Geekbrains

|  |
| --- |
| **Дипломная работа** |
| Анализ и визуализация данных с помощью python на примере данных A/B-тестирования объявлений |

|  |
| --- |
| Выполнила:  Дядькина Анна Анатольевна  Факультет  Разработчик — Аналитик |

|  |
| --- |
| Калининград  2023 |

Оглавление

[Введение 2](#_Toc135949017)

[Глава 1. Визуальный метод анализа данных 3](#_Toc135949018)

[1.1 Применение визуализации данных 3](#_Toc135949019)

[1.2 Средства визуализации данных 10](#_Toc135949020)

[Глава 2. Контекстная реклама и A/B-тестирование объявлений 13](#_Toc135949021)

[2.1 Форматы контекстной рекламы 13](#_Toc135949022)

[2.2 Основные показатели рекламной кампании 14](#_Toc135949023)

[2.3 Методология A/B-тестирования рекламных объявлений 18](#_Toc135949024)

[Глава 3. Сбор, анализ и визуализация данных тестирования 19](#_Toc135949025)

[3.1 Подготовка к анализу данных, установка необходимых библиотек 19](#_Toc135949026)

[3.2 Первичный анализ и визуализация данных 23](#_Toc135949027)

[3.2.1 Первичный анализ данных 23](#_Toc135949028)

[3.2.2 Ответы на анкету 24](#_Toc135949029)

[3.2.3 Анализ по дням 30](#_Toc135949030)

[3.2.4 Исследование часов 33](#_Toc135949031)

[3.2.5 Девайсы 36](#_Toc135949032)

[3.2.6 Операционные системы 44](#_Toc135949033)

[3.2.7 Браузеры 45](#_Toc135949034)

[3.2.8 Подведение итогов первичного анализа базы данных тестирования 52](#_Toc135949035)

[3.3 Статистический анализ 53](#_Toc135949036)

[3.3.1 Формулировка гипотезы 53](#_Toc135949037)

[3.3.2 Критерий Фишера 54](#_Toc135949038)

[3.3.3 Доверительный интервал 55](#_Toc135949039)

[3.3.4 Подведение итогов статистического анализа 59](#_Toc135949040)

[3.4 Подведение итогов тестирования и предложения по настройке рекламной кампании 59](#_Toc135949041)

[Заключение 61](#_Toc135949042)

[Список литературы 62](#_Toc135949043)

[Приложение 64](#_Toc135949044)

Введение

Сегодня стремительно растет количество и виды данных. У современного исследователя есть потребность в гибких, мощных и легко масштабируемых инструментах для сбора и анализа информации.

Рыночные условия в текущее время побуждают компании искать все более эффективные способы продвижения продукции, товаров, услуг. Реклама в интернете - один из лучших способов бизнесу заявить о себе.

При запуске рекламной кампании сложно прогнозировать, какие объявления будут наиболее эффективны. A/B тестирование объявлений позволяет сравнить и определить наиболее перспективные из них. В ходе эксперимента собираются данные, требующие анализа для последующего выбора стратегии продвижения, увеличения конверсии и минимизации затрат на рекламу.

Цель данной работы — изучить инструменты анализа и визуализации данных с помощью библиотек python на примере тестовой базы данных, содержащей результаты A/B-теста, проведенного рекламным агентством.

Для достижения цели необходимо решить следующие задачи:

1. изучить литературу по A/B-тестированию и визуализации данных;
2. рассмотреть основные библиотеки python для анализа и визуализации данных;
3. составить анализ тестовой базы данных, содержащей результаты A/B-теста, проведенного рекламным агентством;
4. визуализировать результаты A/B-тестирования;
5. выявить закономерности в результатах тестирования и разработать предложения по настройке рекламной кампании.

Глава 1. Визуальный метод анализа данных

1.1 Применение визуализации данных

Визуализация данных являет собой представление данных в живописной или графической форме. На протяжении всей своей истории люди зависели от визуального представления данных, чтобы воспринимать информацию легче и быстрее (например, диаграммы, схемы, карты и многое другое).

Визуализация данных относится к методам, используемым для передачи данных или информации с помощью их кодирования как визуальных объектов (например, точки, линии или столбцы), содержащихся в графике. Цель состоит в том, чтобы передавать информацию четко и эффективно для её пользователей. Это один из шагов в анализе данных. Это не означает, что визуализация данных должна выглядеть скучно (быть функциональной или чрезвычайно сложной), чтобы выглядеть красиво. Для эффективной передачи идеи эстетическая форма и функциональность должны идти рука об руку, обеспечивая понимание.

Визуализация данных рассматривается многими дисциплинами как современный эквивалент визуальной коммуникации и, вместе с тем, не принадлежит какой-либо одной области, а находит применение во многих (например, она рассматривается в качестве современного направления описательной статистики, но также в качестве обоснованной теории развития инструментария в других областях). Она включает в себя создание и исследование визуального представления данных, что означает «информацию, которая была в абстрагированном схематическом виде, в том числе, для атрибутов или переменных единиц информации».

Эффективная визуализация помогает пользователям в анализе и обработке данных. Это делает комплексную информацию более доступной, понятной и полезной. Таблицы, как правило, используются там, где пользователи будут смотреть конкретные меры переменных, в то время как диаграммы различных типов используются, чтобы показать тренды или отношения одной или нескольких переменных.

За последнее десятилетие произошло колоссальное увеличение объема цифровых данных, поступающих из различных источников. Системы баз данных позволяют осуществлять массовый сбор данных, которые могут быть использованы для выявления различных проблем и дают возможность прогнозирования путем выявления важных влияющих факторов. Следовательно, можно наблюдать повышенный интерес к использованию этих больших структурированных и неструктурированных наборов данных в целях повышения эффективности принятия обоснованных и аналитических решений.

На сегодняшний момент в мире ежедневно производится 2,5 квинтиллиона байт данных, и 90 % всех данных было создано за последние два года. С таким большим количеством данных становится все труднее управлять и самое главное оперативно принимать решения.

Визуализация может помочь людям увидеть вещи, которые не были очевидны для них прежде. Даже когда объем данных очень большой, тренды могут быть замечены быстро и легко. Визуализация помогает передать информацию универсальным образом и сделать это просто.

Визуализация данных представляет информацию таким образом, что ее можно легко интерпретировать, экономя время и энергию. Интерактивные графики и диаграммы делают восприятие более легким для лиц, принимающих решения:

− определить области, требующие внимания или улучшение; − понимать, какие факторы влияют на поведение ваших клиентов; − знать, какие продукты где размещать;

− прогнозировать объемы продаж;

− узнать, как увеличить доходы или сократить расходы и др.

До недавнего времени визуальный анализ данных для отображения результатов на обычных мониторах использовал только двумерную или очень простую трехмерную графику. Более сложные графические образы отображать в реальном времени было достаточно сложно и дорого. Однако прогресс в области аппаратных средств вывода изображений способствовал и совершенствованию средств визуального анализа данных. В настоящее время существует достаточно большое количество различных видов графических образов, позволяющих представлять результаты анализа в виде, удобном для понимания человеком.

С помощью новых технологий пользователи способны оценивать: большие объекты или маленькие, далеко они находятся или близко. Пользователь в реальном времени может двигаться вокруг объектов или кластеров объектов и рассматривать их со всех сторон. Это позволяет использовать для анализа естественные человеческие перцепционные навыки в обнаружении неопределенных образцов в визуальном трехмерном представлении данных.

Визуальный анализ данных особенно полезен, когда о самих данных мало известно и цели исследования до конца непонятны. За счет того, что пользователь напрямую работает с данными, представленными в виде визуальных образов, которые он может рассматривать с разных сторон и под любыми углами зрения, в прямом смысле этого слова, он может получить дополнительную информацию, которая поможет ему более четко сформулировать цели исследования.

Таким образом, визуальный анализ данных можно представить как процесс генерации гипотез. При этом сгенерированные гипотезы можно проверить или автоматическими средствами (методами статистического анализа или методами Data Mining), или средствами визуального анализа. Кроме того, прямое вовлечение пользователя в визуальный анализ имеет два основных преимущества перед автоматическими методами:

а) визуальный анализ данных позволяет легко работать с неоднородными и зашумленными данными, в то время как не все автоматические методы могут работать с такими данными и давать удовлетворительные результаты;

б) визуальный анализ данных интуитивно понятен и не требует сложных математических или статистических алгоритмов.

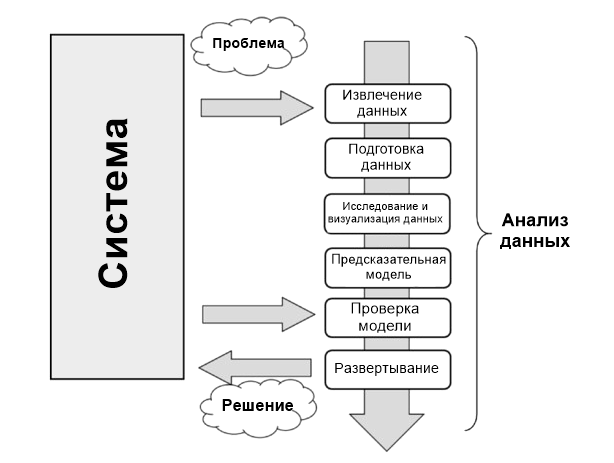
Визуальный анализ данных обычно выполняется в три этапа:

а) беглый анализ — позволяет идентифицировать интересные шаблоны и сфокусироваться на одном или нескольких из них;

б) увеличение и фильтрация — идентифицированные на предыдущем этапе шаблоны отфильтровываются и рассматриваются в большем масштабе;

в) детализация по необходимости — если пользователю нужно получить дополнительную информацию, он может визуализировать более детальные данные.

Визуализацию данных можно представить как один из шагов анализа данных с целью сделать предсказания на основе математической модели.

  
Рисунок 1 - Схема визуального анализа данных

Аналитики применяют более 60 видов визуализации: различные диаграммы, графики, дашборды и карты. Рассмотрим самые популярные методы визуализации.

Диаграммы. Эти средства визуализации информации насчитывают десятки видов. Диаграмму следует выбирать исходя из типа данных, целей и вида анализа.

Виды диаграмм:

* Круговая диаграмма. Идеально подходит для показа частей целого. Но если показателей больше пяти, исчезает наглядность.
* Круговая диаграмма имеет и иные формы, которые помогают лучше решить задачу. Хорошо использовать диаграмму-пончик, поскольку человек проще ориентируется на длину окружности, нежели на площадь, и видит разницу сразу. Многоуровневые диаграммы подходят для показа иерархии и связь между категориями.
* Столбчатая диаграмма более подходит для точного сравнения категорий, когда количество показателей выше пяти. Используется для того, чтобы отображать рейтинг, т.е. представить лучшие и худшие параметры по отношению друг к другу. К этому виду визуализации относятся и гистограммы.
* Географическая диаграмма для представления данных, связанных с географическим положением объектов.
* Пузырьковая диаграмма нужна, когда важно отобразить соотношение различных показателей.

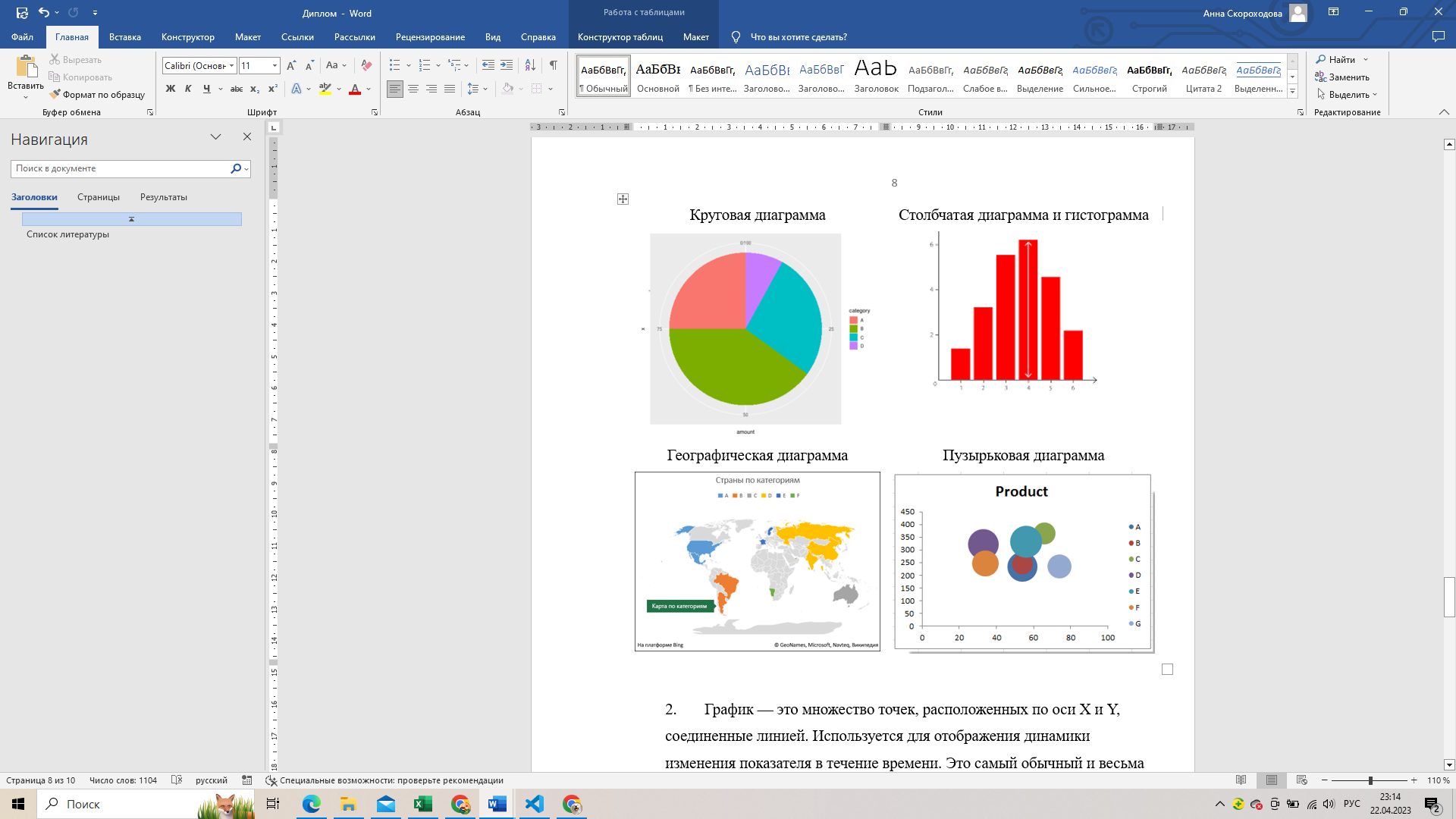


Рисунок - Примеры диаграмм

График — это множество точек, расположенных по оси X и Y, соединенные линией. Используется для отображения динамики изменения показателя в течение времени. Это самый обычный и весьма наглядный вид визуализации, поэтому неизменно пользуется популярностью и широко применяется.

