经典案例分析--A/B test

在互联网公司中，如何通过A/B test对产品升级，如何用数据来形成决策？

**Case:**

At YouTube, recently we’ve conducted a research and found that video playtime negatively correlates with observed latency of the loading page. What’s your recommendation to the product team?

**Framework:**

1. Clarify context

2. Make hypothesis

3. Design experiment

4. Make recommendation

Step 1: Clarify Context.

In academic world, problems are mostly predefined. Business questions are not. Several things to consider:

- What?

- Why? (is it worth the effort?)

Sample questions:

- How do you define latency?

- Why watch time matters?

- Scope of this problem?

- Global or specific country?

- How many users are impacted?

Step 2: Make hypothesis

What do you think might be the reason?

- First all, assume causality (higher latency ->less likely to watch)

- Justify with one or several valid reason

- Next job is to propose A/B test to test your hypothesis

Step 3: Design Experiment 具体步骤

假设是啥 一句话讲明白：

Higher latency is bad user experience and is preventing user from watching. So, if we reduce latency, our users may watch more videos.

Now let’s setup a A/B test to prove our hypothesis. We have 3 steps for this…

3.1 Determine metric (通过做实验，对用户要收集的数据是什么？从数学上来讲，是population mean, 要做出判断的是关于population整体用户的一个性质)

3.2 Design experiment（实验本身包含很多设置，针对哪些用户确定实验样本量，用power反推至少需要多少样本量；分几种不同的测量方法，比如直接衡量每一天用户的表现、或者选定同样的一批用户，来track同一批用户在不同时间的表现，这两种不同的设计可以分别更好地回答某类问题；design principle）

3.3 Interpret results（跑完实验得出结果，计算t-test，如何解释test本身的意义，结果的意义）

Recommended learning

By Diane Tang, head of Data Science at Google

<https://www.udacity.com/course/ab-testing--ud257>

Step 3.1: Define metrics

Select one “success metric” that your experiment will be measured against.

Sometimes we call it “north star” metric 对新的产品或start up，对metrics的设计可能并不完善，学会如何选定指标。

AARM framework for product metrics（可以套用在任何产品，任何服务上。讲述产品的生命周期。新用户到老用户经历的过程）（想象一个漏斗）

通过用户使用产品的生命周期这四个阶段，帮助你brainstorm，如何通过每个阶段来量化、衡量这个产品的表现、健康度。

**A**cquisition: User “lazy” signs up to a website or service. （虚有指标，不要过分关注，只反映获取用户的能力，不反应用户对产品的价值）

**A**ctivation: User takes some key actions to be “active”.（认为定义一个用户行为，该行为反映了用户对从产品的价值，更精确的衡量在该平台会比较活跃的，更有价值的用户数量）（Facebook定义为用户加够7个好友才算是活跃、激活的用户，如果没达到这个milestone，该用户基本上不会回来。）

**R**etention: User continues to use the service.（因其他同类产品吸引走了，或者产品不好对用户没有长久吸引力。Retention不好的话，对公司成本很高。用户走掉很多就要花更多的钱来更aggressive去的拉新用户，或花钱拉回走掉的用户。（拉走掉的用户大部分时候成本比拉新用户更高，但互联网产品拉用户的渠道差不多，都是网上发coupon之类，差别不太大；对非互联网企业，尤其是针对大客户，企业型客户，走掉了很难回来，更该多花心思去留住已经进来的用户。 企业用户：例如zoom的客户几乎都是企业，电话会议。）第二当产品到达一个阶段，市场上新用户已榨干，增长速度小，到达平台期，老用户又流失，用户数量会达到拐点，由上升变为下降，产品gg。）（用户留存率。Facebook的实例：在注册第七天加够了7个好友的用户留存率高达百分之80，没有加够7个好友的用户留存率很低。于是作出假设：如果我们能让用户在第七天之前加够7个人，就能提高retention。 Facebook launch了一个出名的new feature：一注册账号，为你推荐10个你可能认识的人。此举获得了好几百的percentage gain，将留存率提高4倍）设计时间点，根据产品本身的周期，在某段时间内，有多少用户在第一次使用之后又回来过，重复回访产品，就可以说该用户被retain了。 关于周期，是由产品属性决定的。 比如Youtube，视频软件类，7天或30天比较合适。并不期望用户使用产品的频率天天都看。Facebook，社交网络类，因为人有社交属性，可能要看用户一天的留存 daily active user。理论上希望每个用户每天都来。 LinkedIn，比较tricky，除非在找工作或跳槽或recruiter,可能得到较低的周访问频率。可以按月。

**M**onetization: User pays. Total revenue, ARPU（对某些公司特别重要，对一些公司并不是很重要。比如linkedin， Facebook， 注重流量，产品，赚钱或套现的方法并不是来自用户本身的付费，monetization对这些公司不太适用。但对电商，Amazon，O2O（外卖，Uber eats），重视用户下单的行为，用户（平均）花多少钱，这个指标是此类产品最终care的，最关心的。不过其他指标也需要关注。）

\* For companies like LinkedIn or Facebook, who focus on user growth.

