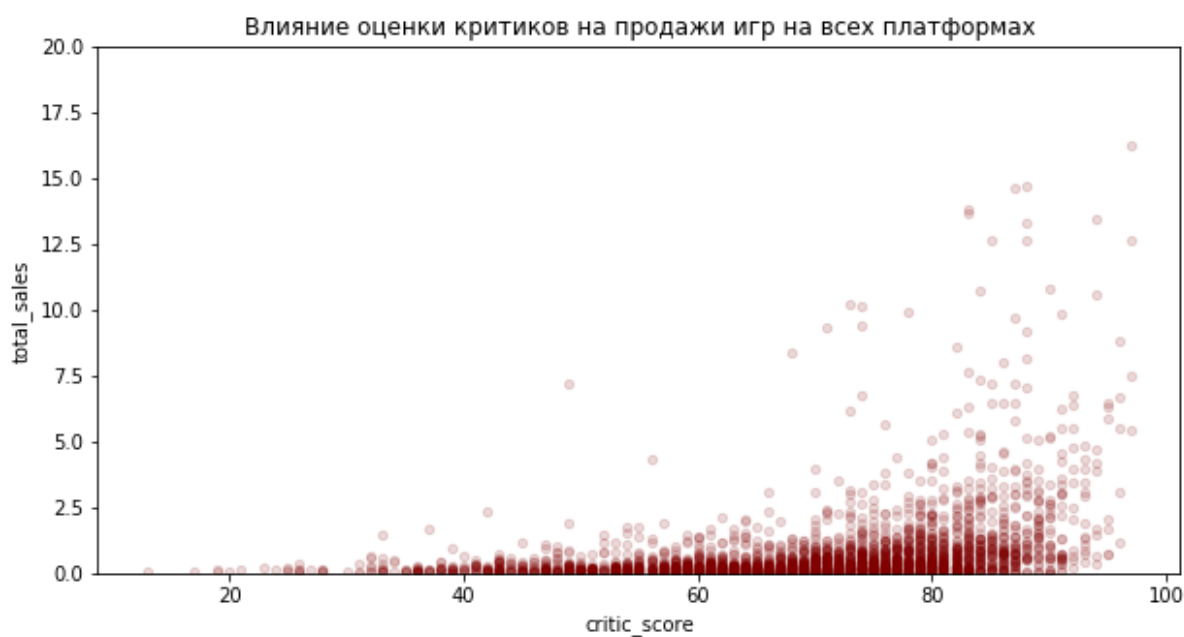


Постройте диаграмму рассеяния и посчитайте корреляцию между отзывами и продажами

```
In [20]: corr_data = pd.DataFrame()
corr_data['total_sales'] = games_data['total_sales']
corr_data['user_score'] = games_data['user_score']
corr_data['critic_score'] = games_data['critic_score']
#print(corr_data.head())

corr_data.plot(x='user_score', y='total_sales', kind='scatter', color='blue',
legend=True, alpha = 0.15, figsize=(10, 5)).set_ylim(0, 20)
plt.title('Влияние оценки пользователей на продажи игр на всех платформах')
plt.show()
corr_data.plot(x='critic_score', y='total_sales', kind='scatter', color='maroon',
legend=True, alpha = 0.15, figsize=(10, 5)).set_ylim(0, 20)
plt.title('Влияние оценки критиков на продажи игр на всех платформах')
plt.show()

corr_data[['total_sales', 'user_score', 'critic_score']].corr().style.format("{:.2%}")
```



Out[20]:

	total_sales	user_score	critic_score
total_sales	100.00%	7.69%	30.29%
user_score	7.69%	100.00%	58.14%
critic_score	30.29%	58.14%	100.00%

Сформулируйте выводы и соотнесите их с продажами игр на других платформах

Диаграммы по PS2 и общая по всем платформам в целом очень схожи

Как правило, чем выше рейтинги критиков и пользователей игры, тем выше ее продажи (не без выбросов, конечно)

Самая слабая зависимость продаж от оценки пользователей - менее 9%

Зависимость продаж от оценки критиков выше - около 25%, но все равно считается слабой

Наблюдается зависимость оценки критиков и оценки пользователей - почти 59%

Комментарий ревьюера Результат получен. Подумай, чем он вызван. Здорово, что рассмотрены несколько платформ.

Посмотрите на общее распределение игр по жанрам. Что можно сказать о самых прибыльных жанрах?

```
In [21]: games_data.pivot_table(index='genre', values='name', aggfunc = 'count').sort_v
        alues(by='name', ascending=False)
```

Out[21]:

	name
genre	
action	1589
misc	750
sports	710
adventure	647
role-playing	617
shooter	453
simulation	322
racing	297
fighting	236
strategy	221
puzzle	192
platform	167

Больше всего игр в жанрах action, misc и sports

```
In [22]: games_data.pivot_table(index='genre', values='total_sales', aggfunc = 'sum').sort_values(by='total_sales', ascending=False)
```

Out[22]:

total_sales	
genre	
action	780.67
shooter	510.81
sports	442.27
role-playing	344.41
misc	307.77
platform	157.02
racing	153.58
fighting	109.02
simulation	102.78
adventure	77.41
strategy	47.00
puzzle	40.38

Вполне ожидаемо, что и продажи по этим жанрам будут высокими

Однако высокие продажи еще у жанра shooter, при том, что кол-во игр в этом жанре не такое высокое. Это говорит о популярности жанра shooter, продается больше копий.

Выделяются ли жанры с высокими и низкими продажами?

Аутсайдеры по продажам - игры в жанрах puzzle и strategy

Комментарий ревьюера Анализ популярности жанров проведен. Однако не стоит забывать, что производство игр в жанрах Action или Shooter обходится сильно дороже, чем производство Puzzle-игр.

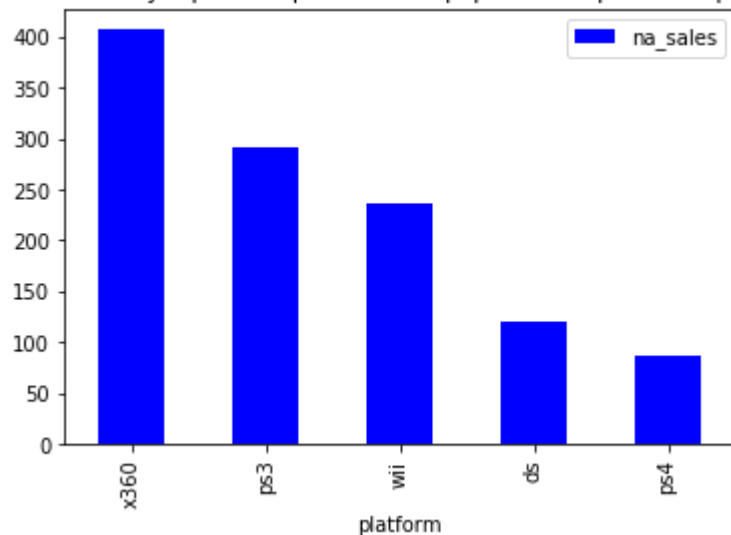
Шаг 5. Составьте портрет пользователя каждого региона

Определите для пользователя каждого региона (NA, EU, JP):

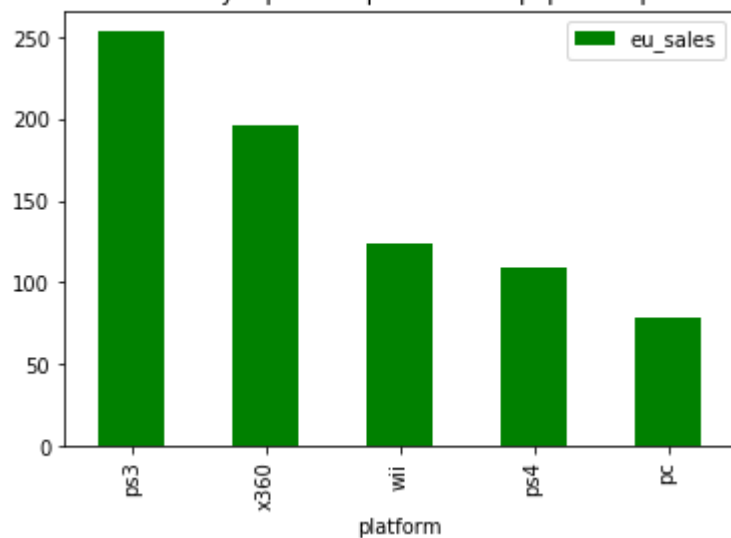
Самые популярные платформы (топ-5). Опишите различия в долях продаж

```
In [23]: games_data.groupby(by='platform').agg({'na_sales':'sum'}).sort_values(by='na_s
ales', ascending=False).head(5).plot(kind='bar', color='blue', legend=True)
plt.title('Топ-5 популярных игровых платформ в Северной Америке')
plt.show()
games_data.groupby(by='platform').agg({'eu_sales':'sum'}).sort_values(by='eu_s
ales', ascending=False).head(5).plot(kind='bar', color='green', legend=True)
plt.title('Топ-5 популярных игровых платформ в Европе')
plt.show()
games_data.groupby(by='platform').agg({'jp_sales':'sum'}).sort_values(by='jp_s
ales', ascending=False).head(5).plot(kind='bar', color='maroon', legend=True)
plt.title('Топ-5 популярных игровых платформ в Японии')
plt.show()
```

