Методология тестов по магазинам

Структура документа

Почему вообще нам нужна какая-то другая методология для проведения тестов на магазинах?

Как минимум, в случае теста по магазинам не выполняется одно из ключевых предположений, необходимое для проведения A/B теста –– i.i.d. (independence and identical distribution) наблюдений (то есть analysis unit'ов). Продемонстрируем на примере, но сначала уточним терминологию:

- Randomization unit – объекты, по которым проводится рандомизация
- Analysis unit – объекты, по которым считается метрика (например, среднее, в случае обычного A/B)

Пример 1:

Допустим, мы проводим A/B на юзерах и хотим оценить эффект на количество заказов.

В таком случае, randomization unit – это юзер (мы рандомно делим юзеров на тест и контроль), analysis unit (/level) – тоже юзер. Мы смотрим изменение в метрике количество заказов на юзера. Сравниваем среднее количество заказов на юзера в тестовой и в контрольной группах.

В таком эксперименте выполнение предпосылки о том, что наблюдения вида "user - количество заказов" i.i.d. довольно вероятно.

Пример 2:

Допустим, мы проводим по-магазинный A/B и хотим оценить эффект на целостность.

Randomization unit – магазин (мы рандомно делим магазины на тест и контроль), analysis unit (/level) – купленный товар. Наши наблюдения имеют вид "line_item - была ли отмена/замена". Итоговую оценку эффекта мы получаем, усредняя метрику целостности по всем товарам, заказанным из из магазинов тестовой группы, и по всем товарам, заказанным из магазинов контрольной группы, а затем сравнивая полученные метрики между собой.

Разумно предположить, что отмены/замены не будут независимы внутри одного магазина, а также то, что их распределение будет отличаться между магазинами. Таким образом, i.i.d. assumption в данном случае не выполняется.

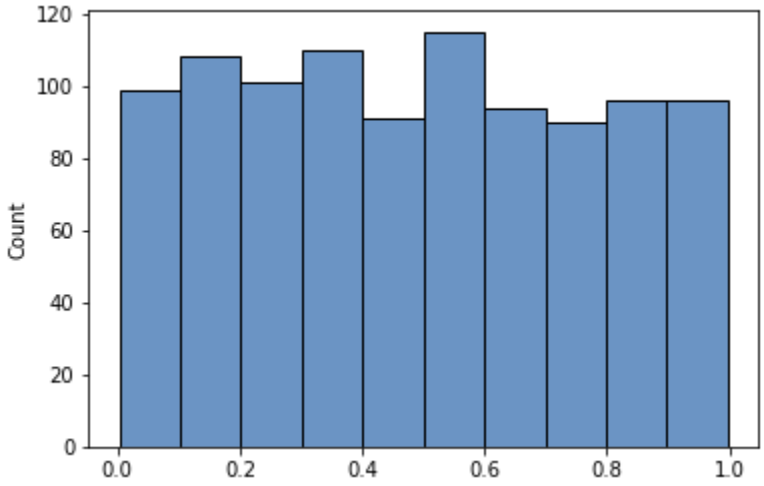
Если мы пренебрежем этим фактом, то при оценке дисперсии эффекта мы рискуем получить заниженное значение, и, как следствие, повышенную вероятность ложного прокрашивания теста.

Ложные прокрашивания тестов

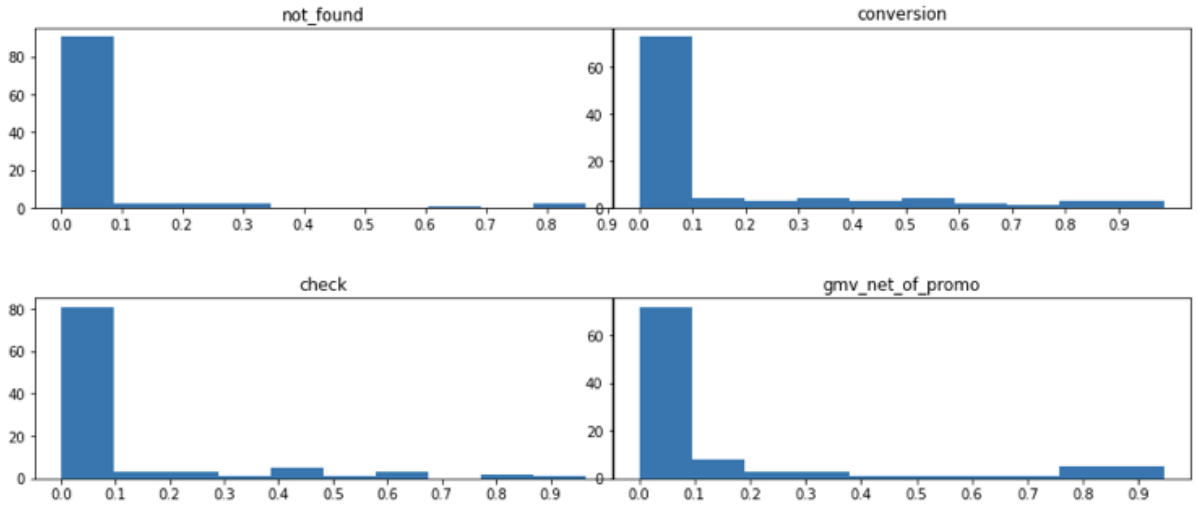
Вероятность ложного прокрашивания вероятность ошибки первого рода вероятность отвергнуть нулевую гипотезу в пользу альтернативной, когда нулевая верна.