AARM for YouTube

- Acquisition: # users sign up （第一次用gmail登录、访问Youtube）

- Activation: # users who watched xx minutes in last 30 days（只有注册后30天内看够xx分钟，才能体验到产品价值，才算真正用了）（或者用户第一次看某个视频，搜索某个视频）（对不同用户采取不同的activation的定义，例如普通用户，activation就可以定义为第一次完整观看一个视频，对content provider，其价值不在看视频，可能定义activation为第一次上传完整视频）

- Retention: # active users (1/7/30 DAU)

- Monetization: Revenue/Subscribers (from Youtube Red, YouTube TV) 想一下该产品有几种赚钱的方法？想一下分类 有广告收入，付费服务

面试中有些题会考指标问题。回答任何BA的问题，要列出框架。Framework

思维严谨缜密 有说服力 列出AARM 显得你很懂 让别人信你所说的东西

显得有条理 有全面的思路来引导答案 对问题分类

比如让你考虑下有哪些指标可以衡量YouTube这个产品，挑一个你认为最重要的，为什么？

我认为activation最重要。只有网站上活跃的用户多了才有广告收入。

或者我认为monetization最重要，YouTube目前是一个收入增长点，以营利为目的，要关注revenue。

按照框架可以给出一系列matrix，衡量产品在不同生命周期的指标。但最终要选一个指标，north star。有时如果你选两个或多个指标，有时两个指标会有冲突。 有些行为提高了activation但降低monetization。 怎么决定这个feature应不应该launch？ 再出现conflict result时可能会争论很久，或产生作弊现象。一个不好一个好，你说要launch 该experiment总有说辞来解释，可以拣好的说。必须要提前确定好criteria，避免事后诸葛亮。比如就要看activation，指标发生冲突也以定好的指标为准，一旦下降就不可以launch。

如果问 “怎么给产品找一个metric”

先列AARM 给出可选择的不同方面的metric 最后选一个north star metric， 并给出原因

对north star的选择，越简单越直白越好测量越好。 measurable！ 每个人在“多少”天内看“多少”分钟，较难定义合理数值。

对于YouTube 可以选 number of watch time 作为north star metric

原因：

YouTube是一个看重用户活跃的产品，赚钱是靠在视频里放广告。 大家看广告的时间长或者广告被播放的次数变多 会带来更多利润，从收入来讲，观看时长或观看一定时间的用户的数量都可以作为north star。但因为题是衡量 watch time 和latency 的关系，绕回到题里，可以把watch time作为north star。

Watching video is a key action that matters at YouTube, so I’d like to choose a metric that can reflect level of watching activities in aggregate. It’s also measurable.测量每个人看的时间 加在一起就可以 不用考虑人数，层数等复杂计算。

AARM只是提供思路的框架，没得答时很好用。但还是多看，培养商业嗅觉和产品嗅觉，知道针对不同的具体产品用什么衡量比较合适。作为一个面试的回答最好按类别来brainstorm多个metric，最后做选择，讲述下选择的原因。

为了衡量产品的performance，从商业角度，我们选定了要衡量的metric是什么。在现实工作中，之后一般要做一下sanity check on the metrics. 从数学和统计的角度，看一下这个metric是否很好测量或是否比较可行。有两方面的考虑：（1）看distribution looks ok吗? 取一些sample的用户，看相关指标的distribution。 (最好是something like bell curve) 看上去make sense. 虽然CLT并不关心population群体满足什么分布。没有特定的要求，但是你选的metric的分布，长的不能太奇怪，太取决于特殊群体。例如两个或多个峰值的，用户行为很不一样，说明选的metric有bias，要对几组切分，分别跑测试。你选的这个指标很可能是对某个群体才有意义，比如男女不一样，土豪和非土豪不一样。分类测验会增大难度，因为还得先去定义什么是土豪不土豪，怎么选择界定标准，容易产生作弊现象：selection bias。很可能认为正好找了一个让土豪和非土豪都显著的阈值。即使在一个组内部，比如土豪内部，样本数量很小，只占百分之1. 再在土豪内部抽样，不剩多少人，sample size很难定。同时在土豪和土豪之间方差也挺大的，花1000和10000的都是土豪，差别很大。不过也可以这样分组考虑，面试算一个好思路，但是实际操作很少这样。

总之选完metric后，像建模一样，建完先去看一下每个feature大概长什么样，会发现一些insight。如果选的靠谱的话，metric大部分都满足bell curve，符合自然界定律。

（2）看不太volatile。波动性不要太大，最好要有小的standard deviation. 如果population的var很大，根据CLT，sample mean服从正态分布的var也会很大，It’s harder to detect impact on a noisier metric.(small power) 太noisy容易测很多都无法reject，power降低，可能需要更多的sample等手段来弥补，更艰难。