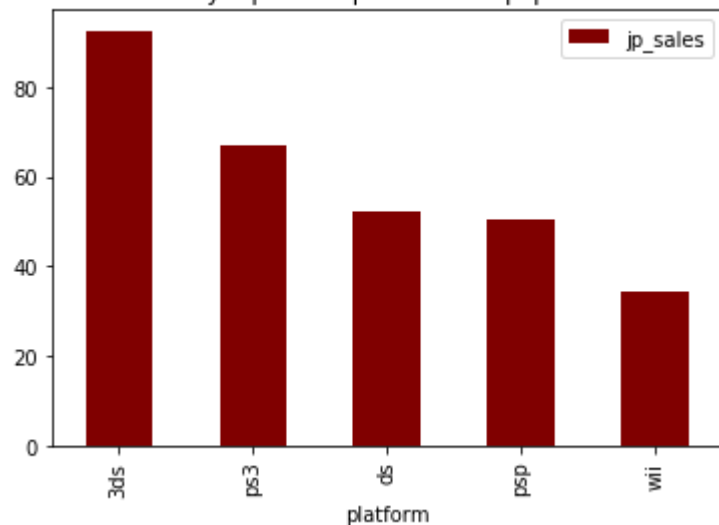
Топ-5 популярных игровых платформ в Северной Америке



Топ-5 популярных игровых платформ в Европе



Топ-5 популярных игровых платформ в Японии



Доля продаж Северной Америки - самая большая, в наблюдаемый период игроки предпочитали Xbox 360 и PS3

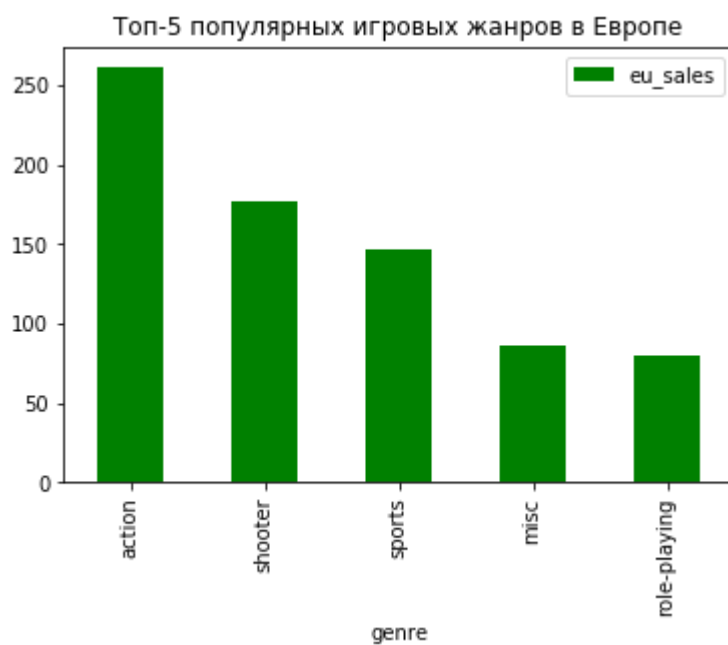
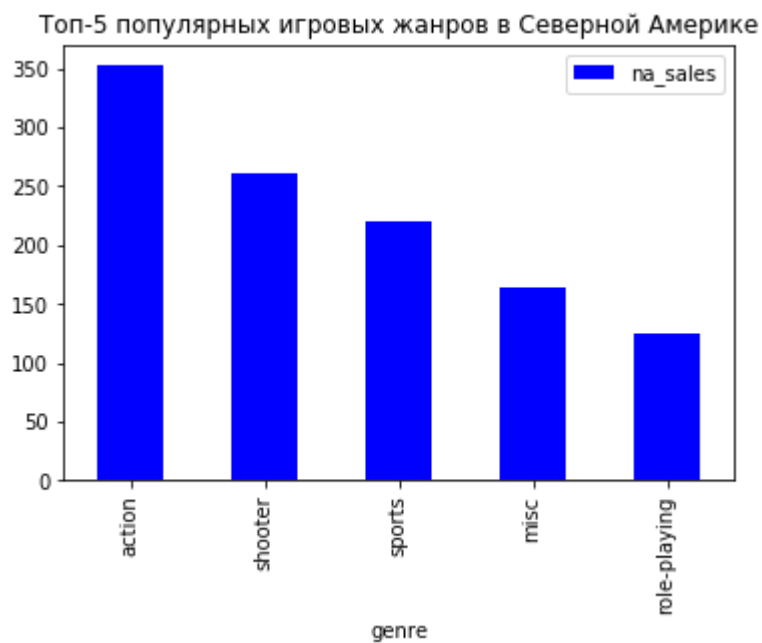
Доля продаж на европейском рынке в почти 2 раза меньше североамериканской, топ по популярности - те же платформы, только в обратном порядке (PS3 и Xbox 360)

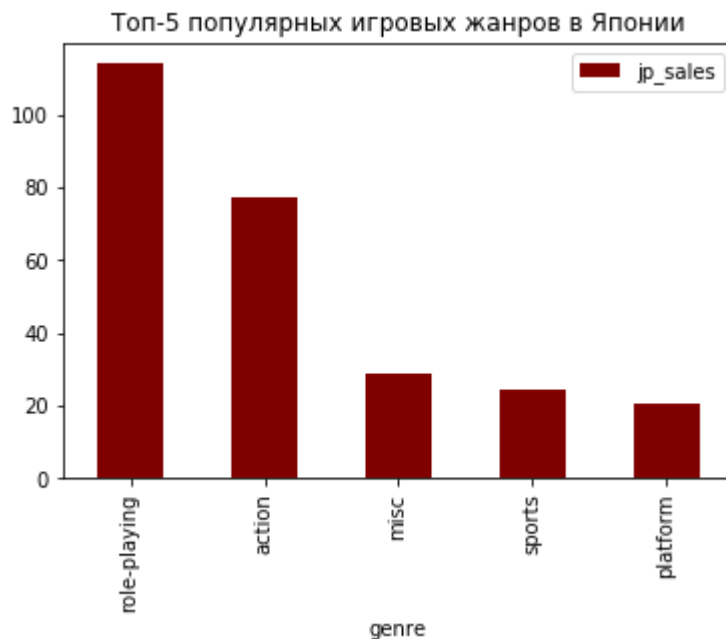
В Японии рынок еще меньше, а самая популярная консоль - Nintendo 3DS, вышедшая в 2011 году

Скорее всего потому, что японцы предпочитают портативные приставки, чтобы играть на ходу или в пути

Самые популярные жанры (топ-5). Поясните разницу

```
In [24]: games_data.groupby(by='genre').agg({'na_sales': 'sum'}).sort_values(by='na_sales', ascending=False).head(5).plot(kind='bar', color='blue', legend=True)
plt.title('Топ-5 популярных игровых жанров в Северной Америке')
plt.show()
games_data.groupby(by='genre').agg({'eu_sales': 'sum'}).sort_values(by='eu_sales', ascending=False).head(5).plot(kind='bar', color='green', legend=True)
plt.title('Топ-5 популярных игровых жанров в Европе')
plt.show()
games_data.groupby(by='genre').agg({'jp_sales': 'sum'}).sort_values(by='jp_sales', ascending=False).head(5).plot(kind='bar', color='maroon', legend=True)
plt.title('Топ-5 популярных игровых жанров в Японии')
plt.show()
```





В Северной Америке и Европе предпочитают жанры action и shooter

Тогда как в Японии наиболее популярны RPG (ну, в прочем, у них там все не как в остальном мире)

Во-первых, игровой рынок Японии - один из самых древних, поэтому зачастую средний возраст игроков может достигать 40-50 лет

Во-вторых, как видно из выводов выше, как раз эти же самые пользователи как играли на Nintendo с конца 80-х, так и продолжают на Nintendo нового поколения

В-третьих, менталитет японцев сильно отличается от менталитета американца или европейца.