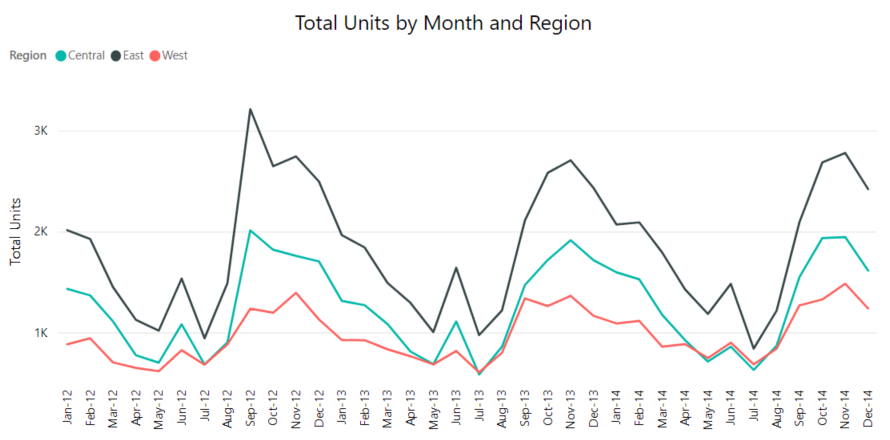


Рисунок - Пример графика

Линейный график. Наиболее понятный вид динамики — это одна или несколько линей. Каждая характеризуется одним показателем, изменяющимся по параметру оси X.

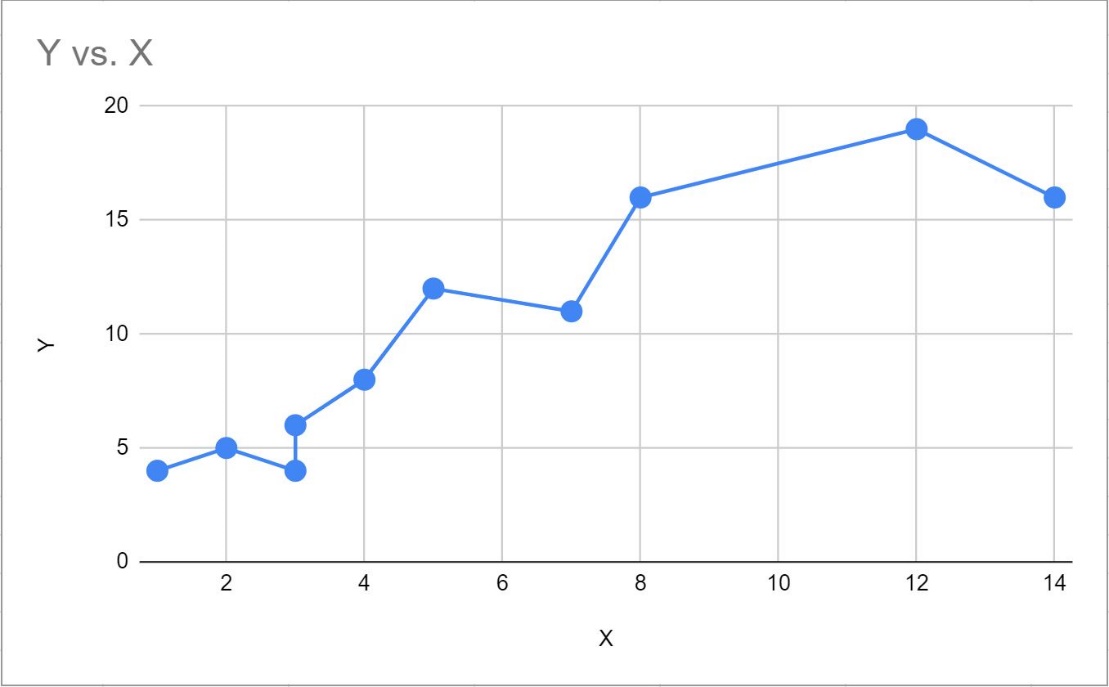


Рисунок - Пример линейного графика

Таблица и матрица. Для сжатия и упрощения информации необходимо, чтобы в таблицах содержалось ограниченное количество строк и столбцов. Таблица является двумерным массивом данных, т.е. на пересечении строки и столбца есть совокупность параметров. Матрица тоже представляет собой таблицу, но она более расширена, поскольку может включать группы строк и/или столбцов.

1.2 Средства визуализации данных

Выбор правильного инструмента для визуализации данных часто зависит от типа данных и уровня знания пользователем языков кодирования, так как многие инструменты специализированы для работы с определенными форматами данных. Рассмотрим некоторые из них:

Tableau — это система, позволяющая провести глубокий и разносторонний анализ информации с последующим представлением результатов в интерактивной форме. При этом это очень гибкая, быстрая и простая в освоении платформа.

Microsoft Power BI - набор инструментов бизнес-аналитики, которые позволяют анализировать и наглядно представлять ценную информацию. Благодаря сервису, можно использовать сторонние приложения и множество источников информации. С его помощью можно создавать кастомизированные визуализации.

Онлайн-сервис Flourish создаёт красивые визуализации данных, чтобы наглядно показать не только цифры, но и аналитику. Инструмент подключает анимацию и создаёт data storytelling.

Устройство для захвата данных Datawrapper – это онлайн-платформа для создания PNG-диаграмм и карт без необходимости кодирования. Он был разработан специально для писателей, журналистов, блогеров.

Plotly – это полностью веб-интерфейс для создания графики. Он не требует каких-либо знаний кодирования. С его помощью аналитики создают интерактивные графики, дашборды, карты. Всем проектам Plotly присваивается уникальный URL — их легко встраивать в блоги. Графики получаются очень детальными и подробными. В библиотеке хранится много уникальных диаграмм.

Excel является эффективным и универсальным средством для любой аналитической задачи. Excel можно объединить ее с Power BI для получения широких возможностей аналитики и визуализации данных.

В Python визуализация данных легко осуществляется с использованием сторонних пакетов. Чаще всего используют библиотеки Matplotlib, Bokeh, pygal, seaborn.

Выбирая те или иные инструменты для визуализации данных, нужно учитывать их плюсы и минусы. Недостатками сервисов могут быть: ограниченный функционал бесплатных версий, невозможность использования всей компанией, плохая защищенность данных, сложности с импортом и обработкой больших объемов данных, высокий порог вхождения для аналитика.

Язык python стал популярен среди специалистов в области data science. Это один из самых простых языков для изучения. Он предлагает множество библиотек, которые применяются на всех этапах анализа данных. С помощью python можно анализировать очень большие массивы данных – нет ограничений по их объему.

Python подходит для следующих задач:

сбор данных;

обработка данных, построение моделей;

визуализация данных.

Выделим некоторые преимущества python для анализа и визуализации данных:

Широкое сообщество, помогающее найти решение для поставленной задачи.

Постой синтаксис и читаемость.

Разнообразие библиотек для разных этапов анализа данных.

Гибкость и масштабируемость, благодаря чему используется в самых разных отраслях.

Регулярные обновления.

Минусами использования можно считать медленную обработку данных и плохая развитость работы с базами данных.

Глава 2. Контекстная реклама и A/B-тестирование объявлений

2.1 Форматы контекстной рекламы

Достаточно крупный сегмент рекламы в интернете — контекстная, которая составляет 55% от общего объема всего интернет-рынка. Этот вид рекламы основывается на анализе тематики сайта, которая привлекла внимание посетителя. Контекстная реклама действует прицельно и предлагает рекламу товаров и услуг, которые интересны посетителю сайта.

Всю рекламу можно условно поделить по месту показа объявлений на два блока: поисковый и сетевой. Поисковая реклама отображается в окне поисковой выдачи. Сетевая реклама размещается на сайтах, которые входят рекламную сеть.

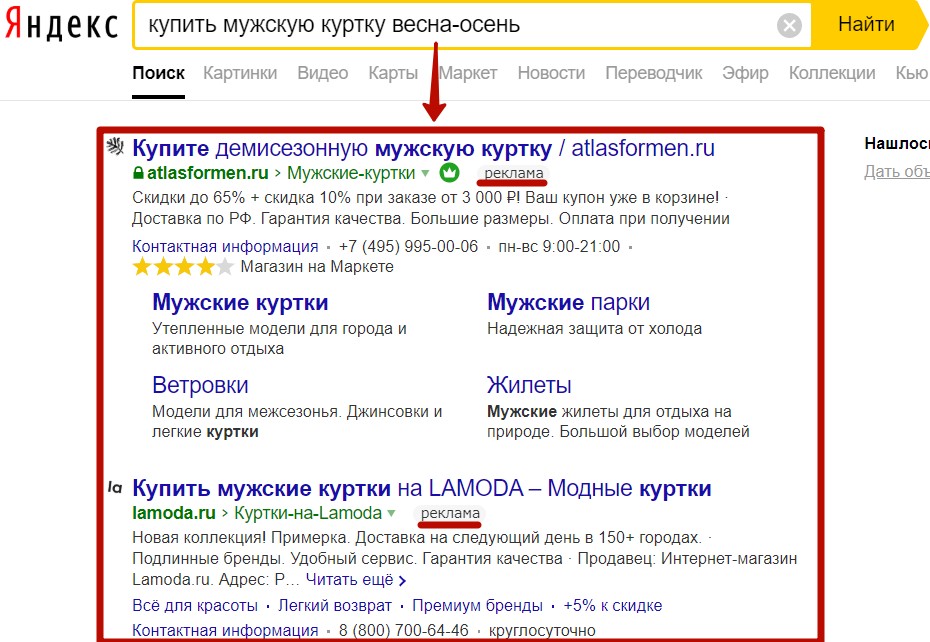
Форматы показа рекламы от Яндекс:

1. Текстовые, которые показываются при использовании поисковой строки. Объявления выглядят как часть поисковой выдачи. Пример представлен на рисунке 6.

2. Текстово-графические. Размещаются обычно слева или справа от основного контента и носят название РСЯ (рекламная сеть Яндекса).

3. Видеоролики.

4. Баннеры на поиске показываются справа от результатов поиска Яндекса по ключевым словам в ответ на поисковый запрос пользователя.

Рисунок - Пример текстовой рекламы

2.2 Основные показатели рекламной кампании

Разработка эффективного контекстного объявления и управления кампанией является сложной задачей, требующей учета многих факторов, которые имеют неоднозначное и нередко ситуативное влияние на конечные результаты для рекламодателя.

