https://bit.ly/ABTestingImperfections

The Imperfections of A/B Testing

Ronny Kohavi
13 Apr 2024

No one pretends that democracy is perfect or all-wise.
Indeed, it has been said that democracy is the worst form of
Government except all those other forms that have been tried from
time to time.

-- Winston Churchill on 11 Nov 1947

TL; DR This document addresses several criticisms of A/B testing. A few are legitimate, but most are based on misunderstandings or poor implementations/execution of A/B testing. I will use "car" analogies throughout the document to argue that despite the "issues" with getting you from A to B with a car, most people prefer it to a horse.

My goal in writing a detailed doc is to have a reference document, so that the next time says: here's my issue with A/B tests, I can refer them to issue X in this doc.

Solomon Kahn posted a 30-minute video making "The Case Against A/B Testing" with 18 things he "thought were SERIOUSLY wrong with A/B testing." He admits it sounds crazy (those words are in the title of the video), and I initially stopped watching after seeing that the first reason was flawed, a common misunderstanding of p-values. He was polite, thanked me for the comment, and asked to check out the remaining 17 issues. In the video he also asked for rebuttals, and I respect that, so I decided to spend the time to address the issues, so that others may benefit. I think enough people might want to review some of these and understand which are valid concerns/pitfalls, and which are incorrect. I'm more than happy to incorporate more "issues" that people will raise.

Paraphrasing Churchill's quote above, A/B tests are fraught with pitfalls that make it hard to establish causality with high confidence; controlled experiments must be the worst methodology, except for all the others that have been tried (post). The same applies to cars: there are issues with fueling, servicing, and obtaining a driver license. These are hard, but most people find enough value to overcome these issues.

At a high level, I will split my responses to two categories of issues:

1. Misunderstandings and poor reasons. Misunderstandings, such as issue #1 about p-values, are important, as he isn't the first person or the last person to misunderstand p-values. Poor reasons include things like "Optimizing for meaningless metrics." By calling something meaningless, it's practically a tautology, hence objectively a poor reason against A/B testing.
   With this category, I will also include poor implementation and execution challenges. These are not limitations of the methodology, but rather the way it is used when we know how to address the issues. If your implementation is buggy, or your A/B testing vendor's quality is poor, that should not be a ding on the methodology. In my career, I focused on making A/B testing trustworthy and reliable. "Trustworthy" in the title of our book (https://experimentguide.com) and a key focus in the online class I Teach (https://bit.ly/ABClassRKLI).

2. Valid Concerns. Mixed with the set of issues raised, there are good valid issues. From what I can tell, all have been addressed in the literature. These are well known and well-understood limitations and requirements to A/B testing, and there are certainly some scenarios where A/B testing is not appropriate. In software web sites,

applications, and services, the data is overwhelmingly strong that the use of A/B testing accelerates innovation. Despite our best efforts to prioritize the best ideas, we are often humbled by the results and adjust our intuition. As shown in https://bit.ly/ABTestingIntuitionBusters, success rates of experiments range from about 8% to 33%, with the median success rate at 10%. Practically every one of those ideas went through a prioritization exercise and was rated high enough to implement, but most were disappointing. With such rates, the value of being able to identify the successful ideas and learning from both the successes and failures, is immense.

Misunderstandings and Poor Reasons

- Issue #1: 95% confidence intervals are inappropriate.
  The claim is that the use of 95% confidence intervals, or equivalently that the use of alpha threshold on p-values of 0.05, implies that the treatment is 95% better when you have a statistically significant result is wrong. P-values are conditioned on (assume) the null hypothesis, so to get the probability that a statistically significant result is a false positive, you need to apply Bayes Rule and compute the false-positive risk (FPR). In https://bit.ly/ABTestingIntuitionBusters, we showed that for the median organization, with a success rate of about 10%, that probability is 22%, not 5% as stated. That's close to the 20-30% suggested.
  Perhaps more important, the alpha level is commonly chosen to be 0.05 as a default when there is no additional information, but businesses can choose their threshold based on their risk tolerance and desire for learning reliable data. For example, at Airbnb Search where I worked, we lowered that threshold to 0.01.
  There was also an incorrect claim in the video that a p-value of 0.95 is a win. It's low p-values (e.g., below 0.05) that are statistically significant, not high ones, but I assume that was just a verbal slip.

- Issue #4: Optimizing changes for meaningless metrics.
  This is analogous to claim #10 on the insurance form: "I collided with a stationary truck coming the other way."
  Don't blame the car manufacturer (or the tree) if you can't drive.
  By definition, if it's meaningless, don't optimize it.