У них своя совершенно уникальная культура, со своими традициями и особенностями. Взять хоть тот же культ аниме, манги и комиксов, игровых автоматов и другие японские "приколы".

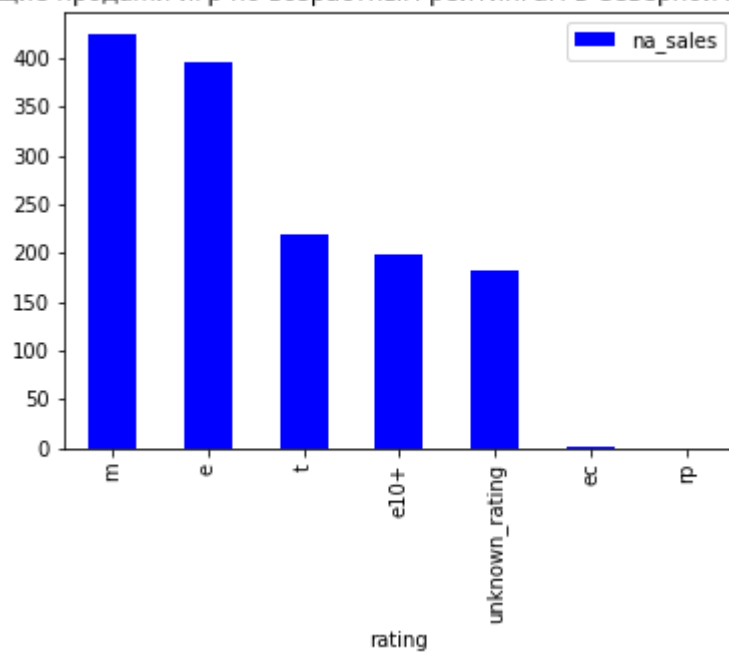
Японцы более мирные и предпочитают гармонию во всем. Поэтому заметно, что шутеры им совсем не интересны, в отличие от пользователей других рассматриваемых рынков.

Комментарий ревьюера Из-за чего японский рынок игр так сильно отличается от других?

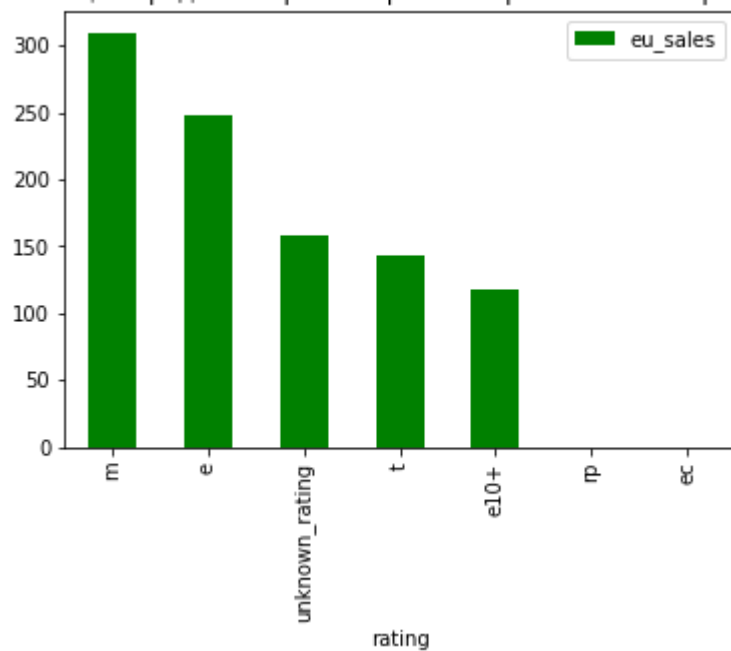
Влияет ли рейтинг ESRB на продажи в отдельном регионе?

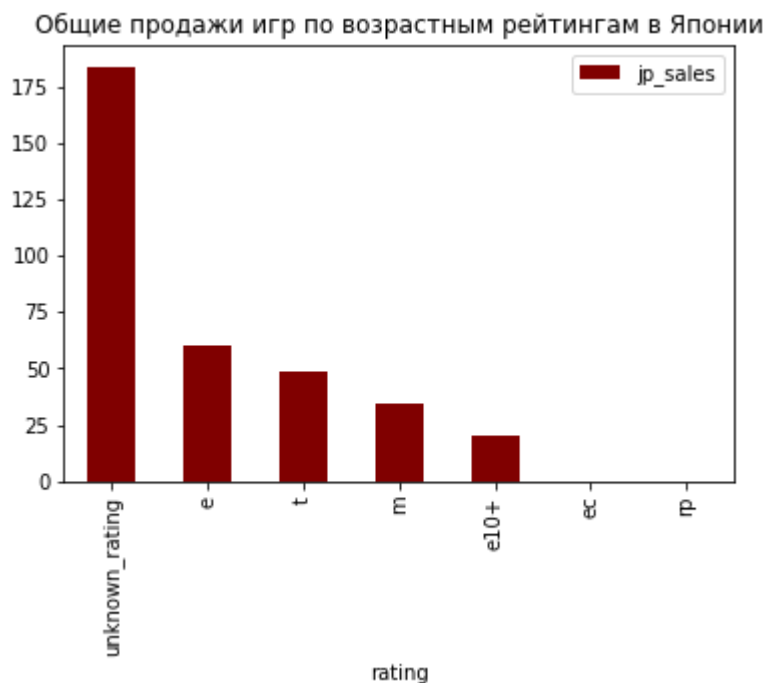

```
In [25]: games_data.groupby(by='rating').agg({'na_sales': 'sum'}).sort_values(by='na_sales', ascending=False).plot(kind='bar', color='blue', legend=True)
plt.title('Общие продажи игр по возрастным рейтингам в Северной Америке')
plt.show()
games_data.groupby(by='rating').agg({'eu_sales': 'sum'}).sort_values(by='eu_sales', ascending=False).plot(kind='bar', color='green', legend=True)
plt.title('Общие продажи игр по возрастным рейтингам в Европе')
plt.show()
games_data.groupby(by='rating').agg({'jp_sales': 'sum'}).sort_values(by='jp_sales', ascending=False).plot(kind='bar', color='maroon', legend=True)
plt.title('Общие продажи игр по возрастным рейтингам в Японии')
plt.show()
```

Общие продажи игр по возрастным рейтингам в Северной Америке



Общие продажи игр по возрастным рейтингам в Европе





Комментарий ревьюера Круто, ты заметил важную особенность данных - большая часть игр японского региона не имеет рейтинга. Как думаешь, почему мы получаем такой результат? Выгрузка не виновата)

В Америке и Европе ситуация примерно одинаковая, самые популярные игры категории E (для всех), далее - игры с неуказанным в данных рейтингом

В Японии же наоборот, на 1-м месте - рейтинг не указан

Возможно, что популярнейшие игры для японского рынка выпускаются самой же Японией, и не особо пользуются популярностью в остальном мире, поэтому и не имеют международного рейтинга

Комментарий ревьюера Портрет типичного пользователя каждого из регионов получен. Приведены все необходимые графики. Здорово, что отмечены индивидуальные особенности каждого региона.

