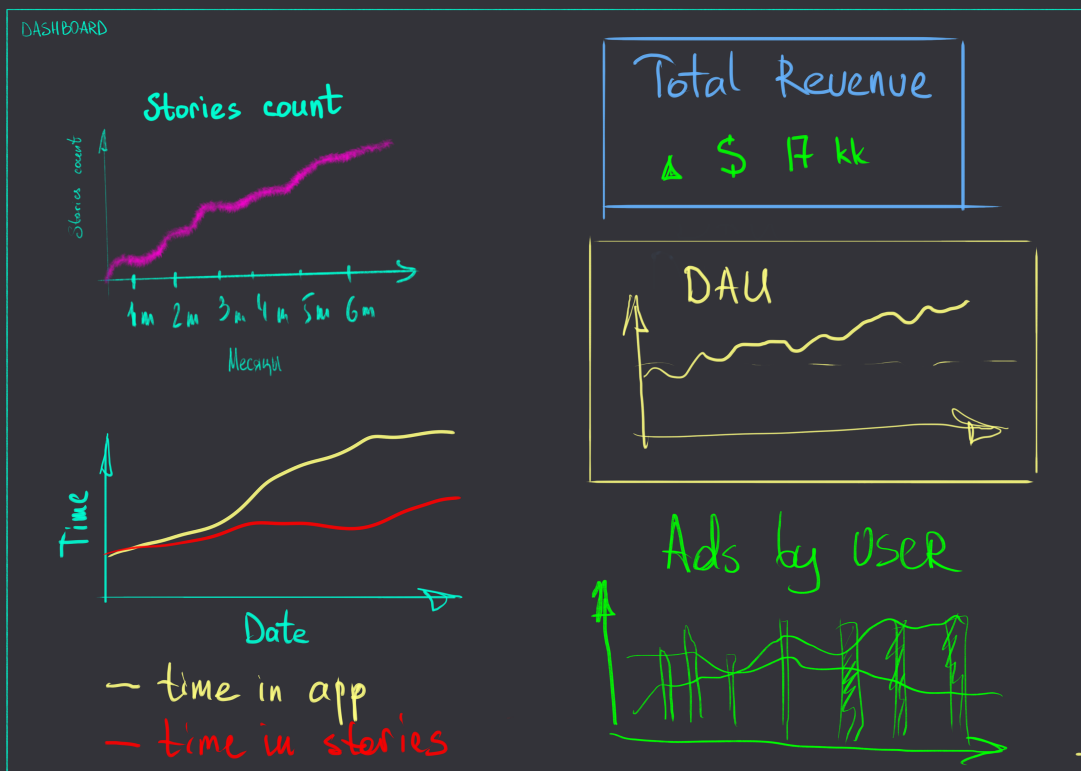
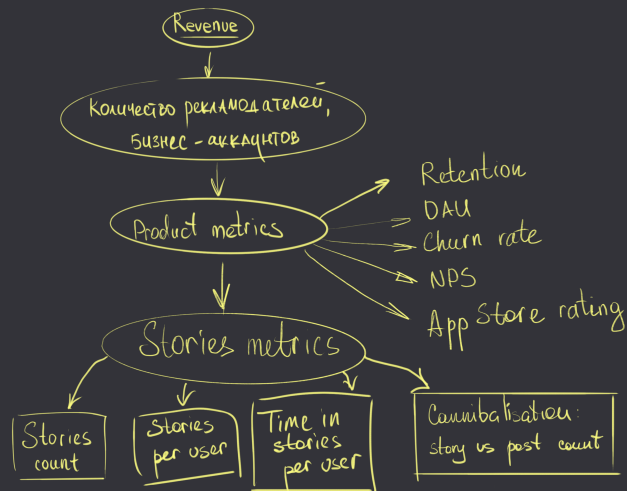


Типичная задача продуктового аналитика:

Представь, что ты продуктовый аналитик в [Instagram](#) и отвечаешь за аналитику продукта [Истории](#) (Stories). Спустя несколько месяцев после его запуска тебя попросили собрать продуктовый дашборд про успешность запуска, первые результаты работы этого продукта и его влияния на весь продукт Instagram.

Напиши, пожалуйста, метрики, которые бы ты отобразил на этом дашборде в порядке убывания важности (здорово если построишь пирамиду/иерархию метрик)



Типичный A/B тест продуктового аналитика:

На сайте запущен A/B тест с целью увеличить доход. В приложенном excel файле ты найдешь сырые данные по результатам эксперимента (user_id), тип выборки (variant_name) и доход, принесенный пользователем (revenue).

Проанализируй результаты эксперимента и напиши свои рекомендации менеджеру.

(приложи скрипт с помощью которого анализировал данные, желательно файл .ipynb на python)

Провел оценку результатов тестирования по критерию Манна-Уитни и методу Bootstrap. По всей выборке (включая нулевые значения) обе методики показали p-value равное 0.2391 и 0.21156 соответственно, что означает отсутствие стат. значимого различия.

Для полноты картины оценил результаты, только по клиентам с revenue $\neq 0$, что не совсем корректно в целом, но можно сделать предварительный вывод о влиянии теста на средний чек, например. На данных выборках критерий Манна-Уитни дает p-value = 0.03962, что < 0.05 , но тем не менее близко к границе значимости. Bootstrap так же показывает отсутствие значимых различий (p-value = 0.15814).