Рассмотрим основные показатели эффективности рекламы:

1. CTR (коэффициент кликабельности, click-through rate), рассчитывается как число кликов по рекламным объявлениям к общему числу показов этих объявлений. Это один из наиболее часто используемых показателей на практике; за счет него можно увидеть, какая доля пользователей не заходит на сайт рекламодателя. В том случае, если показатель кликабельности падает в динамике, можно сделать вывод о нерелевантности (несоответствия) текста или заголовка объявления запросу пользователя;
2. общий расход – общий размер затрат по той или иной рекламной кампании за определенный промежуток времени;
3. средняя цена клика – средняя стоимость перехода пользователя по объявлению рекламодателя. Этот показатель является ключевым при определении эффективности рекламной кампании: если объявление релевантно, отражает потребность пользователя, то ему в поисковой выдаче будет отдаваться больший приоритет, и цена, которую рекламодатель будет платить за переход по этому объявлению, будет ниже. Вторым фактором, определяющим цену клика, является стратегия показов, которую выбирает рекламодатель, а также максимальная цена клика;
4. цена цели (при наличии целей), средняя величина затрат, которые несет рекламодатель для мотивации клиента к достижению заданной им цели (например, при заказе какого-либо товара). На основе данного показателя определяется целесообразность использования некоторых ключевых слов, переходы по которым обходятся дороже стоимости определенных товаров или не выгодны с точки зрения рекламодателя;
5. средняя позиция показа – среднее место, которое занимают все объявления определенной рекламной кампании. То есть в поисковой выдаче, когда пользователь вводит какой-либо запрос, он видит ряд рекламных объявлений и, как правило, выбирает те, которые расположены на первых местах. Обычно чем выше место показа объявления, тем больше пользователей заходят на сайт;
6. посетители – количество посетителей за рассматриваемый период времени. Если наблюдается резкая динамика снижения количества посетителей – это признак того, что в рекламных кампаниях появились какие-либо ошибки, появляется необходимость искать причинно-следственную связь;
7. новые посетители – количество посетителей, которые оказались на сайте впервые;
8. процент отказов – отношение всех посетителей к посетителям, ушедших с первой страницы. Рекламодатели обращает большое внимание этот показатель. В разных нишах может быть различный допустимый порог этого показателя. Универсальным считается интервал в 20-30%. В случае, если этот порог преодолевается, необходимо искать ошибки в рекламных кампаниях;
9. источник трафика. Источником может быть поисковая система, социальная сеть или какой-либо другой интернет-ресурс. Анализируя источники трафика, можно ранжировать маркетинговые каналы по степени их значимости;
10. глубина просмотра – количество просмотренных страниц одним пользователем. На практике, если посетитель твердо решает купить определенный товар и попадает на нужную ему страницу сайта, то он совершает покупку, соответственно, глубина просмотра будет 1. В других случаях, когда пользователь просматривает каталог в поисках нужного ему товара и уходит с 1 страницы, – это может быть признаком того, что пользователь не доверяет сайту либо он ему непонятен;
11. возраст. Возраст посетителей и пол в основном определяют целевую аудиторию. Это сужает круг потенциальных клиентов, но делает их более «качественными», то есть готовыми к покупке товара/услуги;
12. тип устройства (мобильный телефон, планшет, декстоп). Посредством этого показателя становится возможным просмотр отдельной статистики по разным типам устройств и определение целесообразности показа объявлений на определенном типе устройств;
13. конверсия определяет общую эффективность сайта и рекламы, а также отдельных целей рекламы. На практике не существует универсального показателя конверсии, поскольку для разных рыночных ниш этот показатель может меняться;
14. страницы входа – адреса страниц, на которые пользователь заходит в начале сеанса;
15. страницы выхода – адреса страниц, с которых пользователь выходит в конце сеанса;
16. посещаемость по времени суток – время суток, в течение которого наблюдается наибольшее количество пользователей и наоборот. Также можно определить наиболее эффективные конверсионные часы и проводить изменения в рекламных кампаниях именно для этих часов (например, повышать ставки);
17. периодичность визитов – периодичность, с которой клиенты посещают сайт, если не было выполнено никакого целевого действия с первого посещения. То есть данный показатель определяет цикл сделки (какое время обычно затрачивает посетитель на совершение заказа продукции/услуги);
18. время пребывания на сайте. Чем больше времени клиент проводит на сайте, тем выше вероятность совершенствования им покупки и тем сильнее его вовлеченность;
19. география. Этот показатель оценивает эффективность рекламной кампании по разным регионам. По результатам анализа показателя проводятся изменения в рекламных кампаниях (например, могут запрещаться показы рекламы в некоторых неэффективных регионах или повышаться ставки в наиболее эффективных).

2.3 Методология A/B-тестирования рекламных объявлений

A/B тестирование является одним из простейших и распространенных методов маркетинговых исследований. Его предназначение — показывать аудитории несколько разновидностей какой-либо элемента рекламы или сайта при других равных условиях. Измерив статистику, можно определить, какой вариант больше нравится целевой аудитории.

Рассмотрим этапы A/B тестирования:

1. Постановка цели. Тестирование необходимо подстроить в соответствии с целями рекламных кампаний. К примеру, увеличение продаж, повышение узнаваемости марки, привлечение целевой аудитории.
2. Выбор метрики для оценки эффективности рекламных кампаний. В одном тесте выбирается одна метрика.
3. Выбор параметра для тестирования. Каждый эксперимент оценивает одно изменение в отношении основной кампании.
4. Оценка объема аудитории. Тестирование проводят на отдельном сегменте аудитории.
5. Создание разных вариантов креативов.
6. Проведение эксперимента. На этом этапе нельзя прерывать эксперимент и вносить изменения.
7. Анализ результатов. Сравнение метрик и выбор варианта с наилучшими данными. Результаты теста могут отличаться незначительно. В этом случае нужно выбрать новый параметр и повторить алгоритм.

Глава 3. Сбор, анализ и визуализация данных тестирования

3.1 Подготовка к анализу данных, установка необходимых библиотек

В данной работе в качестве объекта анализа будет использована база данных с результатами A/B-тестирования объявлений, размещенная по ссылке <https://www.kaggle.com/datasets/osuolaleemmanuel/ad-ab-testing>. Эта база данных содержит CSV файл AdSmartABdata – AdSmartABdata.csv.

На сайте можно посмотреть описание столбцов базы данных:

* auction\_id: уникальный идентификатор онлайн-пользователя, которому была представлена ​​анкета. В стандартной терминологии это называется идентификатором показа. Пользователь может увидеть анкету BIO, но не отвечать на нее. В этом случае столбцы «да» и «нет» равны нулю.
* experiment: к какой группе относится пользователь — контрольной или выставленной.
* control: пользователи, которым была показана основная реклама;
* exposed: пользователи, которым был показан креатив.
* date: дата в формате ГГГГ-ММ-ДД.
* hour: час дня в формате ЧЧ.
* device\_make: название типа устройства, которое есть у пользователя, например, Samsung.
* platform\_os: идентификатор ОС пользователя.
* browser: имя браузера, который пользователь использует для просмотра рекламы.
* yes: 1, если пользователь выбирает переключатель «Да» для анкеты.
* no: 1, если пользователь выбирает переключатель «Нет» для анкеты.

Для анализа базы данных с помощью языка python будем использовать Jupyter Notebook. Jupyter Notebook — это среда разработки, где сразу можно видеть результат выполнения кода и его отдельных фрагментов.

Если не установлен python, для начала нужно загрузить с официального сайта <https://www.python.org/downloads/> и установить Python актуальной версии. После этого нужно открыть в командную строку и установить сам ноутбук. В этом поможет специальная утилита — pip. Чтобы установить Jupyter Notebook, нужно ввести команду pip install notebook. Если выводится сообщение Successfully installed, то Jupyter-ноутбук готов к запуску. Чтобы начать работу, необходимо ввести команду jupyter notebook. В результате откроется среда разработки (Рисунок 7).



Рисунок - Среда разработки Jupyter Notebook

Далее создадим папку проекта. Для этого нужно нажать на кнопку «New» и выбрать «Folder».



Рисунок - Создание папки проекта

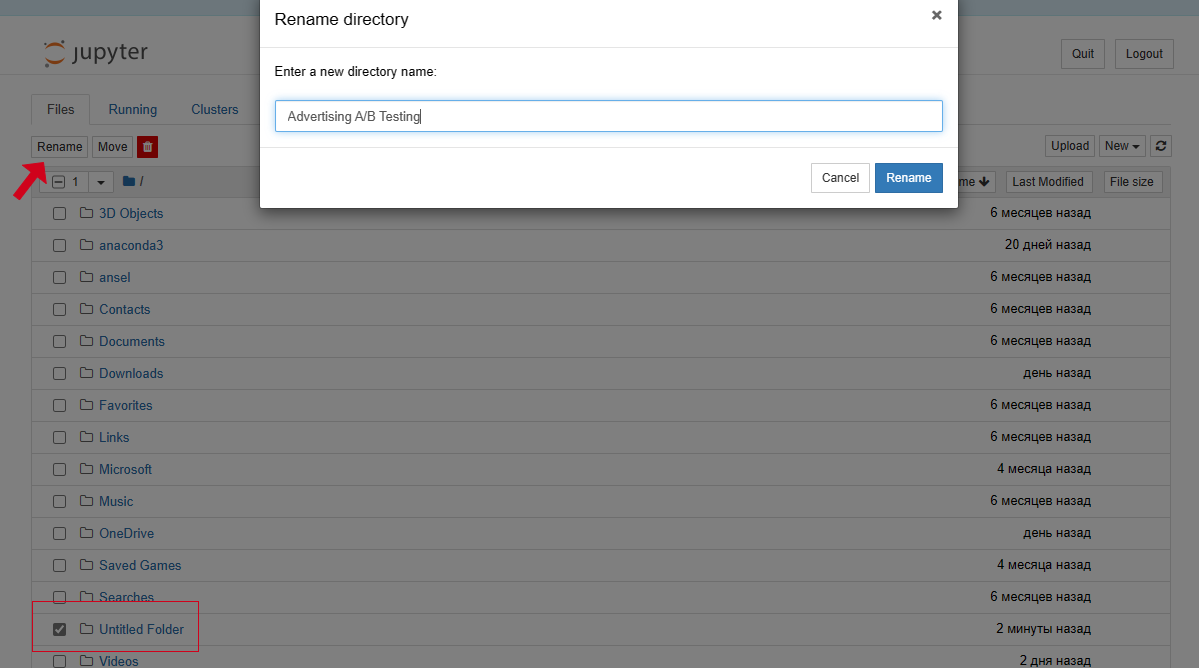
По умолчанию папка будет иметь название «Untitled folder». Чтобы ее переименовать, нужно выбрать ее, нажать на «Rename» и ввести новое название. 

Рисунок - Переименование папки проекта

Для дальнейшей работы необходимо загрузить ранее скачанный файл базы данных в созданную папку. Для этого необходимо нажать в меню «Upload», выбрать файл. После загрузки нужно снова нажать «Upload» в меню файла.

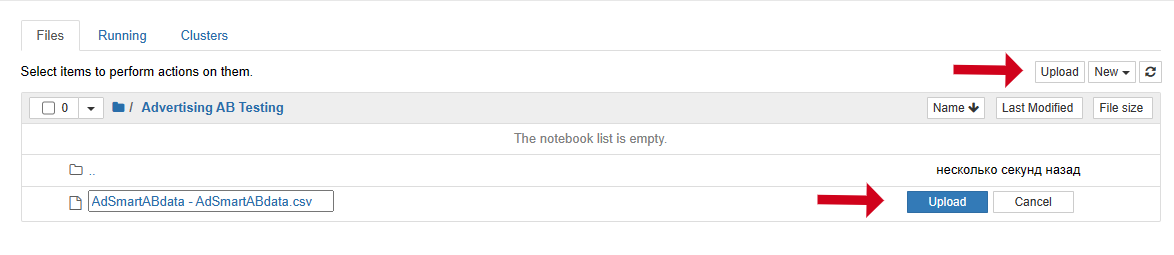


Рисунок - Загрузка базы данных

Чтобы создать ноутбук, нужно снова перейти в «New» и выбрать Python 3.