- Issue #5: Looking at a single metric when the business should look at multiple metrics.
  See issue #4 above. If you are claiming that the business **should** look at multiple metrics, then use multiple metrics as the criteria for A/B testing. See Issue #3 below about the fact that picking an Overall Evaluation Criterion is a hard problem.
  As Lewis Caroll wrote: "If you don't know where you are going, any road will get you there."
  Don't blame the car if you picked a bad destination to drive to.

- Issue #6: Looking at multiple metrics, there is a substantial increase in statistical noise.
  Apply corrections for multiple hypothesis testing. These are well documented.
  Don't tell me that you crashed because there was fog on the windshield. There is a defog/defrost button that you need to learn to press. It's true that horses don't have a front windshield that needs defogging, but that's not a good reason to give up on driving cars.

- Issue #8: Flawed vendor methodology where the results are wrong.
  Incorrect calculations, race conditions, bugs. These are poor reasons not to rely on the gold standard in science.
  Build or buy a trustworthy experimentation platform.
  Don't tell me that a horse is a better transportation vehicle because your cheap car that you didn't maintain broke down.

- Issue #9: A/A tests failing.
  Read Chapter 19 in our book https://experimentguide.com. We have better techniques than running a single A/A test that you discuss, such as looking at the distribution of p-values and seeing if it diverges from uniform. Passing these tests is a prerequisite for a trustworthy experimentation platform.
  Don't tell me that a horse is a better transportation vehicle because your car fails smog tests. Maintain your car properly, and it will pass the required smog tests.

- Issue #10: The experimentation platform needs to support concurrent experiments.
  All good platforms support this. The issue is well understood and addressed; see ref1 (Section 5.2), ref2 (section 11), ref3, ref4, ref5.
  You're complaining that some cars in the 1950s didn't have seat belts, but you acknowledged that "Tooling has gotten better." This is a non-issue.

- Issue #11: A strong result is likely to be a bug.
  You vaguely referenced a "famous paper from Microsoft." I may have been a co-author on that paper, as we had several sayings at Microsoft on the need to be skeptical of extreme results:
  - Twyman's Law: Any figure that looks interesting or different is usually wrong.
  - Getting numbers is easy. Getting numbers you can trust is harder (paper).
  - Extraordinary claims require extraordinary evidence" (ECREE) by Carl Sagan

  That's an implementation problem, not a bug with the A/B testing methodology.
  When someone recently claimed that rounded Call-to-Action buttons drive 17% to 64% more clicks (here), multiple people called Twyman's law (some used stronger, less polite, words). The two strongest supporting experiments were A/B tests, but these were flawed (here); we put little trust in the other experiments based on surveying small samples of graduate students.
  The fact that there are bugs and implementation challenges is not a good reason to stop using a methodology.

- Issue #12: Widespread p-hacking.
  Optimizely encouraged peeking in their early versions, but they fixed the issue many years ago. The problem is well-understood and vendors make it harder to p-hack, including features for group-sequential methods and adaptive techniques that support peeking while controlling the error rates. This isn't a problem with the methodology, but incorrect application of controlled experiments.
  If you drive with your eyes closed, don't complain that you crash often.

- Issue #14: Vendors [and agencies] are incentivized to show wins.
  Your (good) issue #3, short-termism, is relevant here. Vendors that get the statistics wrong get dinged. Here's a great example: How Optimizely (Almost) Got Me Fired (the problem was fixed by Optimizely). I believe this is still partially true today, as Optimizely still defaults to an alpha of 0.10, which is too high given their own customers' low win rate, but it's easily changeable.
  Many cars still have speedometers that over-promise capabilities by showing speeds all the way to 180 MPH (see fun shirt). The fastest horses can't run over 60 MPH.

- Issue #15: If a test comes back with a small but statistically negative result, do you really not ship it?
  Indeed, don't ship it. This appears to be a very strange argument. You know you're harming users (or whatever your OEC is), but you want to ship the feature AND add additional complexity and maintenance costs to the

codebase? Perhaps you're thinking about a scenario where some specific metric is negative, but the OEC includes additional factors, such as strategic or legal reasons. Maybe I'm missing something, but this seems plain wrong.
There is a valid issue sometimes raised about shipping flat results (no statistically significant effect) and thus potentially lowering the OEC over time due to deaths by a thousand cuts, but if something is statistically significantly worse, why would you ship it? In most scenarios, even flat results should not ship (see post), and certainly not negative.