Google Play 实例： north star metric定的都是revenue,每个用户花了多少钱

跑了很多A/B test都是统计不显著的，没法reject hypothesis

最后发现很大一部分原因是因为revenue这个指标非常noisy.用户花费差别特别大

花钱很多的人被分到的组，sample mean会增加很多。

尽管差别变得很大，但因为var也同时变大了，power变小，还是不能reject

推荐 不要去衡量revenue本身，因为商品和商品之间价格差别很大，用户花费也差别很大，导致这个指标非常noisy，方差特别大

改为衡量Buyer week = number of weeks user made a purchase

乘以number of users

只要一个user在七天内购买过一次，就给他的buyer week加1 下一个七天如果也花过钱，再加1 每个用户都有buyer week的metric 最后把实验组中、对照组中每个用户加和，得到一个数据

去除了revenue本身的波动性 衡量了用户在一段时间进行购买的一个非常不容易出现极端值的metric

改了以后在这个指标上很多实验都统计显著了。因为方差降低了，有足够的power来判断出很多有意义的事情。

**一定要在做实验之前就定好要衡量的metric**

Question：为什么var越大，power越小？

从商业上理解：outlier对sample mean的影响很大，造成了很大的sample mean的差别。但这些差别不是真正的sample mean的差别，这完全是由那几个outlier带来的。这使我们对真正的impact的情况的判断的信心降低。

从数学上理解：根据z-score公式，var在分母里，var越大z-score越小，z-score越小，离正态分布的曲线的中心（即均值）越近，越不容易被reject。（Power就是detect impact的能力，对有impact的事情要能够reject H0，现在不容易被reject，power变小。）

Question:工作中的sample size一般多大？

一般都是由power决定的。但是在互联网公司，用户很多

不用担心用户数量不够的问题，一般在所有的population中取1%就可以

Step 3.2: Design A/B test

● Choose population

● Determine sample size

● Sampling method

● Duration vs. Exposure

**Choose population** 搞清楚实验的受众 是容易被忽略的问题 比如amazon.com 在用户进入购物车之后，在购物车页面，在加credit card这里加一个小的feature的改进。 那relevant user是起码进入了购物车页面的人。 压根没进入购物车的人不用去target。 只抓出最相关的，我们真正care的user。 如果不这样做，结果会被稀释掉。

Choose targeting population-相关群体

- Target only the relevant users

- Why? Lots of noise, and result will be highly diluted. 比如如果抽1%的用户，而最后只有1%的人能够进入购物车，看到我们的feature，那相当于抽出的样本数量为万分之一。也就是说绝大部分用户根本不会看到我们的实验，他们的表现对实验组对照组的比较来说是噪音。对实验的观察没有任何的帮助。 所以choose population特别重要。 选定相关的用户作为population。

- How?

- Before-fact: PM/SWE inputs 向产品经理和build了这个feature的工程师（Software Engineer）去要一些inputs:去问你这个feature到底是在哪里加的，是不是有些用户压根不会看到这个。

- Triggering: only when user see something 用户在用我们的产品时，实时判断eligible，判断这个用户是否有资格被选入我们的实验和对照组。 比如我发现你进了购物车，才对你动态分组，实验组或对照组。没进入购物车的就压根不会被选中。提前跟负责这个feature的工程师团队搞清楚具体是怎么做的来更加精确的定位population。

- Post-fact: slice data by the eligibility criteria 可能没做pre-fact和triggering，很多时候做实验只是选了所有的1%的用户。做完实验发现有很多噪音。可能这个feature对应该看到这个feature的用户来说是有很大impact的，但是大部分选中的用户根本没看到这个新的feature，对小部分人的影响（5%）放在大的环境中被稀释掉了（0.5%）。这时可以做一下数据切割，post-fact。想一下当时有资格看到新feature的人有哪些特征。可以把e.g.没有进入购物车的人给去掉。不过要对实验组和对照组使用同样的切割标准。

这个事后的做法不如动态、实时的处理精确。

YouTube example: potentially 选择target population的因素:

- Which countries? 这个feature是否focus在某个国家？或者某个特定的用户群体？只在这些国家取样，只关注我们care 的用户种类。

- If we expose this feature to certain videos only. How do you know who have seen it?

如果只有某些特别热门的视频有相关性，我们可能只关注这些video的观看。选一些看了这些video的用户，让targeting更加精确。

Choose cohort as population -想看长期影响时适用

Let’s define “cohort”: track the same group of users over time.