Шаг 6. Проведите исследование статистических показателей

Как изменяется пользовательский рейтинг и рейтинг критиков в различных жанрах?

```
In [26]: user_genre_pivot = games_data.pivot_table(index='genre', values='user_score', aggfunc = 'mean')
print('Средние оценки пользователей по игровым жанрам')
print(user_genre_pivot.sort_values(by='user_score', ascending=False))
```

Средние оценки пользователей по игровым жанрам

	user_score
genre	
role-playing	7.311296
platform	7.225000
puzzle	7.145000
adventure	7.110169
fighting	6.969032
action	6.835253
strategy	6.752475
misc	6.692105
shooter	6.533521
racing	6.459880
simulation	6.325510
sports	6.173829

Средний рейтинг пользователей для всех жанров лежит в промежутке 6.2-7.3 (1-10)

```
In [27]: critic_genre_pivot = games_data.pivot_table(index='genre', values='critic_score', aggfunc = 'mean')
print('Средние оценки критиков по игровым жанрам')
print(critic_genre_pivot.sort_values(by='critic_score', ascending=False))
```

Средние оценки критиков по игровым жанрам

	critic_score
genre	
platform	73.560748
role-playing	72.434483
strategy	71.585106
shooter	70.597598
sports	70.465190
fighting	70.096552
racing	69.833333
puzzle	69.563636
misc	69.227273
action	67.901985
adventure	65.951220
simulation	65.041667

Средний рейтинг критиков для всех жанров лежит в промежутке 65-73 (1-100)

Что говорит о схожести данных по рейтингам

```
In [28]: games_data.groupby(by='genre').agg({'user_score':'mean'}).sort_values(by='user_score', ascending=False).plot(kind='bar', color='blue', legend=True)
plt.title('Оценки пользователей по игровым жанрам')
plt.show()

games_data.groupby(by='genre').agg({'critic_score':'mean'}).sort_values(by='critic_score', ascending=False).plot(kind='bar', color='maroon', legend=True)
plt.title('Оценки критиков по игровым жанрам')
plt.show()
```



Пользователи охотней ставят высокие баллы жанрам RPG, Platform, Puzzle

Критики охотней ставят высокие баллы почти по тем же жанрам - Platform, RPG, Strategy

Посчитайте среднее количество, дисперсию и стандартное отклонение

```
In [29]: games_data_genre_score = pd.DataFrame()
games_data_genre_score['genre'] = games_data['genre']
games_data_genre_score['critic_score'] = games_data['critic_score']
games_data_genre_score['user_score'] = games_data['user_score']

genre_list = ['action', 'adventure', 'fighting', 'misc', 'platform', 'puzzle',
              'racing', 'role-playing',
              'shooter', 'simulation', 'sports', 'strategy']

for genre in genre_list:
    variance_estimate = np.var(games_data_genre_score[games_data_genre_score[
'genre'] == genre], ddof=1)
    print('Дисперсии по жанру:', genre)
    print(variance_estimate)
    print('Стандартное отклонение по жанру:', genre)
    print(np.sqrt(variance_estimate))
    print('-----')
```

```
Дисперсии по жанру: action
critic_score    188.620195
user_score      1.812026
dtype: float64
Стандартное отклонение по жанру: action
critic_score    13.733907
user_score      1.346115
dtype: float64
-----
Дисперсии по жанру: adventure
critic_score    228.030388
user_score      2.697161
dtype: float64
Стандартное отклонение по жанру: adventure
critic_score    15.100675
user_score      1.642303
dtype: float64
-----
Дисперсии по жанру: fighting
critic_score    204.65728
user_score      1.82332
dtype: float64
Стандартное отклонение по жанру: fighting
critic_score    14.305848
user_score      1.350304
dtype: float64
-----
Дисперсии по жанру: misc
critic_score    170.907476
user_score      2.366128
dtype: float64
Стандартное отклонение по жанру: misc
critic_score    13.073159
user_score      1.538222
dtype: float64
-----
Дисперсии по жанру: platform
critic_score    191.022218
user_score      2.314505
dtype: float64
Стандартное отклонение по жанру: platform
critic_score    13.821079
user_score      1.521350
dtype: float64
-----
Дисперсии по жанру: puzzle
critic_score    119.546801
user_score      2.045103
dtype: float64
Стандартное отклонение по жанру: puzzle
critic_score    10.933746
user_score      1.430071
dtype: float64
-----
Дисперсии по жанру: racing
critic_score    199.900200
user_score      2.296754
```



```
dtype: float64
Стандартное отклонение по жанру: racing
critic_score    14.138607
user_score      1.515505
dtype: float64
-----
Дисперсии по жанру: role-playing
critic_score    152.156592
user_score      1.417805
dtype: float64
Стандартное отклонение по жанру: role-playing
critic_score    12.335177
user_score      1.190716
dtype: float64
-----
Дисперсии по жанру: shooter
critic_score    224.337585
user_score      2.404834
dtype: float64
Стандартное отклонение по жанру: shooter
critic_score    14.977903
user_score      1.550753
dtype: float64
-----
Дисперсии по жанру: simulation
critic_score    165.050877
user_score      3.357384
dtype: float64
Стандартное отклонение по жанру: simulation
critic_score    12.847213
user_score      1.832317
dtype: float64
-----
Дисперсии по жанру: sports
critic_score    226.154340
user_score      2.712877
dtype: float64
Стандартное отклонение по жанру: sports
critic_score    15.038429
user_score      1.647081
dtype: float64
-----
Дисперсии по жанру: strategy
critic_score    150.976550
user_score      2.646919
dtype: float64
Стандартное отклонение по жанру: strategy
critic_score    12.287252
user_score      1.626935
dtype: float64
-----
```

В целом, Стандартное отклонение по отзывам критиков по всем жанрам примерно на уровне 10-15 баллов

Минимальные значения у жанров RPG и Puzzle

Максимальные у жанров Adventure и Sports

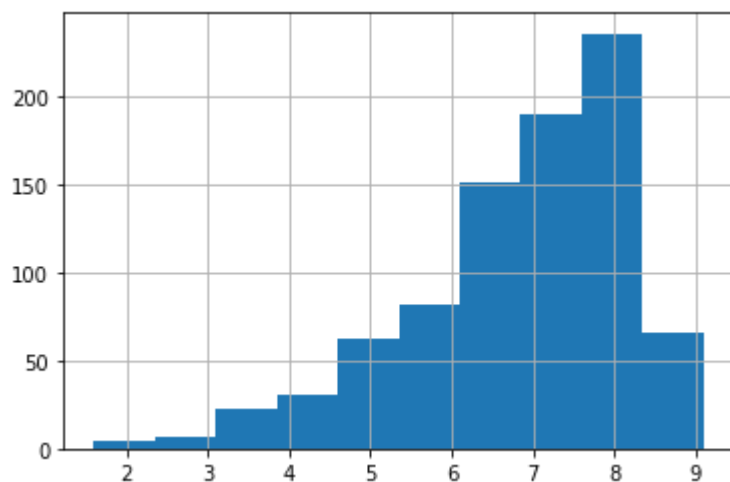
Стандартное отклонение по отзывам пользователей по всем жанрам также примерно на одном уровне- около 1.5

Минимальное выделяющееся значения у жанра RPG, т.е. разброс минимальный

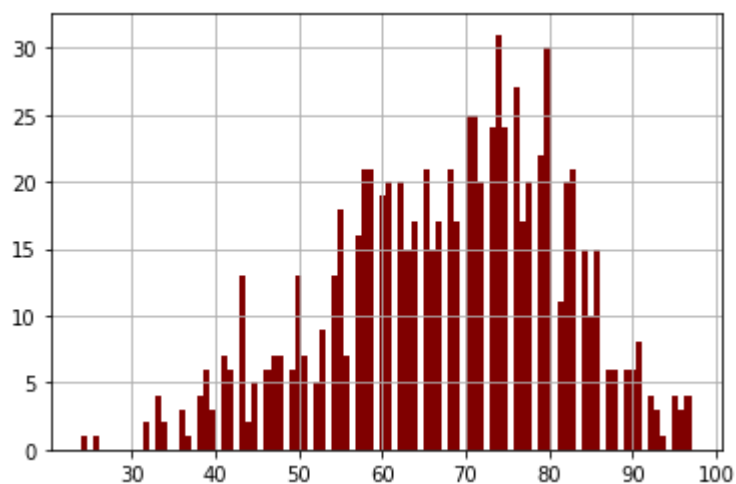
Постройте гистограммы. Опишите распределения

```
In [30]: for genre in genre_list:
          print('Плотность распределения оценок пользователей по жанру:', genre)
          games_data[games_data['genre'] == genre]['user_score'].hist(bins=10)
          plt.show()
          print('-----')
          print('Плотность распределения оценок критиков по жанру:', genre)
          games_data[games_data['genre'] == genre]['critic_score'].hist(bins=100, color='maroon')
          plt.show()
          print('-----')
```