В теории, вероятность ошибки первого рода (альфа) равна трешхолду на p-value. То есть, если мы считаем эксперимент прокрасившимся, если p-value гипотезы меньше 0.05, то вероятность ложного прокрашивания равна 0.05. Это следует из того, что в случае выполнения нулевой гипотезы p-value (при выполнении требуемых assumptions) распределен равномерно (maths, если вы любитель).

Мы можем проверить, что это верно и на практике. Возьмем генеральную совокупность юзеров и рандомно разделим ее на тестовую и контрольную группы. Подождем две недели и с помощью t-test'а сравним средние между тестом и контролем. Полученный p-value сохраним. Повторим эту процедуру 1000 раз для разных разбивок на тест и контроль. Таким образом мы смоделируем 1000 экспериментов (A/A тестов), для которых мы точно знаем, что нулевая гипотеза о равенстве средних выполняется. Нет никакого изолированного воздействия на одну из групп. Построим распределение полученных p-values (1000шт) и убедимся, что эмпирическое распределение соответствует теоретическому.



А теперь повторим ту же самую процедуру, но вместо юзеров будет рандомизировать магазины. Метрики будем считать на уровне заказов/товаров.



Получим вот такое распределение p-value. Что это значит?

Допустим, мы хотим провести эксперимент по магазинам и в качестве трешхолда на p-value выбирает 0.1. Согласно теории, при таком пороге мы должны иметь долю ложных прокрашиваний в районе 10%. По факту, доля ложных прокрашиваний составляет 60-80%. Это означает, что даже если тестируемая фича вообще не работает и никак не влияет на флоу, в 60-80% случаев эксперимент покажет, что имеет место статистически значимый эффект.

Окей, мы убедились, что as-is эксперименты по магазинам никуда не годятся. Давайте разберемся, что с этим можно сделать.

Как проводить эксперименты по магазинам корректно

Уровень наблюдений

Проблема, связанная с несовпадением randomization unit и analysis unit, была описана выше. Мы можем ее частично решить. Идея решения заключается в том, чтобы "поднять" уровень метрик до уровня магазина. Сделать это можно как минимум двумя способами:

1. С помощью линеаризации
2. С помощью суммирования

Линеаризация работает в теории, но на практике это не очень удобно. При линеаризации наши наблюдения соотносятся с магазинами 1:1, а сама линеаризованная метрика в случае магазинов довольно волатильная. Такой формат метрик оказывается очень требовательным к количеством магазинов и размеру эффекта, иначе говоря, линеаризованные метрики в по-магазинных тестах не очень чувствительны. Как бы то ни было, нам все равно придется их применять, поскольку не все метрики корректно агрегируются с помощью суммирования.

Метрики на уровне магазин-день

На текущий момент рабочим способ является агрегация метрик до уровня магазин-день путем суммирования. Таким способом мы не можем оценить часть метрик (например, средний чек), однако большинство метрик считается вполне корректно.

Как перейти к таким метрикам

Пример:

- GMV. Допустим, изначально мы собрали данные по заказам из всех магазинов, участвующих в эксперименте. Затем мы группируем наши метрики по магазинам и дням (датам), а GMV заказов суммируем.

По итогу мы получаем **(количество экспериментальных дней)** x **(количество магазинов)** наблюдений. Это благоприятно сказывается на мощности нашего эксперимента.

Почему это работает

Трансформируя метрики таким образом, мы сильно снижаем зависимость наших наблюдений, но не настолько сильно (как линеаризация) снижаем количество наблюдений.