在进行实验时，每天会有一些人来看视频，每天统计。今天多少人，明天多少人，后天多少人。可能会有overlap，也会有新的人，之前从来没看过。

- Track population: who ever joined: quick, on the spot

一个最简单的方法是直接统计feature对所有人的总体影响。

- Track cohort: same group of users with learning effects 另一种方法是track cohort：指的是同一组人。拿一组人出来，比如这一组人是第一天观看视频的所有用户，之后的每天不去关注其他用户，只关注这一组人之后的表现。假设他们还会回来，之后7天，之后14天，衡量我们这个feature对同样一组人overtime的影响。

这两种做法的唯一区别是 在cohort方法中，所有收据的数据里，没有新的用户来，只是同一拨老用户反复观看，可以观测到feature对具体到某一个用户长期的影响。

第一个整体选取的方法是混在一起的，每天都有老用户也有新用户。 新用户较多的情况下没法衡量产品的feature对用户的长期影响。（看不出长期趋势但确实能测出瞬时的影响，整体上的）

需要track cohort的三种情况：

- Retention

- Adoption of a new feature (**novelty effect**) 产品设计中要考虑的经典因素。 一个新的feature在launch时，用户会因为它比较新鲜而点击，但是可能这个feature本身并不会对产品有改进。比如把所有图标从蓝的改成红的。 大家可能觉得新鲜点了点，但是发现其实没啥区别，以后就不点了。 （甚至新feature可能使产品质量不升反降）在发布feature的第一二天或用户第一次看到这个feature时可能会对我们衡量的metric产生非常大的影响，但是overtime来看，当同一个用户第三四次看到这个feature，可能点击率又下降了或者恢复原来的水平。这时如果你track Population，可能会看到上升趋势，因为每天都会有很多新的用户是第一次看到这个feature，造成一个错觉--我们这个feature会提高点击率（但其实这个feature可能并没有用）。 对于有novelty effect的feature，最好track cohort来避免这个效应带来的bias。

- Run longer: dosage over time e.g. “wanna watch?”

dosage——市场行销 marketing的一种campaign

一些用户不活跃，给这些用户发email，或推送notification，或在产品中加弹窗或小提示。

这些可能都是持续性的处理。第一天发完，没动作，3天后来一次，7天后再来一次。是定剂量的行为。这里需要准确测量定时处理对同一用户长期来讲有什么区别。（有的用户用一次之后反应一下，之后就不反应了；有的用户用第二次之后activation上升）通过track cohort能把我们想观测的结果观测出来。

有很多feature是有延迟的，或者说是有节奏的feature。 在YouTube这个例子中，可以做这样一个提高用户观看频率、观看时间的措施：做推荐，“想看这个吗？”不做的太频繁，不会在用户每次看完视频时就推荐，而是每三天，每七天推荐视频清单。 我们就衡量一下每次推送后 长期的趋势。这样的情况下，需要track cohort（同组用户长期的表现）。Track新旧用户混合的population没法观测feature在时间维度上对用户产生的影响。

实际中到底用population还是cohort，是根据实验的目的具体而定的。 如果是上述三种情况之一，最好用cohort。 一般的情况下，如果只是看revenue，或者conversion rate，或者就是想提高某一个东西的点击率，或者不太认为该feature有novelty effect（新旧体验差别很小），这种情况可以直接track population，比较简单易操作。具体依情况而定。

**Determine sample size**

Power反映了测试有没有意义，如果power很低，没办法把impact找出来，即使Type I error很低，但实验没太大用处。既然power这么重要，如何保证实验有足够大的power来满足要求？一般可以通过增大实验组和对照组的样本数量来保证power。

确定sample size是一个非常重要的问题。

- Standard Deviation ~ 1/sqrt(n)

从CLT角度理解为什么N越大越容易检测出事件的影响：

根据中心极限定理，我们知道sample mean满足的正态分

布的方差等于Population的方差的平方除以N，即，sample mean 满足的正态分布的variance

和1/N成正比。或者说sample mean满足的正态分布的standard deviation和sqrt(n)成反比。n越大，sample mean满足的正态分布的sd就会越小，sample mean本身比较紧，分布比较集中于中心，更容易检测出一个极端情况。测试值更容易落入critical value 两端的区域。

从更直观的角度理解为什么N越大越容易检测数事件有影响:

做A/B test或者one-population hypothesis test的原因就是没办法把population中的每一个人都拿出来测量,只能通过抽样来对population具体长什么样来做出猜测和推断。样本数量越大越能代表整体population的情况，对结果做出更精确判断的power越大。reject掉不太可能的情况的power也越大。最极端的情况就是sample size = population size,已经不是随机变量了，计算了ground truth, power = 100%.

