

Бонус-модуль.

Дополнение результатов А/В-тестирования продвинутыми подходами

Артур Сапрыкин

Data Scientist, Freelancer

Цели модуля

Мы уже поговорили про различные продвинутые способы ускорения A/B-тестирования, но обычно во всех этих методах на каждую гипотезу приходилось одинаковое количество экспериментов. Техники, про которые мы поговорим в этом модуле, более оптимально распределяют трафик так, чтоб на неликвидные варианты тратилось меньше трафика.

Бонус-модуль. Дополнение результатов
A/B-тестирования продвинутыми подходами

Задачи модуля

- Изучим идею оптимизации A/B-тестирования
- Разберём, в каких пропорциях можно распределять трафик
- Смоделируем метод многоруких бандитов
- Посмотрим, как алгоритмы позволяют оптимизировать тестирование

Бонус-модуль. Дополнение результатов
A/B-тестирования продвинутыми подходами

Байесовские многорукие бандиты

Цели урока

Начнём с рассмотрения простого примера, где легко заметить некоторый недочёт в классическом A/B-тестировании. Мы узнаем, как можно ещё ускорять тестирование и получать более значимые результаты, опираясь на перераспределение трафика на примере байесовских многоруких бандитов. Затем мы поговорим про другие байесовские подходы.

Задачи урока

- Рассмотрим простой пример, где A/B-тестирование неразумно
- Разберём такой метод, как многорукие бандиты
- Сравним этот метод с A/B-тестированием
- Эпсилон-жадный алгоритм
- Сравнение с МБ
- Другие байесовские методы оптимизации тестирования

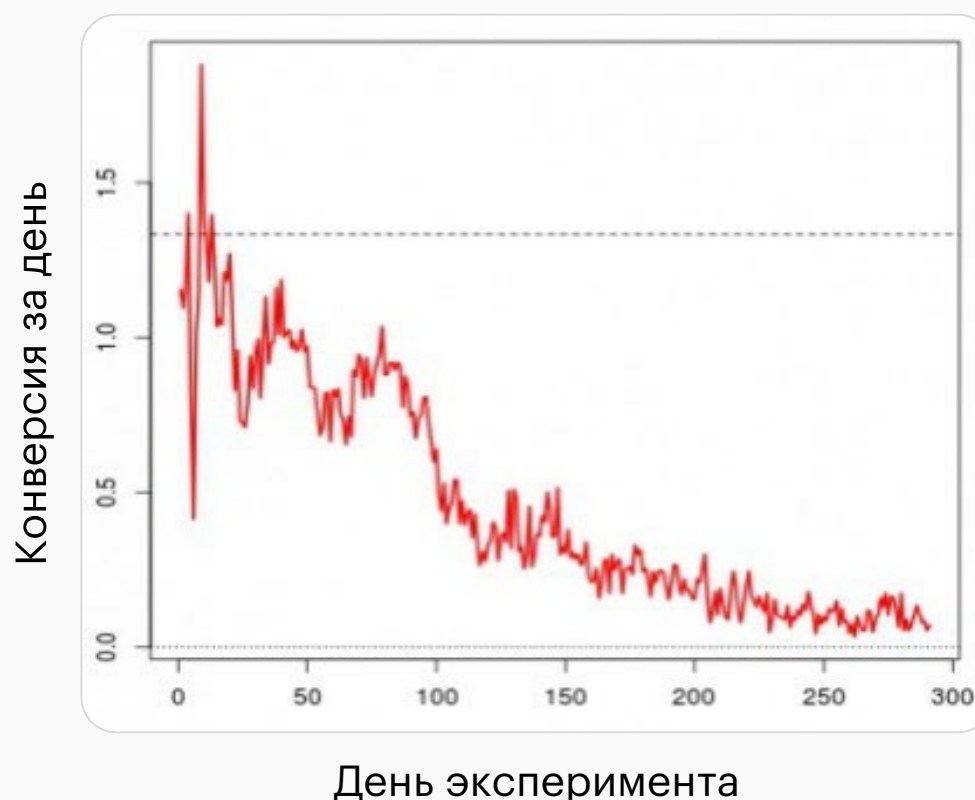
Рассмотрим пример

Вот вы запустили новый дизайн в некотором приложении и теперь ждёте, пока конверсия от потока клиентов подпрыгнет до небес.

Рассмотрим пример

Вот вы запустили новый дизайн в некотором приложении и теперь ждёте, пока конверсия от потока клиентов подпрыгнет до небес.

Мы с вами учились не останавливать A/B-тестирование, пока оно не закончится ранее запланированного срока. Но что делать, если результат тестирования выглядит вот так?