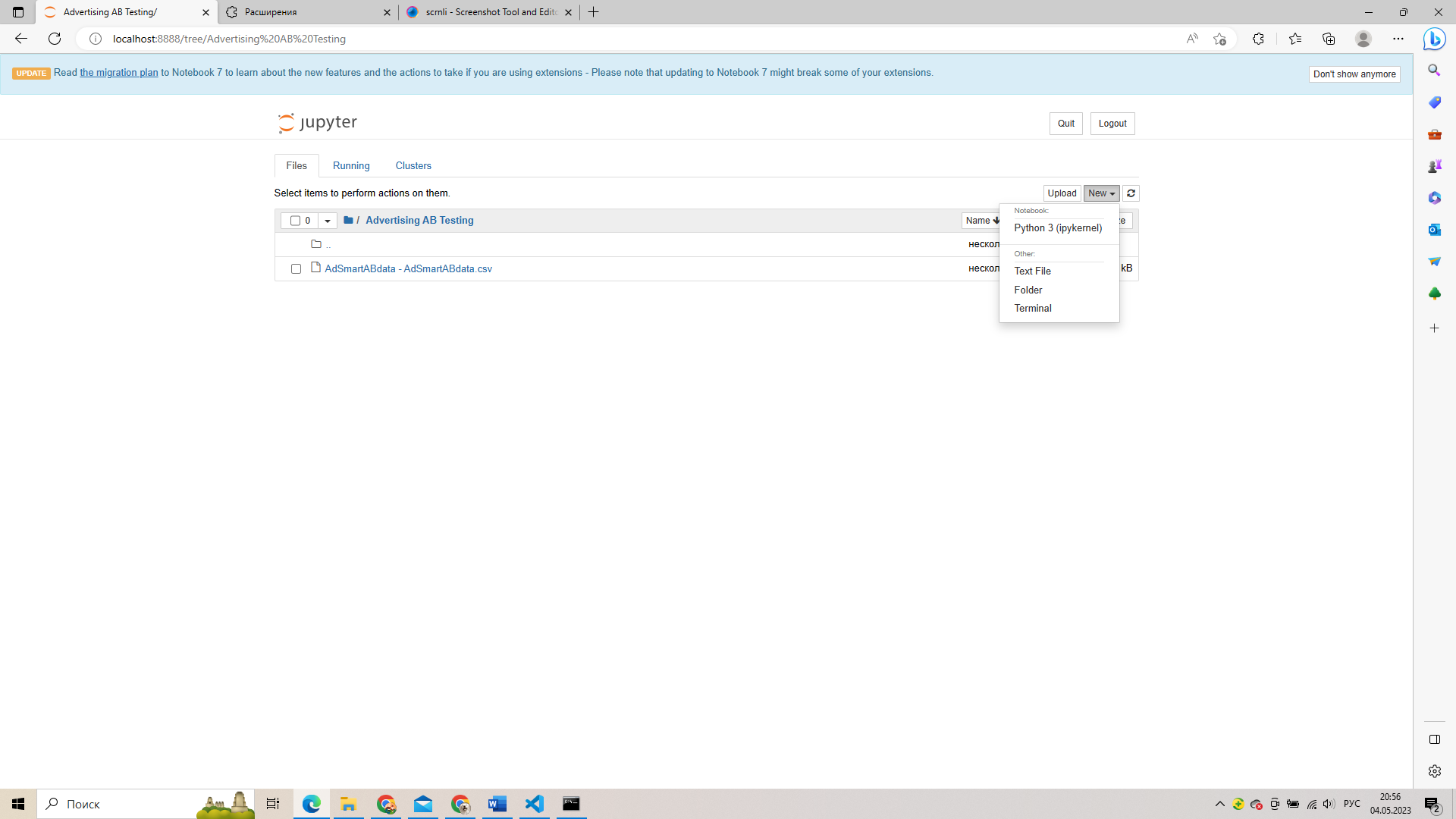


Рисунок - Создание файла ноутбука

Далее необходимо установить необходимые для анализа и визуализации библиотеки:

1. Pandas — это библиотека Python для обработки и анализа структурированных данных;
2. NumPy — это библиотека для работы с многомерными массивами, также включает набор математических функций, которые применяются над ними;
3. Seaborn — библиотека для создания статистических графиков на Python;
4. Matplotlib — библиотека на языке программирования Python для визуализации данных двумерной и трёхмерной графикой;
5. squarify для построения Treemap (древовидной карты);
6. scipy для стастического анализа. Подпакет scipy.stats содержит статистические распределения и функции

Для установки библиотеки нужно в уомандной строке ввести команду pip install название\_библиотеки (pip install squarify).

Далее нужно импортировать библиотеки в ноутбук с помощью команд:

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

import squarify

import scipy.stats as stats

from scipy.stats import norm

3.2 Первичный анализ и визуализация данных

3.2.1 Первичный анализ данных

DataFrame содержит столбцы auction\_id, experiment, date, hour, device\_make, platform\_os, browser, yes, no.

Рассмотрим, какие типы данных содержат столбцы с помощью df.info().

Датафрейм не имеет пустых значений. Столбцы auction\_id, experiment, date, device\_make, browser хранят типы данных object, столбцы hour, platform\_os, yes, no - int64.

Проверка на дубликаты записей. Введем команды:

num\_duplicates = sum(df.duplicated(subset=['auction\_id']))

print(f' Количество дубликатов: {num\_duplicates}')

Вывод команд: Количество дубликатов: 0.

Посмотрим, сколько уникальных значений в каждом столбце с помощью функции df.nunique(). В результате получаем данные, отображенные в таблице 1.

Таблица 1 - Уникальные значения столбцов

|  |  |
| --- | --- |
| auction\_id | 8077 |
| experiment | 2 |
| date | 8 |
| hour | 24 |
| device\_make | 269 |
| platform\_os | 3 |
| browser | 15 |
| yes | 2 |
| no | 2 |

Рассчитаем количество пользователей в каждой группе:

groups = df['experiment'].value\_counts()

groups

В результате в группе control 4071 пользователей, в группе exposed – 4006.

Построим на основе данных круговую диаграмму:

sns.set\_theme(style="whitegrid", palette='Set3')

plt.pie(groups.values, autopct='%.1f%%', labels=groups.index)

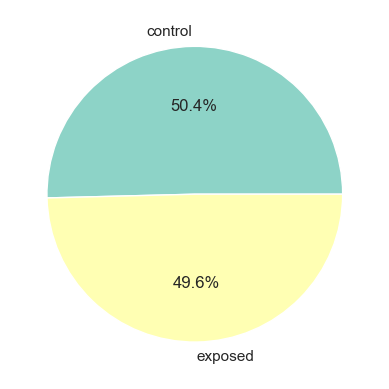


Рисунок 11 - Распределение пользователей в группах

3.2.2 Ответы на анкету

В базе нет столбца, показывающего отсутствие ответа на анкету, если оба столбца yes и no имеют значение 0. Добавим столбец response для явной пометки. Столбец response будет иметь значение «yes», если yes=1 и «no», если no=1. Если столбцы yes и no равны нулю, столбец response имеет значение «no response».

Для этого создадим функцию get\_response, которая принимает массив и возвращает 'yes', 'no' в зависимости от значений в соответствующих столбцах или 'no response', если в обоих столбцах 0.

def get\_response(row):

if row[0] == 1:

res = 'yes'

elif row[1] == 1:

res = 'no'

else:

res = 'no response'

return res

df['response'] = df[['yes', 'no']].apply(get\_response, axis=1)

Рассмотрим соотношение отвеченных и неотвеченных анкет. Для этого посчитаем количество по каждому значению с помощью value\_counts. Эта функция возвращает объект, содержащий количество уникальных значений.

no\_res = df['response'].value\_counts()

no\_res

Таблица 2 - Количество ответов

|  |  |
| --- | --- |
| **no response** | 6834 |
| **no** | 671 |
| **yes** | 572 |

Визуализируем полученные данные с помощью круговой диаграммы. Используем функцию pie библиотеки matplotlib.

sns.set\_theme(style="whitegrid", palette='Set3')

plt.pie(no\_res.values, autopct='%.1f%%', labels=no\_res.index);

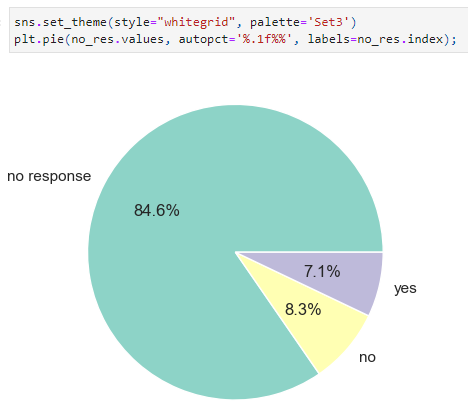


Рисунок 12 - Соотношение ответов

Как видно, преобладают неотвеченные анкеты - 84.6%.

Рассмотрим количество каждого ответа и неотвеченных анкет в каждой группе. Необходимо создать таблицу с кросс-табуляцией, отображающую частоту, с которой появляются определенные группы данных. Применим функцию crosstab(). В качестве данных используем колонки response и experiment.

data = pd.crosstab(df['response'], df['experiment']).reset\_index()

data.head()

Таблица 3 - Соотношение ответов на анкету в разных группах

|  |  |  |  |
| --- | --- | --- | --- |
| **experiment** | **response** | **control** | **exposed** |
| 0 | no | 322 | 349 |
| 1 | no response | 3485 | 3349 |
| 2 | yes | 264 | 308 |

Построим на основе полученных данных столбчатые гистограммы с помощью метода displot библиотеки seraborn.

sns.set\_theme(style="whitegrid", palette=['#fdb462','#80b1d3'])

ax = sns.displot(df.sort\_values(by='experiment'), x='response', col='experiment', hue='experiment', alpha=1)

ax.set\_ylabels('Кол-во')

ax.set\_xlabels('Ответы на анкету')

ax.set\_titles('Количество отвеченных и неотвеченных анкет')

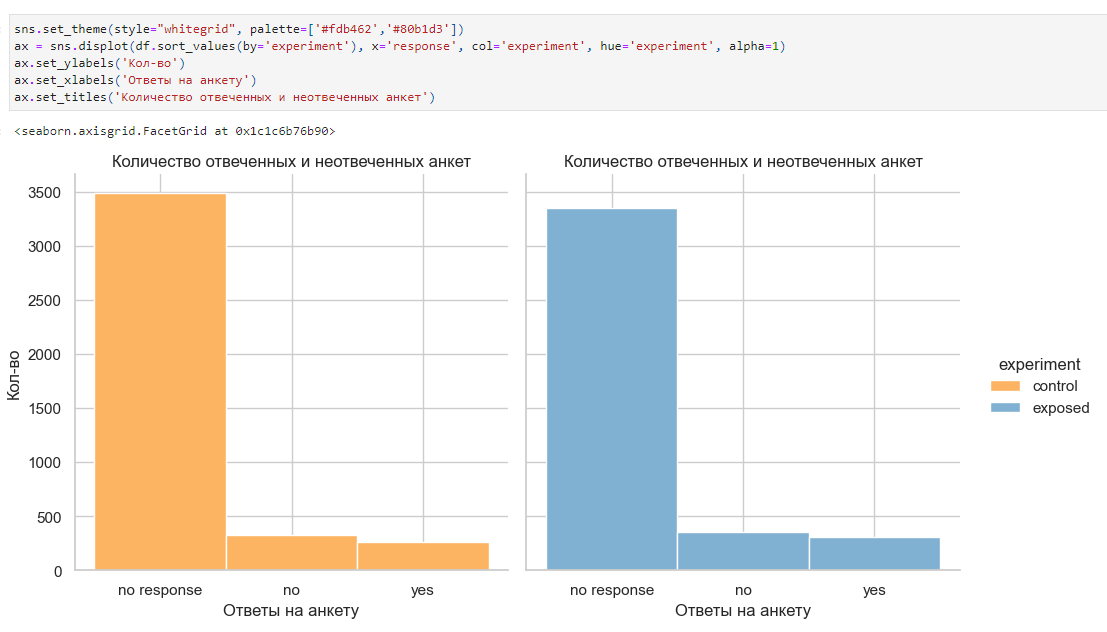


Рисунок 13 - Количество отвеченных и неотвеченных анкет в группах control и exposed

Посчитаем процентное соотношение каждого ответа на анкету и неотвеченных анкет в зависимости от группы. Для этого построим сводную таблицу с помощью pivot\_table. Создадим также функцию get\_category\_percent, которая будет рассчитывать проценты в каждой строке.

def get\_category\_percent(row, decimal=1):

return round(row/max(row)\*100, decimal)

df\_response = df.pivot\_table(index='experiment', columns='response', values='auction\_id', aggfunc='count', fill\_value=0, margins=True, margins\_name='Total')

df\_response = df\_response.apply(get\_category\_percent, axis=1)

df\_response.columns.name = 'response %'

df\_response = df\_response[['yes', 'no', 'no response']]

df\_response

В результате получаем сводную таблицу.

Таблица 4 - Сводная таблица по процентному соотношению ответов

|  |  |  |  |
| --- | --- | --- | --- |
| **experiment** | **yes** | **no** | **no response** |
| control | 6.5 | 7.9 | 85.6 |
| exposed | 7.7 | 8.7 | 83.6 |
| Total | 7.1 | 8.3 | 84.6 |

Из полученной таблицы видно, что в обоих группах количество неотвеченных анкет значительно выше, чем на которые ответили. Как мы видим, обычно (~83-85%) пользователи не отвечают на вопросы анкеты. Сходное количество пользователей (6,5–8,7 %) ответили «да» или «нет».

Рассмотрим соотношение положительных и отрицательных ответов в анкете в каждой группе. Для этого посчитаем соотношение ответов yes и no. Сначала очистим данные от 'no response'.

data\_response = df[(df['yes'] == 1) | (df['no'] == 1)]

exp\_response = data\_response[(data\_response['experiment'] == 'exposed')]['response'].value\_counts()

exp\_response

Построим круговую диаграмму.

sns.set\_theme(style="whitegrid", palette=['#fdb462','#80b1d3'])

plt.pie(exp\_response.values, autopct='%.1f%%', labels=exp\_response.index);

plt.title('Распределение ответов на анкету в группе exposed')

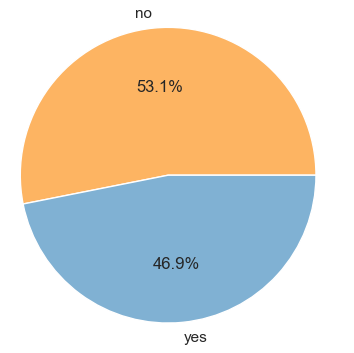


Рисунок - Распределение ответов на анкету в группе exposed

Так же посчитаем и построим диагарамму для группы control.

ctr\_response = data\_response[(data\_response['experiment'] == 'control')]['response'].value\_counts()

ctr\_response

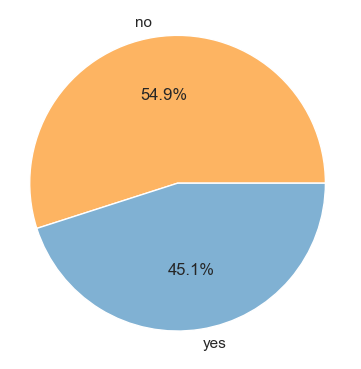


Рисунок - Распределение ответов на анкету в группе control

Распределение ответов в каждой группе одинаковое.