- Larger sample size, always better

- But running a large sample size has some drawbacks-浪费样本空间，不能同时跑很多test

存在限制：工作中为什么不选特别大的sample size（满足要求的power即可，不能再大）：

有两种考量：

- 50%是能够选择的最大的样本数量。（要分两个组，试验组和对照组样本数量要近似）这是max allowable size

- save space。在互联网公司 “growth hacking” “rapid development” 谁都想一年跑几千个实验，迭代速度会很快。但是比如可口可乐公司，要去测量marketing campaign,或者对产品包装做一些测试，目标是增长销量，实体从设计、包装、印刷、寄出、分销、购买，这一套流程走完一年就过去了。对传统公司来说很难快速地做test，因为产品本身和数据的限制不允许做这些快速试验。 而互联网公司都是工程师在后端写代码，开发一个app还算稍微复杂一些，要push一个新的版本，用户要升级更新才能看到新的feature.但如果是一个网页，用户根本不用去做任何更新，我对网站做1000次更改用户都能瞬间看到我的变化，互联网本身又有大量的用户的基数，导致了在互联网企业可以跑很多很多test。而跑很多test，比如一周或一个月之内有七八个实验的idea，每个实验都需要跑一段时间才能看出长期的效果，最后如果规定下来 每个实验的 实验组或对照组 都需要有10%的样本数量 才能有足够的power的话，所有的用户加起来都不够用了，总会有一些用户有重合，在两个实验的实验组里，看到两个新feature。这种情况会造成污染，feature A 和feature B同时被看到导致的影响无法衡量，最后在计算uplift时也会有问题。所以要尽可能节约样本的数量。最好是 每个实验组和对照组只用1%的population，就可以在同一个population中选出很多（实验组-对照组）pairs，来同时做很多不同的实验。

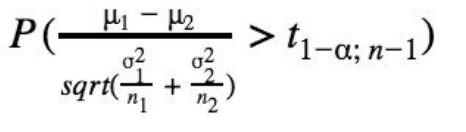
还有一个科学的方法，为了对整体做出更精确的衡量，到底今年比之前improve了多少，可以在所有population中留一个1% Holdout group，任何一个实验都不会碰这1%的用户，他们永远看不到我们这一年作出的任何修改。 我们在年末时所有实验都已经做完了，其他99%的用户看到的是非常新的一个产品版本，1%的holdout group看到的产品版本和去年完全一样。 我们现在从99%的用户中提1%出来，和holdout group相比，目的是做一个cumulative的measurement,比一比今年做了这么多东西，年底和年初的区别是什么。但是如果担心一年没更新会导致1%的用户流失，或者无法放弃这1%的用户带来的好处，（基数大，1%能带来的利润增长也不少，却不对他们launch任何feature），holdout group也可以留一个季度或者一个月。

- So, we want the smallest possible experiment size to achieve maximum power. 并不用去追求99%的power，一般设到80%就可以接受。我们要去找能够满足80% power的最小样本数量。如果取n个样本就能满足80%的power，那么我们就不会特别在乎“抽 >n个样本带来的更大的power”这样的好处。

What can you do to reduce the number of samples？

Power:the ability to detect impact--the probability to reject H0.

Reject H0: t score > the right critical value

Power formula:

在H0本身不成立（事件本身有影响）

的情况下， power就是我们

能检测出影响、即reject H0 的概率

Inputs to calculate power (p):

讨论其他因素不变时，每个单独的因素对power的影响：

- A: Magnitude of effect (μ1 - μ2)

从商业角度，到底什么样的impact对我们来说是meaningful的/really care的?

是人为给定的值，衡量我们对实验的最低要求。我们希望以这个power能检测出多大的差别。

以相同的能力检测出0.01%的差别的难度 > 检测出10%的差别的难度

检测出越小的差别越难

我们需要提前设置这个难度，比如设置成1%。如果这两个sample mean至少有1%的差距，我们希望这个实验有80%的能力把这个差别找出来。如果A和B的差别只有0.5%，那就不期望这个实验以相同的power检测出来。（或许也能检测出来，但检测出来的概率会降低，power可能 < 80%）

Magnitude of effect变大，power会变大，因为测试更容易了（让你检测出一个差别很大的差别），或者说被要求去做一件更简单的事。但具体定多大根据实际情况决定。你觉得这个feature有多大的impact就定多大。盲目定太大的话，ground truth本来就达不到这么大的差别，因此也会检测不出来。

分母：大概相当于把AB两组混在一起计算出的standard deviation。 sd越大，分母越大，左边越小。 小的数 > 某个critical value的概率就变小。即方差变大，power变小。

sigma也很重要。我们希望尽可能找一个metric，其本身的方差比较小。比如从测revenue（方差大，power小）改为测weekly buyer（像是binomial或泊松分布,方差小了，power会变大）

- B: Sample size in each group (n)

前面都在讲为什么n大了power会变大。数学上，n大了，分母变小，分数整体变大，一个更大的数 > critical value 的概率会变大，power就变大了。

- C: Sample standard deviation (σ)

当样本本身的方差变大的时候，因为noise更大了，自然情况下出现极端情况的概率本身就大，（分布比较扁，不容易reject H0）power就变小了。

- D: Type I error (α)

当alpha变大时，右边的critical value会变小，错怪好人的容错率变大，会让实验更加容易找到坏人，power变大了。即损失（提高）alpha也能增大power。但没人这么做。Alpha一般是定死的5%。

至此，我们知道：五个因子，给定四个都能求出第五个；A,B,C,D单独来讲如何影响power。

To calculate sample size n, we use p (normally 80%), μ(assume), σ(we can calculate), and α (normally 5%).