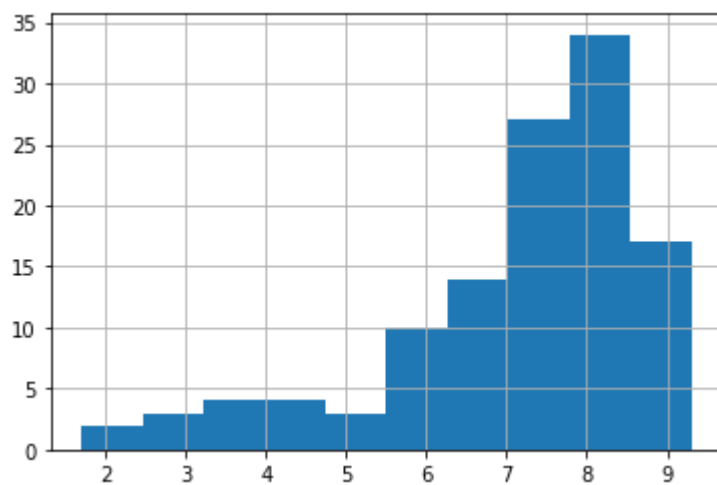
Плотность распределения оценок пользователей по жанру: action



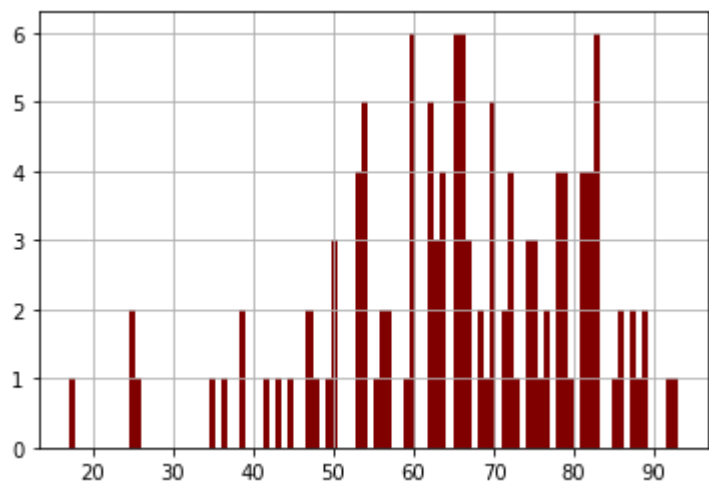
Плотность распределения оценок критиков по жанру: action



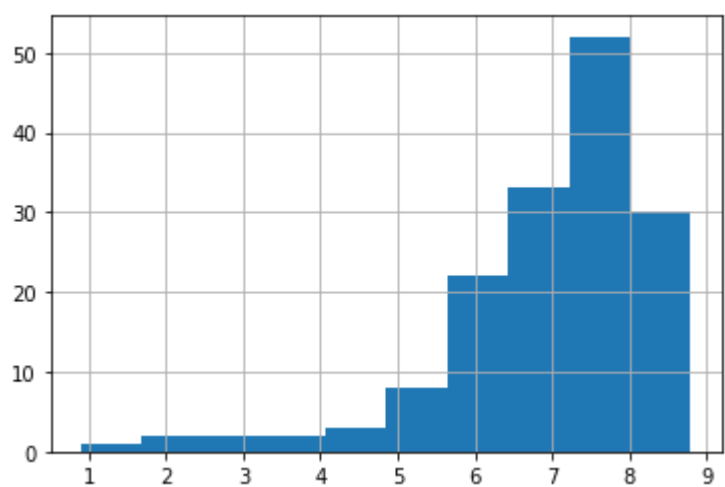
Плотность распределения оценок пользователей по жанру: adventure



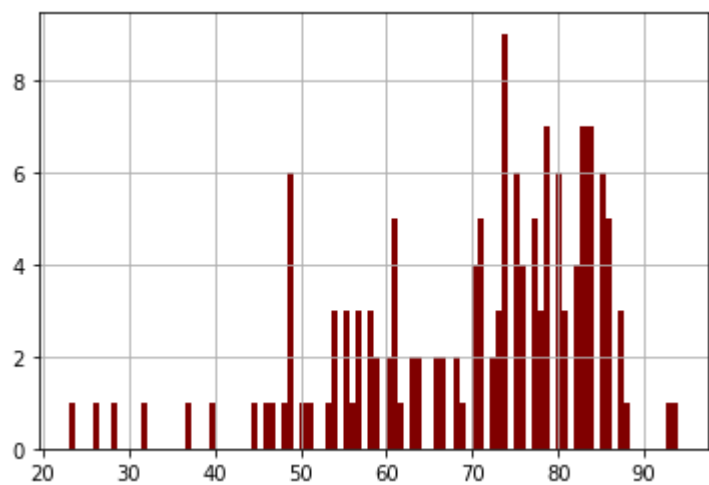
Плотность распределения оценок критиков по жанру: adventure



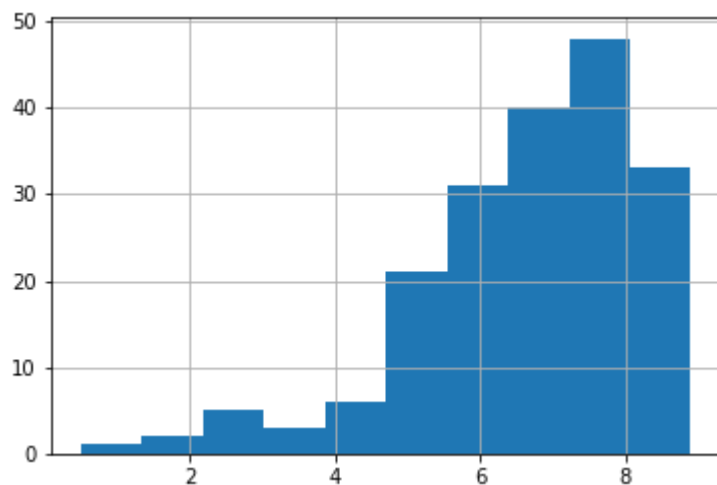
Плотность распределения оценок пользователей по жанру: fighting



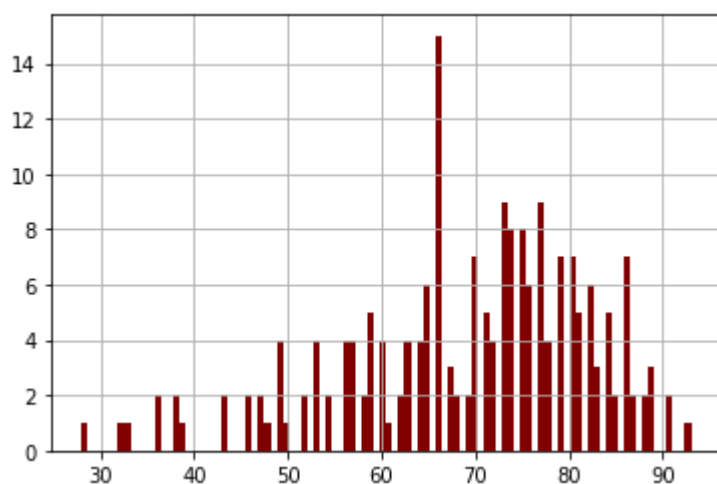
Плотность распределения оценок критиков по жанру: fighting



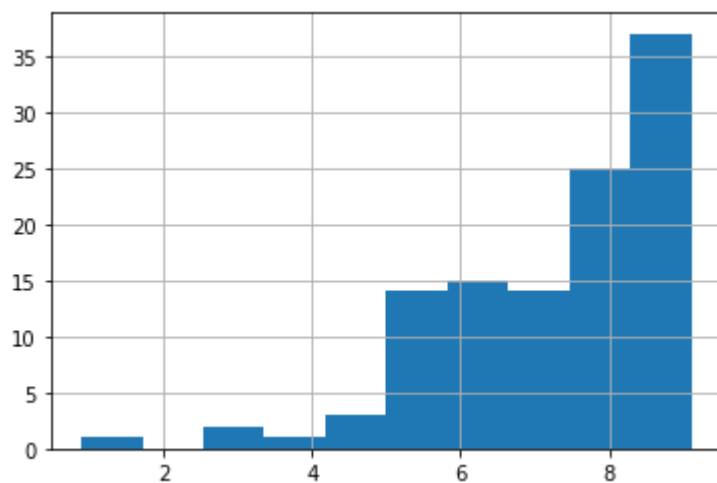
Плотность распределения оценок пользователей по жанру: misc