На что обратить внимание

- Далеко не все метрики можно трансформировать таким образом. Точнее говоря, это можно делать только с теми метриками, в которые уже так или иначе заложен трафик.
    - Пример, как делать не нужно: если мы трансформируем с помощью суммирования отмены-замены, и во время эксперимента в тестовой группе станет больше заказов (/больше позиций в заказе), то такая метрика прокрасится даже при отсутствии эффекта на **долю** отмен-замен. То есть, иначе говоря, агрегированная метрика в этом случае даже не обязательно сонаправленна изначальной метрике.
- Подобная трансформация повышает корректность нашего эксперимента с точки зрения независимости наблюдений, но не делает его абсолютно корректным. Наши наблюдения все еще скорелированы внутри магазина (у нас >1 наблюдения на каждый магазин), и это тоже следует учитывать при оценке статзначимости (например, используя скорректированные ошибки в регрессии, см. cov_type="hac_panel" в statsmodels).
- Есть и другой аспект. Подобная трансформация дает нам дополнительную гибкость. В зависимости от ваших метрик, вы можете использовать метрики более (или менее) гранулированные по времени. Например, группировать не на уровне магазин-день, а на уровне магазин-час.

Рандомизация

В случае проведения по-юзерного теста мы совсем не думаем о рандомизации. Всю работу за нас выполняет A/B платформа и эконометрическая теория, согласно которой при большом размере выборки и случайном разделении на тест и контроль группы в среднем не будут отличаться друг от друга ни по наблюдаемым, ни по ненаблюдаемым признакам.

Если мы хотим делать тест по магазинам, то генерировать разбиение на тестовую и контрольную группу нам приходится самостоятельно. Да и простая рандомизация нам не поможет: из-за того, что выборки в разы меньше, а магазины гораздо меньше похожи друг на друга, чем юзеры, простая рандомизация зачастую будет давать очень несбалансированные выборки (тест и контроль будут сильно отличаться друг от друга еще до начала эксперимента). Это приводит и к ложных прокрасам и к неверным оценкам эффекта (bias).

Как рандомизировать

- Стратификация (подробнее тут)
- KNN и подобные методы (подробнее тут, раздел "подбор похожих групп")

Оценка эффекта и статистической значимости

И "хитрая" рандомизация, и трансформация метрик "лечат" наши по-магазинные эксперименты, но не до конца. У нас все еще остается некоторая автокорреляция в наблюдениях (внутри магазина), распределения метрик разнится в зависимости от магазина, группы не идеально похожи друг на друга, а метрики не настолько чувствительны, как бы нам хотелось. Поэтому обычный t-test для оценки эффекта нам подходит.

Как корректно оценивать эффект и статистическую значимость

1. Используем регрессии с корректировкой стандартных ошибок (тык, смотрим в раздел Notes: нам подходит "hac-panel", "cluster")
2. Используем CUPED для повышения чувствительности (тык)

3. Multilevel modeling (тык)

Что делать дальше

Мы обсудили основные проблемы, которые возникают при проведение экспериментов на магазинах. Причем на каждую проблему мы предлагаем >1 подхода. Итоговый дизайн эксперимента мы собираем из этого конструктора. Как убедиться, что полученный дизайн – корректен и удовлетворяет нашим требованиям: на долю ложных прокрашиваний, мощность, а также отсутствие bias?

Фреймворк

Вспомните подраздел этого текста про ложные прокрашивания. Там частично описана методология, которую мы и будем применять для того, чтобы понять, хороший ли наш дизайн или нет. То, насколько наш эксперимент хорош, мы будем описывать с помощью трех метрик:

- Ошибка первого рода (alpha)
- Мощность (1 - beta)
- Bias

Иначе говоря, мы хотим знать, с какой вероятностью наш дизайн обнаруживает эффект, когда его на самом деле нет **(alpha)**, с какой вероятностью наш дизайн обнаруживает эффект, когда он на самом деле есть **(power = 1-beta),** а также то, насколько оцененный эффект в среднем отклоняется от реального (**bias**).

Как тестировать дизайн эксперимента

Общая идея оценки дизайна эксперимента такая: дизайн эксперимента в базовом случае можно описать тремя метриками доля ложных прокрашиваний (FPR), мощность (power), степень смещение оценки эффекта (bias). Все эти метрики мы сможем оценить с помощью исторических данных и синтетических A/B тестов.

Допустим, наш дизайн выглядит следующим образом:

1. мы стратифицированно разбиваем магазины на тест и контроль (используя данные за последние две недели)
2. затем в течение двух недель проводим эксперимент
3. затем с помощью CUPED'a и линейной регрессии с cov_type="cluster" оцениваем эффект и статистическую значимость.

