

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ "СКИЛБОКС"

Финальная работа по курсу
«Аналитик данных с нуля 2.0»

Проведение А/В-теста

Выполнила:

Тешаева Анастасия

Новосибирск, 2022

Оглавление

Раздел 1. Цель проекта	3
Раздел 2. Анализ источников	4
Раздел 3. Очистка данных	6
Раздел 4. Использование статистических методов	13
4.1. Определение соотношения количества участников в группах по каждой игровой платформе	15
4.2. Построение доверительных интервалов для каждой метрики от средних значений с точностью 95%	16
Раздел 5. Формирование отчета	24
5.1. Расчет и сравнение метрик ARPU (средняя прибыль на игрока), ARPPU (средняя прибыль на платящего игрока) и траты внутриигровой валюты между различными группами игроков	24
5.1.1. Расчет и сравнение метрик для платформы PC	25
5.1.2. Расчет и сравнение метрик для платформы PS4	28
5.1.3. Расчет и сравнение метрик для платформы XBox	31
5.2. Графики сравнения метрик по дням (Power BI)	34
5.2.1. Графики сравнения метрик по дням для игровой платформы PC	35
5.2.2. Графики сравнения метрик по дням для игровой платформы PS4	36
5.2.3. Графики сравнения метрик по дням для игровой платформы XBox	37
5.3. Исходники программы Python для построения всех графиков и таблиц	39
5.4. Сводная таблица в Excel с ARPU по группам и платформам	41
Заключение	43
Список используемых источников	45

Раздел 1. Цель проекта

Проведен А/В-тест¹. Объект теста — бесплатный командный онлайн-шутер. В игре есть внутриигровая валюта, которую возможно выигрывать, побеждая в матчах, либо покупать за настоящие деньги. На днях в игре прошёл А/В-тест — некоторые игроки могли приобрести премиумную броню по скидке. Руководство хочет узнать, как это повлияло на ARPU (средняя прибыль на игрока), ARPPU (средняя прибыль на платящего игрока) и траты внутриигровой валюты.

Цель работы: выяснить, стоит ли проводить акцию в дальнейшем. Если игроки, участвовавшие в акции, принесли больше денег, чем игроки, у которых акции не было, то стоит повторять акцию и при этом уже на всех игроках.

Важное уточнение: Среди игроков есть читеры — игроки, которые с помощью взлома игры начисляют себе большие объёмы внутриигровой валюты. Имеется список уже известных читеров, но есть и ещё не пойманные читеры, чьи результаты могут повлиять на выводы.

Порядок проведения анализа.

Сначала требуется выявить еще не пойманных читеров. Далее, вычистить перечень игроков от читеров, выявленных до проведения тестирования, и от читеров, выявленных в процессе текущего анализа.

Получив перечень добропорядочных игроков (user_id), сформировать таблицы с совокупной информацией по user_id, группе (test/control), типу игровой платформы и тратам (как у.е. валюты, так и внутриигровой валют). Рассчитать метрики ARPU (средняя прибыль на игрока), ARPPU (средняя прибыль на платящего игрока) и траты внутриигровой валюты. Сформировать сводную таблицу в Excel с ARPU по группам и платформам, а также построить графики сравнения метрик по дням (Power BI).

Далее, для подведения окончательных выводов, построить доверительные интервалы от средних значений с точностью 95% для каждой метрики.

Заключительные выводы об успешности проведения акции и возможные рекомендации о повторном проведении для конкретных игровых платформ.

¹ [Ссылка на веб-страницу](https://habr.com) (Сайт <https://habr.com>)

Раздел 2. Анализ источников

В качестве инструмента работы с данными выбран Python (Jupyter Notebook² (anaconda3)). Он более универсален в рамках конкретного запланированного порядка проведения анализа (более быстрая загрузка данных файлов csv, построение графиков).

Кроме этого, для освоения Python в рамках пройденного обучения выделялось больше времени с учетом программы пройденного курса «Аналитик данных с нуля 2.0».

Исходники кодов по загрузке данных в Jupyter Notebook.

Загружены все исходные данные в Jupyter Notebook (рис. 1).

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.api as sm
import seaborn as sns
from scipy import stats
```

Загрузка csv-файла Money (таблица с платежами).

```
df_money = pd.read_csv('Money.csv')
```

Загрузка csv-файла Cheaters (таблица с обнаруженными читерами).

```
df_cheaters = pd.read_csv('Cheaters.csv');
```

Загрузка csv-файла Platforms (таблица с игровыми платформами (PC, PS4, Xbox)).

```
df_platforms = pd.read_csv('Platforms.csv')
```

Загрузка csv-файла Cash (таблица с тратами внутриигровой валюты).

```
df_cash = pd.read_csv('Cash.csv')
```

Загрузка csv-файла ABgroup (таблица с распределением игроков по группам теста).

```
df_abgroup = pd.read_csv('ABgroup.csv')
```

Рис.1. Скрипт загрузки данных csv-файлов в Jupyter Notebook

Прежде, чем приступить к выполнению расчетов, необходимо проанализировать все загруженные таблицы с данными.

Таблица Money состоит из 3 столбцов (user_id, date, money). В таблице 8640000 строк, при этом столбец user_id содержит повторяющиеся значения каждого user_id для каждого из 8 дней проведения акции с приведением платежей по игроку в конкретный день (рис.76 в разделе 5).

² [Ссылка на веб-страницу](https://habr.com) (Сайт <https://habr.com>)

Таблица Cheaters состоит из 2 столбцов (user_id, cheaters). В таблице также 8640000 строк, столбец user_id также содержит повторяющиеся значения каждого user_id для каждого из 8 дней проведения акции с указанием признака читера (1), либо (0) в обратном случае (рис.77 в разделе 5).

Таблица Platforms состоит из 2 столбцов (user_id, platform). В таблице также 8640000 строк, столбец user_id также содержит повторяющиеся значения каждого user_id для каждого из 8 дней проведения акции с указанием игровой платформы, на которой зарегистрирован игрок (user_id) (рис.78 в разделе 5).

Таблица Cash состоит из 3 столбцов (user_id, cash, date). В таблице 8 640 000 строк, при этом столбец user_id содержит повторяющиеся значения каждого user_id для каждого из 8 дней проведения акции с приведением размера трат внутриигровой валюты по каждому игроку в конкретный день (рис.в разделе 5).

И, наконец, таблица ABgroup состоит из 2 столбцов (user_id, group). В таблице также 8640000 строк, столбец user_id также содержит повторяющиеся значения каждого user_id для каждого из 8 дней проведения акции с указанием группы, к которой отнесен игрок (test/control) (рис.в разделе 5).

В результате можно заключить, что все таблицы содержат по 8 640 000 строк, данные в них приведены в разрезе одного дня для каждого игрока (повтор user_id в каждой таблице 8 раз).

Акция проводилась с 10.07.2021 по 17.07.2021 включительно и общее количество участников = 1 080 000 человек, из них 539 653 участвовали в акции (group = test) (рис.).

Согласно полученным данным (рис.) на момент проведения тестирования выявлено 353 читера (таблица Cheaters) (рис.75 в разделе 5).

Раздел 3. Очистка данных

Прежде, чем выявлять читеров, хотелось бы оценить диапазон объемов внутриигровой валюты, которые с помощью взлома себе начислили уже выявленные читеры.

Для этого отберем уже выявленных читеров в таблице Cheaters и соединим их с таблицей Cash (посредством метода `.groupby()`³). Сгруппируем данные по `user_id` и просуммируем объемы начислений для каждого `user` за весь период проведения акции (рис. 2).

```
# Датафрейм с пользователями-читерами, выявленными до проведения теста
cheat_1 = df_cheaters[df_cheaters['cheaters'] == 1].groupby('user_id').count()
# получена таблица с начисленными объемами внутренней валюты выявленных читеров до:
cheat_1_cash = pd.merge(cheat_1, df_cash, how = 'left', left_on='user_id', right_on='user_id')

# сумма объемов начисленной внутр валюты за 8 дней по каждому ранее выявленному читеру
cheat_1_cash = cheat_1_cash.groupby('user_id').agg('sum')[['cash']].sort_values('cash')
cheat_1_cash
```

	cash
user_id	
GLGM0P-VMHL3W	46700
3WOSJF-UC2DAO	50150
IALNLO-F5DDS6	55800
9YG4CS-DIEGHQ	70300
AYLSIY-0XGKR5	70700
...	...
57T98Z-SYFLSH	222950
N9HQ50-Y2L88S	226500
VTOLQS-23AL7R	231750
PDOMHG-6CPEZ4	237600
6Q3KB0-DF2SIX	253150

353 rows × 1 columns

Рис.2. Датафрейм `cheat_1_cash`

Видим, что объемы начисления внутриигровой валюты с помощью взлома по ранее выявленным читерам варьируются от 46 700 до 253 150 у.е. по каждому из читеров (суммарно за рассматриваемый период 8 дней).

Далее перейдем к поиску пока не выявленных читеров.

Для выявления читеров подготовим сводную таблицу `notcheat_cash` (рис. 3):

- отфильтруем тех игроков, которые имеют признак 0 в таблице Cheaters (сохраним в датафрейм `cheat_2`);
- полученный перечень «не читеров» соединим с таблицей Cash (получим таблицу с начисленными объемами внутренней валюты по игрокам = не читерам (в т.ч. пока не выявленным));

³ [Ссылка на веб-страницу](https://rukovodstvo.net) (Сайт <https://rukovodstvo.net>)

- сгруппируем данные в полученной таблице по user_id и просуммируем объемы начислений внутриигровой валюты по каждому пользователю за весь период проведения акции.

```
# Датафрейм с пользователями-"не читерами" (с учетом не выявленных пока читеров)
cheat_2 = df_cheaters[df_cheaters['cheaters'] == 0].groupby('user_id').count()
# получена табл с начисленными объемами внутр валюты НЕ читерам (в т.ч. пока не выявленным):
notcheat_cash = pd.merge(cheat_2, df_cash, how = 'left', left_on='user_id', right_on='user_id')
notcheat_cash = notcheat_cash.groupby('user_id').agg('sum')[['cash']]
notcheat_cash.sort_values('cash')
```

user_id	cash
DTU9N4-QS35SQ	50
INBV7M-9S58NQ	50
4KAX70-CHAJX2	50
S6F3GG-MUGQG3	100
2CJYT7-AXT4GP	100
...	...
BZRZ88-435FHY	225650
OCRQ1L-GZH0BP	230600
BGTT4M-1ZFLMA	235100
0R9XB0-4J1EVJ	238500
4HCVCA-V3M19Y	240950

1079647 rows × 1 columns

Рис.3. Датафрейм notcheat_cash

Видим, что разброс велик, от 50 до 240 950 у.е. суммарно по каждому игроку за рассматриваемый период. Очевидно, что требуется выявить читеров и убрать их из списка для последующей оценки эффективности проведения ab-теста.

Перейдем к поиску читеров (аномалий).

Рассмотрим точечные оценки выборки трат внутриигровой валюты не читерами (в т.ч. пока не выявленными читерами) (cash в таблице notcheat_cash). Для этого подготовлена функция basic_research(), с помощью которой будут рассчитаны точечные оценки выборки, а также построен график распределения (библиотека matplotlib (plt)⁴) (рис. 4).

```
def basic_research (df = notcheat_cash, column = 'cash'):
    print("Базовые метрики")
    print(df[column].describe())
    print("-----")
    print("Самые популярные значения cash, топ 5")
    print(df[column].value_counts().nlargest(5))
    print("-----")
    sns.distplot(df[column])
    plt.title("Распределение трат внутриигровой валюты по игрокам")
```

Рис.4. Функция basic_research() для анализа точечные оценки выборки трат внутриигровой валюты

⁴ Уэс Маккини. Python и анализ данных. – 2020 - №2 – С. 274

```

basic_research()

Базовые метрики
count    1.079647e+06
mean     6.061209e+03
std      3.004201e+03
min      5.000000e+01
25%      5.050000e+03
50%      6.000000e+03
75%      6.950000e+03
max      2.409500e+05
Name: cash, dtype: float64
-----
Самые популярные значения cash, топ 5
6050    15629
6000    15600
5850    15510
6100    15501
5900    15499
Name: cash, dtype: int64
-----

```

Рис.5. Результат выполнения функции `basic_research()`

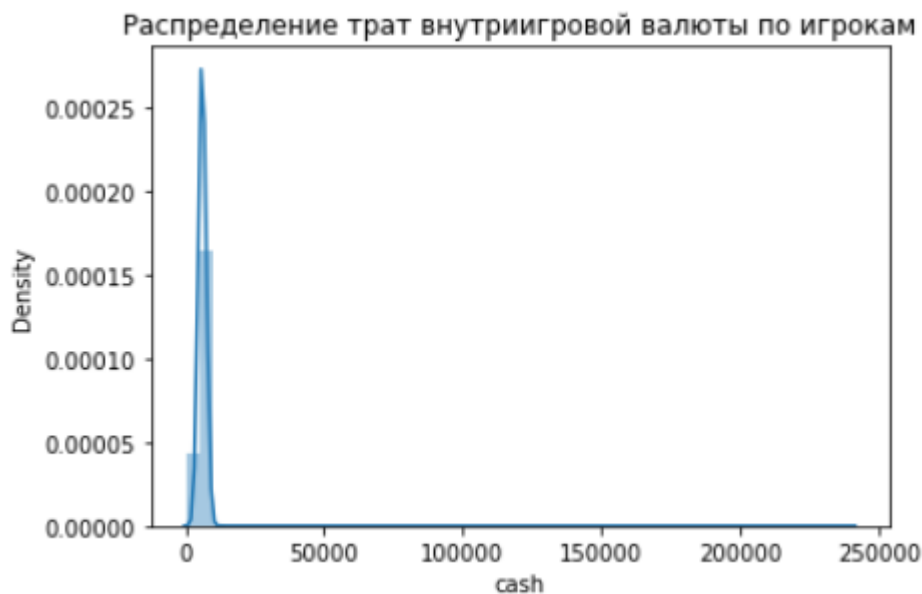


Рис.6. Результат выполнения функции `basic_research()`, построение графика распределения трат внутриигровой валюты

Анализируя данные, полученные в результате выполнения функции `basic_research()` (рис.5, рис.6) очевидно, что распределение `cash` (трат внутриигровой валюты) ассиметричное (наличие читеров в нем делает диаграмму смещенной вправо).

Построим график `QQ plot` (рис.7) (который сравнивает квантили выборки `cash` и квантили нормального распределения с таким же средним и сигмой).