- Issue #18: We lose something. We take away from the deep understanding of users and products.
  It's the other way around. We *gain* so much knowledge and develop better intuition when we are given reliable and trustworthy evaluations of our ideas.
  Are doctors better or worse than they were 50 years ago when the government started to mandate that drugs can only be approved through controlled experiments, called Randomized Clinical Trials (RCTs)? No, I think doctors were helped by clear objective data.
  See the history and progress made with the 1970 regulation.

- Issue #19: the math is far more complicated than most people expect and they often get it wrong.
  This was raised in response to me pointing out that issue #1 was a misinterpretation of p-values.
  I think it's another weak argument. The thousands of developers, PMs, and Testers at Bing used the ExP experimentation platform without necessarily understanding the intricacies of A/B testing, but my team built a solid decision-making process that was justified given our deeper understanding of the concepts.
  For our car analogy, I don't understand how the fuel injectors in my car work, but I trust that the engineers who programmed the computer inside the car, and lo and behold, the car does accelerate when I press the accelerator pedal, and I'm able to get from point A to point B. Sorry, I'm not changing to a horse.

- [Not numbered, but mentioned several times, such as in relation to peeking]
  Something that is statistically significant may not be business significant. That 0.2% statistically significant result means nothing to the business.
  We cover this scenario in Chapter 2 of https://experimentguide.com, but I think this is the result of a misunderstanding, as I think it's rare for something to be statistically significant and not ship.
  - You built the feature, QA'ed it, took it through an A/B test, and it's 0.2% statistically significantly better on your key metrics. We know you wanted it to be 5% better, but you just got a dose of reality: your baby is not as beautiful as you thought. The development and QA is sunk cost. You can turn the feature on since it ran in a live A/B test. The question is whether the maintenance cost of the feature is more expensive than the value of the 0.2%. I worked at Bing and net at revenue at Bing in fiscal year 2023 was $6.2 billion. That 0.2% improves that revenue by $12M, so we definitely want to know whether a feature is helping or hurting by millions of dollars. If you're a small startup with revenues of $5M, maybe you don't care about $10,000, but I think even that small amount can improve morale by funding a nice annual dinner for the team.
  - If you truly don't care about 0.2%, set your MDE (minimum Detectable Effect) such that you'll only be finding statistically significant results that matter for the business, perhaps 0.5%. Increasing the MDE by a factor of 2.5 (0.2% to 0.5%) means you can run the experiment 2.5^2=6 times faster. Instead of waiting two months, you'll reach the desired sample size in 10 days.

The reality is that I have never seen this be an issue. There are never enough users and we always want more sensitivity, which is why variance reduction techniques like CUPED have been developed (Microsoft CUPED paper, Netflix paper).

- [Not numbered, but mentioned at the end] The Alternative: test quickly and look for big shifts.
  This is a pitch for the Inter Ocular Trauma (IOT) test mentioned here. The effect is so large, that when you look at the time-series graph, it hits you between the eyes.
  This is a weak argument for most innovative ideas due to multiple reasons:
  - Organizations that have run proper A/B tests have shown that big effects are rare.
    The most successful experiment in Bing's history was the opening example in our book, which improved revenue by 12%. Most ideas have small impacts that add up.
    Look at the results shown in the Fat Tails paper in Table 1. The highest improvement to the success rate metric from the 1,450 experiments analyzed in the paper was 0.28%. This is not something you can see in any time series graph.
    Every year Bing improved revenue per search by about 20%, but evaluating thousands of ideas, and launching a small percentage of the successful ones.
  - Large organizations work on many ideas concurrently. Microsoft (Office) 365 releases its Office application update every month with 100s of new features, all protected by A/B tests for safe deployments. It is wonderful to see how when one of those features increases crashes, it is quickly identified and disabled in real-time. Try that without A/B testing infrastructure? All those 100 features passed QA, but somehow something was missed.
  - The sensitivity you get from A/B testing allows you to look at the impact to segments. The idea may be flat on average, but the fact that it was successful in this segment and hurt this other segment provides insights for the next iteration.

  For our car analogy, this reminds me of Ken Thompson's car (Turing award winner and early Computer Science pioneer):

  > Ken Thompson has an automobile which he helped design.
  > Unlike most automobiles, it has neither speedometer, nor gas gauge, nor any of the numerous idiot lights which plague the modern driver.
  > Rather, if the driver makes any mistake, a giant "?" lights up in the center of the dashboard.
  > "The experienced driver", he says, "will usually know what's wrong" ☺

  I assume you don't have such a car and prefer more specific insights with a clear light or message for the 50 things that usually go wrong.