面试中不用计算power。非常重要的是知道 power是什么。用非统计术语，直观解释出来。

会计算sample size—— 黑板上写出 power怎么算的。有四个影响power的因子。每个稍微解释一下，算sample size就是用这四个反推出来。

回答问题：What can you do to reduce the number of samples？

1. Increase the level of uplift we want to detect.(detect 10% not 5%)

power不变的情况下，难度降低了，所需的样本数量就也变少了

1. Change to less volatile metric (sd is smaller)

其他都不变时，var减小，power会增大。当固定power不动，所需的n会变小。

1. *Decrease power (just want 70% confidence,instead of 80%) 要求的power变少了，那需要的n也会变小。只是一种思路但一般会要求power固定为80%。*

----从四个因素的角度考虑。Alpha是固定的不能动，动另外三个。但是一般power也不动。

1. Subset on population (filter out ineligible users) （可能本来取得样本有些是不eligible的，feature可能只影响到某一些用户，将取样范围缩小在only eligible的用户，过滤掉本来没用的人）
2. Run longer (assume not too many repeated users)极端情况是每天来的人是同一拨人。但如果天与天之间没有很多overlap的话，就可以通过跑久一点来减少每一天对population取样的percentage。（比如今天对1%的user跑实验，明天又对1%的user跑实验，一共得到几乎2%，再跑一天变成3%，但由于会有overlap，percentage不会随着天数线性增加，可能跑7天得到了5%的user。可以增加天数来减少每天取样的数量）

Q1: How to pick magnitude of effect? (business assumption)

需要商量，大概多大的effect是meaningful/make sense的？见A下面的解释。

Q2: When calculating the two sample t test sample size, what if the two sample sizes are different? (to reverse-calculate N, we need to assume sample size is the same)

实际操作中不会出现两组样本数量不一样，公式中都会让n1 = n2，实验组对照组样本数量相等。

Q3: How to determine the sample variance? (use control as “baseline”)

现在有两个不太一样的组，一个A，一个B，可能方差不同，如何计算sigma1和sigma2?

做实验之前就把它算好，在所有人情况都一样时计算好sample variance.

或者说以control组为准。

**Sampling method**

● Simple 最直白的随机抽取：每人同样的机会

○ Every user has equal chance of likelihood of being selected

● Stratified 分层抽样法

○ Problem: only 1% of our users are high spenders.

○ Strata: low/medium/high spender

在三个阶层中，每个都用simple sampling method去抽1%，可以在理论上完全确保我们选的样本中 low,medium,high spender的比例跟整个population的比例是完全一样的

○ Ensures an equal representation from each strata

比较特殊的情况是在某个维度上，可能我们非常关心一小个群体，由于他们的数量比较少，可能抽样时实验组抽了几个，对照组没能抽到，会对观测结果有影响。我们就可以把high spender单独拿出来，保证在实验组和对照组中都能抽到这样的用户。

不过这个在工作中很少需要做，都是工程师写好了，按照seed随机抽取。

Quiz: Sometimes, we want to check if our sampling method is correct

As BA, how do you check if our sampling is actually random?

通过数据来观测一下抽的样是否合理。在真实工作中可能会遇到。

把数据按国家（分层/strata）画直方图,看在每个层上,组A和组B样本数是否相似。

（也可以和整体作比较，看strata间的比例在实验组、对照组和population中是否相似，或者看分布是否相似）

设计实验要考虑的最后一问：

**How long should I run the experiment?**

**Duration vs. exposure**

- **Duration**: # days you want it to run

- **Exposure**: % of users who see this experiment on a day. 实验组在每天占百分之几

- Duration \* exposure = N（样本数量） 如果不考虑不同天有重复用户

- Duration and exposure are trade-offs. 假设样本数量已经确定了，N为定值

一般来讲肯定不会把duration调得太长，毕竟要快速，短期内做很多实验。有的结果就好，没有好的结果就迅速move out。这是基本原则，所以duration不能太长，3个月——不行。

但duration也不能太短，3天——不行。 存在很多问题：

- Exercise: Why not load everything on the same day?

- Reduce seasonality

- Reduce risk

10%对照组，10%实验组，跑一天行吗？No

因为（1）季节性，产品本身有周期性。对YouTube来说，周末看视频的用户表现和周中看视频的用户表现可能很不一样。可能压根就是不一样的两个人群。如果Feature只是对某类用户比较有效，（可能你的feature只对周中看视频的用户适用，对周末看视频的用户反而有不好的影响），如果你的test只在周一到周三跑三天，看不出全面的影响。至少应该从周一到周日跑七天，对实验的影响作出全面判断。所以一般的test至少要跑一周。