```
sm.qqplot(notcheat_cash['cash'], fit = True)
plt.title('QQ plot')
plt.show()
```

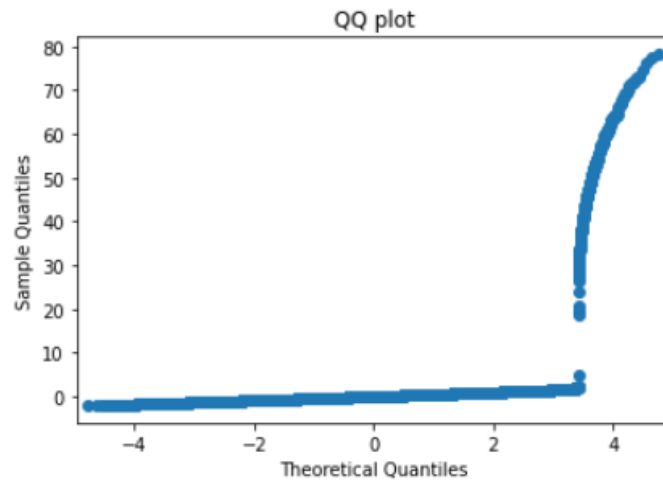


Рис.7. Построение графика QQ plot

Видим (рис.7), что cash имеет не очевидно нормальное распределение, левая часть графика не под 45 градусов (это объемы трат внутриигровой валюты читерами), но вторая часть графика похожа на нормальное распределение.

Рассчитаем доверительный интервал при доверительной вероятности 95% (рис.8).

```
def my_norm_confidence(df=notcheat_cash, column='cash', alpha = 0.95):
    interval = stats.norm.interval(alpha, loc=df[column].mean(), scale = df[column].std())
    return interval
```

```
confidence = my_norm_confidence()
print(confidence)
```

```
(173.08327322169953, 11949.334517037525)
```

Рис.8. Функция my_norm_confidence() и результат ее выполнения

Отметим полученный коридор на графике cash (рис.9).

```
notcheat_cash['cash'].plot()
plt.axhline(y = confidence[1], color = 'g', linestyle = '-')
plt.axhline(y = confidence[0], color = 'r', linestyle = '-')
plt.title('Доверительный интервал')
```

```
Text(0.5, 1.0, 'Доверительный интервал')
```



Рис.9. График распределения трат внутриигровой валюты (cash) и отмеченный доверительный интервал

Все, что выше верхней границы доверительного интервала = 11 949, можно смело называть аномалией или, в данном конкретном случае, проделками читеров.

Отметим читеров (аномальные события, выходящие за верхнюю зеленую границу) (рис.10).

```
notcheat_cash['anomaly_big'] = np.where(notcheat_cash['cash'] > confidence[1], 1, 0)
```

```
found_cheaters = notcheat_cash[notcheat_cash['anomaly_big'] == 1]
found_cheaters.describe()
```

	cash	anomaly_big
count	362.000000	362.0
mean	144094.889503	1.0
std	45314.555978	0.0
min	11950.000000	1.0
25%	122362.500000	1.0
50%	147525.000000	1.0
75%	171387.500000	1.0
max	240950.000000	1.0

Рис.10. Добавление в df notcheat_cash столбца 'anomaly_big' с пометкой аномалий

С помощью доверительного интервала выявлены и отобраны 362 читера. Итого, с учетом выявленных ранее 353 читеров, сумма всех читеров составила = 715 читеров.

Теперь получим перечень добропорядочных игроков (notcheat), который дальше будем использовать для анализа результатов ab-тестирования (рис.11). Фильтр df notcheat_cash по признаку аномалии = 0.

```
notcheat = notcheat_cash[notcheat_cash['anomaly_big'] == 0]
notcheat.sort_values('user_id')
```

	cash	anomaly_big
user_id		
00036L-1T885I	7600	0
0004MK-MRBUKF	6150	0
0004MK-VW0O9L	6300	0
0005S4-UEC6BE	5000	0
0007I5-EWUD6O	6350	0
...
ZZZPEX-P2T8BH	6600	0
ZZZTEN-0QGZ3X	4600	0
ZZZTR5-QSI6D3	5900	0
ZZZV9K-NM9B4Z	3900	0
ZZZVH1-2OJT7I	5250	0

1079285 rows × 2 columns

Рис.11. Получение перечня добропорядочных игроков (notcheat)

Итого: 1 079 285 игроков отображены для проведения анализа результатов аб-тестирования.

Проверим полученный список не читеров на нормальное распределение (Гауссово распределение) для подтверждения гипотезы, что распределение трат внутриигровой валюты (cash) после исключения объемов читеров станет нормально распределенным. Построим гистограмму распределения 'cash' оставшихся игроков = не читеров (рис. 12).

```
# построим гистограмму распределения
notcheat['cash'].hist(bins = 100)
plt.title('Распределение V внутр валюты, полученной НЕ читерами')
```

Text(0.5, 1.0, 'Распределение V внутр валюты, полученной НЕ читерами')

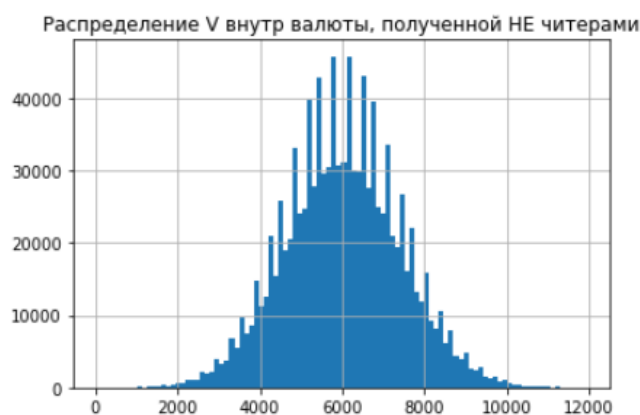


Рис.12. Гистограмма распределения 'cash' оставшихся игроков = не читеров

```
notcheat['cash'].plot()
plt.title('Распределение V внутр валюты, полученной НЕ читерами')
Text(0.5, 1.0, 'Распределение V внутр валюты, полученной НЕ читерами')
```

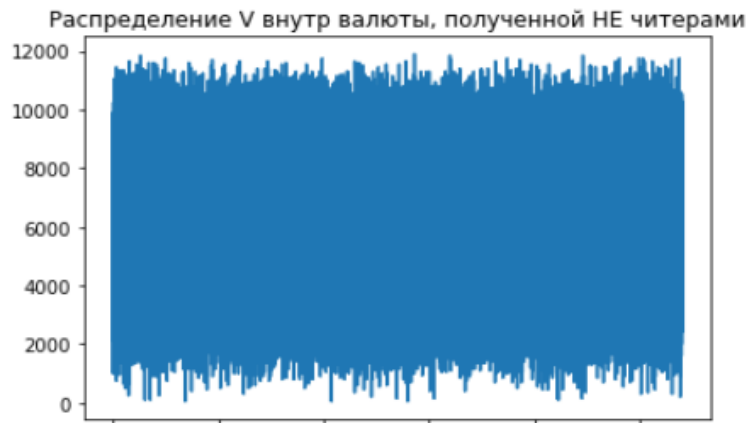


Рис.13. Распределение объемов трат внутренней валюты игроками не читерами

```
# построим график QQ plot
sm.qqplot(notcheat['cash'], fit = True)
plt.title('QQ plot')
plt.show()
```

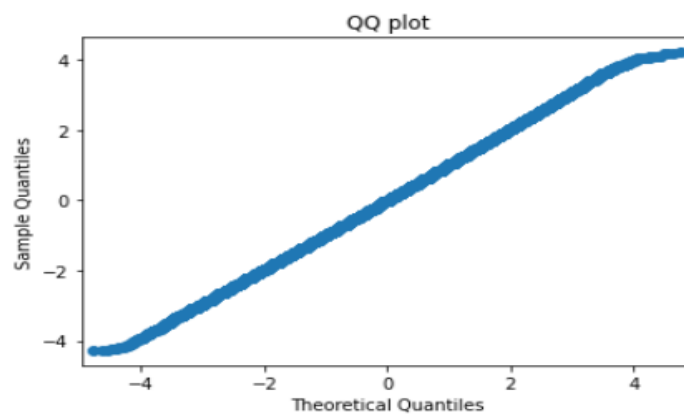


Рис.14. Построение графика QQ plot объемов трат внутренней валюты игроками не читерами

Теперь видим (рис. 12-14), что данные очень близки к Гауссову распределению, график QQ plot расположен под 45 градусов.

Раздел 4. Использование статистических методов

Для построения доверительных интервалов ниже приведены сводные таблицы (рис. 15-17) с рассчитанными метриками по каждой из игровых платформ, которые являются исходными данными для построения доверительных интервалов (подробный расчет метрик приведен в разделе 5).

```
PC_metrics = pd.merge(ARPU_PC, ARPPU_PC, left_on='date', right_on='date')
PC_metrics = pd.merge(PC_metrics, cash_PC, left_on='date', right_on='date')
PC_metrics
```

	AR_t_PC	AR_c_PC	perc_PC	ARPP_t_PC	ARPP_c_PC	perc_ARPP	cash_t	cash_c	perc_cash
date									
10.07.2021	0.783	0.717	9.3	1.229	1.208	1.7	848.750	749.533	13.2
11.07.2021	0.782	0.713	9.7	1.229	1.207	1.8	848.167	748.794	13.3
12.07.2021	0.779	0.712	9.4	1.228	1.208	1.6	843.568	745.380	13.2
13.07.2021	0.762	0.697	9.3	1.224	1.202	1.8	822.536	726.588	13.2
14.07.2021	0.724	0.660	9.7	1.211	1.188	1.9	768.224	679.558	13.0
15.07.2021	0.670	0.605	10.7	1.193	1.169	2.1	677.087	593.841	14.0
16.07.2021	0.592	0.539	9.7	1.163	1.138	2.2	546.502	478.081	14.3
17.07.2021	1.176	1.003	17.3	1.953	1.737	12.4	1026.585	865.326	18.6

Рис.15. Получение сводной таблицы по рассчитанным метрикам игровой платформы PC

Результаты сравнения метрик ARPU, ARPPU и трат внутриигровой валюты на платформе PC (рис.15) показывают, что приход денег/трат внутриигровой валюты от игроков группы участников, которые участвовали в тестировании (test, сокращенно в имени метрики - _t), больше второй группы, которая не участвовала в акции (control, сокращенно в имени метрики - _c). Что позволяет предварительно сделать вывод об успешности проведения акции на игровой платформе PC.

Метрики ARPU (по дням) группы test (AR_t_PC) выше аналогичной метрики группы control (AR_c_PC) ежедневно более чем на 9%.

Метрики ARPPU (по дням) группы test (ARPP_t_PC) также выше аналогичной метрики группы control (ARPP_c_PC), но уже на меньший процент (в среднем 1,5%), но сохраняется ежедневная тенденция превышения метрики группы test над аналогичной метрикой группы control.

Такая же тенденция в тратах внутриигровой валюты на платформе PC, группа test (cash_t) показывает превышение над группой control (cash_c) в среднем на 13,3%.

Отметим рост увеличения по группе test относительно группы control по всем метрикам площадки PC в последний день проведения акции. Это можно объяснить психологией человека «успеть в последний вагон».

Сравнение метрик указывает на то, что проведение акции уже на всех игроках платформы PC принесет пользу (увеличение прибыли, увеличение трат внутриигровой

валюты). Окончательное решение возможно принять после построения доверительных интервалов.

	AR_t_PS4	AR_c_PS4	perc_PS4	ARPP_t_PS4	ARPP_c_PS4	perc_ARPP	cash_t	cash_c	perc_cash
date									
10.07.2021	0.750	0.717	4.6	1.218	1.208	0.8	799.965	750.972	6.5
11.07.2021	0.753	0.714	5.4	1.222	1.207	1.2	799.857	750.073	6.6
12.07.2021	0.747	0.710	5.2	1.216	1.207	0.7	794.961	744.970	6.7
13.07.2021	0.735	0.693	6.0	1.216	1.201	1.3	780.440	724.262	7.8
14.07.2021	0.703	0.663	6.1	1.204	1.190	1.2	734.040	679.014	8.1
15.07.2021	0.648	0.611	6.1	1.183	1.170	1.1	653.338	598.475	9.2
16.07.2021	0.578	0.546	5.8	1.156	1.142	1.3	532.290	487.906	9.1
17.07.2021	1.172	1.085	8.0	1.921	1.824	5.3	1035.495	946.264	9.4

Рис.16. Получение сводного датафрейма по рассчитанным метрикам игровой платформы PS4

Результаты сравнения метрик ARPU, ARPPU и трат внутриигровой валюты на платформе PS4 (рис.16) также можно назвать положительными.

Метрики ARPU (по дням) группы test (AR_t_PS4) выше аналогичной метрики группы control (AR_c_PS4) ежедневно более чем на 6% (среднее за 7 дней, без последнего дня).

Метрики ARPPU (по дням) группы test (ARPP_t_PS4) также выше аналогичной метрики группы control (ARPP_c_PS4), но уже на меньший процент (в среднем 1,1%), но сохраняется ежедневная тенденция превышения метрики группы test над аналогичной метрикой группы control.

Такая же тенденция в тратах внутриигровой валюты на платформе PS4, группа test (cash_t) показывает ежедневное превышение над группой control (cash_c) в среднем на 7,9%.

Также отмечается рост увеличения по группе test относительно группы control площадки PS4 в последний день проведения акции, но уже не по всем метрикам, а только по ARPU, ARPPU.

Сравнение метрик указывает на то, что проведение акции уже на всех игроках платформы PC принесет увеличение прибыли, увеличение трат внутриигровой валюты, но в меньшей степени, нежели чем на платформе PS4. Окончательное решение возможно принять после построения доверительных интервалов.

	AR_t_Xb	AR_c_Xb	perc_Xb	ARPP_t_Xb	ARPP_c_Xb	perc_ARPP	cash_t	cash_c	perc_cash
date									
10.07.2021	0.750	0.755	-0.7	1.219	1.221	-0.1	798.724	801.320	-0.3
11.07.2021	0.751	0.753	-0.3	1.218	1.219	-0.1	800.721	800.393	0.0
12.07.2021	0.749	0.748	0.1	1.220	1.219	0.1	795.484	795.144	0.0
13.07.2021	0.734	0.736	-0.4	1.215	1.215	-0.0	777.967	778.912	-0.1
14.07.2021	0.704	0.702	0.2	1.205	1.204	0.1	737.205	732.771	0.6
15.07.2021	0.649	0.648	0.1	1.186	1.187	-0.0	651.932	648.521	0.5
16.07.2021	0.581	0.576	0.8	1.157	1.154	0.2	537.521	530.738	1.3
17.07.2021	1.216	1.184	2.8	1.964	1.935	1.5	1076.715	1043.961	3.1

Рис.17. Получение сводного датафрейма по рассчитанным метрикам игровой платформы XBox

Результаты сравнения метрик ARPU, ARPPU и трат внутриигровой валюты на платформе XBox (рис.16) не утешительные.

Метрики ARPU (по дням) группы test (AR_t_Xb) в 5 из 8 дней имеют превышение над аналогичной метрикой группы control (AR_c_Xb) и это превышение не достигает даже 1%.

Метрики ARPPU (по дням) группы test (ARPP_t_Xb) уже в 4 из 8 дней имеют превышение над аналогичной метрикой группы control (ARPP_c_Xb), и это превышение крайне незначительно (0,1%).

Такая же тенденция в тратах внутриигровой валюты на платформе PS4, группа test (cash_t) имеет незначительное превышение над группой control (cash_c) только в 6 из 8 дней проведения акции.

Также отмечается рост увеличения по группе test относительно группы control площадки PS4 в последний день проведения акции, но уже не по всем метрикам, а только по ARPU, ARPPU.

Сравнение метрик указывает на то, что акция не принесла успеха (прибыли) на игровой платформе Xbox, проведение акции повторно не целесообразно. Окончательное решение возможно принять после построения доверительных интервалов.

4.1. Определение соотношения количества участников в группах по каждой игровой платформе

Определим сколько человек в группе test PC и сколько в группе control PC для понимания сопоставимости метрик по группам платформ (рис.18).