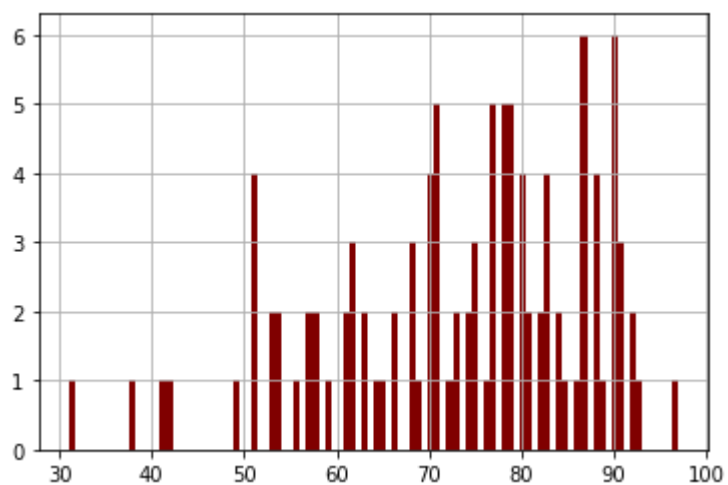
Плотность распределения оценок критиков по жанру: misc



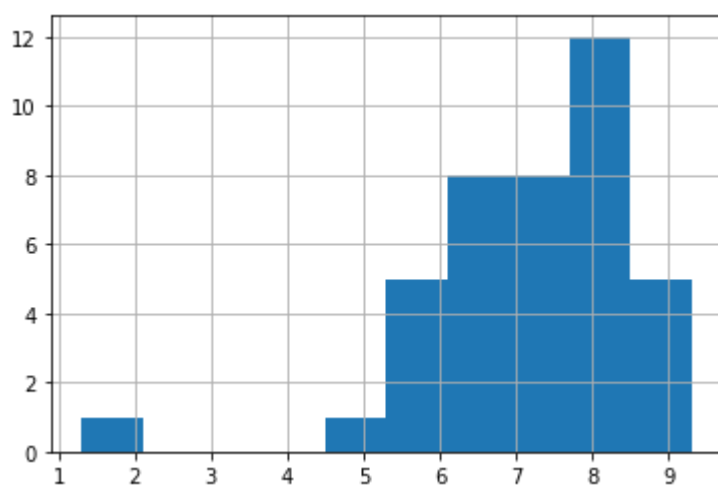
Плотность распределения оценок пользователей по жанру: platform



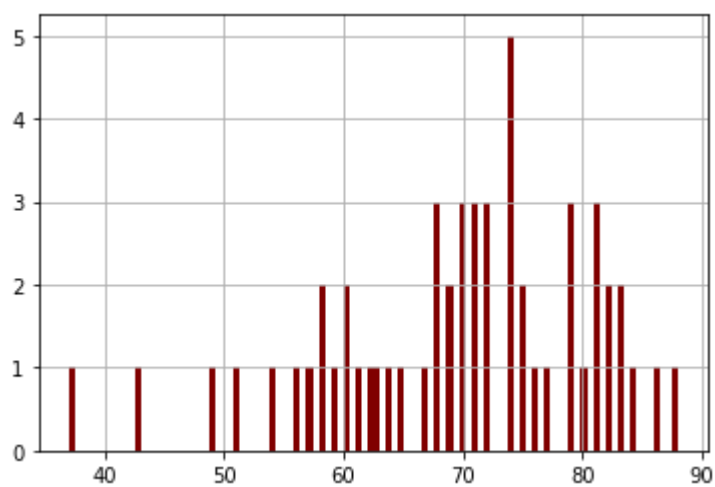
Плотность распределения оценок критиков по жанру: platform



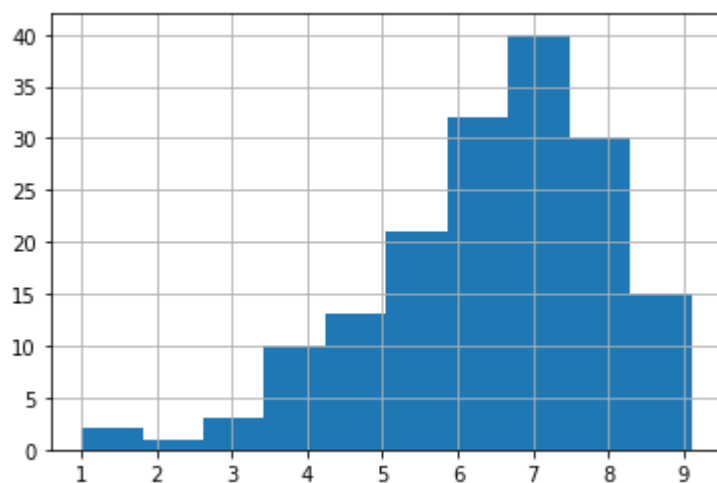
Плотность распределения оценок пользователей по жанру: puzzle



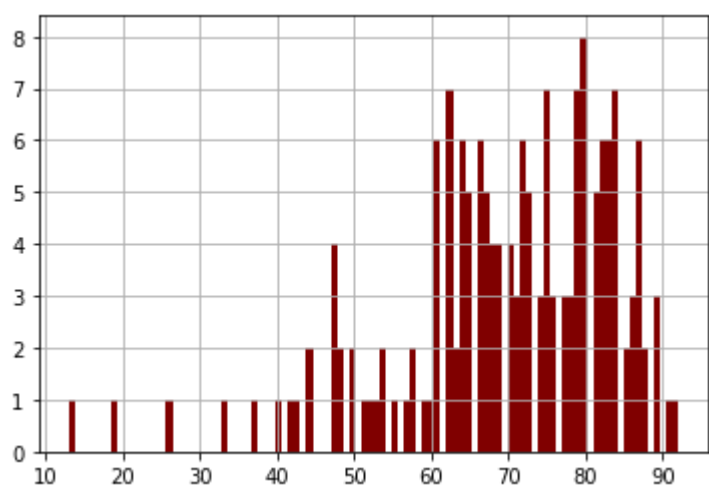
Плотность распределения оценок критиков по жанру: puzzle



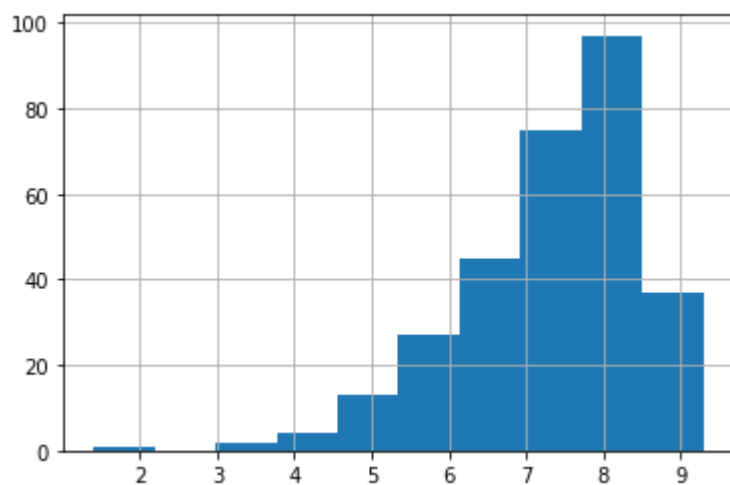
Плотность распределения оценок пользователей по жанру: racing



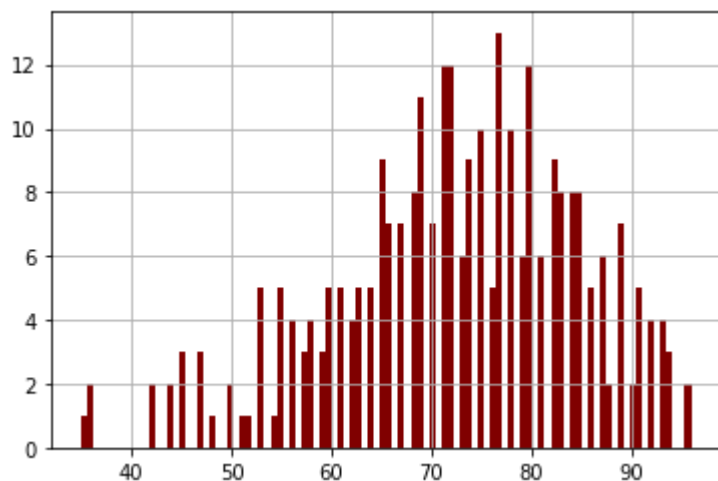
Плотность распределения оценок критиков по жанру: racing