В итоге, я считаю, что эксперимент не показал стат. значимые различия между контрольной и тестовой группой.

Рекомендации:

1. Обратить внимание на условия эксперимента и увеличить время его проведения, если это возможно
2. Для контроля результатов использовать более явные и зависимые от эксперимента метрики.

Обычная SQL задача 1

Departments

id	name
1	Finance
2	Operations
3	Deployment

Employees

id	name	Dep_id	Manager_id	Salary
1	John Smith	1		2000
2	Jack Smith		1	1500
3	Becky Smith	1	2	2000
4	Rebecca Smith	2	2	700
5	Sonny Smith	3	1	3000

Используя таблицы выше, напиши следующие SQL запросы:

- а) Для каждого сотрудника найти его департамент, включая тех, у кого департамента нет

```
1 SELECT
2   id          AS employee_id,
3   name        AS employee_name,
4   Department_name
5
6 FROM at7sharp_employees AS l
7
8 LEFT JOIN
9
10 (SELECT
11    id,
12    name AS Department_name
13 FROM at7sharp_departments) AS r
14
15 ON l.Dep_id = r.id
```

- б) Найти наибольшую зарплату по департаментам и отсортировать департаменты по убыванию максимальной зарплаты

```
1 SELECT
2   Department_name,
3   max(employee_salary) AS max_salary
4 FROM
5   (SELECT
6     id          AS employee_id,
7     Salary      AS employee_salary,
8     Department_name
9
10    FROM at7sharp_employees AS l
11
12    LEFT JOIN
13
14    (SELECT
15       id,
16       name AS Department_name
17    FROM at7sharp_departments) AS r
18
19    ON l.Dep_id = r.id)
20
21 GROUP BY Department_name
22 ORDER BY max_salary DESC
```

Обычная SQL задача 2

Purchases

user_id	user_gender	items	price
101	f	3	100
102	female	0	0
103	m	0	0
101	f	2	100
105	male	2	100
103	m	0	0
...

Используя таблицу выше, напиши следующие SQL запросы:

а) Посчитай доход с женской аудитории (доход= сумма price*items)

```
4 SELECT
5     startsWith(user_gender, 'f') AS gender,
6     sum (items*price) AS female_sum
7 FROM at7sharp_purchases
8 WHERE gender = 1
9 GROUP BY gender
```

б) Сравни доход по группе мужчин и женщин

```
4 SELECT
5     startsWith(user_gender, 'f') AS gender,
6     sum (items*price) AS female_sum
7 FROM at7sharp_purchases
8 GROUP BY gender
```

	gender	female_sum
1	0	200
2	1	500



- с) Посчитай кол-во уникальных пользователей-мужчин, заказавших более чем три айтема (суммарно за все заказы).

```
1  SELECT
2    count(DISTINCT user_id) AS male_id_count
3  FROM (
4    SELECT
5      user_id,
6      sum(items) AS total_items
7    FROM at7sharp_purchases
8    WHERE startsWith(user_gender, 'm') = 1
9    GROUP BY user_id
10   )
11  WHERE total_items > 3
```

```
SELECT
  count(DISTINCT user_id) AS male_id_count
FROM (
  SELECT
    user_id,
    sum(items) AS total_items
  FROM at7sharp_purchases
  WHERE startsWith(user_gender, 'm') = 1
  GROUP BY user_id
)
WHERE total_items > 3
```

Не самая обычная SQL задача 3

Transactions

transaction_ts	user_id	transaction_id	item
2016-06-18 13:46:51.0	13811335	1322361417	glove
2016-06-18 17:29:25.0	13811335	3729362318	hat
2016-06-18 23:07:12.0	13811335	1322363995	vase
2016-06-19 07:14:56.0	13811335	7482365143	cup
2016-06-19 21:59:40.0	13811335	1322369619	mirror
2016-06-17 12:39:46.0	3378024101	9322351612	dress
2016-06-17 20:22:17.0	3378024101	9322353031	vase
2016-06-20 11:29:02.0	3378024101	6928364072	tie
2016-06-20 18:59:48.0	13811335	1322375547	mirror
...

Каждый пользователь имеет набор транзакций с определенным временем. Используя таблицы выше, напиши НАИБОЛЕЕ ОПТИМАЛЬНЫЕ SQL запросы:

- а) Выведи для каждого пользователя первое наименование, которое он заказал (первое по времени транзакции)

```
3  SELECT
4      user_id,
5      argMin(item, transaction_ts) AS first_item
6  FROM at7sharp_transactions
7  GROUP BY user_id
8
```

	user_id	first_item
1	3378024101	dress
2	13811335	glove

```
SELECT
    user_id,
    argMin(item, transaction_ts) AS first_item
FROM at7sharp_transactions
GROUP BY user_id
```

b) Посчитай сколько транзакций в среднем делает каждый пользователь в течении 72х часов с момента первой транзакции

```
SELECT
    avg(transaction_count) AS avg_transactions
FROM
    (WITH t AS (
        SELECT
            user_id,
            (min(transaction_ts) + interval '72' hour) AS tr_upper_bound
        FROM transactions
        GROUP BY user_id
    )

    SELECT
        a.user_id AS user_id,
        count(a.transaction_id) AS transaction_count
    FROM
        transactions AS a
    INNER JOIN
        t
    ON a.user_id = t.user_id AND a.transaction_ts <= t.tr_upper_bound
    GROUP BY a.user_id
    ) AS useless_alias
```