```
len(ab_money[(ab_money['platform'] == 'PC') & (ab_money['group'] == 'test')])/8
```

```
179810.0
```

```
len(ab_money[(ab_money['platform'] == 'PC') & (ab_money['group'] == 'control')])/8
```

```
179732.0
```

Рис.18. Численность групп test и control платформы PC

Количество групп практически одинаковое, группа, участвующая в акции, больше контрольной на 78 человек (0,02% от всех игроков платформы PC). Этой погрешностью можно пренебречь при сравнении метрик между группами платформы PC.

Определим сколько человек в группе test PS4 и сколько в группе control PS4 (рис.19).

```
len(ab_money[(ab_money['platform'] == 'PS4') & (ab_money['group'] == 'test')])/8  
179412.0
```

```
len(ab_money[(ab_money['platform'] == 'PS4') & (ab_money['group'] == 'control')])/8  
179804.0
```

Рис.19. Численность групп test и control платформы PS4

Количество групп незначительно отличается, группа, участвующая в акции, меньше контрольной на 392 человека (0,1% от всех игроков платформы PS4). Этой погрешностью можно пренебречь при сравнении метрик между группами платформы PS4.

Определим сколько человек в группе test Xbox и сколько в группе control Xbox (рис.20).

```
len(ab_money[(ab_money['platform'] == 'XBox') & (ab_money['group'] == 'test')])/8  
180073.0
```

```
len(ab_money[(ab_money['platform'] == 'XBox') & (ab_money['group'] == 'control')])/8  
180454.0
```

Рис.20. Численность групп test и control платформы Xbox

Количество групп незначительно отличается, группа, участвующая в акции, меньше контрольной на 381 человека (0,1% от всех игроков платформы Xbox). Этой погрешностью можно пренебречь при сравнении метрик между группами платформы Xbox.

4.2. Построение доверительных интервалов для каждой метрики от средних значений с точностью 95%

4.2.1. Построение доверительных интервалов для метрик игровой платформы PC.

Напишем функцию `tu_norm_confidence()`, внутри которой будем использовать функцию из библиотеки `scipy`⁵, для нахождения доверительного интервала, которая на вход будет принимать `df` и имя столбца (для значений которого требуется рассчитать

⁵ [Ссылка на веб-страницу](https://docs.scipy.org) (Сайт <https://docs.scipy.org>)

доверительный интервал) (рис.21). Все доверительные интервалы по другим метрикам будут рассчитаны далее с помощью этой функции.

Нахождение доверительных интервалов для метрики ARPU по обеим группам игровой платформы PC:

```
def my_norm_confidence(df=PC_metrics, column='AR_t_PC', alpha = 0.95):  
    interval = stats.norm.interval(alpha, loc=df[column].mean(), scale = df[column].std())  
    return interval  
confidence_t = my_norm_confidence()  
print(confidence_t)  
  
(0.44594045103257385, 1.121059548967426)
```

```
confidence_c = my_norm_confidence(df=PC_metrics, column='AR_c_PC', alpha = 0.95)  
print(confidence_c)  
  
(0.4399892288403603, 0.9715107711596397)
```

Рис.21. Функция my_norm_confidence() и ее результат для метрики ARPU по обеим группам платформы PC

Отобразим на графике доверительные интервалы для метрики ARPU по группам платформы PC (рис.22):

```
plt.axhline(y = confidence_t[1], color = 'g', linestyle = '-')  
plt.axhline(y = confidence_t[0], color = 'g', linestyle = '-')  
plt.axhline(y = confidence_c[1], color = 'r', linestyle = '-')  
plt.axhline(y = confidence_c[0], color = 'r', linestyle = '-')  
plt.title('Доверительный интервал')  
  
Text(0.5, 1.0, 'Доверительный интервал')
```

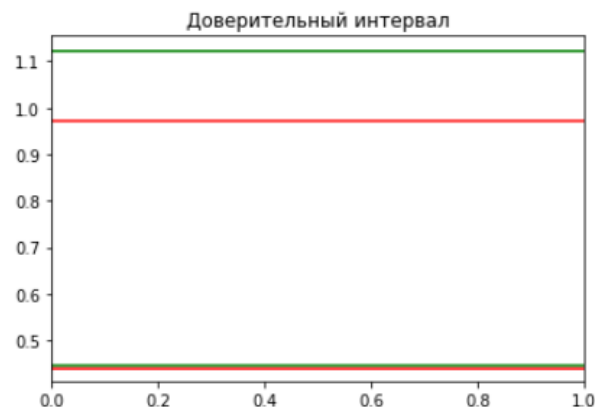


Рис.22. Доверительные интервалы на графике для ARPU по обеим группам платформы PC

Нахождение доверительных интервалов для метрики ARPPU по обеим группам игровой платформы PC и отображение доверительных интервалов на графике (рис.23):

```
confidence_t = my_norm_confidence(df=PC_metrics, column='ARPP_t_PC', alpha = 0.95)
print(confidence_t)
```

```
(0.7875908572344139, 1.819909142765586)
```

```
confidence_c = my_norm_confidence(df=PC_metrics, column='ARPP_c_PC', alpha = 0.95)
print(confidence_c)
```

```
(0.8740586651515616, 1.6401913348484385)
```

```
plt.axhline(y = confidence_t[1], color = 'g', linestyle = '-')
plt.axhline(y = confidence_t[0], color = 'g', linestyle = '-')
plt.axhline(y = confidence_c[1], color = 'r', linestyle = '-')
plt.axhline(y = confidence_c[0], color = 'r', linestyle = '-')
plt.title('Доверительный интервал')
```

```
Text(0.5, 1.0, 'Доверительный интервал')
```

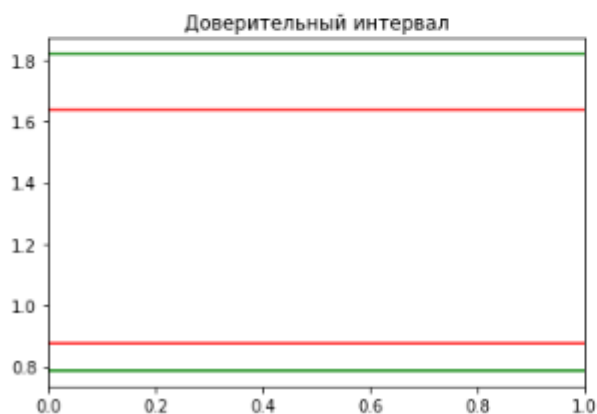


Рис.23. Расчет доверительных интервалов и их отображение на графике для ARPPU PC

Нахождение доверительных интервалов для трат внутренней валюты по обеим группам на игровой платформе PC и отображение доверительных интервалов на графике (рис.24):

```
confidence_t = my_norm_confidence(df=PC_metrics, column='cash_t', alpha = 0.95)
print(confidence_t)
```

```
(521.7081366314266, 1073.6466133685735)
```

```
confidence_c = my_norm_confidence(df=PC_metrics, column='cash_c', alpha = 0.95)
print(confidence_c)
```

```
(469.04171366919445, 927.7335363308057)
```

```
plt.axhline(y = confidence_t[1], color = 'g', linestyle = '-')
plt.axhline(y = confidence_t[0], color = 'g', linestyle = '-')
plt.axhline(y = confidence_c[1], color = 'r', linestyle = '-')
plt.axhline(y = confidence_c[0], color = 'r', linestyle = '-')
plt.title('Доверительный интервал')
```