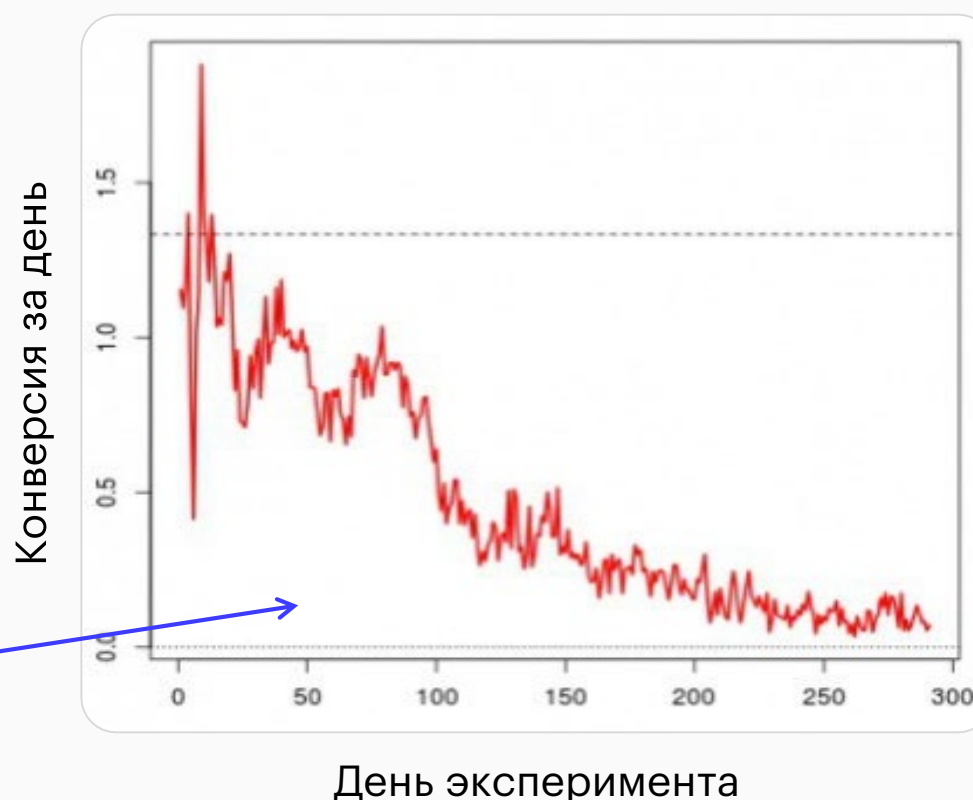


Рассмотрим пример

Вот вы запустили новый дизайн в некотором приложении и теперь ждёте, пока конверсия от потока клиентов подпрыгнет до небес.

Мы видим, что конверсия неумолимо падает со временем. Что можно в таком случае сделать?

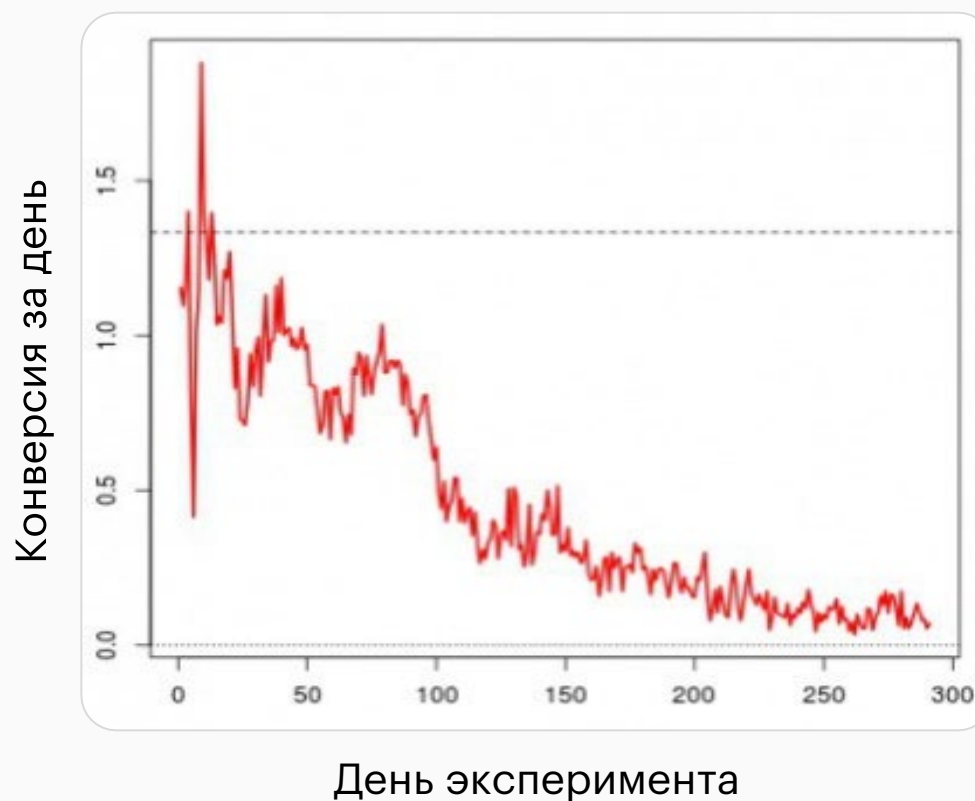
Ждать или всё-таки остановить тест?



Байесовский подход

Если бы мы выключили этот тест, то сэкономили бы время. Ведь, как видно по итогу, конверсия только упала.

Получается, чтобы решить эту проблему, нам нужен какой-то метод, который будет с хорошей уверенностью снижать поток трафика на малоэффективные гипотезы и перенаправлять этот поток на более перспективные. Такие методы давно придуманы и называются **байесовскими методами**.



Байесовский подход

Вообще, байесовских методов много. Мы начнём с классических **байесовских многоруких бандитов** (этот метод ещё называют частотным подходом). Такое название пошло от игровых автоматов, где, дёргая за ручку, вы могли получить некоторый выигрыш. При этом дёргая за ручки правильных автоматов, эту прибыль можно было максимизировать.



Общая схема

Каждый бандитский алгоритм работает по одному принципу. Многократно повторяется ряд действий:

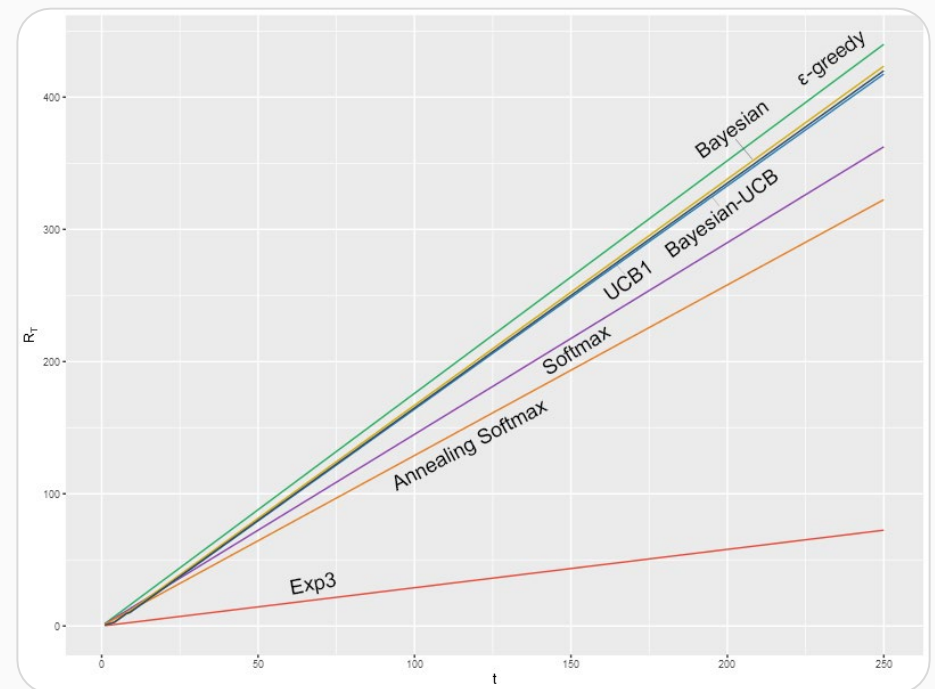
1. Выбрать руку
2. Получить вознаграждение
3. Запомнить выбранную руку и полученное вознаграждение

Все алгоритмы отличаются в основном только способом, как выбрать руку, и параметрами.



Виды алгоритмов

- Байесовские бандиты
- Эпсилон-жадный алгоритм
- Softmax-алгоритм
- Softmax-алгоритм с имитацией отжига
- Алгоритм UCB1
- Байесовский алгоритм UCB
- Алгоритм EXP3



Перед тем как углубляться в то, как устроены эти методологии, давайте разберём несколько примеров, чтобы оценить преимущество.

Кейс с баннерами

Представим, что вы начали тестировать три рекламных баннера для конкурса, который начнётся через месяц. Ваша цель — привлечь максимальное количество людей, принявших участие в конкурсе. Времени выбирать нет, поэтому вы решаете запускать сразу три баннера.

CTR 20 %



Cup of Champions

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

CTR 10 %



The Game Will Begin

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

CTR 30 %



Time to Min

Lorem ipsum dolor sit amet, consectetur adipiscing ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

Кейс с баннерами

Уже спустя две недели у вас получились три разные конверсии для трёх баннеров. С каким бы баннером вы бы хотели работать? Разумеется с тем, который 30 %.

20 %



Cup of Champions

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

10 %



The Game Will Begin

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

30 %

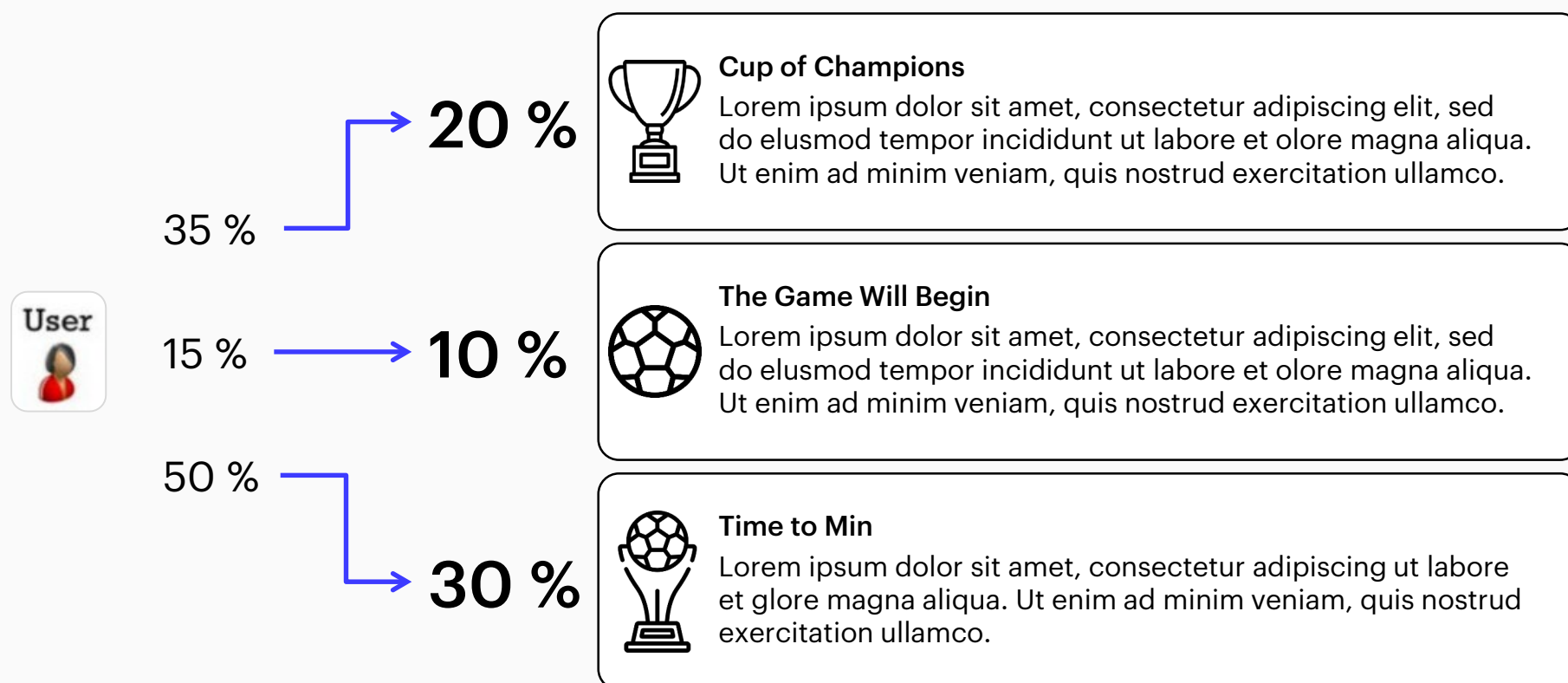


Time to Min

Lorem ipsum dolor sit amet, consectetur adipiscing ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