3.2.3 Анализ по дням

Определим длительность эксперимента.

df['date'].min()

Вывод: '2020-07-03'

df['date'].max()

Вывод: '2020-07-10'

Эксперимент длился 7 дней - с 03.07.2020 по 10.07.2020.

Построим график, отображающий показы рекламы по дням с помощью метода lineplot библиотеки seaborn.

sns.set\_theme(style="whitegrid", palette="Set3")

fig, ax = plt.subplots(figsize=(10,5))

g = sns.lineplot(data=df.groupby(['experiment', 'date'])['date'].count()['control'], ax=ax, label='control', color='#80b1d3', linewidth = 4)

g= sns.lineplot(data=df.groupby(['experiment', 'date'])['date'].count()['exposed'], ax=ax, label='exposed', color='#fdb462', linewidth = 4)

ax.set\_ylabel('Кол-во')

ax.set\_xlabel('Даты')

ax.set\_title('Количество пользователей в зависимости от даты')



Рисунок 16 - Количество пользователей по дням

В первый день эксперимента количество пользователей в контрольной группе было намного больше, чем в экспонированной группе. В другие дни количество пользователей было немного выше в экспериментальной группе.

Рассмотрим соотношение положительных и отрицательных ответов в зависимости от дня.

plt.figure(figsize=(10,8))

sns.set\_theme(style="whitegrid", palette=['#fdb462','#80b1d3'])

ax = sns.countplot(x="date", data=data\_response, hue='yes', order=['2020-07-03', '2020-07-04', '2020-07-05', '2020-07-06', '2020-07-07', '2020-07-08', '2020-07-09', '2020-07-10'] )

ax.set\_title('Количество положительных и отрицательных ответов по датам')

ax.set\_xlabel('Дни')

ax.set\_ylabel('Кол-во')

plt.legend(['Нет', 'Да'], title='Ответ')

plt.show()

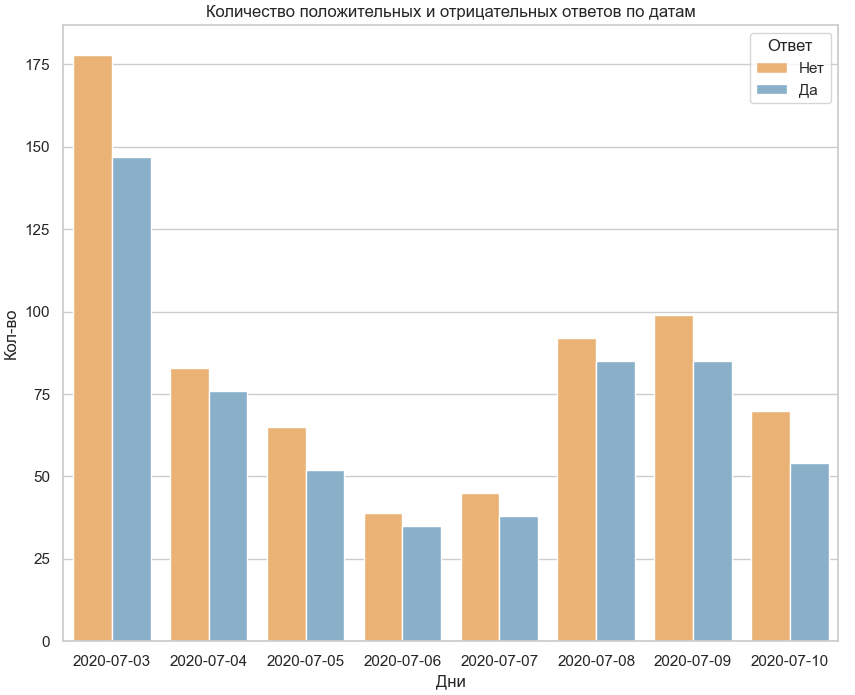


Рисунок 17 - Количество положительных и отрицательных ответам по дням

Построим гистограмму, отображающую распределение отвеченных и неотвеченных анкет в зависимости от дня.

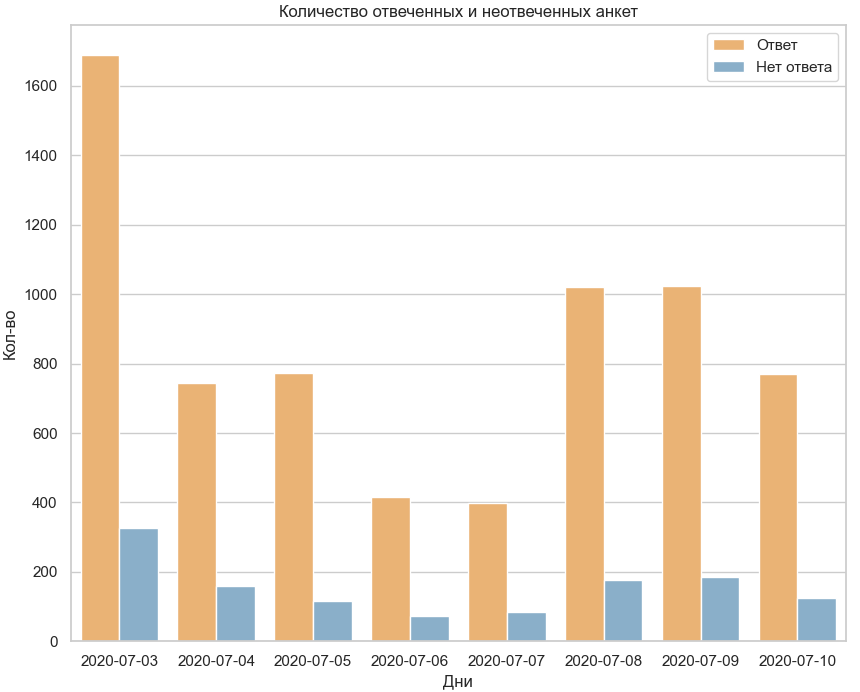


Рисунок 18 - Количество отвеченных и неотвеченных анкет

Как видно из гистограмм, во все дни преобладали отрицательные ответы на анкету, а также неотвеченные анкеты над отвеченными.

3.2.4 Исследование часов

Проведем исследование эксперимента по часам.

Рассмотрим количество показов анкеты в каждой группе по часам.

ax = sns.displot(df.sort\_values(by='experiment'), x='hour', col='experiment', bins=range(24), hue='experiment', palette=['#fdb462','#80b1d3'], alpha=1)

plt.text(-25, 1500, "Среднее = {:.0f} показов"\

.format(df[df['experiment']=='control']\

.groupby('hour')['experiment'].count().mean()))

plt.text(-27, 1100, "Аномальное количество в 15 ч:\n{:.0f} показов"\

.format(df[df['experiment']=='control']\

.groupby('hour')['experiment'].count().max()))

plt.text(0, 1500, "Среднее = {:.0f} показов"\

.format(df[df['experiment']=='exposed']\

.groupby('hour')['experiment'].count().mean()))

ax.set\_ylabels('Кол-во')

ax.set\_xlabels('Часы')

ax.set\_titles('Показы анкеты')

plt.show()

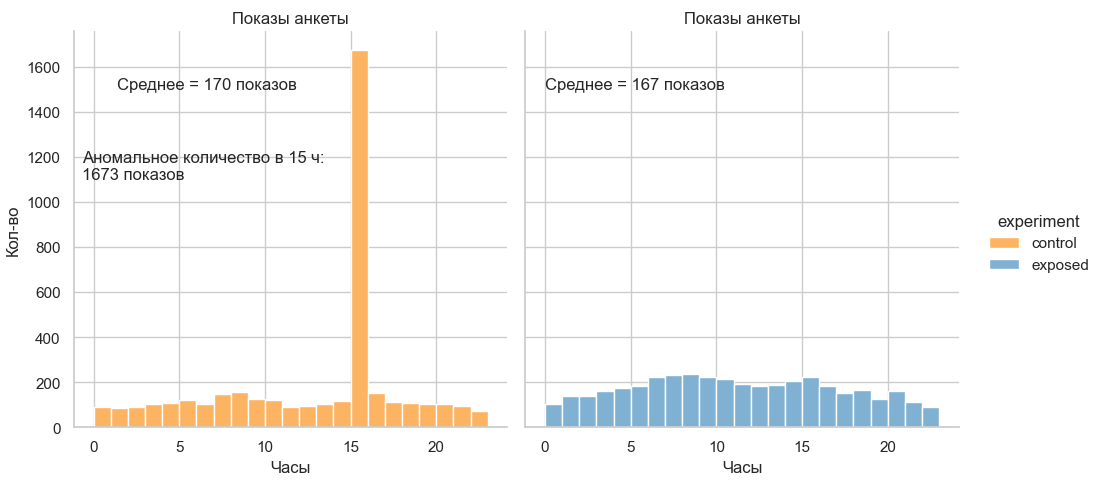


Рисунок 19 - Показы анкеты по часам

Построим графики показов анкеты по часам для каждого дня.

ax = sns.displot(df.sort\_values(by=['date','experiment']), x='hour', col='date', bins=range(24), hue='experiment', palette=['#fdb462','#80b1d3'], alpha=1, multiple="dodge", col\_wrap=3, facet\_kws=dict(sharey=True))

plt.ylim([0, 100])

ax.set\_ylabels('Кол-во')

ax.set\_xlabels('Часы')

plt.show()



Рисунок - Показы анкеты по часам для каждого дня

Имеются некоторые различия в часах между двумя группами в разные дни, но в первый день эксперимента мы получили экстремальные значения пользователей в контрольной группе в течение 15 часов.

3.2.5 Девайсы

Рассмотрим, какими девайсами использовались пользователями для просмотра анкеты. При первоначальном анализе данных выявлено, что всего 269 уникальных значений в столбце "device\_make". Выясним, сколько уникальных девайсов в выборке отвеченных анкет.

data\_response.device\_make.nunique()

В итоге 155 уникальных девайсов. Как видно, не все девайсы попали в эту выборку. Чтобы выявить уникальные значения, используем метод unique().

data\_response.device\_make.unique()

Для простоты дальнейшего анализа сгруппируем девайсы по основным маркам.

data\_response['general\_device'] = data\_response['device\_make'].apply(lambda x: x if x == 'Generic Smartphone' else x.split()[0])

data\_response.general\_device.nunique()

Узнаем, какой был самы популярные девайсы в каждой группе, с кототорго пользователи просматривали рекламу. Также посторим сводную таблицу по столбцам experiment и device\_make.

device = data\_response.pivot\_table(index='general\_device', columns='experiment', values='auction\_id', aggfunc='count', fill\_value=0)

device

Проанализируем получившиеся данные с помощью метода describe().

device.describe()

В итоге получаем сводную таблицу.

Таблица 5 - Сводная таблица по девайсам

|  |  |  |
| --- | --- | --- |
| **experiment** | **control** | **exposed** |
| **count** | 43.000000 | 43.000000 |
| **mean** | 13.627907 | 15.279070 |
| **std** | 55.426495 | 66.041591 |
| **min** | 0.000000 | 0.000000 |
| **25%** | 0.500000 | 1.000000 |
| **50%** | 1.000000 | 1.000000 |
| **75%** | 3.000000 | 3.500000 |
| **max** | 317.000000 | 402.000000 |

Видно, что самые популярным девайсом в контрольной группе пользовалось 317 пользователей и 402 в экспериментальной. При этом, отрыв самого популярного девайса от остальных очень большой - 75% не превышает 3 в контрольной группе и 3.5 - в экспериментальной.