```
Text(0.5, 1.0, 'Доверительный интервал')
```

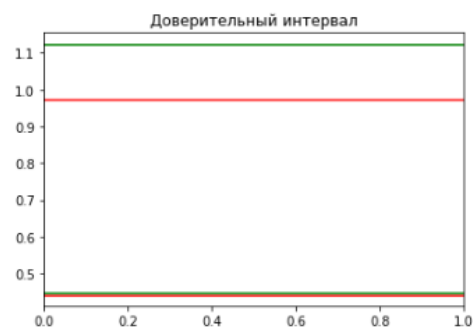


Рис.24. Расчет доверительных интервалов и их отображение на графике для cash PC

Полученные доверительные интервалы группы test по всем метрикам игровой платформы PC значительно превышают доверительные интервалы группы control, что подтверждает сделанные ранее выводы об успешности проведения акции для данной игровой платформы (при сравнении рассчитанных метрик). Считаю, что повторное проведение акции уже на всех игроках платформы PC принесет пользу (увеличение прибыли, увеличение трат внутриигровой валюты).

4.2.2. Построение доверительных интервалов для метрик игровой платформы PS4

Нахождение доверительных интервалов для метрики ARPU по обеим группам игровой платформы PS4 и отображение доверительных интервалов на графике (рис.25):

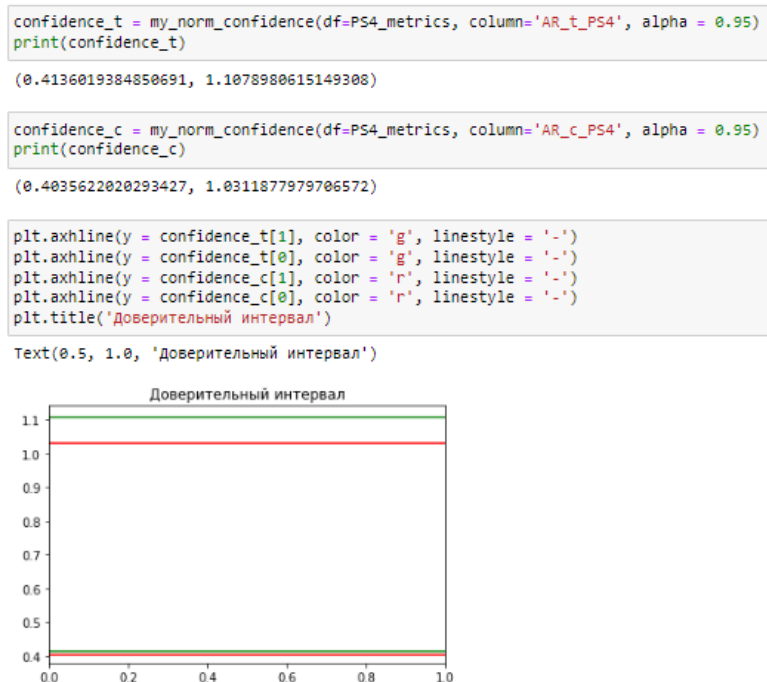


Рис.25. Расчет доверительных интервалов и их отображение на графике для ARPU PS4

Нахождение доверительных интервалов для метрики ARPPU по обеим группам игровой платформы PS4 и отображение доверительных интервалов на графике (рис.26):

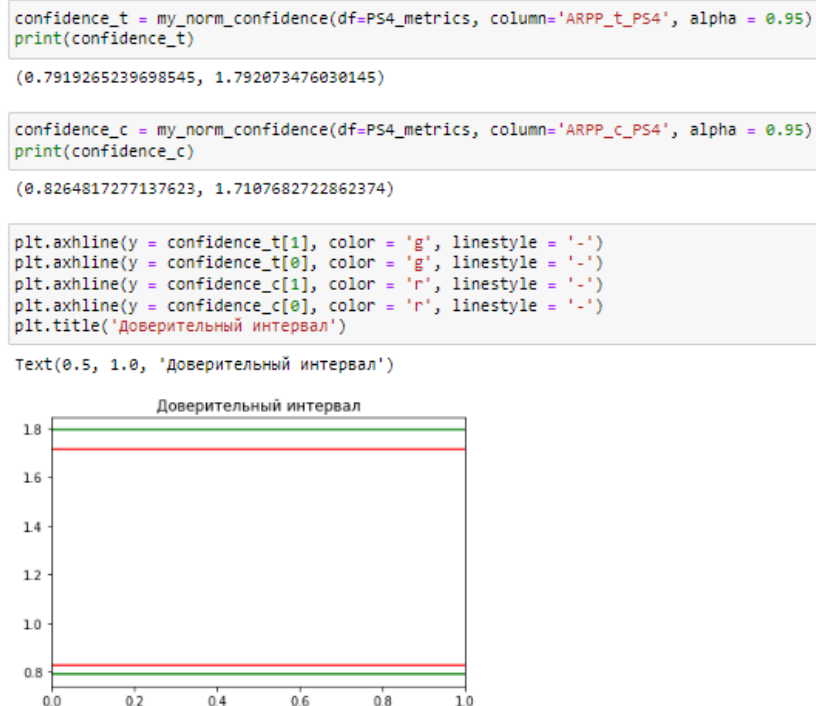


Рис.26. Расчет доверительных интервалов и их отображение на графике для ARPPU PS4

Нахождение доверительных интервалов для трат внутренней валюты по обеим группам на игровой платформе PS4 и отображение доверительных интервалов на графике (рис.27):

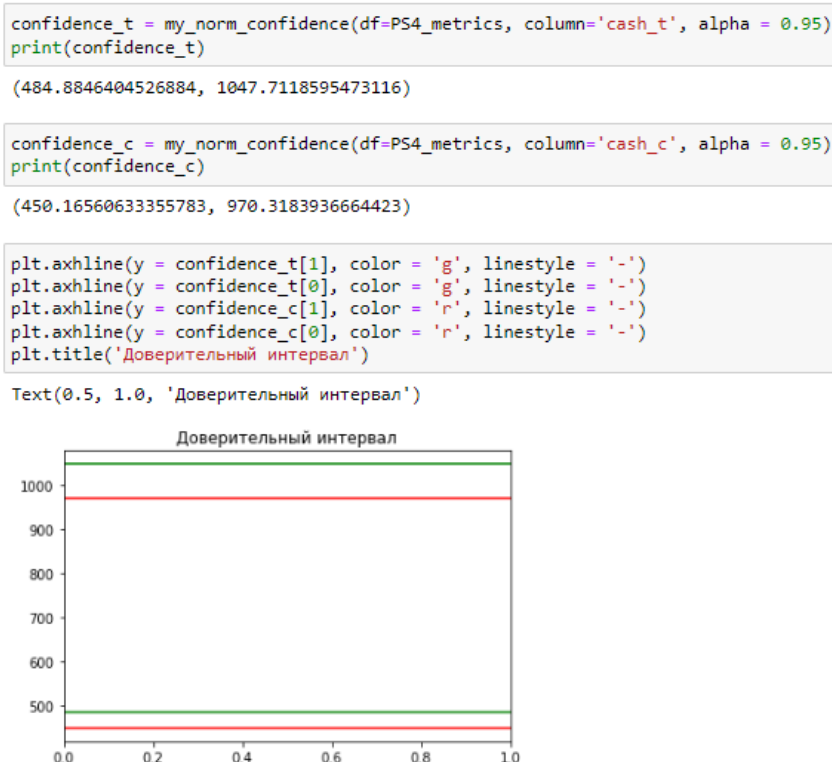


Рис.27. Расчет доверительных интервалов и их отображение на графике для cash PS4

Полученные доверительные интервалы группы test по всем метрикам игровой платформы PS4 также превышают доверительные интервалы группы control, но в меньшей степени, нежели рассмотренная выше платформа PC.

Сопоставление полученных доверительных интервалов по метрикам игровой платформы PS4 подтверждает сделанные ранее выводы об успешности проведения акции для данной игровой платформы (при сравнении рассчитанных метрик).

Считаю, что повторное проведение акции уже на всех игроках платформы PS4 принесет пользу (увеличение прибыли, увеличение трат внутриигровой валюты).

4.2.3. Построение доверительных интервалов для метрик игровой платформы Xbox

Нахождение доверительных интервалов для метрики ARPU по обеим группам игровой платформы Xbox и отображение доверительных интервалов на графике (рис.28):

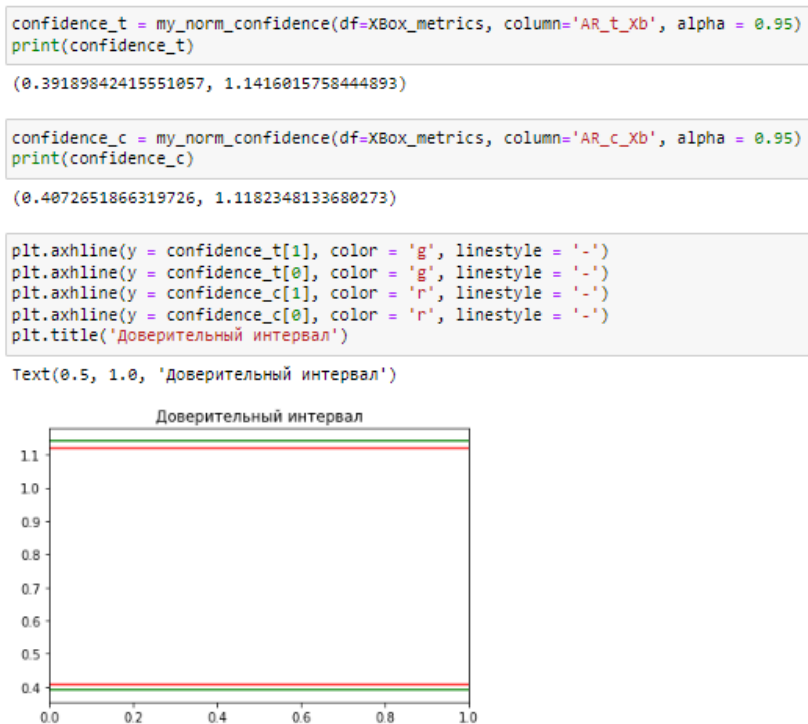


Рис.28. Расчет доверительных интервалов и их отображение на графике для ARPU Xbox

Нахождение доверительных интервалов для метрики ARPPU по группам игровой платформы Xbox и отображение доверительных интервалов на графике (рис.29):

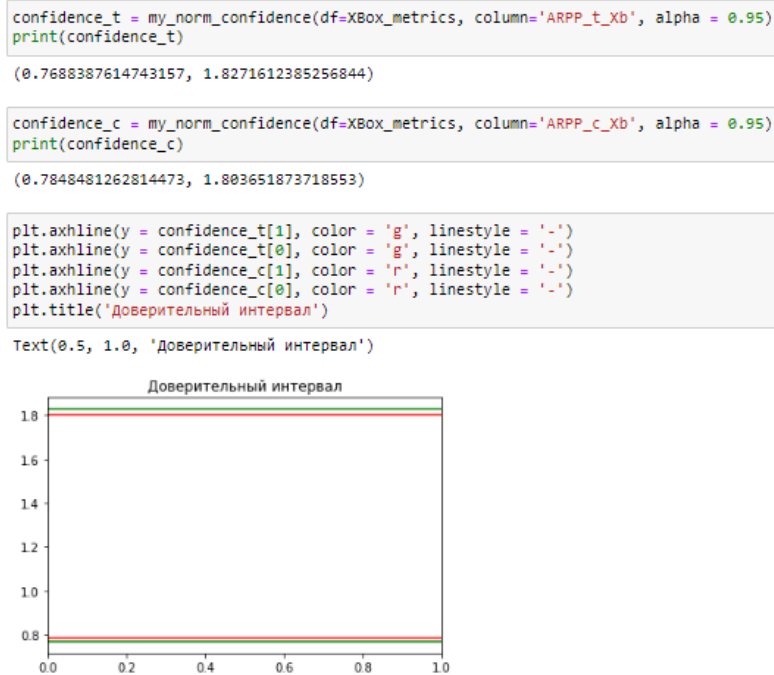


Рис.29. Расчет доверительных интервалов и их отображение на графике для ARPPU XBox

Нахождение доверительных интервалов для трат внутренней валюты по группам на игровой платформе XBox и отображение доверительных интервалов на графике (рис.30):

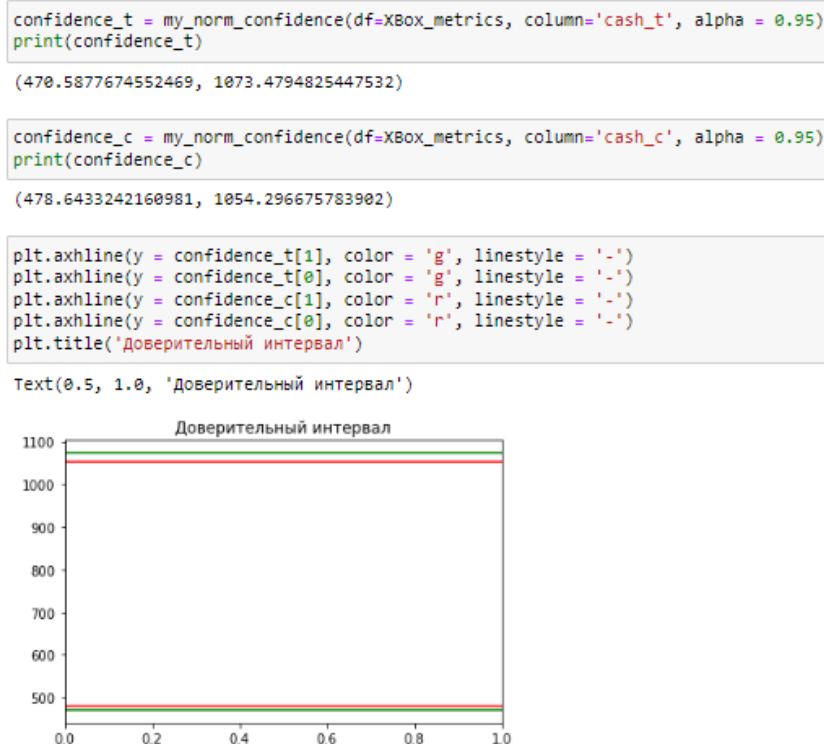


Рис.30. Расчет доверительных интервалов и их отображение на графике для cash XBox

Полученные верхние границы доверительных интервалов группы test по всем метрикам игровой платформы XBox превышают, но не верхние границы доверительных интервалов группы control.

Получаем и по платформе Xbox подтверждение сделанных ранее выводов о не успешности проведения акции для данной игровой платформы (при сравнении рассчитанных метрик).

Однозначно, повторное проведение акции уже на всех игроках платформы Xbox не целесообразно.

Раздел 5. Формирование отчета

5.1. Расчет и сравнение метрик ARPU (средняя прибыль на игрока), ARPPU (средняя прибыль на платящего игрока) и траты внутриигровой валюты между различными группами игроков

Для расчета метрик подготовим 2 сводные таблицы ab_money, ab_cash по добропорядочным игрокам (notcheat).

Для этого:

- сгруппируем таблицу Platforms по user_id (platforms);
- сгруппируем таблицу ABgroup по user_id (abgroup);
- оставим в таблице notcheat только перечень неповторяющихся user_id игроков - не читеров (удалим столбцы⁶ anomaly_big и cash);
- соединим таблицы platforms и табл notcheat (с user_id = "не читерами"), получим перечень уникальных user_id без читеров с указанием платформы (получим df = users_platforms);
- соединим таблицы users_platforms и таблицу abgroup (получим df = users_group) (рис.31).

```
platforms = df_platforms.groupby('user_id').agg('max')[['platform']]
abgroup = df_abgroup.groupby('user_id').agg('max')[['group']]
notcheat = notcheat.drop(columns=['anomaly_big', 'cash'])
users_platforms = pd.merge(notcheat, platforms, left_on='user_id', right_on='user_id')
users_group = pd.merge(users_platforms, abgroup, left_on='user_id', right_on='user_id')
```

Рис.31. Получение сводного df users_group

Получен датафрейм users_group (рис.32) с уникальным перечнем user_id (без читеров), который содержит данные по группе (test/control) и типу игровой платформы игрока (PC/PS4/Xbox).

```
users_group.sort_values('user_id')
```

user_id	platform group	
	platform	group
00036L-1T885I	XBox	control
0004MK-MRBUKF	PS4	control
0004MK-VW009L	PC	test
0005S4-UEC6BE	PC	test
0007I5-EWUD6O	PC	test
...
ZZZPEX-P2T8BH	PC	test
ZZZTEN-0QGZ3X	XBox	test
ZZZTR5-QSI6D3	XBox	test
ZZZV9K-NM9B4Z	PS4	control
ZZZVH1-2OJT7I	PC	control

1079285 rows × 2 columns

⁶ [Ссылка на веб-страницу](https://pythonpip.ru) (Сайт <https://pythonpip.ru>)

Рис.32. Датафрейм users_group

Создание сводной таблицы ab_money соединением таблиц users_group и Money. В результате получим полную информацию по игрокам с учетом платежей игроком в каждый из 8 дней (рис.33).

```
ab_money = pd.merge(users_group, df_money, left_on='user_id', right_on='user_id')
ab_money.sort_values(['user_id', 'date']).head(10)
```

	user_id	platform	group	date	money
0	00036L-1T885I	XBox	control	10.07.2021	0.00
2	00036L-1T885I	XBox	control	11.07.2021	0.00
3	00036L-1T885I	XBox	control	12.07.2021	0.99
6	00036L-1T885I	XBox	control	13.07.2021	0.00
4	00036L-1T885I	XBox	control	14.07.2021	0.00
7	00036L-1T885I	XBox	control	15.07.2021	0.00
1	00036L-1T885I	XBox	control	16.07.2021	0.99
5	00036L-1T885I	XBox	control	17.07.2021	1.99
9	0004MK-MRBUKF	PS4	control	10.07.2021	0.00
10	0004MK-MRBUKF	PS4	control	11.07.2021	1.99

Рис.33. Датафрейм ab_money

Аналогично соединим таблицу users_group и таблицу Cash. Получим таблицу ab_cash с полной информацией по игрокам с учетом трат внутриигровой валюты игроком в каждый из 8 дней (рис.34).

```
ab_cash = pd.merge(users_group, df_cash, left_on='user_id', right_on='user_id')
ab_cash.sort_values(['user_id', 'date']).head(10)
```

	user_id	platform	group	date	cash
5	00036L-1T885I	XBox	control	10.07.2021	700
1	00036L-1T885I	XBox	control	11.07.2021	0
3	00036L-1T885I	XBox	control	12.07.2021	1500
2	00036L-1T885I	XBox	control	13.07.2021	850
4	00036L-1T885I	XBox	control	14.07.2021	650
6	00036L-1T885I	XBox	control	15.07.2021	700
0	00036L-1T885I	XBox	control	16.07.2021	550
7	00036L-1T885I	XBox	control	17.07.2021	2650
10	0004MK-MRBUKF	PS4	control	10.07.2021	550
14	0004MK-MRBUKF	PS4	control	11.07.2021	1100

Рис.34. Датафрейм ab_cash

5.1.1. Расчет и сравнение метрик для платформы PC

Далее расчет метрик. Чтобы уменьшить загрузку системы, при расчете каждой метрики произвожу предварительное разделение таблиц ab_money, ab_cash по типам игровых платформ и типу группы. Иначе мой ПК не справляется.

В процессе создания таблицы со значением метрик и их сравнением необходимо присваивать столбцам названия, используя для этого метод `rename()`⁷.

Расчет метрики ARPU(средняя прибыль на игрока) для двух групп на платформе PC (рис. 35-36).

```
PC_test_m = ab_money[(ab_money['platform'] == 'PC') & (ab_money['group'] == 'test')]
ARPU_PC_test = PC_test_m.groupby('date').agg('mean')[['money']]
ARPU_PC_test.rename(columns={'money': 'ARP_t_PC'}, inplace=True)
ARPU_PC_test
```

Рис.35. Расчет ARPU для группы test платформы PC

```
PC_control_m = ab_money[(ab_money['platform'] == 'PC') & (ab_money['group'] == 'control')]
ARPU_PC_control = PC_control_m.groupby('date').agg('mean')[['money']]
ARPU_PC_control.rename(columns={'money': 'ARP_c_PC'}, inplace=True)
ARPU_PC_control
```

Рис.36. Расчет ARPU для группы control платформы PC

Соединим в один df метрики ARPU для двух групп площадки PC и рассчитаем эффект в % (рис.37):

```
ARPU_PC = pd.merge(ARPU_PC_test, ARPU_PC_control, left_on='date', right_on='date')
ARPU_PC['perc_ARP'] = ARPU_PC['ARP_t_PC'] / ARPU_PC['ARP_c_PC'] * 100 - 100
ARPU_PC = ARPU_PC.round({'ARP_t_PC': 3, 'ARP_c_PC': 3, 'perc_ARP': 1})
ARPU_PC
```

	ARP_t_PC	ARP_c_PC	perc_ARP
date			
10.07.2021	0.783	0.717	9.3
11.07.2021	0.782	0.713	9.7
12.07.2021	0.779	0.712	9.4
13.07.2021	0.762	0.697	9.3
14.07.2021	0.724	0.660	9.7
15.07.2021	0.670	0.605	10.7
16.07.2021	0.592	0.539	9.7
17.07.2021	1.176	1.003	17.3

Рис.37. df метрики ARPU PC и их сравнение

Рассчитаем ARPPU(средняя прибыль на платящего игрока) для двух групп на платформе PC (рис. 38-39).

⁷ Кристиан Хилл. Научное программирование на Python. – 2021. - С. 510

```
PC_test_m_2 = ab_money[(ab_money['platform'] == 'PC') & (ab_money['group'] == 'test')
                        & (ab_money['money'] > 0)]
ARPPU_PC_test = PC_test_m_2.groupby('date').agg('mean')[['money']]
ARPPU_PC_test.rename(columns={'money': 'ARPPU_t_PC'}, inplace=True)
ARPPU_PC_test
```

Рис.38. Расчет ARPPU для группы test платформы PC

```
PC_control_m_2 = ab_money[(ab_money['platform'] == 'PC')
                           & (ab_money['group'] == 'control') & (ab_money['money'] > 0)]
ARPPU_PC_control = PC_control_m_2.groupby('date').agg('mean')[['money']]
ARPPU_PC_control.rename(columns={'money': 'ARPPU_c_PC'}, inplace=True)
ARPPU_PC_control
```

Рис.39. Расчет ARPPU для группы control платформы PC

Соединим в один df метрики ARPPU для двух групп площадки PC и рассчитаем эффект в % (рис.40):

```
ARPPU_PC = pd.merge(ARPPU_PC_test, ARPPU_PC_control, left_on='date', right_on='date')
ARPPU_PC['perc_ARPPU'] = ARPPU_PC['ARPPU_t_PC'] / ARPPU_PC['ARPPU_c_PC'] * 100 - 100
ARPPU_PC = ARPPU_PC.round({'ARPPU_t_PC': 3, 'ARPPU_c_PC': 3, 'perc_ARPPU': 1})
ARPPU_PC
```

	ARPPU_t_PC	ARPPU_c_PC	perc_ARPPU
date			
10.07.2021	1.229	1.208	1.7
11.07.2021	1.229	1.207	1.8
12.07.2021	1.228	1.208	1.6
13.07.2021	1.224	1.202	1.8
14.07.2021	1.211	1.188	1.9
15.07.2021	1.193	1.169	2.1
16.07.2021	1.163	1.138	2.2
17.07.2021	1.953	1.737	12.4

Рис.40. df метрики ARPPU PC и их сравнение

Рассмотрим траты внутриигровой валюты для двух групп на платформе PC (рис. 41-42).