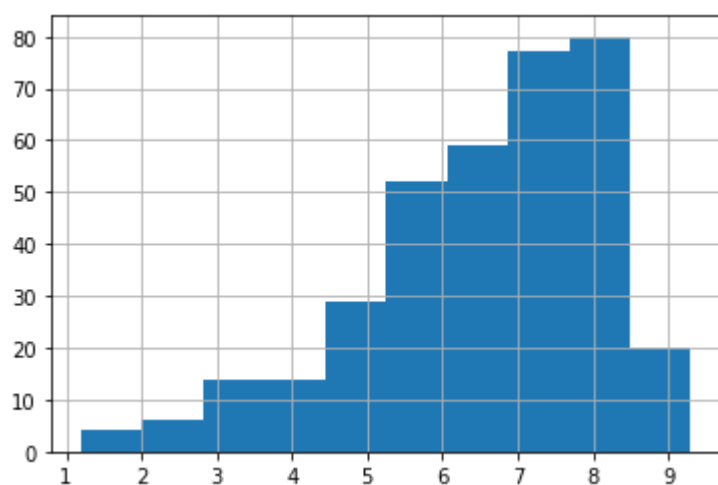
Плотность распределения оценок пользователей по жанру: role-playing



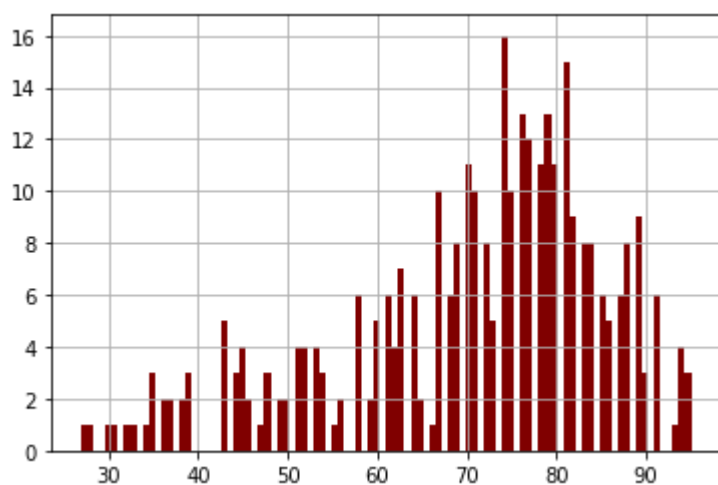
Плотность распределения оценок критиков по жанру: role-playing



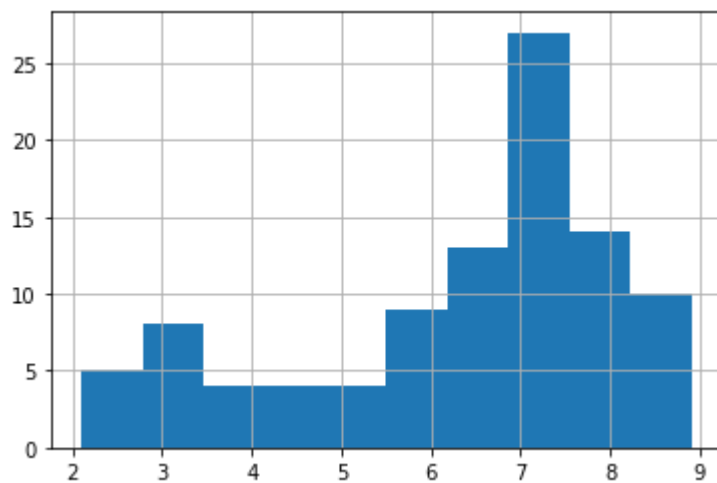
Плотность распределения оценок пользователей по жанру: shooter



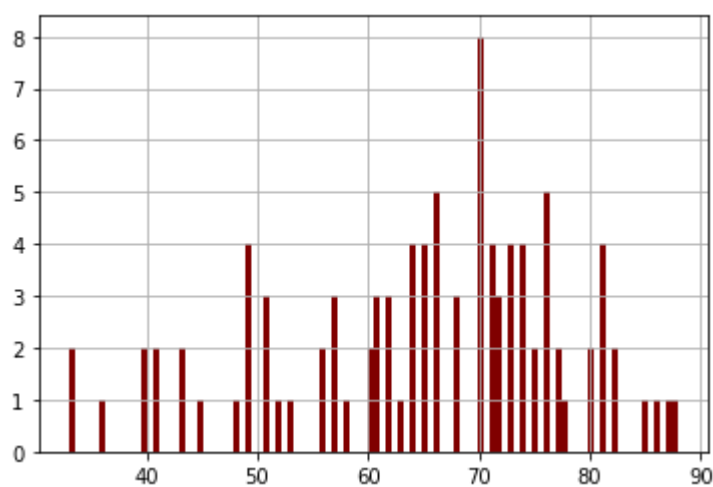
Плотность распределения оценок критиков по жанру: shooter



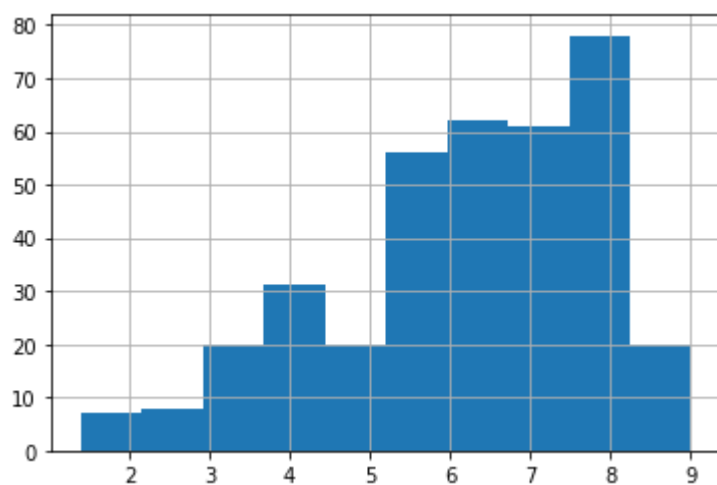
Плотность распределения оценок пользователей по жанру: simulation



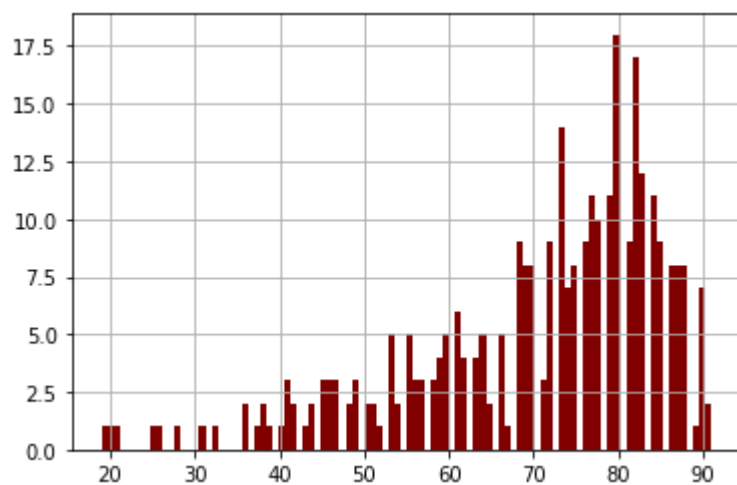
Плотность распределения оценок критиков по жанру: simulation