Weekly seasonality. 要覆盖一整个周期。(2) Risk风险。做实验其实是很危险的，一定要非常小心，每做一步，每把新的feature展现给1%的用户，都要确保这一步是work的。万一这个实验失败了，或者影响非常差，那你如果一开始就expose这个new feature to 10% users，可能直接损失10%的用户利益。比如电商在shopping cart里launch一个小feature，但这个小的修改里有个bug，导致用户无法下单成功，那直接expose 10% user，相当于当天10%的revenue就没有了。 预防方案是选1%的用户，跑两天，要实时观测这个东西表现如何，如果发现metric走势异常，比如revenue降为0，那就赶紧叫停这个实验，快速roll back，找出实验到底出了什么问题，来修bug。

产品的周期性是由产品本身的特性决定的。YouTube一般就一周一周期。 有的产品以月为周期，那就应该跑一个月来覆盖所有情况。

至此，实验都设计好了。可以告诉大家：

我要衡量哪些metric，track population还是track cohort,每天需要选多少样本，如何取样，一共跑多久。

现在可以开始跑实验。跑完预设天数，拿到一个结果。

Step 3.3: Interpret results

This is where you make recommendations so it is important. 根据拿到的数据，向商业伙伴作出recommendation-实验到底跑得怎么样，实验得出了什么结果。

Briefly describe how to calculate hypothesis testing

- (Almost always) Two sample t-tests

- Calculate t-score

- State your decision criteria决策衡量标准

在工作中不用给人讲t-score,t-test，甚至不用展示出来。没人关注t-score具体是多少，P-value具体是多少，我们只需要得到一个decision criteria，只需要告诉别人A比B到底是统计显著的大/小，还是A和B没有统计显著的区别。（自己通过t-test算出t-score，根据t-score和critical value作比较或者算个p-value来做出决定。）

\*需要注意的一种情况：

可以有多个指标，但是一定要提前定好一个primary metric，其他的指标只作为参考，防止作弊行为（挑表现好的metric为launch标准）。比如我们可以以revenue为主要指标，conversion rate为次要指标，如果revenue不变但conversion rate上升了，虽然我们不能launch这个feature（因为对north star metric没有影响），但我们能看出一些有用的东西，对我们的下一步产生一些指导。可以进一步去转化流程中看具体是哪一步上升的比较明显，可以对这个步骤做一个实验。至少可以argue这个feature对用户转化率有帮助。至于revenue没升，可能是因为重复购买率降低了或者客单价减少了。总之观测多个metric可以让我们有很多follow up可以做。

Multiple metrics

是有帮助的。会带来很多信息。但不能用这个作弊-挑表现好的metric作为launch一个feature的标准。还是要有一个主metric。

- Idea: the more stuff you check, the more likely you are going to see something.

- Solution:

~~- Repeat this, if it happens by chance, it won’t show up~~

- Multiple comparisons - reduce the alpha

(https://en.wikipedia.org/wiki/Multiple\_comparisons\_problem)

- Should have one “north star” metric

Type I error-alpha:描述的是A和B本来没有区别，但你碰巧检测到了一个很极端的结果，说它们有区别 的概率。首先要把这个概率降到很低，但这个概率还是存在的，我们定为5%。这5%意味着，如果把这个实验做20次，20次中可以出现一次false positive。如果我们看20个metric,概率上20个里面可能会有一个metric有一个是significant up（即有很大差别）。如果你看了20个metric，只要有一个有好的结果，就认为这件事影响很好，肯定是错的。会极大增加false positive rate.（见下一个quiz）

解决的方法是wiki/Multiple\_comparisons\_problem，一个定律：

如果你要看10个metric，对每个metric做相应计算时需要把alpha降低，new\_alpha = alpha/10 = 0.5%，才能保证最终整体实验的alpha是5%。

以防作弊--“cherry picking”

（或者重复做实验，如果是碰巧significant/出现极端值，下次做不太可能再次发生一样的up）

Quiz: Track 3 metrics. Alpha = 0.05. 假设三个metric互相独立，不会互相影响。Probability that at least one metric will show up as significant, when it is not（本身并无significant的impact）?

帮助理解如果用这种方法(三个metric里只要有一个显示significant，就认为significant)来做实验的话，会产生多大的错误率（false positive-因为本身并无impact，却shows sig impact）

Since alpha = 0.05, true negative = 1- false positive = 95%.

for any of the above metric:

P(shows up sig) = 5%

P(not sig) = 95%

P(at least one metric sig) = 1- P(all metrics are not sig) = 1 - 95%\*95%\*95% = 14.26%

Ans: 1-P(no metric show up as stat sig) = 1 - (1-0.05)^3 = 14%

如果用这种作弊的方法来做rejection，false positive会上升到14%！

Hypothesis testing among multiple groups(面试不考，知道就行)

Assume we have 3 groups A, B, C to compare.