```
PC_test_c = ab_cash[(ab_cash['platform'] == 'PC') & (ab_money['group'] == 'test')]
cash_PC_test = PC_test_c.groupby('date').agg('mean')[['cash']]
cash_PC_test.rename(columns={'cash': 'cash_t'}, inplace=True)
cash_PC_test
```

Рис.41. Расчет cash для группы test платформы PC

```
PC_control_c = ab_cash[(ab_cash['platform'] == 'PC') & (ab_money['group'] == 'control')]
cash_PC_control = PC_control_c.groupby('date').agg('mean')[['cash']]
cash_PC_control.rename(columns={'cash': 'cash_c'}, inplace=True)
cash_PC_control
```

Рис.42. Расчет cash для группы control платформы PC

Соединим в один df средние по тратам внутренней валюты для двух групп площадки PC и рассчитаем эффект в % (рис.43).

```
cash_PC = pd.merge(cash_PC_test, cash_PC_control, left_on='date', right_on='date')
cash_PC['perc_cash'] = cash_PC_test['cash_t'] / cash_PC_control['cash_c'] * 100 - 100
cash_PC = cash_PC.round({'cash_t': 3, 'cash_c': 3, 'perc_cash': 1})
cash_PC
```

	cash_t	cash_c	perc_cash
date			
10.07.2021	848.750	749.533	13.2
11.07.2021	848.167	748.794	13.3
12.07.2021	843.568	745.380	13.2
13.07.2021	822.536	726.588	13.2
14.07.2021	768.224	679.558	13.0
15.07.2021	677.087	593.841	14.0
16.07.2021	546.502	478.081	14.3
17.07.2021	1026.585	865.326	18.6

Рис.43. df метрики cash PC и их сравнение

Соединим в один df полученные метрики и их сравнение по игровой платформе PC (рис.44).

```
PC_metrics = pd.merge(ARPU_PC, ARPPU_PC, left_on='date', right_on='date')
PC_metrics = pd.merge(PC_metrics, cash_PC, left_on='date', right_on='date')
PC_metrics
```

	ARP_t_PC	ARP_c_PC	perc_ARP	ARPPU_t_PC	ARPPU_c_PC	perc_ARPPU	cash_t	cash_c	perc_cash
date									
10.07.2021	0.783	0.717	9.3	1.229	1.208	1.7	848.750	749.533	13.2
11.07.2021	0.782	0.713	9.7	1.229	1.207	1.8	848.167	748.794	13.3
12.07.2021	0.779	0.712	9.4	1.228	1.208	1.6	843.568	745.380	13.2
13.07.2021	0.762	0.697	9.3	1.224	1.202	1.8	822.536	726.588	13.2
14.07.2021	0.724	0.660	9.7	1.211	1.188	1.9	768.224	679.558	13.0
15.07.2021	0.670	0.605	10.7	1.193	1.169	2.1	677.087	593.841	14.0
16.07.2021	0.592	0.539	9.7	1.163	1.138	2.2	546.502	478.081	14.3
17.07.2021	1.176	1.003	17.3	1.953	1.737	12.4	1026.585	865.326	18.6

Рис.44. Метрики и их сравнение по игровой платформе PC

5.1.2. Расчет и сравнение метрик для платформы PS4

Расчет метрики ARPU(средняя прибыль на игрока) для двух групп на платформе PS4 (рис. 45-46).

```
PS4_test_m = ab_money[(ab_money['platform'] == 'PS4') & (ab_money['group'] == 'test')]
ARPU_PS4_test = PS4_test_m.groupby('date').agg('mean')[['money']]
ARPU_PS4_test.rename(columns={'money': 'ARP_t_PS4'}, inplace=True)
ARPU_PS4_test
```

Рис.45. Расчет ARPU для группы test платформы PS4

```
PS4_control_m = ab_money[(ab_money['platform'] == 'PS4') & (ab_money['group'] == 'control')]
ARPU_PS4_control = PS4_control_m.groupby('date').agg('mean')[['money']]
ARPU_PS4_control.rename(columns={'money': 'ARP_c_PS4'}, inplace=True)
ARPU_PS4_control
```

Рис.46. Расчет ARPU для группы control платформы PS4

Соединим в один df метрики ARPU для двух групп площадки PS4 и рассчитаем эффект в % (рис.47):

```
ARPU_PS4 = pd.merge(ARPU_PS4_test, ARPU_PS4_control, left_on='date', right_on='date')
ARPU_PS4['perc_ARP'] = ARPU_PS4['ARP_t_PS4'] / ARPU_PS4['ARP_c_PS4'] * 100 - 100
ARPU_PS4 = ARPU_PS4.round({'ARP_t_PS4': 3, 'ARP_c_PS4': 3, 'perc_ARP': 1})
ARPU_PS4
```

	ARP_t_PS4	ARP_c_PS4	perc_ARP
date			
10.07.2021	0.750	0.717	4.6
11.07.2021	0.753	0.714	5.4
12.07.2021	0.747	0.710	5.2
13.07.2021	0.735	0.693	6.0
14.07.2021	0.703	0.663	6.1
15.07.2021	0.648	0.611	6.1
16.07.2021	0.578	0.546	5.8
17.07.2021	1.172	1.085	8.0

Рис.47. df метрики ARPU PS4 и их сравнение

Рассчитаем ARPPU(средняя прибыль на платящего игрока) для двух групп на платформе PS4 (рис. 48-49).

```
PS4_test_m_2 = ab_money[(ab_money['platform'] == 'PS4')
                        & (ab_money['group'] == 'test') & (ab_money['money'] > 0)]
ARPPU_PS4_test = PS4_test_m_2.groupby('date').agg('mean')[['money']]
ARPPU_PS4_test.rename(columns={'money': 'ARPP_t_PS4'}, inplace=True)
ARPPU_PS4_test
```

Рис.48. Расчет ARPPU для группы test платформы PS4

```
PS4_control_m_2 = ab_money[(ab_money['platform'] == 'PS4')
                           & (ab_money['group'] == 'control') & (ab_money['money'] > 0)]
ARPPU_PS4_control = PS4_control_m_2.groupby('date').agg('mean')[['money']]
ARPPU_PS4_control.rename(columns={'money': 'ARPP_c_PS4'}, inplace=True)
ARPPU_PS4_control
```

Рис.49. Расчет ARPPU для группы control платформы PS4

Соединим в один df метрики ARPPU для двух групп площадки PS4 и рассчитаем эффект в % (рис.50):

```
ARPPU_PS4 = pd.merge(ARPPU_PS4_test, ARPPU_PS4_control, left_on='date', right_on='date')
ARPPU_PS4['perc_ARPP'] = ARPPU_PS4['ARPP_t_PS4'] / ARPPU_PS4['ARPP_c_PS4'] * 100 - 100
ARPPU_PS4 = ARPPU_PS4.round({'ARPP_t_PS4': 3, 'ARPP_c_PS4': 3, 'perc_ARPP': 1})
ARPPU_PS4
```

	ARPP_t_PS4	ARPP_c_PS4	perc_ARPP
date			
10.07.2021	1.218	1.208	0.8
11.07.2021	1.222	1.207	1.2
12.07.2021	1.216	1.207	0.7
13.07.2021	1.216	1.201	1.3
14.07.2021	1.204	1.190	1.2
15.07.2021	1.183	1.170	1.1
16.07.2021	1.156	1.142	1.3
17.07.2021	1.921	1.824	5.3

Рис.50. df метрики ARPPU PS4 и их сравнение

Рассмотрим траты внутриигровой валюты для двух групп на платформе PS4 (рис. 51-52).

```
PS4_test_c = ab_cash[(ab_cash['platform'] == 'PS4')
                     & (ab_money['group'] == 'test')]
cash_PS4_test = PS4_test_c.groupby('date').agg('mean')[['cash']]
cash_PS4_test.rename(columns={'cash': 'cash_t'}, inplace=True)
cash_PS4_test
```

Рис.51. Расчет cash для группы test платформы PS4

```
PS4_control_c = ab_cash[(ab_cash['platform'] == 'PS4')
                       & (ab_money['group'] == 'control')]
cash_PS4_control = PS4_control_c.groupby('date').agg('mean')[['cash']]
cash_PS4_control.rename(columns={'cash': 'cash_c'}, inplace=True)
cash_PS4_control
```

Рис.52. Расчет cash для группы control платформы PS4

Соединим в один df средние по тратам внутренней валюты для двух групп площадки PS4 и рассчитаем эффект в % (рис.53).

```
cash_PS4 = pd.merge(cash_PS4_test, cash_PS4_control, left_on='date', right_on='date')
cash_PS4['perc_cash'] = cash_PS4_test['cash_t'] / cash_PS4_control['cash_c'] * 100 - 100
cash_PS4 = cash_PS4.round({'cash_t': 3, 'cash_c': 3, 'perc_cash': 1})
cash_PS4
```

Рис.53. df метрики cash PS4 и их сравнение

Соединим в один df полученные метрики и их сравнение по игровой платформе PS4 (рис.54).

```
PS4_metrics = pd.merge(ARPU_PS4, ARPPU_PS4, left_on='date', right_on='date')
PS4_metrics = pd.merge(PS4_metrics, cash_PS4, left_on='date', right_on='date')
PS4_metrics
```

	ARP_t_PS4	ARP_c_PS4	perc_ARP	ARPP_t_PS4	ARPP_c_PS4	perc_ARPP	cash_t	cash_c	perc_cash
date									
10.07.2021	0.750	0.717	4.6	1.218	1.208	0.8	799.965	750.972	6.5
11.07.2021	0.753	0.714	5.4	1.222	1.207	1.2	799.857	750.073	6.6
12.07.2021	0.747	0.710	5.2	1.216	1.207	0.7	794.961	744.970	6.7
13.07.2021	0.735	0.693	6.0	1.216	1.201	1.3	780.440	724.262	7.8
14.07.2021	0.703	0.663	6.1	1.204	1.190	1.2	734.040	679.014	8.1
15.07.2021	0.648	0.611	6.1	1.183	1.170	1.1	653.338	598.475	9.2
16.07.2021	0.578	0.546	5.8	1.156	1.142	1.3	532.290	487.906	9.1
17.07.2021	1.172	1.085	8.0	1.921	1.824	5.3	1035.495	946.264	9.4

Рис.54. Метрики и их сравнение по игровой платформе PS4

5.1.3. Расчет и сравнение метрик для платформы Xbox

Расчет метрики ARPU(средняя прибыль на игрока) для двух групп на платформе Xbox (рис 55-56).

```
XBox_test_m = ab_money[(ab_money['platform'] == 'XBox')
                        & (ab_money['group'] == 'test')]
ARPU_XBox_test = XBox_test_m.groupby('date').agg('mean')[['money']]
ARPU_XBox_test.rename(columns={'money': 'AR_t_XBo'}, inplace=True)
ARPU_XBox_test
```

Рис.55. Расчет ARPU для группы test платформы Xbox

```
XBox_control_m = ab_money[(ab_money['platform'] == 'XBox')
                           & (ab_money['group'] == 'control')]
ARPU_XBox_control = XBox_control_m.groupby('date').agg('mean')[['money']]
ARPU_XBox_control.rename(columns={'money': 'AR_c_XBo'}, inplace=True)
ARPU_XBox_control
```

Рис.56. Расчет ARPU для группы control платформы Xbox

Соединим в один df метрики ARPU для двух групп площадки Xbox и рассчитаем эффект в % (рис.57):


```

ARPU_XBox = pd.merge(ARPU_XBox_test, ARPU_XBox_control, left_on='date', right_on='date')
ARPU_XBox['perc_XBo'] = ARPU_XBox['AR_t_XBo'] / ARPU_XBox['AR_c_XBo'] * 100 - 100
ARPU_XBox = ARPU_XBox.round({'AR_t_XBo': 3, 'AR_c_XBo': 3, 'perc_XBo': 1})
ARPU_XBox

```

	AR_t_XBo	AR_c_XBo	perc_XBo
date			
10.07.2021	0.750	0.755	-0.7
11.07.2021	0.751	0.753	-0.3
12.07.2021	0.749	0.748	0.1
13.07.2021	0.734	0.736	-0.4
14.07.2021	0.704	0.702	0.2
15.07.2021	0.649	0.648	0.1
16.07.2021	0.581	0.576	0.8
17.07.2021	1.216	1.184	2.8

Рис.57. df метрики ARPU XBox и их сравнение

Рассчитаем ARPPU(средняя прибыль на платящего игрока) для двух групп на платформе Xbox (рис.58-59).