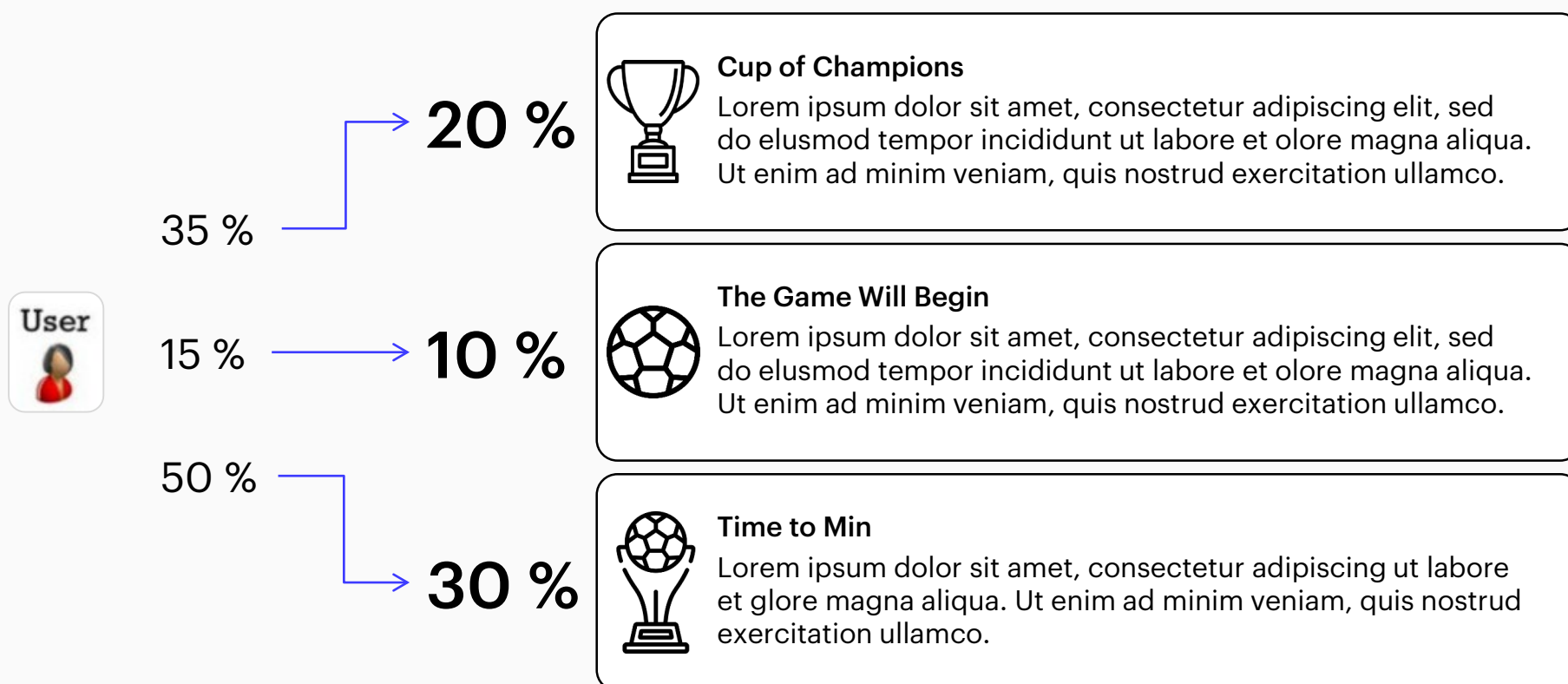
Байесовские многорукие бандиты

Если говорить про БМБ-метод, то стоит не выключать два малорезультативных баннера, а направлять на них меньше трафика, чтобы максимизировать количество участников в конкурсе.



Байесовские многорукие бандиты

Как итог, не дожидаясь конца конкурса, мы уже отправляем пользователей на более результативный баннер.

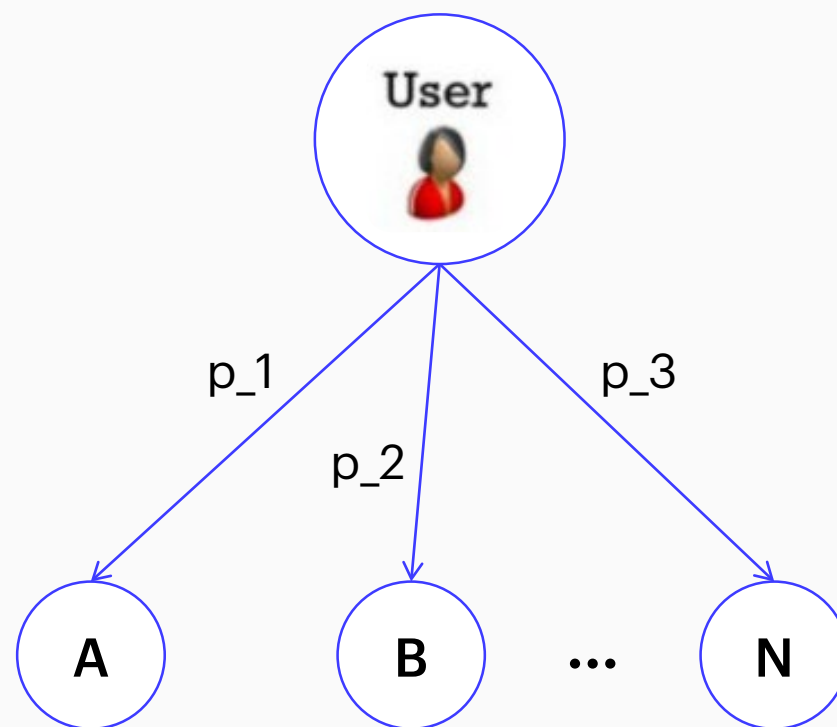


Байесовский подход и вероятности

Идея метода — направлять меньше трафика на гипотезы, приносящие меньший результат.