A/A тесты на исторических данных

Сначала оценим вероятность ошибки первого рода (FPR). Для этого:

1. Выбираем n дат. Эти даты будут соответствовать датам начала наших исторических A/A тестов и синтетических A/B. Ограничение на эти даты следующие: каждая дата начала ≤ today() - 4 недели (две недели на сбор данных для стратификации (observational period) и две недели – длительность псевдо-эксперимента (experimental period))
2. Для каждой даты начала эксперимента генерируем q сплитов (разбивок на тест и контроль) с помощью стратификации.
3. Для каждого сплита собираемые данные за следующие 2 недели после момента начала соответствующего эксперимента.
4. Для каждого сплита оцениваем эффект и статистическую значимость с помощью CUPED'a и линейной регрессии с cov_type="cluster" на двух неделях, соответствующих "экспериментальному периоду".

Таким образом мы получаем n*q p-values и n*q точечных оценок эффекта. Строим распределения. Распределение p-values должно быть равномерным, а распределение точечных оценок ~нормальным со средним 0.

FRP для фиксированного уровня значимости x% оцениваем как x-й квантиль p-values.

Bias можем оценить как средний оцененный эффект - средний фактический эффект (0 в данном случае).

Синтетические A/B тесты на исторических данных

Для описания нашего дизайна осталось оценить его мощность. Мощность будем оценивать для эффектов конкретного размера.

В синтетических A/Б тестах мы будем "накидывать" на нашу экспериментальную группу синтетический эффект. Делать это можно мультипликативно или аддитивно. Для начала нужно определиться, каков наш ожидаемый эффект (в процентах). Допустим, мы ожидаем, что наша фича увеличит GMV на 2%.

Если мы хотим накидывать эффект мультипликативно (это проще, но менее реалистично), то нам нужно будет умножать метрику (GMV, в нашем случае) в наблюдениях тестовой группы на (1 + ожидаемый эффект).

Если мы хотим накидывать эффект аддитивно, то нам понадобится оценить параметры распределения нашего эффекта. Эффект мы будем моделировать нормальным распределением, поэтому нам нужно оценить его м.о. и дисперсию.

Будем переиспользовать сплиты, которые мы сгенерировали в пунктах 1-3 раздела **A/A тесты на исторических данных**. Рандомно выберем некоторые из них и по ним посчитаем среднее и дисперсию нашей метрики (GMV) (по наблюдениям соответствующей гранулярности). Тогда наш эффект мы будем сэмплировать из нормального распределения со средним = средний GMV * ожидаемый эффект (в процентах) и дисперсией = 1/100 (или 1/10) дисперсии GMV.

Теперь у нас есть распределение эффекта по всем метрикам, для которых мы хотим оценить мощность нашего эксперимента, а также n*q сплитов, сгенерированных ранее.

Осталось провернуть пункт 4 из раздела **A/A тесты на исторических данных,** но в каждом таком синтетическом эксперименте мы будем приплюсовывать к метрикам тестовой группы сэмплы из нормального распределения с соответствующим средним и дисперсией. Таким образом мы синтетически сгенерируем данные, которые ожидаем увидеть в реальном эксперименте.

По итогу этих махинаций у нас на руках будет, как в предыдущем разделе, n*q p-values и n*q точечных оценок эффекта.

Мощность для данного размера эффекта и уровня значимости мы оценим, посчитав долю p-value < уровня значимости (например, долю p-value < 0.05).

Bias можем оценить как (средний оцененный эффект - средний фактический эффект) (средний фактический в данном случае – это просто математическое ожидание распределения эффекта, который мы накидывали). Для большей интепретируемости мы можем посчитать bias в процентах как (средний оцененный эффект - средний фактический эффект)/средний фактический эффект.

Вуаля!

По итогам всех этих хитрых и не очень манипуляций мы можем описать наш дизайн эксперимент с помощью FPR и мощности для фиксированного уровня значимости, а также bias.

В среднем, хорошим экспериментом считается эксперимент со следующими характеристиками: FPR = 5%, мощность = 80% на уровне значимости 5%, bias = 0.

На практике перевод метрик дизайна в “хороший дизайн” / “плохой дизайн” может зависить от многих обстоятельств.

Примеры с кодом

TBA, пока можно посмотреть на эти два документа (но обязательно на оба, а не только на первый 🙂) и вложенные ноутбуки:

Дизайн ухудшающего A/B теста OOS [2022]

Результаты ухудшающего теста out-of-stock модели [февраль-март 2022]