```

XBox_test_m_2 = ab_money[(ab_money['platform'] == 'XBox')
                        & (ab_money['group'] == 'test') & (ab_money['money'] > 0)]
ARPPU_XBox_test = XBox_test_m_2.groupby('date').agg('mean')[['money']]
ARPPU_XBox_test.rename(columns={'money': 'ARPP_t_XBo'}, inplace=True)
ARPPU_XBox_test

```

Рис.58. Расчет ARPPU для группы test платформы Xbox

```

XBox_control_m_2 = ab_money[(ab_money['platform'] == 'XBox')
                           & (ab_money['group'] == 'control') & (ab_money['money'] > 0)]
ARPPU_XBox_control = XBox_control_m_2.groupby('date').agg('mean')[['money']]
ARPPU_XBox_control.rename(columns={'money': 'ARPP_c_XBo'}, inplace=True)
ARPPU_XBox_control

```

Рис.59. Расчет ARPPU для группы control платформы Xbox

Соединим в один df метрики ARPPU для двух групп площадки Xbox и рассчитаем эффект в % (рис.60):


```
ARPPU_XBox = pd.merge(ARPPU_XBox_test, ARPPU_XBox_control, left_on='date', right_on='date')
ARPPU_XBox['perc_ARPP'] = ARPPU_XBox['ARPP_t_XBo'] / ARPPU_XBox['ARPP_c_XBo'] * 100 - 100
ARPPU_XBox = ARPPU_XBox.round({'ARPP_t_XBo': 3, 'ARPP_c_XBo': 3, 'perc_ARPP': 1})
ARPPU_XBox
```

	ARPP_t_XBo	ARPP_c_XBo	perc_ARPP
date			
10.07.2021	1.219	1.221	-0.1
11.07.2021	1.218	1.219	-0.1
12.07.2021	1.220	1.219	0.1
13.07.2021	1.215	1.215	-0.0
14.07.2021	1.205	1.204	0.1
15.07.2021	1.186	1.187	-0.0
16.07.2021	1.157	1.154	0.2
17.07.2021	1.964	1.935	1.5

Рис.60. df метрики ARPPU Xbox и их сравнение

Рассмотрим траты внутриигровой валюты для двух групп на платформе Xbox (рис.61-62).

```
XBox_test_c = ab_cash[(ab_cash['platform'] == 'Xbox')
                      & (ab_money['group'] == 'test')]
cash_XBox_test = XBox_test_c.groupby('date').agg('mean')[['cash']]
cash_XBox_test.rename(columns={'cash': 'cash_t'}, inplace=True)
cash_XBox_test
```

Рис.61. Расчет cash для группы test платформы Xbox

```
XBox_control_c = ab_cash[(ab_cash['platform'] == 'Xbox')
                        & (ab_money['group'] == 'control')]
cash_XBox_control = XBox_control_c.groupby('date').agg('mean')[['cash']]
cash_XBox_control.rename(columns={'cash': 'cash_c'}, inplace=True)
cash_XBox_control
```

Рис.62. Расчет cash для группы control платформы Xbox

Соединим в один df средние по тратам внутренней валюты для двух групп площадки Xbox и рассчитаем эффект в % (рис.63).

```
cash_XBox = pd.merge(cash_XBox_test, cash_XBox_control, left_on='date', right_on='date')
cash_XBox['perc_cash'] = cash_XBox_test['cash_t'] / cash_XBox_control['cash_c'] * 100 - 100
cash_XBox = cash_XBox.round({'cash_t': 3, 'cash_c': 3, 'perc_cash': 1})
cash_XBox
```

	cash_t	cash_c	perc_cash
date			
10.07.2021	798.724	801.320	-0.3
11.07.2021	800.721	800.393	0.0
12.07.2021	795.484	795.144	0.0
13.07.2021	777.967	778.912	-0.1
14.07.2021	737.205	732.771	0.6
15.07.2021	651.932	648.521	0.5
16.07.2021	537.521	530.738	1.3
17.07.2021	1076.715	1043.961	3.1

Рис.63. df метрики cash Xbox и их сравнение

Соединим в один df полученные метрики и их сравнение по игровой платформе Xbox (рис.64).

```
XBox_metrics = pd.merge(ARPU_XBox, ARPPU_XBox, left_on='date', right_on='date')
XBox_metrics = pd.merge(XBox_metrics, cash_XBox, left_on='date', right_on='date')
XBox_metrics
```

	AR_t_XBo	AR_c_XBo	perc_XBo	ARPP_t_XBo	ARPP_c_XBo	perc_ARPP	cash_t	cash_c	perc_cash
date									
10.07.2021	0.750	0.755	-0.7	1.219	1.221	-0.1	798.724	801.320	-0.3
11.07.2021	0.751	0.753	-0.3	1.218	1.219	-0.1	800.721	800.393	0.0
12.07.2021	0.749	0.748	0.1	1.220	1.219	0.1	795.484	795.144	0.0
13.07.2021	0.734	0.736	-0.4	1.215	1.215	-0.0	777.967	778.912	-0.1
14.07.2021	0.704	0.702	0.2	1.205	1.204	0.1	737.205	732.771	0.6
15.07.2021	0.649	0.648	0.1	1.186	1.187	-0.0	651.932	648.521	0.5
16.07.2021	0.581	0.576	0.8	1.157	1.154	0.2	537.521	530.738	1.3
17.07.2021	1.216	1.184	2.8	1.964	1.935	1.5	1076.715	1043.961	3.1

Рис.64. Метрики и их сравнение по игровой платформе Xbox

5.2. Графики сравнения метрик по дням (Power BI)

Для построения графиков сравнения метрик по дням в Power BI сохраним в excel-файлы датафреймы с метриками по каждой из игровых платформ (рис.65). Далее полученные excel-файлы загружены в Power BI и используются в качестве источников для построения графиков.

Графики экспортировать из Power BI в word не представляется возможным, такой опции нет⁸. Поэтому экспортированы графики в pdf, затем pdf -> word через prt sc.

⁸ [Ссылка на веб-страницу](https://community.powerbi.com) (Сайт <https://community.powerbi.com>)