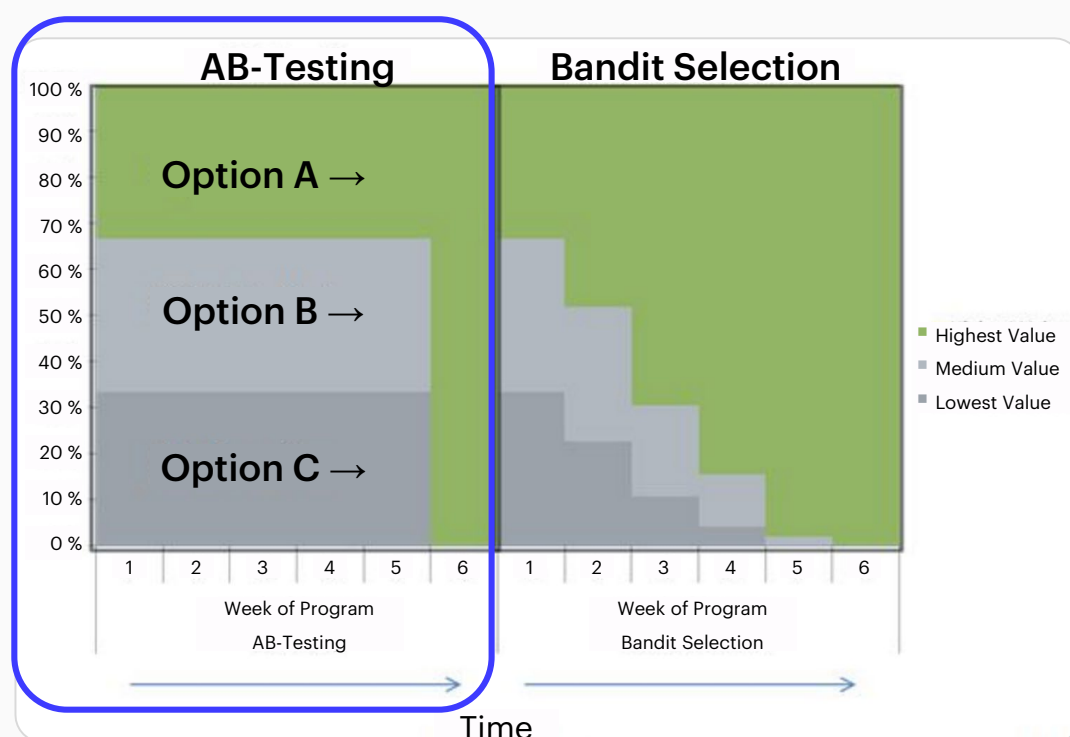
То есть в этом подходе у нас **вероятность** попасть в определённую группу **пропорциональна результативности** этой группы.

При этом малоэффективные варианты остаются «в игре» до конца тестирования. У них остаётся шанс показать свою эффективность.



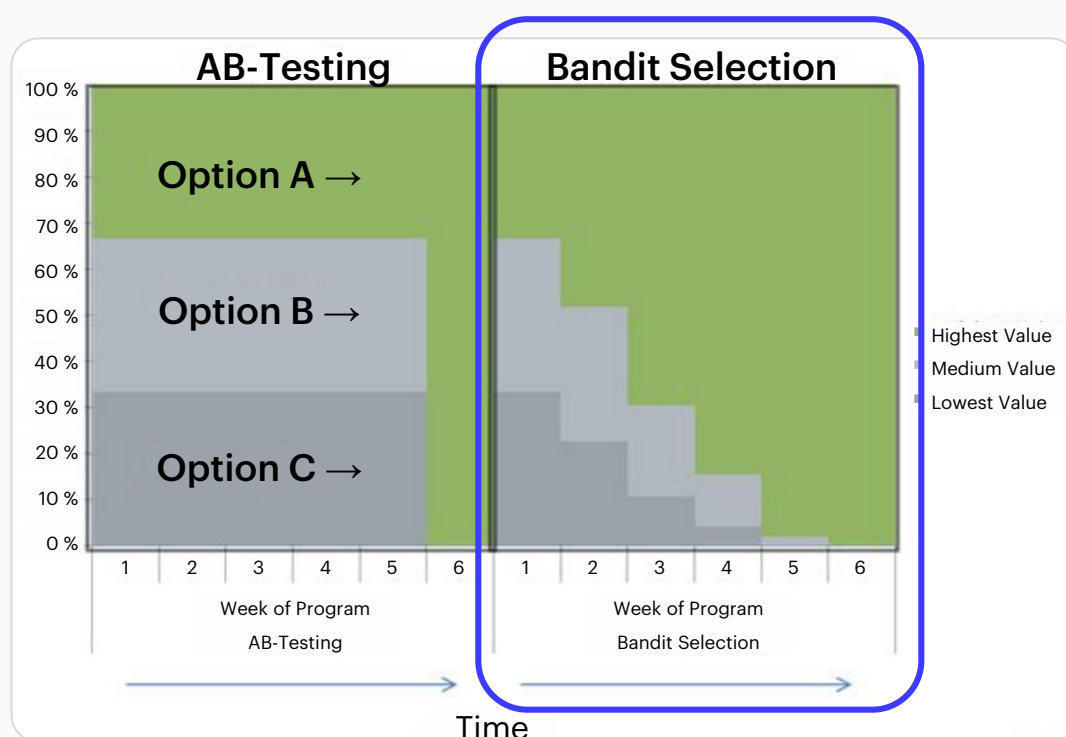
Байесовский подход: многорукие бандиты

Посмотрим теперь пример изменения соотношения трафика между группами во времени. На первом этапе в этом подходе накапливается некоторая статистика для каждой группы. То есть сперва это обычное A/B/n-тестирование.