Узнаем 10 самых популярных девайсов, с которых просматривали рекламу. Отсортируем результаты сначала по столбцу control, а затем по столбцу exposed. Представим полученные данные по каждой группе с помощью древовидной карты.

device\_ctr = device.sort\_values(by='control', ascending=False)[0:10][['control']] device\_ctr

Таблица 6 - Популярные девайсы в контрольной группе

|  |  |
| --- | --- |
| **general\_device** | **control** |
| Generic Smartphone | 317 |
| Samsung | 190 |
| iPhone | 14 |
| POT-LX1 | 6 |
| LG-$2 | 6 |
| XiaoMi | 5 |
| Nokia | 5 |
| MAR-LX1A | 5 |
| Pixel | 4 |
| ANE-LX1 | 4 |

Визуализируем полученные данные.

plt.figure(figsize=(12, 6))

squarify.plot(sizes=device\_ctr['control'], label=device\_ctr.index, alpha=.6, text\_kwargs={'fontsize': 12})

plt.axis('off')

plt.title('Популярные девайсы в контрольной группе', fontsize=14)

plt.show()

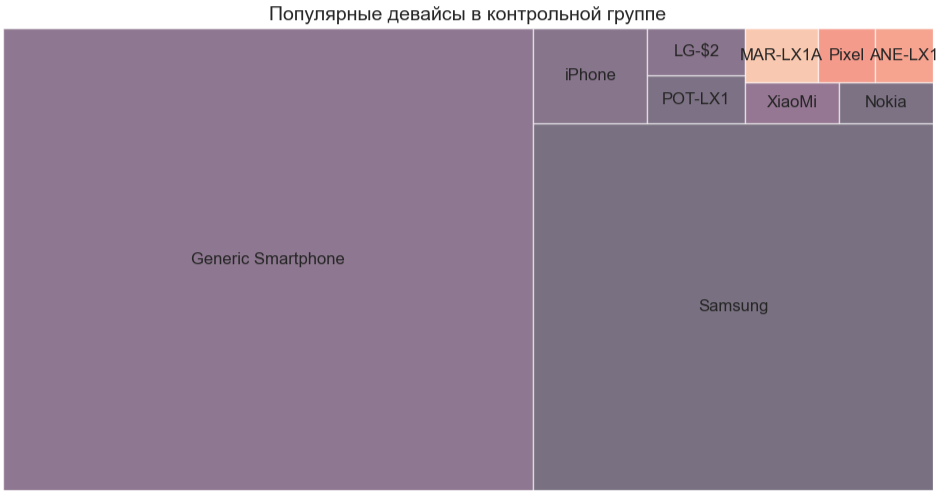


Рисунок 21 - Популярные девайсы в контрольной группе

Проведем аналогичный анализ для экспериментальной группы.

device\_exp = device.sort\_values(by='exposed', ascending=False)[0:10][['exposed']]

device\_exp

Таблица 7 – Популярные девайсы в экспериментальной группе

|  |  |
| --- | --- |
| **general\_device** | **exposed** |
| Generic Smartphone | 402 |
| Samsung | 177 |
| Pixel | 7 |
| Nokia223 | 6 |
| Nokia | 6 |
| I3312 | 5 |
| LG-$2 | 5 |
| iPhone | 4 |
| OnePlus | 4 |
| Moto | 4 |

plt.figure(figsize=(12, 6))

squarify.plot(sizes=device\_exp['exposed'], label=device\_exp.index, alpha=.6, text\_kwargs={'fontsize': 12})

plt.axis('off')

plt.title('Популярные девайсы в экспериментальной группе', fontsize=14)

plt.show()

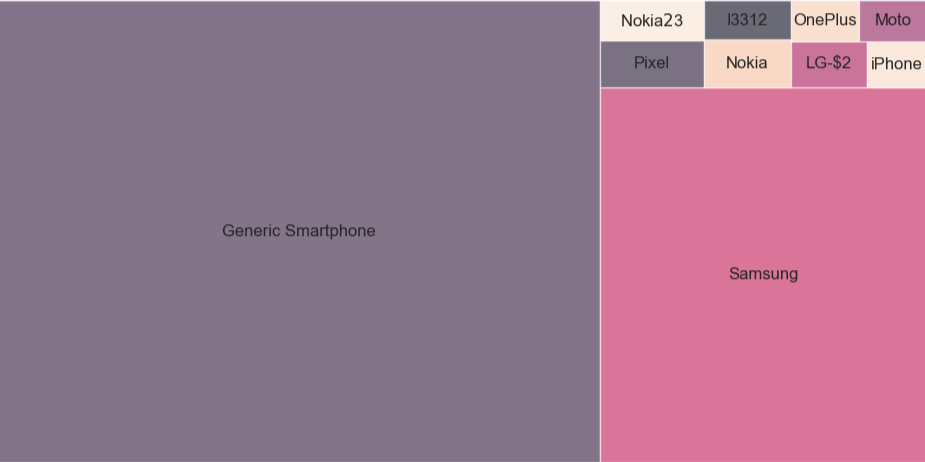


Рисунок 22 - Популярные девайсы в экспериментальной группе

Можем сделать вывод, что существенных отличий в используемых девайсах нет. Как видно, Generic Smartphone занимает наибольшую долю в общих данных, экспериментальной группе и контрольной группе. Второе место в обоих группах занимает Samsung. В контрольной группе iPhone находится на 3 месте, однако в экспериментальной он занимает 8 место и в этой группе его использовали в 3,5 раза реже.

Рассмотрим, с каких девайсов чаще положительно отвечали на анкеты.

device = data\_response.pivot\_table(index='general\_device', columns='response', values='auction\_id', aggfunc='count', fill\_value=0, margins=True)

device

Полученную сводную таблицу отсортируем по итоговому столбцу и выделим первые 10 строк.

device\_response = device.sort\_values(by='All', ascending=False)[0:10].drop(index=['All'])

device\_response

Таблица 8 - Сводная таблица по девайсам и ответам на анкету

|  |  |  |  |
| --- | --- | --- | --- |
| **general\_device** | **no** | **yes** | **All** |
| Generic Smartphone | 398 | 321 | 719 |
| Samsung | 187 | 180 | 367 |
| iPhone | 13 | 5 | 18 |
| LG-$2 | 2 | 9 | 11 |
| Nokia | 5 | 6 | 11 |
| Nokia | 4 | 7 | 11 |
| Nokia223 | 4 | 5 | 9 |
| POT-LX1 | 5 | 3 | 8 |
| XiaoMi | 3 | 3 | 6 |

Построим столбчатую многорядную диаграмму.

plt.figure(figsize=(10, 8))

sns.set\_theme(style="whitegrid", palette=['#fdb462','#80b1d3'])

n\_ticks = np.arange(len(device\_response.index))

offset = 0.1

w = 0.2

plt.bar(n\_ticks - offset, device\_response['yes'], width=w)

plt.bar(n\_ticks + offset, device\_response['no'], width=w)

plt.title('Ответы на анкету в зависимости от девайса')

plt.xlabel('Девайс')

plt.ylabel('Кол-во')

plt.legend(['Да', 'Нет'])

plt.xticks(n\_ticks, device\_response.index)

plt.xticks(rotation = 45)

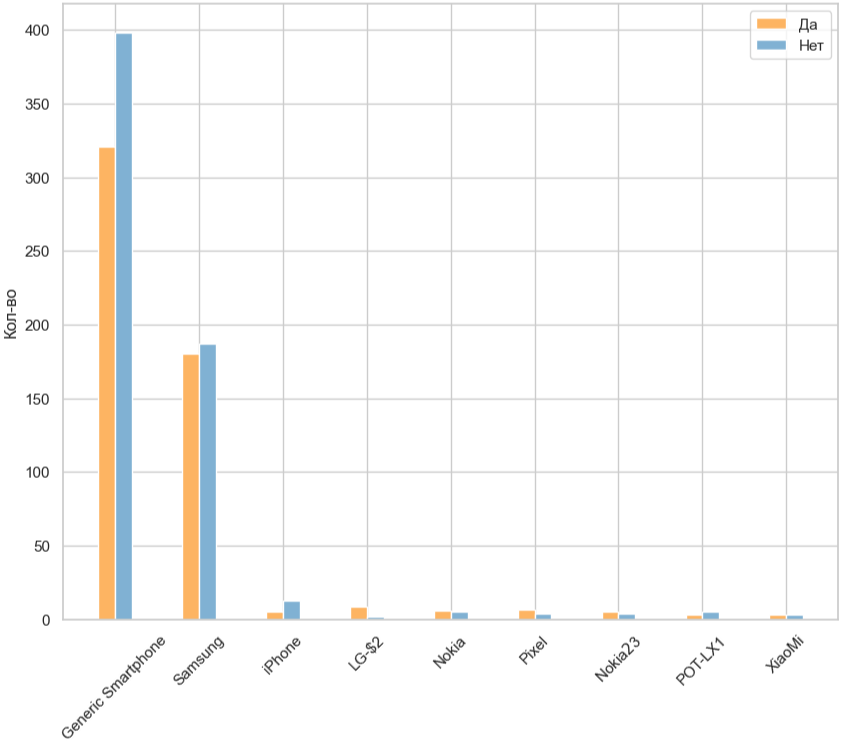


Рисунок - Ответы на анкету в зависимости от девайса

Как видно, чаще всего положительно на анкету отвечали пользователи Generic Smartphone, Samsung и LG-$2. Построим гистограммы для каждой группы.

plt.figure(figsize=(10,5))

ax = sns.histplot(data=data\_response.loc[df["experiment"] == "exposed"], x="general\_device", hue="response", shrink=.8, multiple="dodge", palette=['#fdb462','#80b1d3'])

ax.set\_ylabel('Кол-во')

ax.set\_xlabel('Девайс')

ax.set\_title('Девайсы и ответы на анкету в экспериментальной группе')

ax.set\_xticklabels(ax.get\_xticklabels(), rotation = 80)

plt.legend(['Да', 'Нет'])

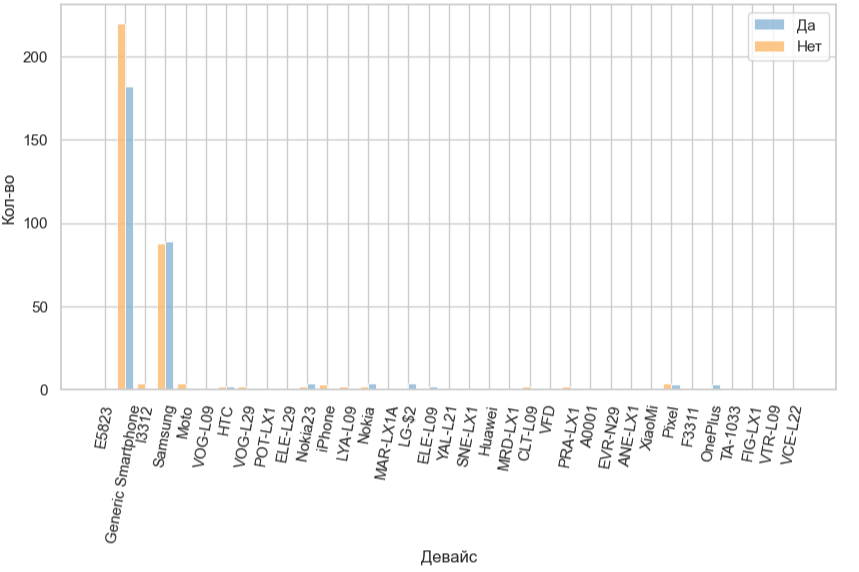


Рисунок 24 - Девайсы и ответы на анкету в экспериментальной группе

plt.figure(figsize=(10,5))

ax = sns.histplot(data=data\_response.loc[df["experiment"] == "control"], x="general\_device", hue="response", shrink=.8, multiple="dodge", palette=['#fdb462','#80b1d3'])

ax.set\_ylabel('Кол-во')

ax.set\_xlabel('Девайс')

ax.set\_title('Девайсы и ответы на анкету в контрольной группе')

ax.set\_xticklabels(ax.get\_xticklabels(), rotation = 80)

plt.legend(['Да', 'Нет'])

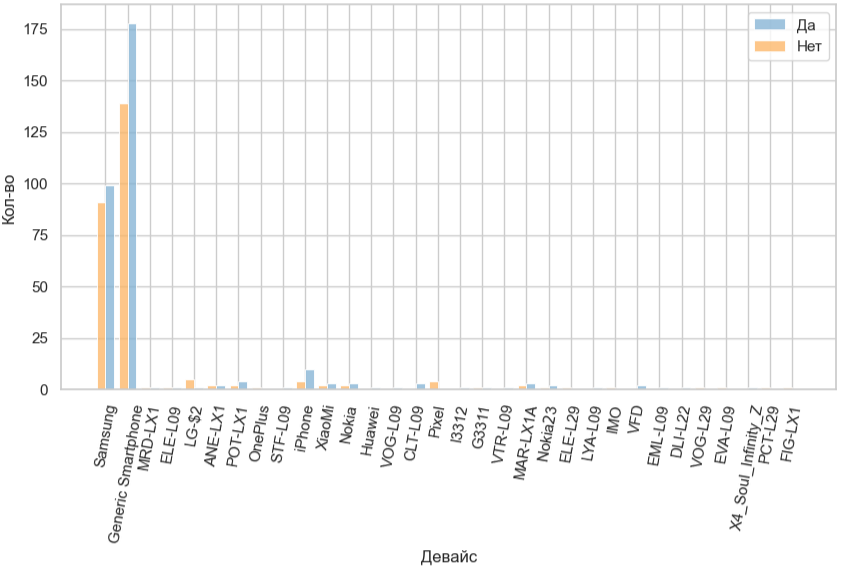


Рисунок 25 - Девайсы и ответы на анкету в контрольной группе

Можно заметить, что для Generic Smartphone и Samsung в контрольной группе преобладали положительные ответы.

3.2.6 Операционные системы

Рассмотрим, какими операционными системами пользовались в ходе эксперимента все пользователи и те, кто ответил на анкету.

df.platform\_os.unique()

Получаем массив array([6, 5, 7].

data\_response.platform\_os.unique()

Получаем массив array([6, 5].

Всего в базе 3 вида операционных систем - 6, 5 и 7. Пользователи с операционной системой 7 не отвечали на анкету.