```
PC_metrics.to_excel('PC.xlsx')
PS4_metrics.to_excel('PS4.xlsx')
XBox_metrics.to_excel('XBox.xlsx')
```

Рис.65. Сохранение датафреймов с метриками в excel-файлы с помощью метода .to_excel()

5.2.1. Графики сравнения метрик по дням для игровой платформы PC

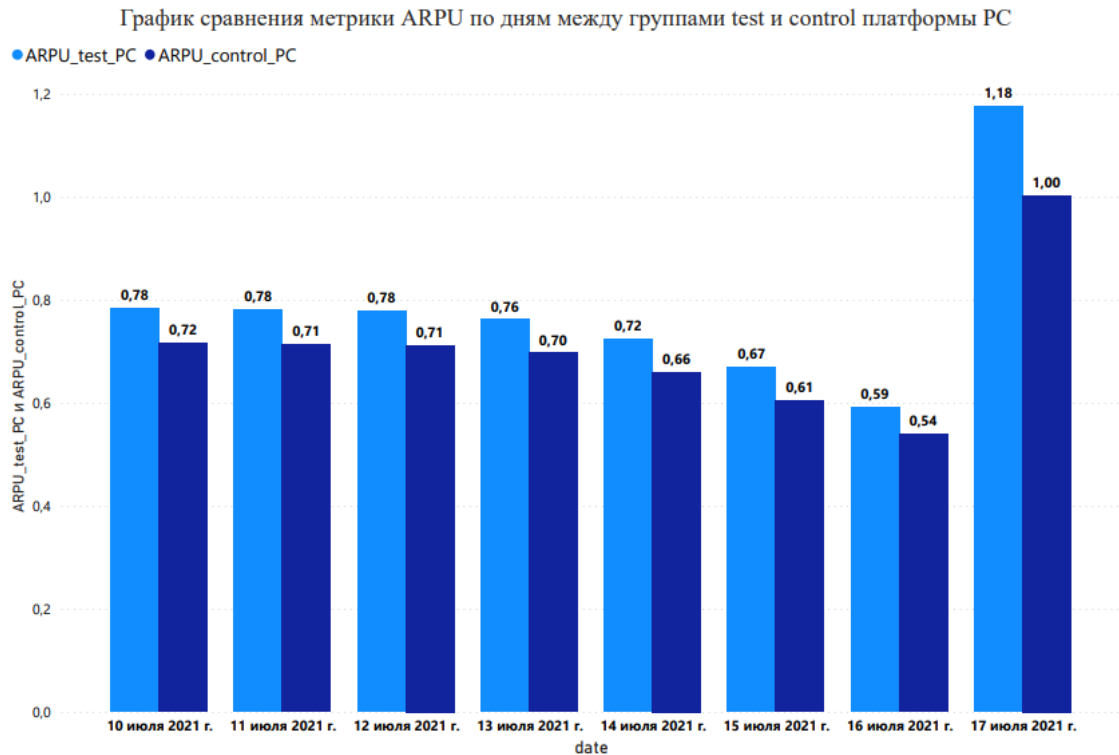


Рис.66. График сравнения ARPU по дням для платформы PC

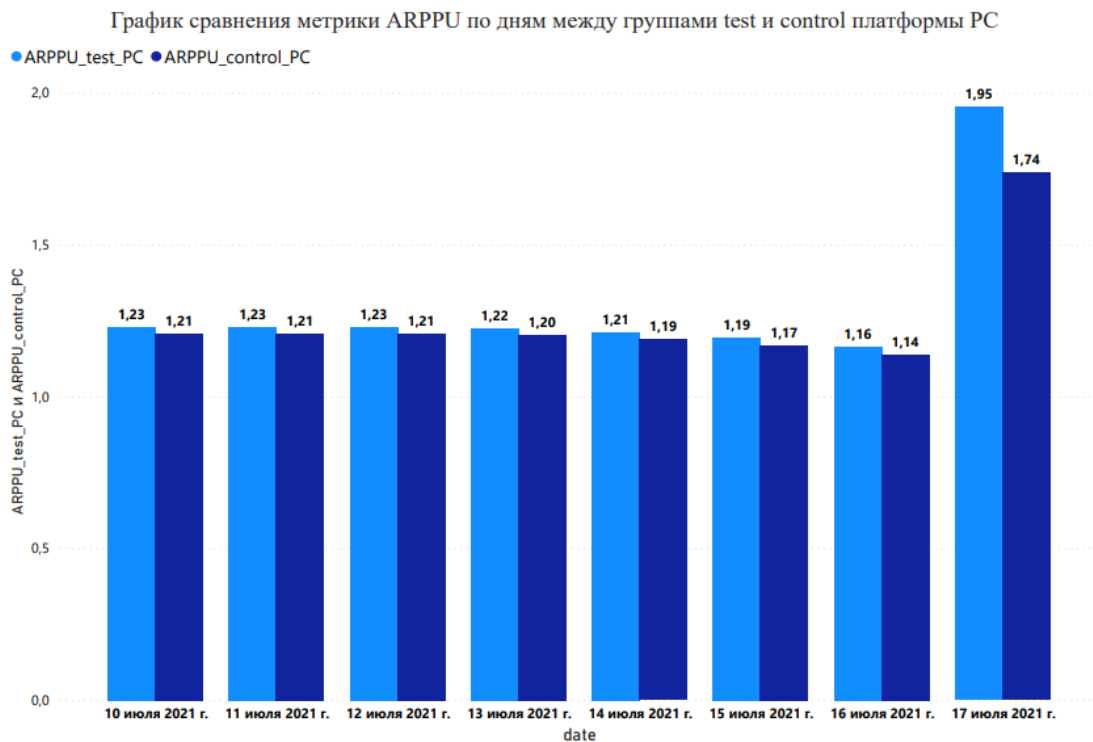


Рис.67. График сравнения ARPPU по дням для платформы PC

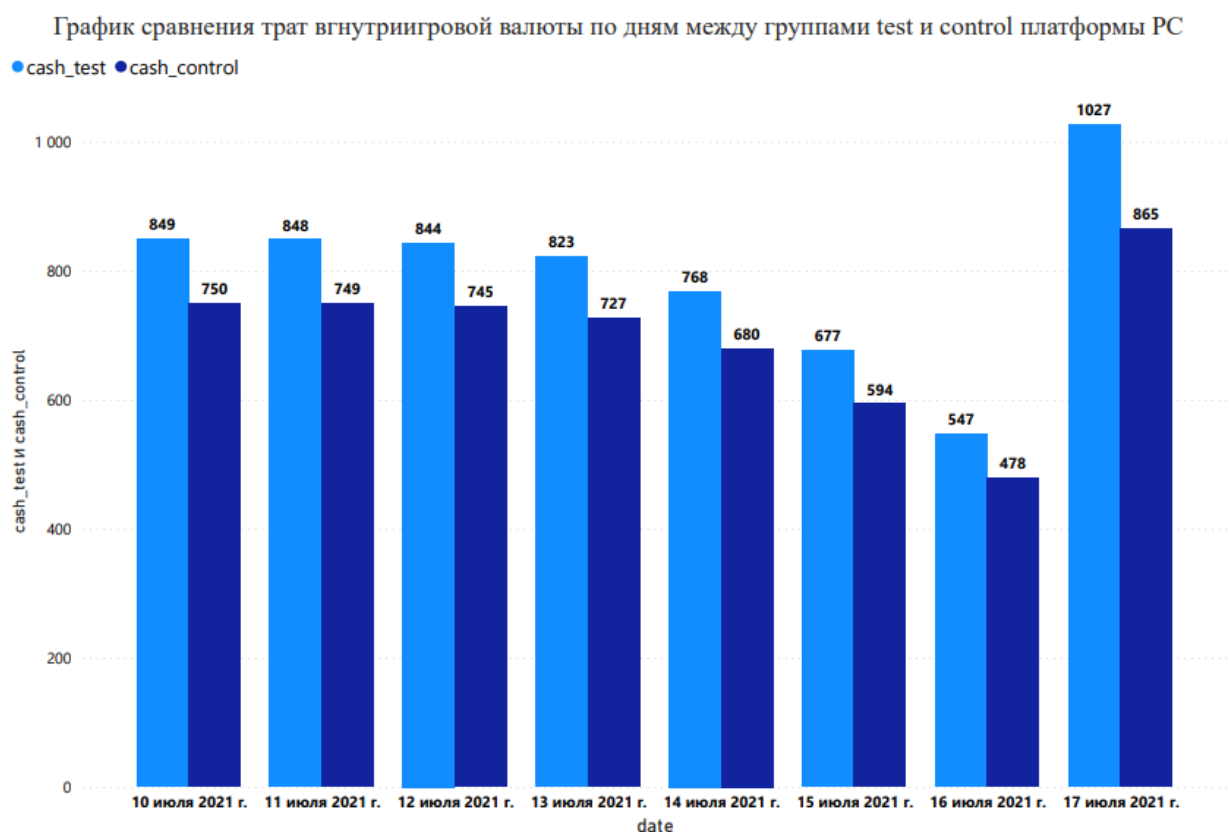


Рис.68. График сравнения трат внутриигровой валюты по дням для платформы PC

5.2.2. Графики сравнения метрик по дням для игровой платформы PS4

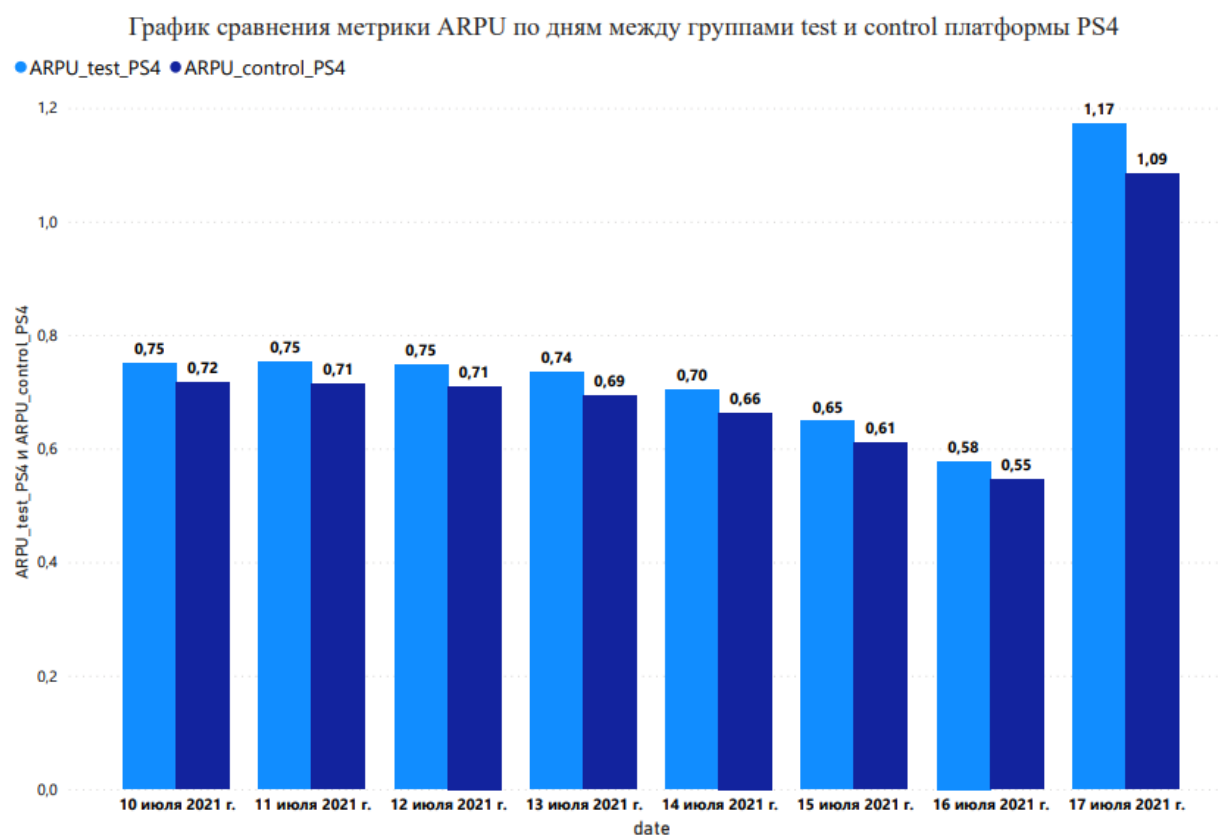


Рис.69. График сравнения ARPU по дням для платформы PS4

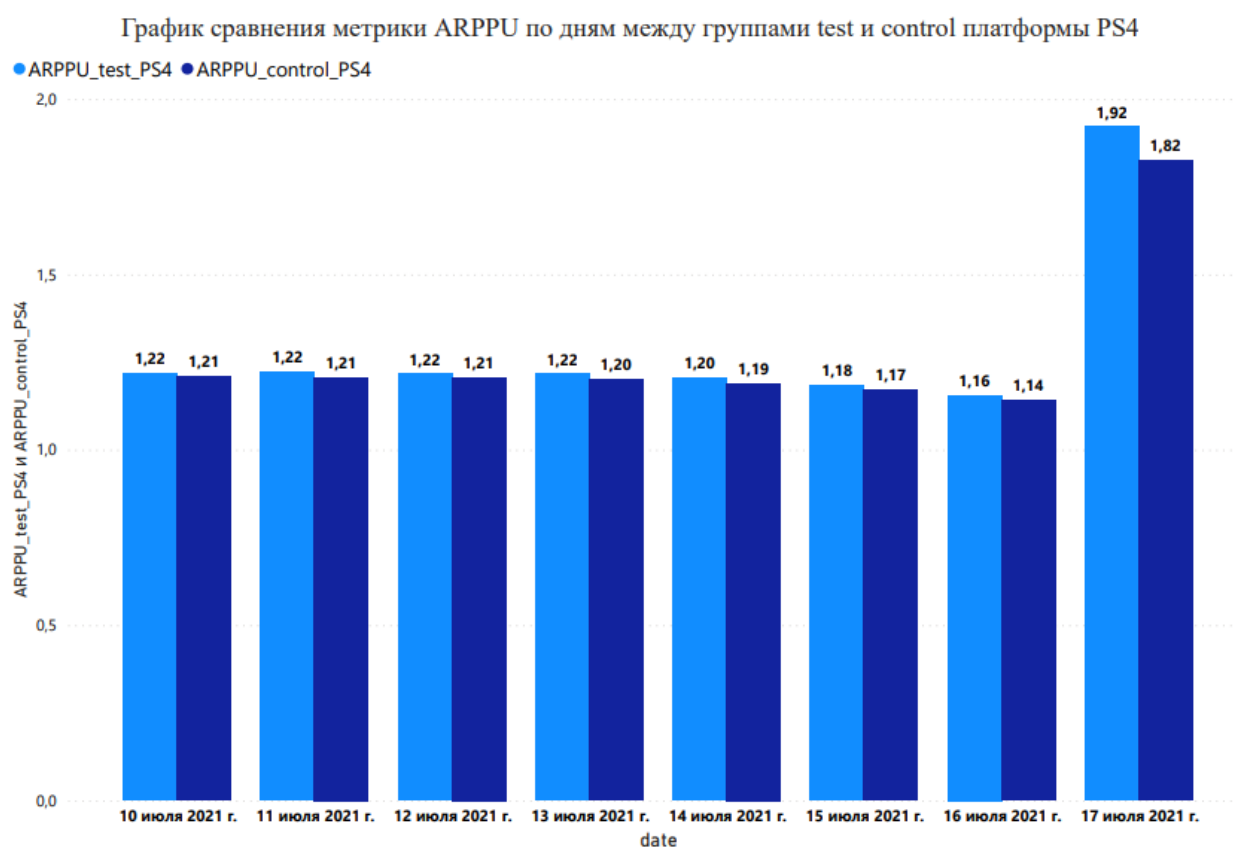


Рис.70. График сравнения ARPPU по дням для платформы PS4

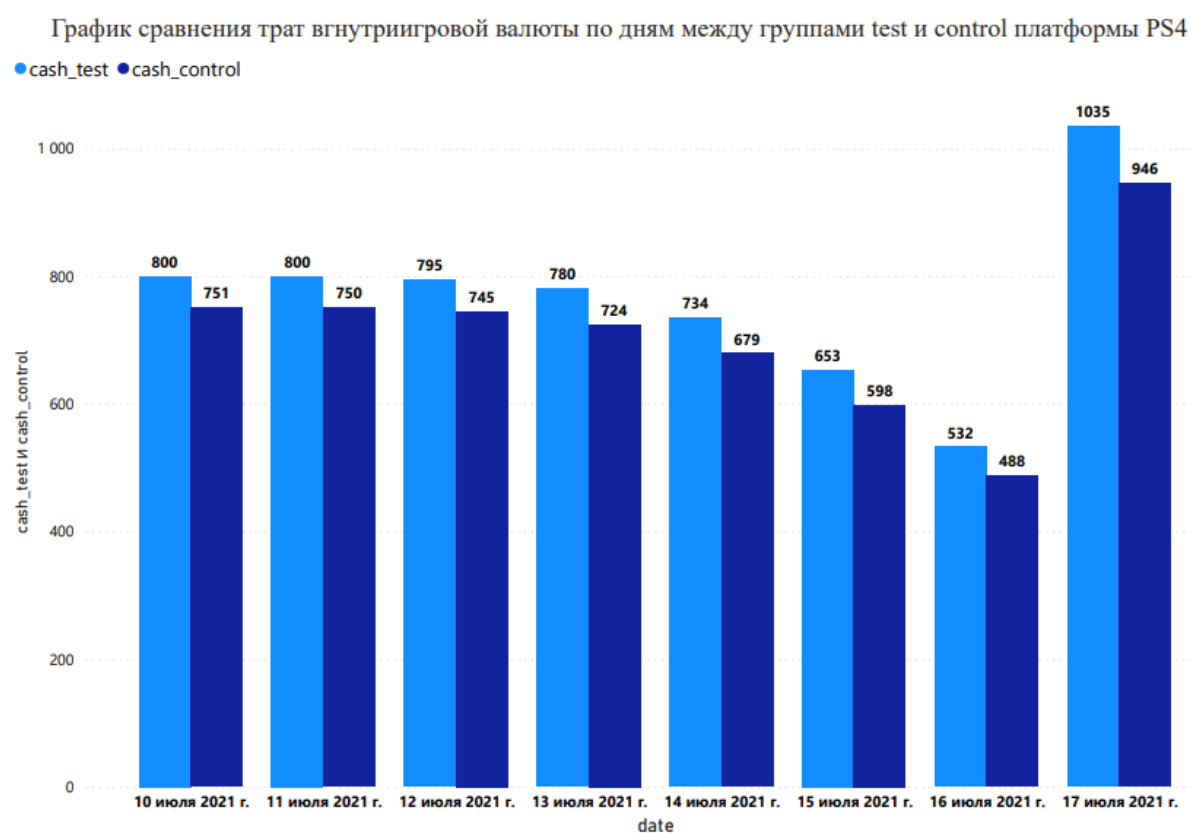


Рис.71. График сравнения трат внутриигровой валюты по дням для платформы PS4

5.2.3. Графики сравнения метрик по дням для игровой платформы XBox

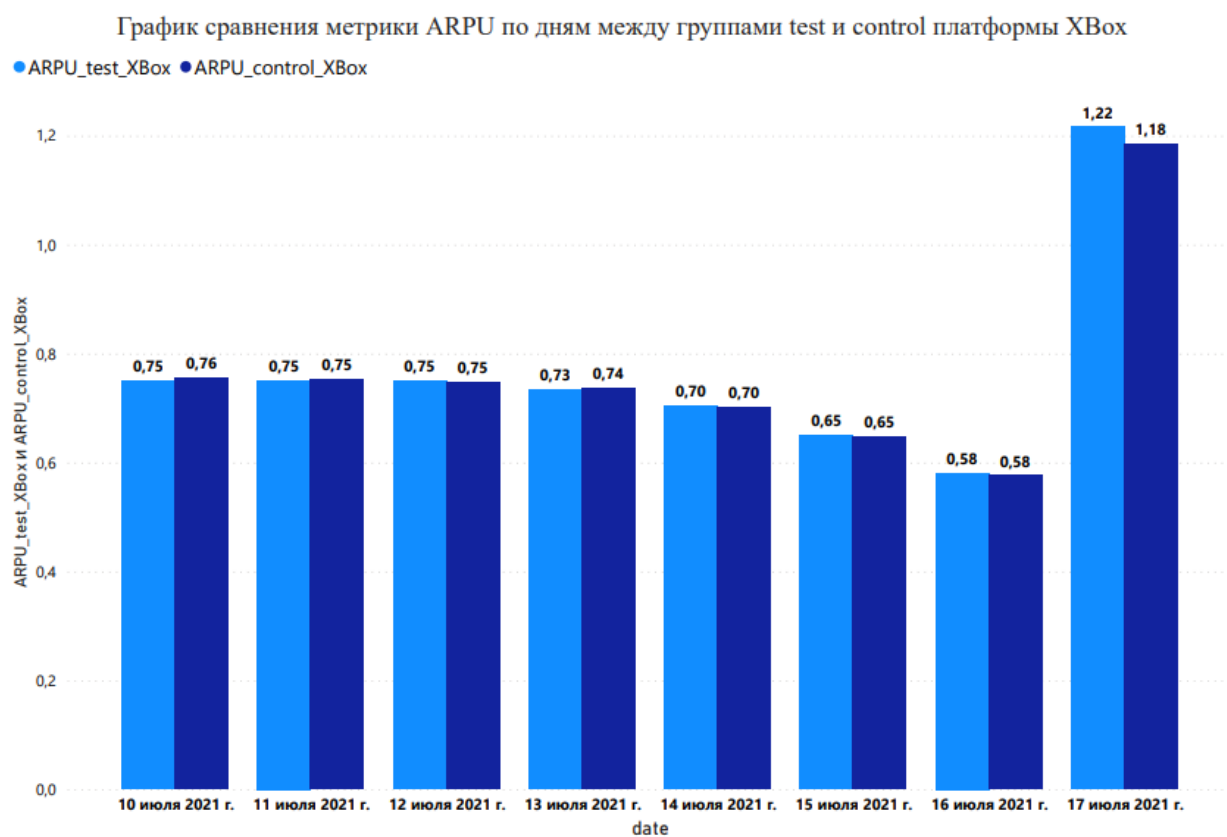


Рис.72. График сравнения ARPU по дням для платформы XBox

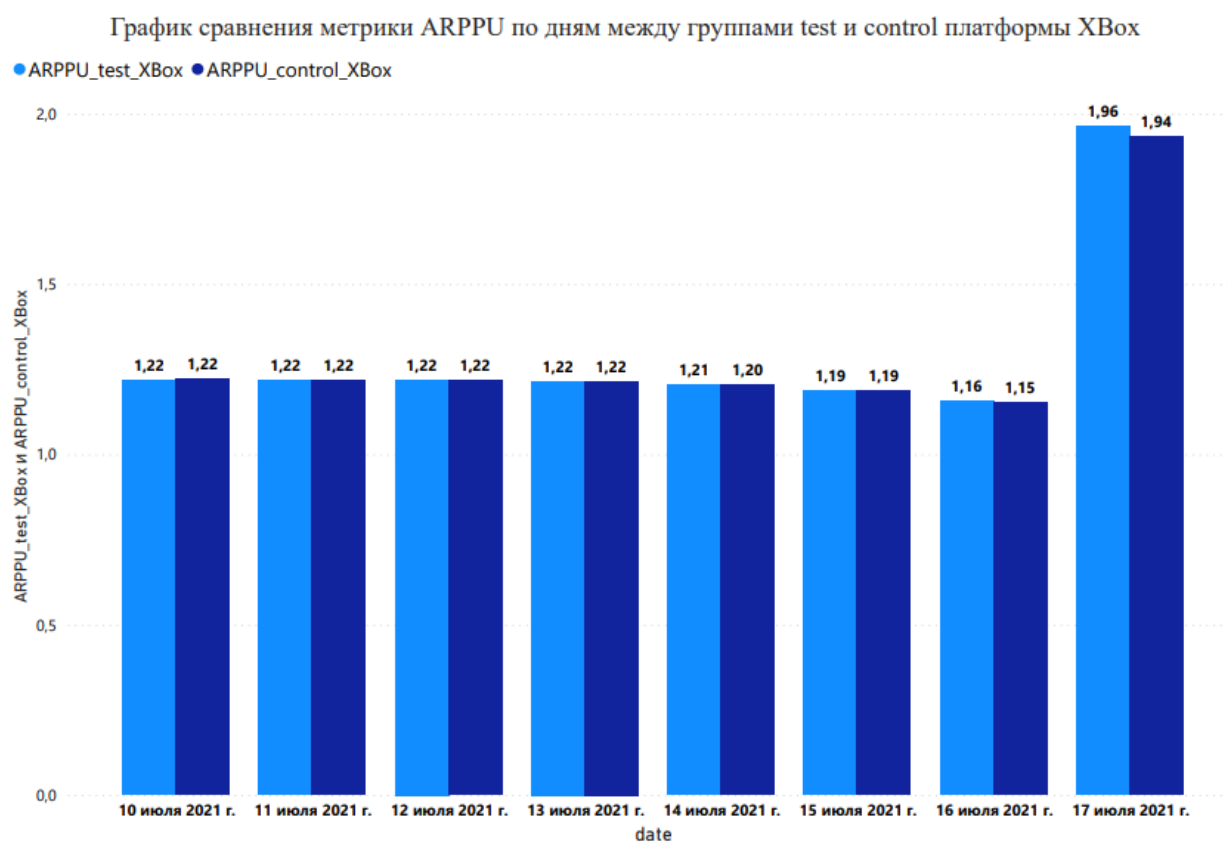


Рис.73. График сравнения ARPPU по дням для платформы XBox

График сравнения трат внутриигровой валюты по дням между группами test и control платформы XBox

● cash_test ● cash_control

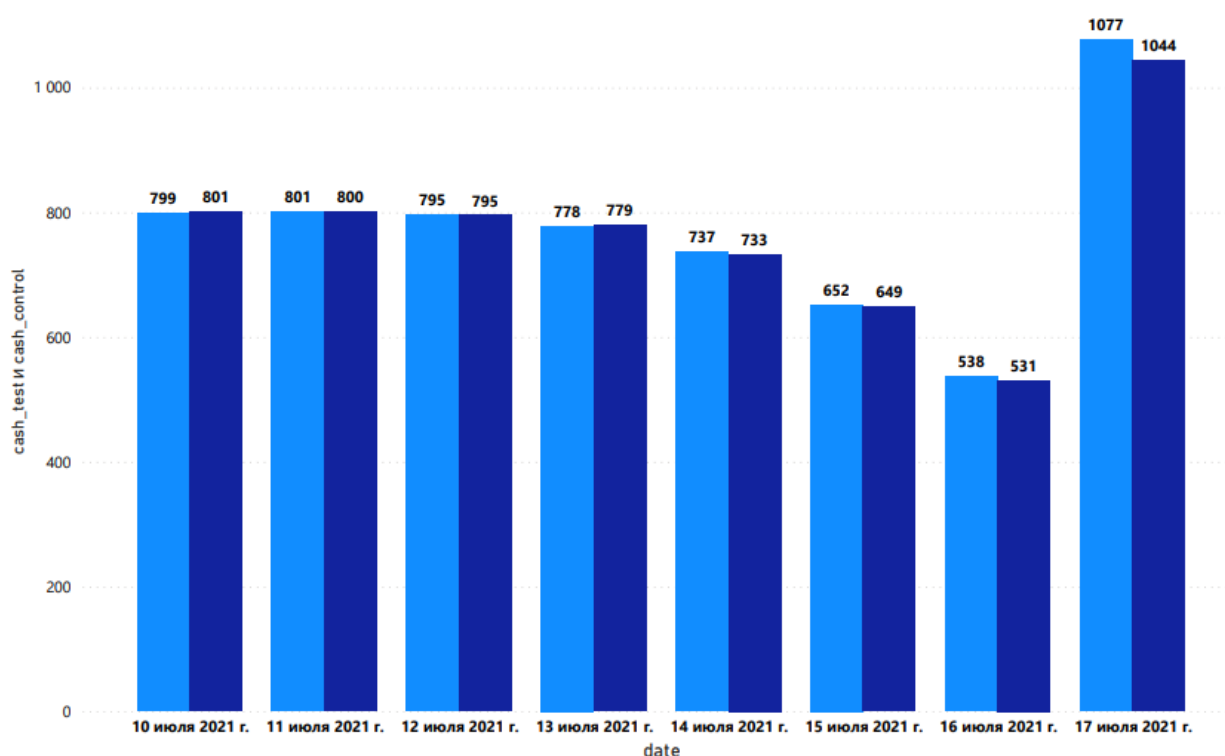


Рис.74. График сравнения трат внутриигровой валюты по дням для платформы XBox

5.3. Исходники программы Python для построения всех графиков и таблиц

В этот подраздел добавлены исходники из Jupyter Notebook, которые остались не добавленными в отчет. Остальные исходники приведены в работе по тематике разделов/подразделов.