Байесовский подход: многорукие бандиты

На первом этапе в этом подходе накапливается некоторая статистика. Далее трафик перераспределяется пропорционально успешности. При этом продолжается изучение эффективности кандидатов.



Байесовский подход: кейс с заголовками

Подходы типа многоруких бандитов позволяют начать получать больше преимущества от результативных кандидатов уже на стадии тестирования. Мы уже видели такой случай с баннерами ранее. Рассмотрим пример с заголовками.



Байесовский подход: кейс с заголовками

Заголовки — отличный пример использования «бандитских алгоритмов». Зачем применять классический сплит-тест для заголовков, если к тому времени, когда вы узнаете, какой вариант лучше, актуальный момент для этого заголовка уже пройдёт!



Байесовский подход: кейс с заголовками

Заголовки — отличный пример использования «бандитских алгоритмов». Зачем применять классический сплит-тест для заголовков, если к тому времени, когда вы узнаете, какой вариант лучше, актуальный момент для этого заголовка уже пройдёт!

Новости характеризуются коротким циклом жизни, а «бандитские алгоритмы» быстро определяют, какой заголовок эффективнее. При этом мы знаем, что важно, чтобы пользователи увидели лучший заголовок в первые моменты публикации.

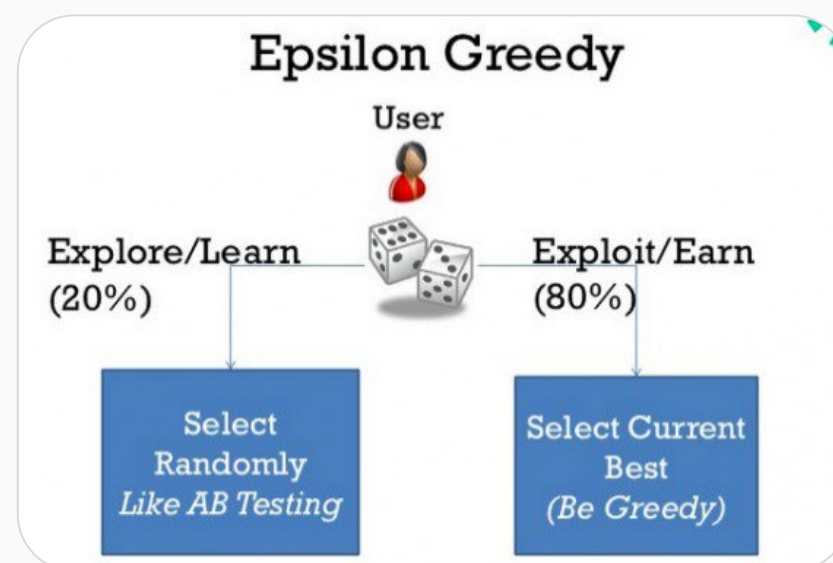


A/B/n-тестирование vs многорукие бандиты

- **Стоимость.** Сбор данных — это затраты, а «бандитский алгоритм» позволяет вам минимизировать эти затраты в ходе оптимизации. Иногда данные стоят очень дорого, если речь идёт про тестирование какого-либо препарата
- **Автоматизация.** «Бандитский алгоритм» — это естественный путь автоматизации оптимизации с применением машинного обучения, особенно с учётом потребностей пользователя (A/B-тестирование в этой ситуации весьма затруднительно)
- **Контроль ситуации.** Разрешая алгоритму иногда выбирать менее удачные опции, вы даёте ему возможность пересмотреть их эффективность. Это позволяет создать работающий инструмент, непрерывно определяющий наиболее эффективные варианты и заменяющий малоэффективные новыми

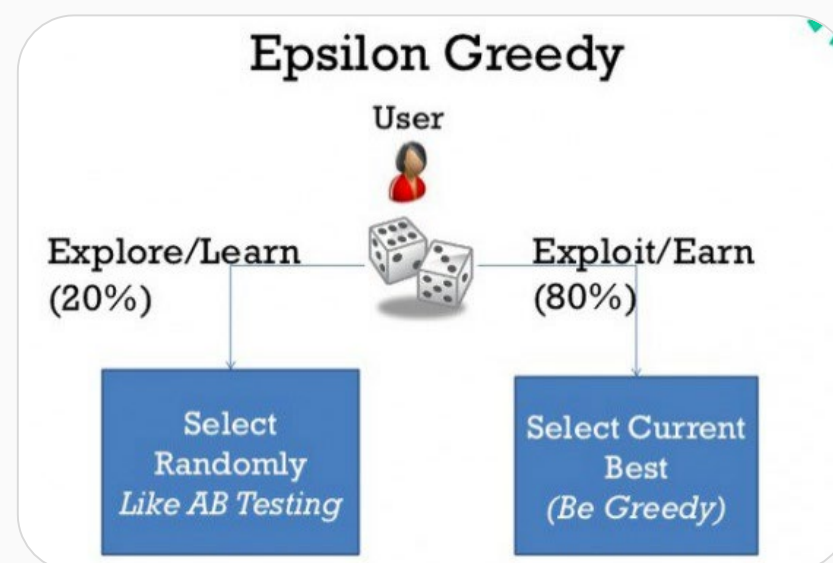
Другой байесовский ПОДХОД: эпсилон-жадный алгоритм

Рассмотрим простейший и наиболее доступный метод — эпсилон-жадный алгоритм (epsilon-greedy algorithm). Действие данного алгоритма показывает в общих чертах, как работают «бандитские алгоритмы».