H0: μ1=μ2=μ3

H1: at least one of them is different

Approach 1: do many two-sample t-tests. 做多个测试，两两比较，只要reject掉一个，就可以说这三个group不是两两一致的。

H0: μ1=μ2

H1: μ1!=μ2

Then,

H0: μ2=μ3

H1: μ2!=μ3

方法简单但要注意，我们用了两个或多个实验去判断原本的H0，需要降低每次实验的Type I error，否则你reject掉整体H0的criteria变成了“至少reject掉一个”,会增大Type I error，使实验整体的alpha大于5%。

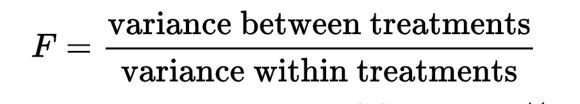
Remember that we need a smaller type I error.

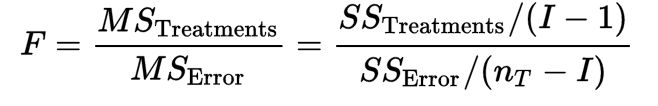
Approach 2: ANOVA (analysis of variance)

用于判断多个组之间的miu是否相等.在Two-population test中也可以用。

Intuition: Assume H0 is correct. Then these groups should have very small difference/variation among themselves.直观思想就是如果三个组的均值都差不多，那组跟组之间的差别相比于各组内本身（自己跟自己）的差别应该会比较小。可以做个F test：F也是一个random variable.

（A组和B组和C组之间的差别/各组组内100个人的variance）





So, if F-statistics is big（组间差别很大）, then we are more likely to reject H0.

(F-statistics也满足于一个类似于bell curve的分布，如果观测值很大，也是要reject H0)

Step 4: Make recommendation

Your recommendation:

- Non stat sig? Oh no. But maybe there’s strategic value?

实验如果不成功，想一下what have we learned? 这个test对我们当前的metric没有影响，但对其他指标是否会有影响？观测一下其他指标有无变化？如果对别的指标有影响，未来可以研究一下原因是什么，可以促进我们brainstorm出一个新的idea。

- Stat sig? Good, but talk about trade-offs 好事，但还要有几点考虑：

- Benefit: Small impact?

- Cost: Hard to implement?

- Opportunity cost (prioritization)

假设这个test检测出了1%的positive impact,但我们在同个时间段可能在跑很多test，由于精力有限，时间有限，我们需要做出取舍，这1%的impact到底是否值得，不一定。要和其他test做一个prioritization，排个序。比如可以画图：横轴是impact，纵轴是difficulty/Cost. 大impact&小的cost一定要做，大的impact&高成本、长周期可能最后不会做。

BA在工作中要做很多prioritization,不光是针对A/B test,还要帮助产品组做路线图，根据数据、用户表现来做估算，应该在哪些地方着重想办法改进我们的产品。 Think about漏斗，用户在逐渐流失，有很多步骤，从哪个步骤入手impact最大？——BA初期需要做的事。

一般来讲在漏斗上端改进影响比较大。稍微平移桶边，进来的人就会一下变多，可以继续往下增大宽度。不同的问题都需要在初期做个排序，看做哪些事情对我们综合影响比较好。

——————有影响的情况，考虑成本，做prioritization(幸福的烦恼)—————————

Quiz: 一个test跑了7天，发现it is actually stat sig. Two days later, not stat sig. What do you think happened?

What’s your recommendation?

Ans: - Seasonality (solution: long-term holdback)可以观测时间长一点，把前7天和后两天的用户单独拿出来，去看一下他们的行为有什么变化来间接验证。

- Novelty effect (cohort) 可能前七天还是热乎劲，七天之后新鲜感结束了就回到不统计显著的阶段了。 解决方法是做cohort，去抓一下前七天表现非常好的用户，看他们之后两天是否下降了（可以在不重新做实验的情况下去验证是否是这个原因造成的）。

最终的结论如果是novelty effect,证明这个feature没什么效果。

如果是seasonality,跑的时间长一些，再过一个周期可能长期来讲还是一个uplift.

Quiz: A marketing manager ran an email campaign to 5% of users but forgot to set up control. What

would you do? 工作中容易遇到的问题，尤其是做marketing时

忘记设置对照组，事后还是想测量这个email campaign到底有没有用/有多大影响,怎么做？

只看发了email的用户的前后表现是不行的，需要做横向对比。找control。

Ans: - Time series based analysis

统计方法。用时间序列forecast出那5%的用户，如果当时我们没给他们发邮件，他们 会什么样。用forecast的结果和现实的结果做对比。（但这个方法说服性不高，除非time series非常整齐，前后差别的确很大）

- Retrospectively find out the control:(segmentation)

往回构造对照组，切割用户，给用户分组，从剩下的95% user中找5%，确保新的5% 和旧的发过邮件的5%两组成分差不多（在各方面都很相似）。

可以近似理解为新找的5%和之前的5%相似。看这两组在发邮件那个时间之后变化的表 现。或者说新的5%和旧的5%在发邮件之前有个difference，在发邮件之后又有一个 difference。 要去比较difference of difference.本来有个区别，发邮件这件事可能导致这 个difference变大或变小。衡量这件事。