Рассмотрим распределение используемых операционных систем в каждой группе среди отвеченных анкет.

os = data\_response.pivot\_table(index='platform\_os', columns='experiment', values='auction\_id', aggfunc='count', fill\_value=0)

os

Таблица 9 - Операционные системы в каждой группе среди отвеченных анкет

|  |  |  |
| --- | --- | --- |
| **platform\_os** | **control** | **exposed** |
| 5 | 13 | 4 |
| 6 | 573 | 653 |

В обоих группах преобладает platform\_os = 6. Визуализируем полученные данные с помощью столбчатой диаграммы.

ax = sns.displot(data\_response.sort\_values(by='experiment'), x='platform\_os', col='experiment', hue='experiment', palette=['#fdb462','#80b1d3'], alpha=1).set(xticks=list(range(5,7)))

ax.set\_ylabels('Кол-во')

ax.set\_xlabels('Операционная система')

ax.set\_titles('Используемые операционные системы')

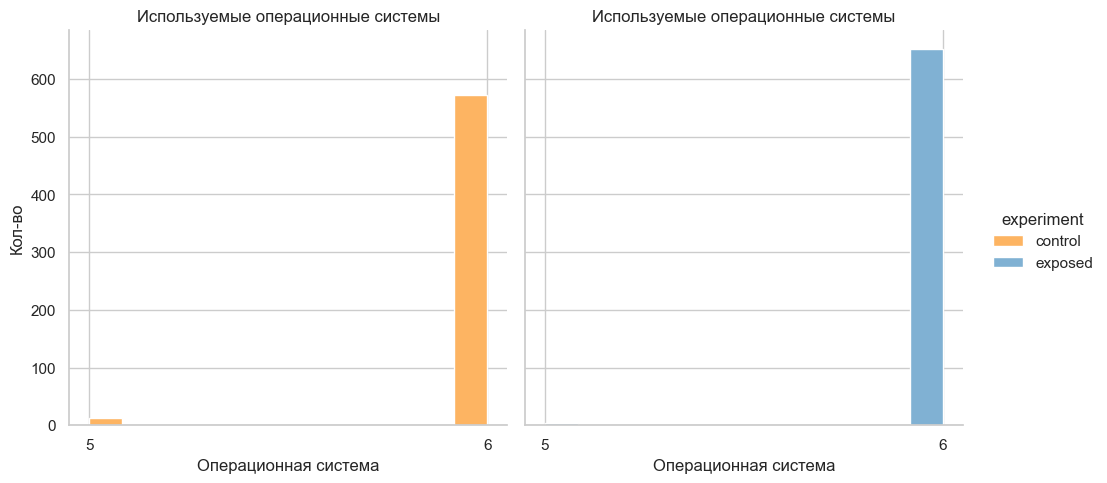


Рисунок - Операционные системы в каждой группе

3.2.7 Браузеры

Выясним, сколько всего разных браузеров в базе данных и в выборке с отвеченными анкетами.

df.browser.nunique()

data\_response.browser.nunique()

Всего 15 разных браузеров и 8 из них попали в выборку с отвеченными анкетами.

Рассмотрим браузеры, с которых отвечали на анкету.

data\_response.browser.unique()

В итоге получаем следующий массив:

array(['Chrome Mobile WebView', 'Chrome Mobile', 'Facebook',

'Samsung Internet', 'Mobile Safari', 'Chrome',

'Mobile Safari UI/WKWebView', 'Chrome Mobile iOS'], dtype=object)

Построим сводную таблицу по используемым браузерам для отвеченных анкет и используем метод describe.

browser\_response = data\_response.pivot\_table(index='browser', columns='experiment', values='auction\_id', aggfunc='count', fill\_value=0, margins=True)

browser\_response

Таблица 10 - Сводная таблица по используемым браузерам

|  |  |  |  |
| --- | --- | --- | --- |
| **browser** | **control** | **exposed** | **All** |
| Chrome | 1 | 1 | 2 |
| Chrome Mobile | 324 | 371 | 695 |
| Chrome Mobile WebView | 47 | 180 | 227 |
| Chrome Mobile iOS | 1 | 0 | 1 |
| Facebook | 112 | 44 | 156 |
| Mobile Safari | 10 | 4 | 14 |
| Mobile Safari UI/WKWebView | 3 | 0 | 3 |
| Samsung Internet | 88 | 57 | 145 |
| All | 586 | 657 | 1243 |

browser\_response.describe()

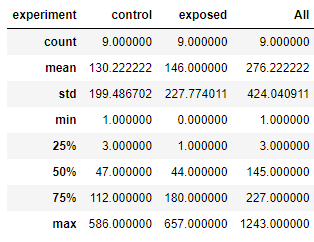


Рисунок 27 - Описание сводной таблицы по браузерам

Как видно, в группе exposed минимальное значение равно 0. Браузеры Chrome Mobile iOS и Mobile Safari UI/WKWebView не использовались в экспонированной группе. Есть существенная разница в обоих группах между самыми популярными браузерами и остальным. в контрольной группе 75% браузеров встречалось не более 112 раз, в экспонированной - 180.

Объединим браузеры в основные группы. Для этого напишем функцию group\_browser, которая будет принимать строку и возвращать «Chrome» или «Safari», если строка содержит соответственно это слово.

def group\_browser(row):

if 'Chrome' in row:

return 'Chrome'

elif 'Safari' in row:

return 'Safari'

else:

return row

data\_response['general\_browser'] = data\_response['browser'].apply(group\_browser)

Снова посмотрим уникальные значения браузеров.

data\_response.general\_browser.unique()

В итоге получаем:

array(['Chrome', 'Facebook', 'Samsung Internet', 'Safari'], dtype=object)

Выявим соотношение используемых браузеров и определим самые популярные из них. Воспользуемся сводной таблицей.

browser = data\_response.pivot\_table(index='general\_browser', columns='experiment', values='auction\_id', aggfunc='count', fill\_value=0, margins=True)

browser

Таблица 11 - Сводная таблицам по браузерам и группам

|  |  |  |  |
| --- | --- | --- | --- |
| **general\_browser** | **control** | **exposed** | **All** |
| Chrome | 373 | 552 | 925 |
| Facebook | 112 | 44 | 156 |
| Safari | 13 | 4 | 17 |
| Samsung Internet | 88 | 57 | 145 |
| All | 586 | 657 | 1243 |

В обоих группах самый популярный браузер Chrome. В контрольной группе на втором месте Facebook, но в экспериментальной группе он использовался в 2,5 раза реже. Построим гистограмму распределения используемых браузеров.

plt.figure(figsize=(10,5))

ax = sns.histplot(data=data\_response, x="general\_browser", hue="experiment", shrink=.8, multiple="dodge", palette=['#fdb462','#80b1d3'])

ax.set\_xlabel('Кол-во')

ax.set\_ylabel('Браузер')

ax.set\_title('Распределение используемых браузеров')

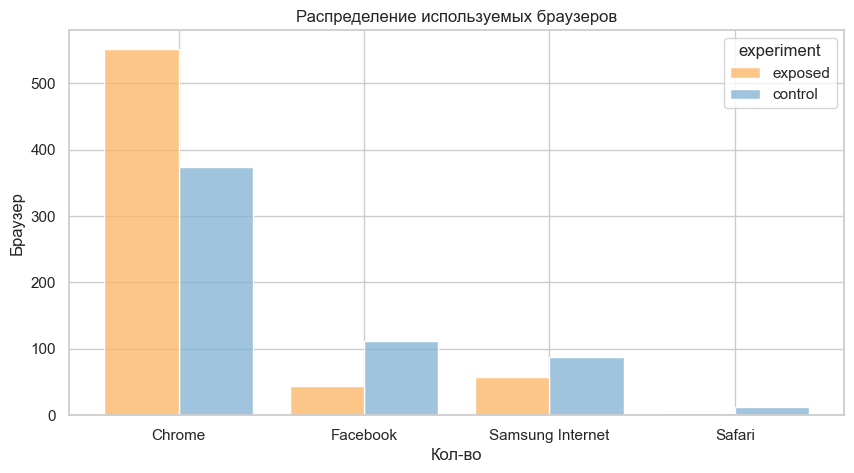


Рисунок 28 - Распределение используемых браузеров в группах

Также рассмотрим, какое было распределение браузеров в зависимости от ответа на анкету с помощью сводной таблицы.

browser = data\_response.pivot\_table(index='general\_browser', columns='response', values='auction\_id', aggfunc='count', fill\_value=0)

browser

Таблица - Сводная таблица по браузерам и ответам на анкету

|  |  |  |
| --- | --- | --- |
| **general\_browser** | **no** | **yes** |
| Chrome | 493 | 432 |
| Facebook | 87 | 69 |
| Safari | 13 | 4 |
| Samsung Internet | 78 | 67 |

Для всех браузеров преобладал отрицательный ответ. Рассмотрим соотношение ответов на анкету для каждого браузера в экспериментальной и контрольной группе.

plt.figure(figsize=(10,5))

ax = sns.histplot(data=data\_response.loc[df["experiment"] == "control"], x="general\_browser", hue="response", shrink=.8, multiple="dodge", palette=['#fdb462','#80b1d3'])

ax.set\_ylabel('Кол-во')

ax.set\_xlabel('Браузер')

ax.set\_title('Браузеры и ответы на анкету в контрольной группе')

plt.legend(['Да', 'Нет'])

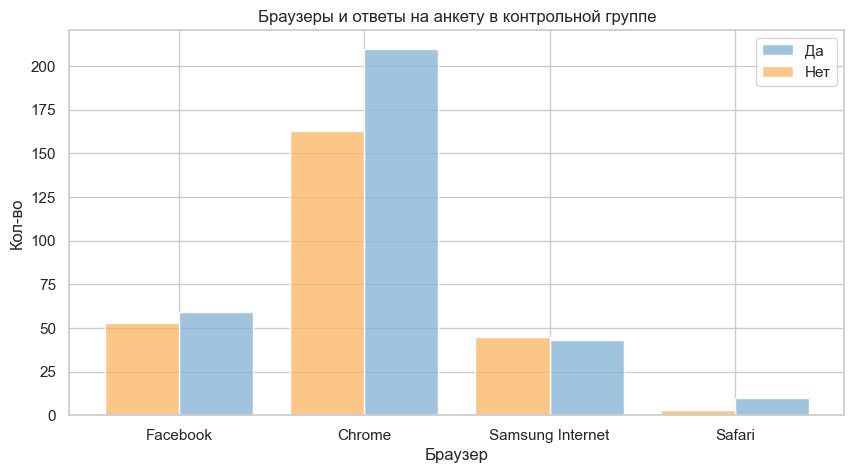


Рисунок 29 - Браузеры и ответы на анкету в контрольной группе

plt.figure(figsize=(10,5))

ax = sns.histplot(data=data\_response.loc[df["experiment"] == "exposed"], x="general\_browser", hue="response", shrink=.8, multiple="dodge", palette=['#fdb462','#80b1d3'])

ax.set\_ylabel('Кол-во')

ax.set\_xlabel('Браузер')

ax.set\_title('Браузеры и ответы на анкету в экспериментальной группе')

plt.legend(['Да', 'Нет'])

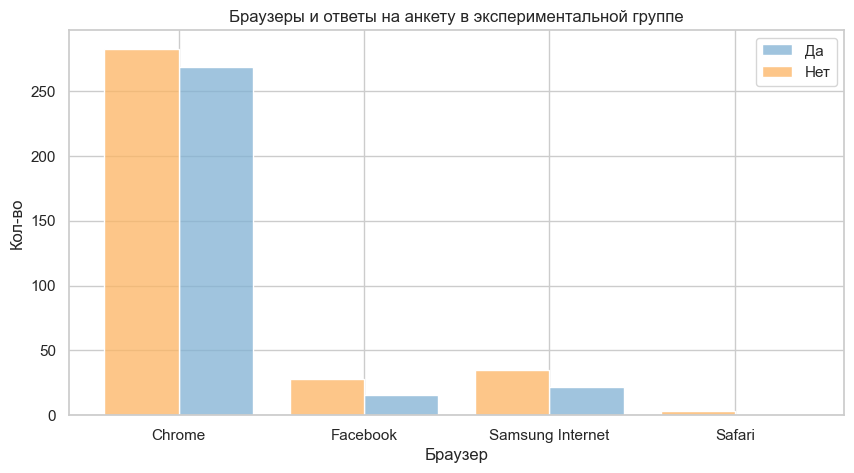


Рисунок 30 - Браузеры и ответы на анкету в экспериментальной группе

Можно сделать вывод, что чаще всего положительно отвечали на анкету в обоих группах при просмотре с браузеров Chrome, Facebook и Samsung Internet. В контрольной группе для браузера Chrome преобладают положительные ответы. В экспериментальной группе для всех браузеров преобладают отрицательные ответы.

3.2.8 Подведение итогов первичного анализа базы данных тестирования

Можно сделать следующие выводы:

1. Преобладают неотвеченные анкеты в обоих группах - обычно (~83-85%) пользователи не отвечают на вопросы анкеты. Разницы в ответах на анкету в обоих группах не наблюдается - 45-47% приходится на ответ "yes" и 55-53% - на ответ "no".
2. Размеры контрольной и экспонированной групп аналогичны.
3. В течение всего эксперимента, кроме первого дня, количество пользователей было немного выше в экспериментальной группе.
4. На 03.07.2020 в 15 часов в контрольной группе было аномальное количество пользователей (1673), хотя среднее значение равно 170.
5. Самыми популярными девайсами являются Generic Smartphone, Samsung. Существенных отличий в группах по девайсам нет.
6. Чаще всего пользователи в обоих группах использовали операционную систему 6.
7. Самые популярные браузеры: Chrome, Samsung, Facebook и Safari.