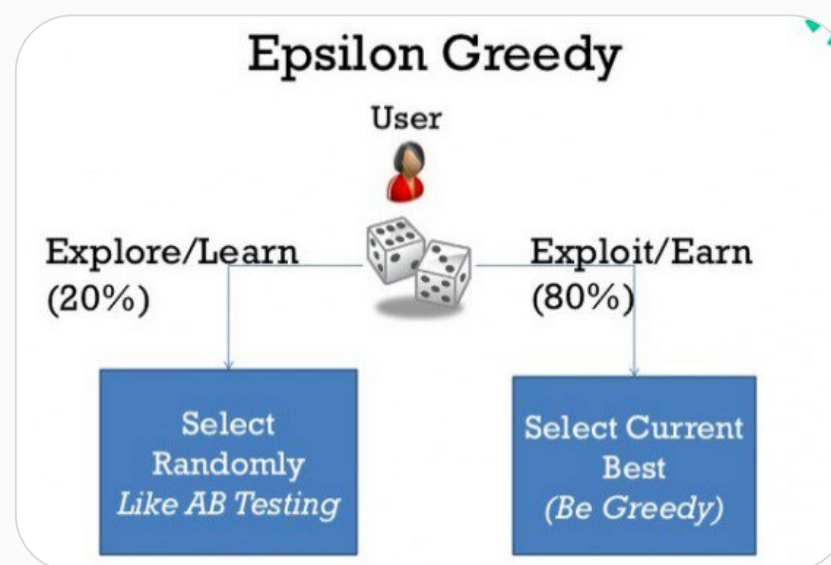
Другой байесовский ПОДХОД: эпсилон-жадный алгоритм

Жадный алгоритм — это алгоритм, всегда выбирающий наилучшее на данный конкретный момент времени действие. Так, эпсилон-жадный алгоритм оказывается в большинстве случаев обычным жадным алгоритмом, каждый раз выбирая наиболее целесообразный в данный момент вариант.



Другой байесовский ПОДХОД: эпсилон-жадный алгоритм

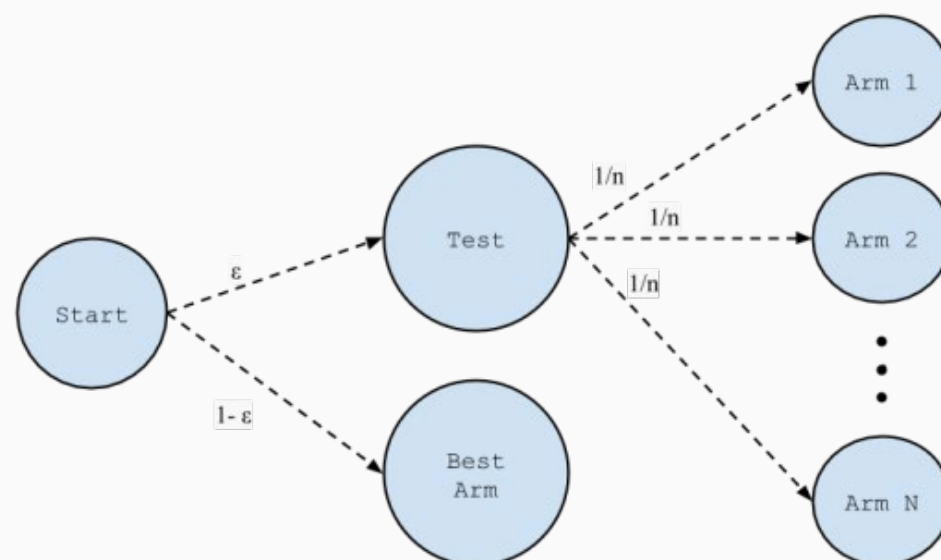
Но время от времени
эпсилон-жадный алгоритм
отдаёт предпочтение другим
доступным вариантам с целью
изучения их эффективности.



Другой байесовский ПОДХОД: эпсилон-жадный алгоритм

Как эпсилон-жадный алгоритм направляет пользователя между тестируемыми версиями:

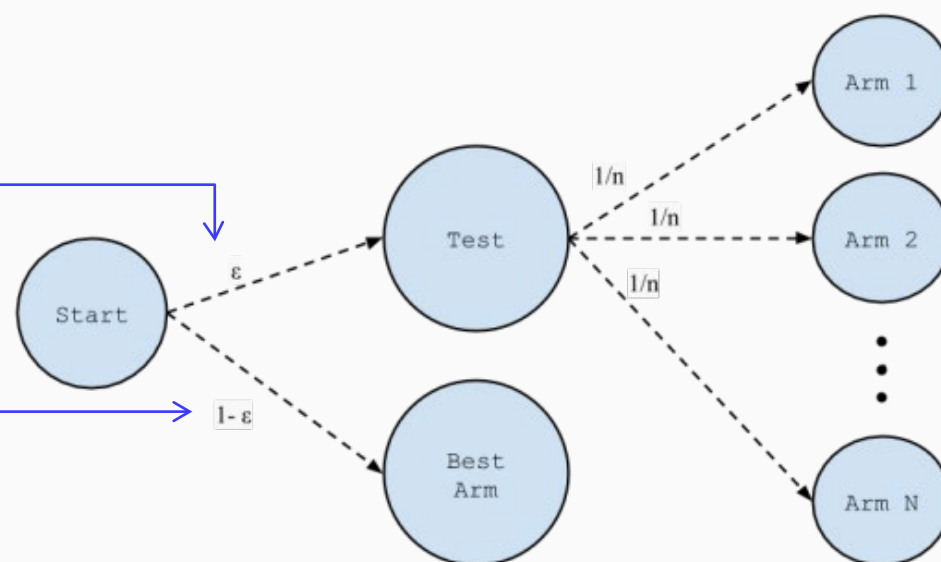
- 20 % — случайный выбор (как A/B-тестирование)
- 80 % — выбор текущего лучшего варианта («жадность»)



Другой байесовский ПОДХОД: эпсилон-жадный алгоритм

Как эпсилон-жадный алгоритм
направляет пользователя между
тестируемыми версиями:

- 20 % — случайный выбор
(как A/B-тестирование)
- 80 % — выбор текущего
лучшего варианта
(«жадность»)