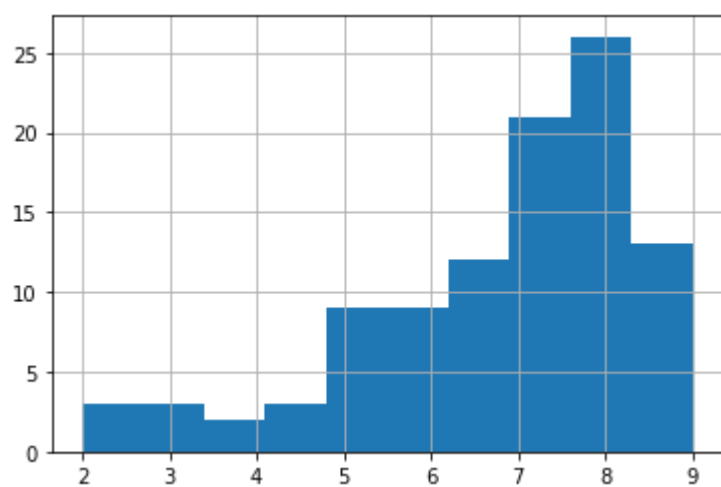
Плотность распределения оценок пользователей по жанру: sports



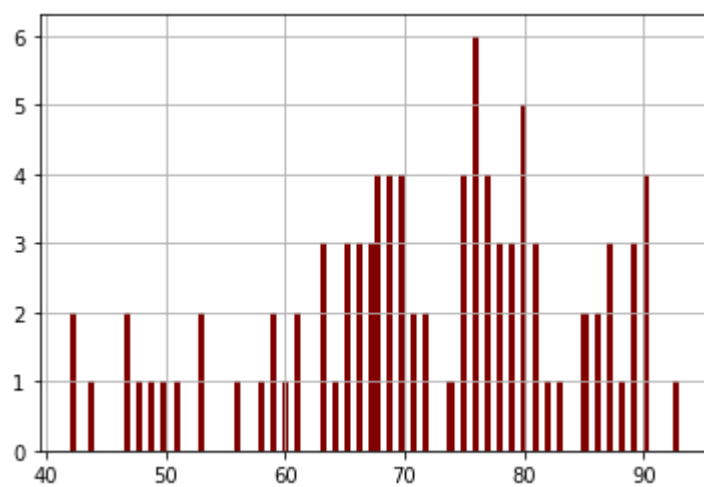
Плотность распределения оценок критиков по жанру: sports



Плотность распределения оценок пользователей по жанру: strategy



Плотность распределения оценок критиков по жанру: strategy



Гистограммы по оценкам и пользователей, и критиков по всем жанрам схожи тем, что имеют отрицательную скошенность из-за того, что средняя оценка выше 5 баллов из 10.

Комментарий ревьюера Стат. показатели рассчитаны. Помимо них нас просят также построить и гистограммы по отзывам критиков и пользователей для всех рассматриваемых жанров. Пожалуйста, дополни данный раздел.

Комментарий ревьюера 2 Необходимые графики приведены. Как думаешь, почему по большинству жанров мы получаем скошенные влево распределения?

Шаг 7. Проверьте гипотезы

Средние пользовательские рейтинги платформ Xbox One и PC одинаковые:

H0: Средние пользовательские рейтинги платформ Xbox One и PC одинаковые

H1: Средние пользовательские рейтинги платформ Xbox One и PC различаются

Задайте самостоятельно пороговое значение alpha:

0.05 - стандартное значение для такого рода исследований (не слишком жесткое)

```
In [31]: alpha = 0.05

xbox_one = games_data[(games_data['platform'] == 'xone') & (games_data['user_score'] > 0)][['user_score']]
pc = games_data[(games_data['platform'] == 'pc') & (games_data['user_score'] > 0)][['user_score']]

results = st.ttest_ind(xbox_one, pc)

print('p-значение:', results.pvalue)

if (results.pvalue < alpha):
    print("Отвергаем нулевую гипотезу")
else:
    print("Не получилось отвергнуть нулевую гипотезу")

p-значение: 0.1708077691700714
Не получилось отвергнуть нулевую гипотезу
```

А это значит, что мы не смогли отвергнуть гипотезу, что средние пользовательские рейтинги платформ Xbox One и PC одинаковые

И все же это не означает, что мы доказали, что эти рейтинги одинаковые

Просто по нашей выборке похоже на то, что эти рейтинги схожи

Частенько происходит так, что на одной консоли игра получается успешной без серьезных багов, тогда как на другой платформе основные баги могут не исправляться годами

Но, видимо, качество игр на PC и Xbox One примерно одинаково

Комментарий ревьюера Гипотезы сформулированы и проверены верно.

Средние пользовательские рейтинги жанров Action (англ. «действие») и Sports (англ. «виды спорта») разные:

H0: Средние пользовательские рейтинги жанров Action (англ. «действие») и Sports (англ. «виды спорта») одинаковые

H1: Средние пользовательские рейтинги жанров Action (англ. «действие») и Sports (англ. «виды спорта») разные

```
In [32]: alpha = 0.05

action = games_data[(games_data['genre'] == 'action') & (games_data['user_score'] > 0)][ 'user_score' ]
sports = games_data[(games_data['genre'] == 'sports') & (games_data['user_score'] > 0)][ 'user_score' ]

results = st.ttest_ind(action, sports)

print('p-значение:', results.pvalue)

if (results.pvalue < alpha):
    print("Отвергаем нулевую гипотезу")
else:
    print("Не получилось отвергнуть нулевую гипотезу")
```

p-значение: 4.707667517418828e-13

Отвергаем нулевую гипотезу

Мы отвергаем гипотезу о том, что средние пользовательские рейтинги жанров Action и Sports одинаковые
По нашей выборке получается, что пользователи в среднем оставляют разные оценки играм в жанрах Action и Sports

Комментарий ревьюера

Проверка второй гипотезы также осуществлена без нареканий.

Как вы сформулировали нулевую и альтернативную гипотезы:

H0 - всегда на равенство, либо на отсутствие каких-либо изменений

H1 - альтернативная (противоположная)

Какой критерий применили для проверки гипотез и почему:

T-критерий Стьюдента, потому что работаем с выборкой, а не генеральной совокупностью

Выводы:

Игры начали издаваться еще в начале 1980-х годов.

Но понадобилось целых 15-20 лет развития техники, чтобы началось массовое производство игр.

Основной пик выхода игр попадает на 2008-2010 годы.

Далее последовал спад игровой индустрии, связанный с развитием смартфонов и планшетов, из-за чего многие пользователи перешли на мобильные устройства.

В среднем, цикл актуальности платформ 8-10 лет.

Затем актуальность платформы пропадает, либо выходит следующее поколение платформ.

Есть одна особенная приставка, которая оставалась актуальной аномально долгий период - 28 лет Nintendo не отпускало игроков, в основном японцев.

Для построения прогнозов на 2017 год мы взяли данные за период спада индустрии 2009-2015 гг.

Заметно, что с эволюцией конкретной платформы растут и продажи по играм, как и расходы на производство игр.

Больше всего игр выпускалось в жанрах Action, Miscellaneous и Sports.

На разных рынках предпочтения пользователей относительно игровых платформ и жанров могут отличаться.

К примеру, как уже упоминалось, японцы больше всего предпочитают Nintendo и RPG.

Вообще рынок Японии сильно отличается от остальных рынков в основном из-за всего раннего развития еще с 80-х годов и совершенно уникального менталитета японцев.

Комментарий ревьюера Финальный вывод и есть главный результат твоей работы. Стоит писать его подробно по результатам проведенной работы. В нем можно приводить полученные в ходе работы значения. Также можно расписать все, что было сделано в работе.

Комментарий ревьюера 2 Выводы описаны грамотно. Приведены ответы на главные вопросы проекта. В выводах можно приводить полученные ранее значения. Также можно расписать все, что было сделано в ходе проведения работы. Еще лучше будет, если приведешь рекомендации для компании по дальнейшим действиям.

Комментарий ревьюера Если тебе нравится тема визуализации, то можешь изучить методы библиотеки `seaborn`. Она позволяет строить довольно презентабельные графики. Ты выполнил все пункты работы, молодец! Критических замечаний немного. Однако с ними важно поработать. Также есть достаточное число желтых комментариев, которые стоит исправить. Думаю, ты справишься с этим быстро. Жду твою работу :)

Комментарий ревьюера 2 Помарки исправлены, и теперь работа выполнена хорошо. У тебя получился классный проект, молодец. Поздравляю со сданным проектом. Надеюсь, он был интересен и познавателен. Спасибо за оставленные комментарии по исправлениям. Успехов в дальнейшем пути :)