```
len(df_cheaters[df_cheaters['cheaters'] == 1])/8
```

353.0

Количество выявленных читеров (353).

```
len(df_abgroup[df_abgroup['group'] == 'test'])/8
```

539653.0

Количество в группе тестируемых (539 653 чел).

```
len(df_abgroup[df_abgroup['group'] == 'control'])/8
```

540347.0

Количество в контрольной группе, без изм (540 347 чел).

Рис.75. Определение количества читеров, выявленных до теста

```
df_money.sort_values(['user_id', 'date'])
```

	user_id	date	money
612508	00036L-1T885I	10.07.2021	0.00
1089789	00036L-1T885I	11.07.2021	0.00
1663770	00036L-1T885I	12.07.2021	0.99
7136619	00036L-1T885I	13.07.2021	0.00
4001126	00036L-1T885I	14.07.2021	0.00
...
3707582	ZZZVH1-2OJT7I	13.07.2021	0.99
403488	ZZZVH1-2OJT7I	14.07.2021	0.99
1351622	ZZZVH1-2OJT7I	15.07.2021	0.00
2307820	ZZZVH1-2OJT7I	16.07.2021	0.00
114364	ZZZVH1-2OJT7I	17.07.2021	0.00

8640000 rows × 3 columns

Рис.76. Содержание таблицы Money

```
df_cheaters.sort_values('user_id').head(10)
```

	user_id	cheaters
1252809	00036L-1T885I	0
3722504	00036L-1T885I	0
8050836	00036L-1T885I	0
7144435	00036L-1T885I	0
7763404	00036L-1T885I	0
7418240	00036L-1T885I	0
5600738	00036L-1T885I	0
7360158	00036L-1T885I	0
8431545	0004MK-MRBUKF	0
1262083	0004MK-MRBUKF	0

Рис.77. Содержание таблицы Cheaters

```
df_platforms.sort_values('user_id')
```

	user_id	platform
1205248	00036L-1T885I	XBox
5053411	00036L-1T885I	XBox
4528844	00036L-1T885I	XBox
2096927	00036L-1T885I	XBox
880380	00036L-1T885I	XBox
...
2941753	ZZZVH1-2OJT7I	PC
3809963	ZZZVH1-2OJT7I	PC
1636571	ZZZVH1-2OJT7I	PC
4326129	ZZZVH1-2OJT7I	PC
1601136	ZZZVH1-2OJT7I	PC

8640000 rows × 2 columns

Рис.78. Содержание таблицы Platforms


```
df_cash.sort_values(['user_id', 'date'])
```

	user_id	date	cash
7288533	00036L-1T885I	10.07.2021	700
3638659	00036L-1T885I	11.07.2021	0
4102936	00036L-1T885I	12.07.2021	1500
4099102	00036L-1T885I	13.07.2021	850
6900680	00036L-1T885I	14.07.2021	650
...
4151151	ZZZVH1-2OJT7I	13.07.2021	650
4594405	ZZZVH1-2OJT7I	14.07.2021	1400
6263715	ZZZVH1-2OJT7I	15.07.2021	100
6999729	ZZZVH1-2OJT7I	16.07.2021	550
2322594	ZZZVH1-2OJT7I	17.07.2021	550

8640000 rows × 3 columns

Рис.79. Содержание таблицы Cash

```
df_abgroup.sort_values('user_id')
```

	user_id	group
516029	00036L-1T885I	control
4282472	00036L-1T885I	control
1530349	00036L-1T885I	control
4015064	00036L-1T885I	control
4841272	00036L-1T885I	control
...
522760	ZZZVH1-2OJT7I	control
777139	ZZZVH1-2OJT7I	control
8528190	ZZZVH1-2OJT7I	control
7827472	ZZZVH1-2OJT7I	control
2287724	ZZZVH1-2OJT7I	control

8640000 rows × 2 columns

Рис.80. Содержание таблицы ABgroup

5.4. Сводная таблица в Excel с ARPU по группам и платформам

Создадим сводную таблицу с метрикой ARPU (средняя прибыль на игрока) по группам и платформам, сохраним ее в excel (рис. 81-82).

```
metrics = pd.merge(ARPU_PC, ARPU_PS4, left_on='date', right_on='date')
metrics = pd.merge(metrics, ARPU_XBox, left_on='date', right_on='date')
metrics
```

	AR_t_PC	AR_c_PC	perc_PC	AR_t_PS4	AR_c_PS4	perc_PS4	AR_t_Xb	AR_c_Xb	perc_Xb
date									
10.07.2021	0.783	0.717	9.3	0.750	0.717	4.6	0.750	0.755	-0.7
11.07.2021	0.782	0.713	9.7	0.753	0.714	5.4	0.751	0.753	-0.3
12.07.2021	0.779	0.712	9.4	0.747	0.710	5.2	0.749	0.748	0.1
13.07.2021	0.762	0.697	9.3	0.735	0.693	6.0	0.734	0.736	-0.4
14.07.2021	0.724	0.660	9.7	0.703	0.663	6.1	0.704	0.702	0.2
15.07.2021	0.670	0.605	10.7	0.648	0.611	6.1	0.649	0.648	0.1
16.07.2021	0.592	0.539	9.7	0.578	0.546	5.8	0.581	0.576	0.8
17.07.2021	1.176	1.003	17.3	1.172	1.085	8.0	1.216	1.184	2.8

```
metrics.to_excel('metrics.xlsx')
```

Рис.81. Соединение таблиц с метриками ARPU по трем платформам и выгрузка итоговой таблицы в excel-файл (метод .to_excel())

date	AR_t_PC	AR_c_PC	perc_PC	AR_t_PS4	AR_c_PS4	perc_PS4	AR_t_Xb	AR_c_Xb	perc_Xb
10.07.2021	0,783	0,717	9,3	0,75	0,717	4,6	0,75	0,755	-0,7
11.07.2021	0,782	0,713	9,7	0,753	0,714	5,4	0,751	0,753	-0,3
12.07.2021	0,779	0,712	9,4	0,747	0,71	5,2	0,749	0,748	0,1
13.07.2021	0,762	0,697	9,3	0,735	0,693	6	0,734	0,736	-0,4
14.07.2021	0,724	0,66	9,7	0,703	0,663	6,1	0,704	0,702	0,2
15.07.2021	0,67	0,605	10,7	0,648	0,611	6,1	0,649	0,648	0,1
16.07.2021	0,592	0,539	9,7	0,578	0,546	5,8	0,581	0,576	0,8
17.07.2021	1,176	1,003	17,3	1,172	1,085	8	1,216	1,184	2,8

Рис.82. Сводная таблица в Excel с ARPU по группам и платформам

Закключение

В работе проведен анализ результатов выполнения А/В-теста, а именно – проведение акции в компании по разработке игр. Объектом проведения акции был бесплатный командный онлайн-шутер. Тест проводился на трех игровых платформах: PC, PS4, Xbox.

Акция длилась 8 дней, с 10.07.2021 по 17.07.2021 включительно. Общее количество участников = 1 080 000 человек.

По заданию необходимо было выяснить, стоит ли проводить акцию в дальнейшем. Для этого требовалось сравнить результаты тестовой и контрольной групп по каждой из игровых платформ, а именно: рассчитать метрики ARPU (средняя прибыль на игрока), ARPPU (средняя прибыль на платящего игрока) и сопоставить траты внутриигровой валюты, а также построить доверительные интервалы от средних значений с точностью до 95%.

До проведения теста была выявлена часть читеров (353 игрока), по заданию требовалось выявить вторую часть и уже после, получив перечень добропорядочных игроков, проводить анализ результатов А/В-теста.

Первым была выполнена очистка данных = поиск второй части читеров. Поиск проводился с помощью построения доверительного интервала на графике распределения трат внутриигровой валюты (cash из исходной таблицы Cash) и отметки аномальных значений (читеров), которые размещены выше верхней границы доверительного интервала. В результате выявлено 362 читера. Итого, с учетом выявленных ранее 353 читеров, сумма всех читеров составила = 715 читеров.

Далее расчет всех метрик. Сравнение метрик дало первые результаты. Самым успешным было проведение акции на игровой платформе PC, по ней отмечено самое высокое превышение показателей метрик группы test (участники акции) относительно группы control. Метрики ARPU (по дням) группы test выше аналогичной метрики группы control ежедневно более чем на 9%.

Результаты сравнения метрик на платформе PS4 также можно назвать положительными. По всем метрикам на платформе PS4 наблюдается ежедневная тенденция превышения метрик группы test над аналогичными метриками группы control. Метрики ARPU (по дням) группы test выше аналогичной метрики группы control ежедневно более чем на 6%.

Результаты сравнения метрик на платформе XBox не утешительные. По всем метрикам нет стабильно положительного эффекта, часть дней группа test имеет превышение над группой control, но и есть дни, когда наблюдается обратная динамика (и так по всем метрикам). Метрики ARPU (по дням) группы test в 5 из 8 дней имеют превышение над аналогичной метрикой группы control и это превышение не достигает даже 1%.

Кроме этого был проведен анализ соотношения количества участников в группах по каждой игровой платформе для заключения о корректности сопоставления данных между группами. Значимых отличий не выявлено (не более 0,1% между группами каждой игровой платформы).

И, наконец, построение доверительных интервалов для каждой метрики по разным платформам. Полученные доверительные интервалы полностью подтвердили сделанные ранее выводы об успешности проведения акции для каждой из игровых платформ на основании сравнения рассчитанных метрик.

Полученные верхние границы доверительных интервалов группы test по всем метрикам игровой платформы PC значительно превышают верхние границы доверительных интервалов группы control, что подтверждает сделанные ранее выводы об успешности проведения акции для данной игровой платформы (при сравнении рассчитанных метрик).

Полученные доверительные интервалы группы test по всем метрикам игровой платформы PS4 также превышают доверительные интервалы группы control, но в меньшей степени, нежели рассмотренная выше платформа PC.

Полученные верхние границы доверительных интервалов группы test по всем метрикам игровой платформы XBox превышают, но не значительно, верхние границы доверительных интервалов группы control.

В результате я могу заключить, что повторное проведение акции уже на всех игроках принесет пользу только на двух игровых платформах: PC и PS4. При этом большую прибыль принесет акция на платформе PC. На игровой платформе Xbox проведенная акция не дала положительных результатов, и повторное проведение акции будет не целесообразным.

Список используемых источников

1. [Ссылка на веб-страницу](https://habr.com) (Сайт <https://habr.com>).
2. [Ссылка на веб-страницу](https://habr.com) (Сайт <https://habr.com>).
3. [Ссылка на веб-страницу](https://rukovodstvo.net) (Сайт <https://rukovodstvo.net>)
4. Уэс Маккини. Python и анализ данных. – 2020 - №2 – С. 274.
5. [Ссылка на веб-страницу](https://docs.scipy.org) (Сайт <https://docs.scipy.org>)
6. [Ссылка на веб-страницу](https://pythonpip.ru) (Сайт <https://pythonpip.ru>).
7. Кристиан Хилл. Научное программирование на Python. – 2021. - С. 510.
8. [Ссылка на веб-страницу](https://community.powerbi.com) (Сайт <https://community.powerbi.com>).