Постановка задач

Задачи

- Написать функцию, которая будет считать retention игроков (по дням от даты регистрации игрока).
- Определить какой набор предложений можно считать лучшим по имеющимся результатам A/B теста.
- Предложите метрики для оценки результатов последнего прошедшего тематического события в игре.

Предоставленные данные:

- *problem1-reg_data.csv* – данные о времени регистрации
- *problem1-auth_data.csv* – данные о времени захода пользователей в игру
- *problem2.csv* – результаты A/B теста

Выводы исследования

- Стоит подумать об акциях и/или новых механиках игры, позволяющих увеличить удержание пользователей после седьмого дня.
  (Максимальное удержание пользователей происходит на 4-7 дни использования продукта, после чего показатель retention начинает падать)

- Изменения из тестовой группы рекомендуется запустить в прод.
  (По результатам A/B теста статистически значимо удалось увеличить ARPU и ARPPU)

- Список метрик для оценки результатов прошедшего тематического события в игре:
  - **ASD** (Average Session Duration),
  - **MAU/DAU** (Daily Active Users / Monthly Active Users),
  - **Stickiness rate**,
  - **Users Online,**
  - **ATV** (Average Transaction Value).

Общие результаты исследования

Поиск аномалий данных

- Были обнаружены и убраны из датасета регистрации пользователей за период 1998-2013 годы (1.76% от общего числа регистраций) — не релевантны исследованию.
- Порядка 7% данных имели аномальную частоту заходов в игру — меньше 5 минут между сеансами (возможно боты или случайные заходы пользователей в игру, без самого процесса игры — были удалены из датасета).
- При анализе данных для A/B-тестирования были обнаружены 123 аномально высоких значения выручки с пользователя в контрольной группе — эти значения были удалены из данных, т.к. с наибольшей вероятностью они имеют ошибочную природу происхождения (технический баг, проблемы со сплитованием и т. д.).

Общие результаты исследования

Результаты A/B-тестов*

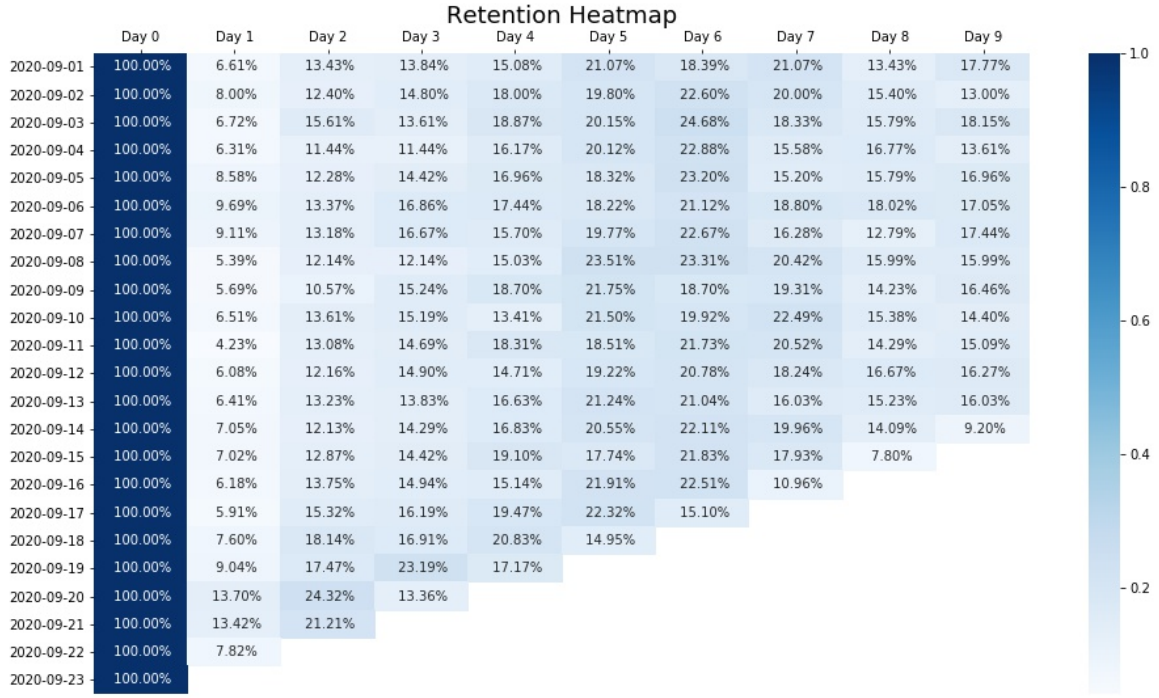
- Конверсия в покупку и равенство распределений значений выручки с пользователя в тестовой и контрольной группах не имеют статистически значимых различий.
- ARPU, ARPPU и распределения значений выручки с платящих пользователей в тестовой и контрольной группах статистически значимо различаются.

** Для проверки гипотез использовались proportions_ztest (statsmodels.stats), непараметрический статистический U-критерий Манна-Уитни и bootstrap средних значений.*

Общие результаты исследования

Пример работы функции расчёта Retention

```python
def retention(data, reg_pos=2, auth_pos=0, uid_pos=1, min_date=None, max_date=None):
    # делаем выборку нужных колонок
    df = data.iloc[:, [uid_pos, reg_pos, auth_pos]].copy(deep=True)

    # проверка наличия и определение параметра min_date
    if min_date is None:
        min_date = df.reg_ts.min()

    # проверка наличия и определение параметра max_date
    if max_date is None:
        max_date = df.reg_ts.max()

    # выносим даты регистрации и входа в приложение в отдельные колонки
    df['date_reg'] = df.reg_ts.dt.date
    df['date_auth'] = df.auth_ts.dt.date

    # делаем выборку по диапазону дат
    df = df[(df['reg_ts'] >= min_date) & (df['reg_ts'] <= max_date)]

    # посчитаем количество дней между датой регистрации и входом в приложение
    df['days'] = df['date_auth'] - df['date_reg']

    # сгруппируем по дням регистрации и количеству дней использования
    cohorts = df.groupby(['date_reg', 'days'], as_index=False) \
                .agg({'uid': 'nunique'})  # агрегируем по уникальным пользователям

    # сформируем сводную таблицу
    cohorts_pivot = cohorts.pivot(index='date_reg', columns='days', values='uid')

    # посчитаем retention
    retention = pd.DataFrame()  # пустой датафрейм к которому будем прибавлять построково
    for i in range(cohorts_pivot.shape[0]):  #  итерируемся по строкам
        a = cohorts_pivot.iloc[i, 0]  # 100% число пользователей для этой даты
        temp = cohorts_pivot.iloc[i, :].divide(a, axis=0).round(4)  # получаем пересчет на проценты
        retention = retention.append(temp)  # построково добавляем посчитанные значения в итоговую таблицу
            retention

    return retention
```



Общие результаты исследования

Список метрик

- Использование предложенного набора метрик (ASD, MAU/DAU, Stickiness rate, Users Online, ATV) оптимально, чтобы как можно точнее следить за тем, как происходящие изменения в игровых механиках отражаются на поведении пользователей и самом продукте.
- Рекомендуется дополнительно провести исследование как перечисленные метрики коррелируют (или даже предсказывают) траты пользователей и LTV. Это будет полезно для последующей приоритизации используемых метрик.

Дополнительно

Подробности исследования, технические детали и код в Jupyter Notebook