3.3 Статистический анализ

Чтобы понять, как новая реклама меняет коэффициент конверсии, мы должны оценить, значительна ли разница между результатами старой и новой рекламы.

3.3.1 Формулировка гипотезы

**Нулевая гипотеза Hₒ: p = pₒ** — нет существенной разницы между показателем успешности рекламы в обеих группах.

**Альтернативная гипотеза Hₐ: p ≠ pₒ** — существует значительная разница между показателями успешности рекламы в обеих группах.  
Поскольку мы не знаем, будет ли новый дизайн работать лучше/хуже/равно нашему текущему дизайну, мы проведем двусторонний тест.  
Уровень достоверности: 95% (α=0,05)  
p и pₒ обозначают коэффициент конверсии нового и старого дизайна.

3.3.2 Критерий Фишера

**Обзор**

Точный критерий Фишера можно использовать, чтобы показать, как переменные различаются между группами.

Есть несколько предположений для теста Фишера:

1. Случайная выборка - в каждой группе точка данных должна быть из случайной выборки.
2. Независимость - каждое наблюдение должно быть независимым.
3. Взаимоисключающие группы — каждая категория должна быть взаимоисключающей.

Все эти предположения справедливы для нашего случая:

1. Все пользователи из каждой группы случайны без какой-либо предвзятости,
2. Каждый пользователь принял самостоятельное решение,
3. Ответ взаимный — либо «да», либо «нет».

**Расчеты**

contingency = pd.pivot\_table(data=data\_response, index='experiment', columns='response', aggfunc='count')['auction\_id']

contingency

Таблица 13 - Сводная таблица по ответам в группах

|  |  |  |
| --- | --- | --- |
| **experiment** | **no** | **yes** |
| control | 322 | 264 |
| exposed | 349 | 308 |

odds\_ratio, p\_value = stats.fisher\_exact(contingency, alternative="two-sided")

print('P-значение теста Фишера: {:.2f}'.format(p\_value))

Результат: P-значение теста Фишера: 0.53

**Вывод:**

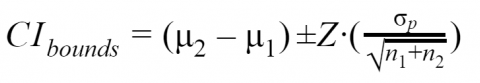
* P-значение теста Фишера равно 0,53, поэтому мы не можем отвергнуть гипотезу Hₒ.
* Результат: нет существенной разницы между коэффициентом конверсии в контрольной и экспонированной группах.
  + 1. Доверительный интервал

**Обзор**

Мы можем оценить A/B-тестирование, используя доверительные интервалы.

Предел погрешности рассчитывается на основе предполагаемого стандартного отклонения параметра, умноженного на Z-оценку, соответствующую выбранному уровню достоверности.

Границы доверительного интервала для абсолютной разницы между двумя средними значениями рассчитываются по следующему уравнению:



где:

μ1 – среднее значение базовой или контрольной группы.  
μ2 – среднее значение группы обработки.  
n1 – размер выборки базовой или контрольной группы.  
n2 – размер выборки группы обработки.  
σp – объединенное стандартное отклонение двух групп.  
Z является оценочной статистикой, соответствующей желаемому уровню достоверности.

Границы интервала зависят от:

* 1. Случайная ошибка, вызванная наблюдаемой разницей в средних значениях;
  2. Ошибка в оценке стандартного отклонения указанной разницы.

**Расчеты**

* Рассчитать z-показатель для доверительного интервала 95%;
* По сути, z-показатель для определенного интервала определяет, насколько мы должны отклоняться от среднего значения нормального распределения в обоих направлениях, чтобы покрыть 95% значений данных вокруг среднего.

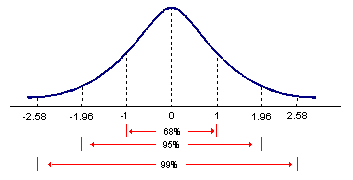


Рисунок 31 - График доверительного интервала

zscore = norm.ppf(1 - .05/2)

print('Z-показатель = {:.2f}'.format(zscore))

Z-показатель = 1.96

Расчет среднего значения для обеих групп:

e\_mean = data\_response.query('experiment=="exposed"')['yes'].mean()

c\_mean = data\_response.query('experiment=="control"')['yes'].mean()

Теперь необходимо рассчитать объединенную дисперсию для двух групп.

e\_std = data\_response.query('experiment=="exposed"')['yes'].std()

c\_std = data\_response.query('experiment=="control"')['yes'].std()

e\_num = data\_response.query('experiment=="exposed"')['yes'].shape[0]

c\_num = data\_response.query('experiment=="exposed"')['yes'].shape[0]

var\_pooled = (((e\_num-1)\*e\_std\*\*2 + (c\_num-1)\*c\_std\*\*2)/(e\_num+c\_num-2))\*\*0.5

Расчет верхней и нижней границы доверительного интервала:

CI\_low = (e\_mean - c\_mean) - zscore \* var\_pooled \* (1/e\_num + 1/c\_num)\*\*.5

CI\_high = (e\_mean - c\_mean) + zscore \* var\_pooled \* (1/e\_num + 1/c\_num)\*\*.5

**Построение графика**

Узнаем биномиальное распределение на основе рассчитанных групповых средних:

e\_binom = np.random.binomial(e\_num, e\_mean, 100000)/e\_num

c\_binom = np.random.binomial(c\_num, c\_mean, 100000)/c\_num

diff\_binom = e\_binom - c\_binom

Рассчитаем нормальное распределение:

norm\_dist = np.random.normal(0, np.std(diff\_binom), len(diff\_binom))

Построим на основе полученных данных график.

z = np.arange(-3,3, 0.1)

plt.figure(figsize=(6,4))

plt.plot(z, norm.pdf(z), label = 'Стандартное нормальное распределение', linewidth = 2.5, color ='#80b1d3')

plt.fill\_between(z[z>zscore], norm.pdf(z[z>zscore]), label = 'Правая область отклонения', color ='#fdb462' )

plt.fill\_between(z[z<(-1)\*zscore], norm.pdf(z[z<(-1)\*zscore]), label = 'Левая область отклонения', color ='#fdb462' )

plt.title("95%-ный доверительный интервал")

plt.legend(loc='upper right', bbox\_to\_anchor=(1.8, 0.8))

plt.show()

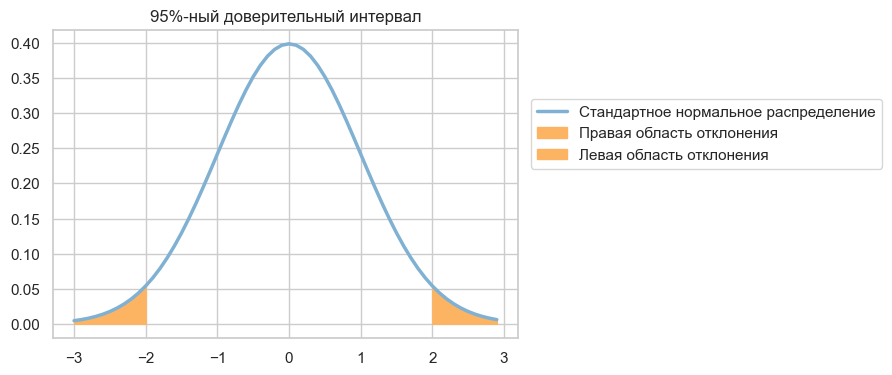


Рисунок 32 - 95%-ный доверительный интервал

Рассчитаем 95%-ный доверительный интервал:

print('95%-ный доверительный интервал: [{:.2f}, {:.2f}]'.format(CI\_low, CI\_high))

Результат:

95%-ный доверительный интервал: [-0.04, 0.07]

Доверительный интервал говорит нам, что с 95%-ной уверенностью разница между двумя средними находится между [-0,04, 0,07], что означает, что мы не можем отвергнуть нулевую гипотезу.

Мы можем сделать вывод, что нет существенной разницы между показателями конверсии для старой и новой рекламы.

* + 1. Подведение итогов статистического анализа

По результатам проведенного статистического анализа А/В теста с точным критерием Фишера и доверительным интервалом можно сделать вывод, что новый дизайн рекламы не изменил коэффициент конверсии.

* 1. Подведение итогов тестирования и предложения по настройке рекламной кампании

После того, как мы удалили из набора данных наблюдения без ответов (столбцы «yes» и «no» == 0), у нас осталось только 1243 наблюдения (что составляет 15,4%). Эта значительная потеря данных приводит к отсутствию наблюдений, чтобы гарантировать обнаружение значительных различий.

Тот факт, что P-значение теста Фишера равно 0,53 и намного выше установленного порога (95%, α = 0,05), означает, что мы не можем отвергнуть «нулевую гипотезу» (Hₒ: p = pₒ), которая утверждает, что нет существенной разницы между двумя группами.

Это указывает на то, что различия между «фиктивной рекламой», показанной «контрольной» группе, и «креативной рекламой», показанной «экспонированной» группе, не приводят к повышению эффективности рекламы. Эти результаты показывают, что нет веских бизнес-причин для внедрения нового дизайна рекламы вместо старого, поскольку это не принесет дополнительных преимуществ.

Пользователи, использующие Chrome в экспериментальной и контрольной группах, показывают разные пропорции «да» и «нет». Только в контрольной группе для браузера Chrome преобладали положительные ответы на анкету.

Также мы наблюдали, что есть значительные отрывы в показателях для самых популярных устройств и браузеров от остальных.

Эти результаты показывают, что интерактивная онлайн-реклама с брендом SmartAd может быть более эффективной для определенных девайсов и браузеров, и что таргетинг на эти конкретные типы устройств и браузеры может привести к более высокому уровню вовлеченности.

Заключение

Проведен анализ литературы по применению визуализации данных как метода для передачи данных или информации с помощью их кодирования как визуальных объектов (например, точки, линии или столбцы), содержащихся в графике.

Рассмотрены средства для визуализации данных, их преимущества и недостатки.

Изучены инструменты анализа и визуализации данных с помощью библиотек python.

Проведен анализ тестовой базы данных, содержащей результаты A/B-теста, проведенного рекламным агентством.

Сделаны выводы по проведенному A/B-тестированию рекламной кампании на основе статистического анализа и предложены рекомендации по настройке рекламы для увеличения конверсии.

Список литературы

Аганина Р. Н. Андронова Т. А. Интернет-реклама в эпоху цифровизации [Статья]. - М. : Вестник Университета имени О. Е. Кутафина, 2020 г.. - 7 (71), с. 8.

Антошина Т. Н. Применение визуализации данных в автоматизированных информационных системах [Статья]. - СПб. : Научно-аналитический журнал «Вестник Санкт-Петербургского университета Государственной противопожарной службы МЧС России», 2020 г.. - 2, с. 4.

Беляев Н. А. Визуализация данных: инфографика как инструмент маркетинга [Статья]. - Челябинск : [б.н.], 2015 г.. - Вестник ЮУрГУ. Серия «Экономика и менеджмент». - Т. 9, с. 5.

Визуализация данных — что это: виды, способы и средства визуального представления информации [В Интернете] // Блог Алексея Колоколова. - https://alexkolokolov.com/ru/blog/vizualizaciya-dannyh-chto-eto.

Визуализация данных: применение в работе, основные принципы, способы и инструменты для использования [В Интернете] // Яндекс Практикум. - https://practicum.yandex.ru/blog/vizualizaciya-dannyh/#id4.

Демкина О. В. Марушко Л. Ю. Анализ возможности использования веб-аналитики в управлении доходностью организации [Статья]. - М. : Московский экономический журнал, 2019 г.. - 9, с. 9.

Захаров Л.И. Сбор и визуальный анализ больших объемов данных, с помощью инструментов python и power bi.. - Казань, с. 30 : Институт математики и механики им. Н. И. Лобачевского, 2018 г..

Как и зачем использовать Python для анализа данных [В Интернете] // pythonru. - https://pythonru.com/baza-znanij/python-dlya-analiza-dannyh.

Как провести А/Б тестирование в Директе [В Интернете] // Яндекс. - https://yandex.ru/adv/news/1544.

Поисковые и контекстные объявления [В Интернете] // Яндекс. - https://yandex.ru/adv/products/platforms/text-image-ads.

Прохоров А. В. Владимирская Е. Н. Концептуальная модель мультиагентной системы управления контекстной рекламой [Статья]. - Харьков : Восточно-Европейский журнал передовых технологий, 2010 г.. - 9(46), с. 9.

Процесс анализа данных [В Интернете] // https://pythonru.com/. - https://pythonru.com/baza-znanij/process-analiza-dannyh.

Разработка модели прогнозирования эффективности контекстной рекламы [Отчет].

Фетисова М. С. Информационно-аналитическая система поддержки принятия решений в области рекламы. - Барнаул, с. 68 : Международный институт экономики, менеджмента и информационных систем, 2019 г..

Приложение

[Ссылка на репозиторий с файлом jupyter notebook](https://github.com/AnnaDyadkina/graduate_work)