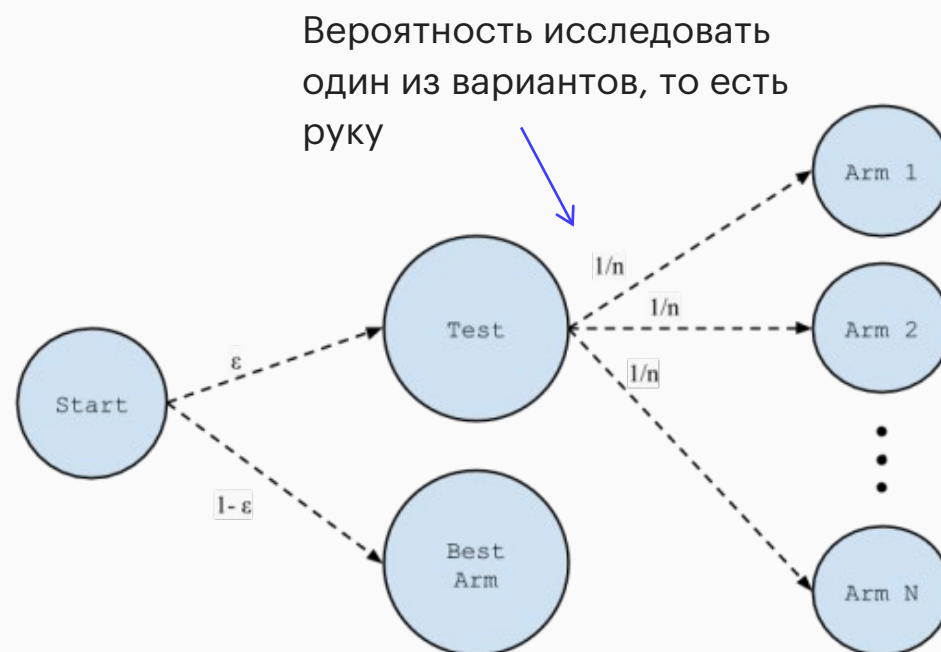


Параметр ϵ (обычно малое число, которое должно лежать в пределах между 0 и 1) означает вероятность, с которой алгоритм выполняет исследование, а не использование

Другой байесовский ПОДХОД: эпсилон-жадный алгоритм

Как эпсилон-жадный алгоритм направляет пользователя между тестируемыми версиями:

- 20 % — случайный выбор (как A/B-тестирование)
- 80 % — выбор текущего лучшего варианта («жадность»)



Параметр ϵ (обычно малое число, которое должно лежать в пределах между 0 и 1) означает вероятность, с которой алгоритм выполняет исследование, а не использование

Эпсилон-жадный алгоритм: плюсы и минусы

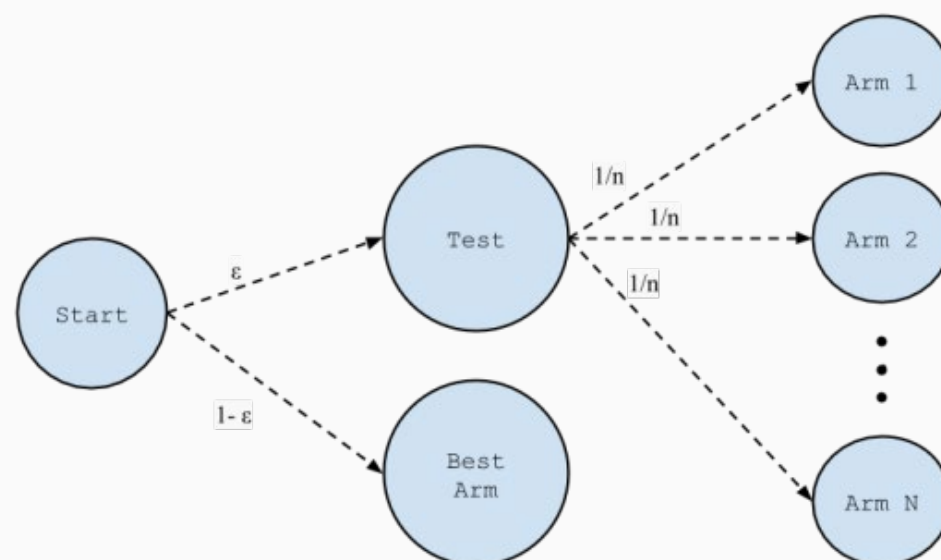
Метод эпсилон-жадный алгоритм имеет определённые недостатки и достоинства.

Среди преимуществ метода:

- лёгкость в применении
- высокая эффективность в большинстве случаев
- эффективность метода не зависит от сезонности

Из недостатков метода следует выделить:

- изменчивость стратегии
- открытый вопрос: стоит ли сокращать период изучения со временем?



Другие байесовские подходы

Другие «бандитские алгоритмы»

Другие «бандитские алгоритмы» решают те же проблемы иными путями. Среди них:

- верхний доверительный предел (UCB1)
- метод выборки Томпсона
- байесовский «бандитский алгоритм»

Все эти методы пытаются найти идеальный баланс между длительностью периода изучения/сбора данных и практического применения полученных данных в эксперименте (выполнение действий, основываясь на актуальных данных о лучшем выборе).

Итоги и выводы урока

Итак, кроме обычного разделения трафика на равные доли, можно делить его на доли пропорционально успешности «кандидата». Изучай и оптимизируй одновременно — это основная идея многоруких бандитов. В ряде случаев «бандитские алгоритмы» дают отличные результаты. Однако, при наличии конкретного предмета исследования, в ходе которого необходимо узнать эффективность применяемой методики, также имеется определённая относительно решения задачи. Тогда стандартное A/B-тестирование станет лучшим вариантом.

ИТОГИ И ВЫВОДЫ УРОКА

- Многорукие бандиты могут позволить сэкономить ещё больше ресурсов и времени
- Идея оптимизации очень проста, при этом у неё много модификаций
- Реализация байесовских подходов уже реализована под капотом многих инструментов для проведения тестирования в web-сервисах