Valid Concerns

- Issue #2: Assuming that the results will continue in the long term.
  Technically this is called external generalizability. We know the value today, but there may be novelty effects, primacy effects, and other factors that will change the effect over time. It is a concern that everyone should be aware of, but after having done many replication-runs on features that we thought would certainly diminish over time, most of the time, the concerns were overblown. The winning long-ad title in the opening of our book is unlikely to be influenced by interest rates (the example used in the video) and neither are the improvements to search relevance. If anything, controlled experiments provide the sensitivity to check if the treatment effect is decreasing over time during the experiment, hinting at some novelty effect, as discussed in our book.

- Issue #3: Short-termism around metrics vs. long-term impact.
  Picking the OEC, the Overall Evaluation Criterion, is a hard problem. We have two chapters on metrics in our book, and I have a large session in my class about this. For any quantitative methodology, you need to decide what you're optimizing for, and the driver metrics that you will use and that are measurable. Whether you are looking at a time-series graphs or any quasi-experimental design methodology, you need to decide what you're optimizing for. History has shown that reliance on the Highest Paid Person's Opinion (the HiPPO approach) does not work well. For example, in medicine (see history), the 1938 regulation appointed physicians as the judges of which drugs were appropriate for individuals to take, but by the 1970, the regulation said that physicians were not able to discriminate efficacious drugs from useless ones (Temin 1980, p. 138).

- Issue #7: Adding up A/B tests isn't close to the business results.
  Statistically significant results are optimistically biased. This is well documented, and it implies you should apply a haircut to the results (e.g., post). This is a lower bound on the haircut if everything is executed perfectly. In practice, there will be other external generalizability problems that imply a higher haircut should be used. We used 20% at Bing and it matched our results well (see two-year example in Chapter 1 of https://experimentguide.com).
  We had a trustworthy A/B testing platform. If you're not there and have many issues, it is likely that your results will be more optimistically biased.
  The friend discussed in the video, who claimed a series of 100% improvements is clearly not using a trustworthy system. At Microsoft we ran 20,000 A/B testing treatments per year and never had a single result that improved our OEC by more than 12% (the opening example in our book). But we are big believers in Twyman's Law and every time we saw an improvement over 5%, we worked hard to find what's wrong with it, and in 90% of the time we found the issue. The remaining 10% were smaller effects that were indeed breakthroughs.

- Issue #16. Even if you use 95% threshold with hundreds of tests, then you'll have many tests that have statistically significant results that are incorrect. The XKCD cartoon with 20 Jellybean colors (not lollipops as incorrectly mentioned in the video) is at https://xkcd.com/882/.
  This is indeed a valid concern, and that percentage is the FPR, not the alpha rate (see misunderstanding of issue #1), so for a median org with 10% success rate, about 22% of the statistically significantly positive results are likely to be false positives. This is well understood, so let me share a few thoughts:
  - Statistically significantly positive results that are false positives are very highly likely flat (close to zero, or closer to zero than your statistically significant threshold), but not statistically significantly negative, if the experiment is properly run with 80% power. An "S error" (sign error) in Gelman and Carlin's paper, is extremely rare for properly powered experiments, as shown in Table 2 left of that paper. At 80% power, the probability of an S error (sign error) is less than 1 in 800,000. So yes, you will have some results that you think were statistically significantly positive, but the effect is smaller, or slightly negative, but not statistically significantly negative, except perhaps one in 40 years, if you run 20,000 experiments per year.
  - You have the option of validating successes to increase the trust level and reduce the false positive risk. At the relevance team in Bing, it was a standard procedure to do a replication run for every statistically positive result. If your false-positive-risk is 22%, the probability of two false positives is 22%^2, or 4.8%. An alternative is to lower the alpha threshold to 0.01, which we have done at Airbnb.
  - What's the alternative? Human decisions have a much higher error rate with well-documented biases.

- Issue #17: A/B testing slows you down.
  I agree. That's the price you must pay for trustworthy data. Running in the right direction is more important

than just running fast. The video says you can't A/B test A/B testing, but there is some evidence in this paper about startup performance (PDF).

My experience is that most experiments run for about two weeks. This is wall-clock time and people aren't idling, but working on the next feature, as data is gathered. There is a tradeoff between evaluation mechanisms cost and fidelity. You could ask the HiPPO, you could do a quick lab study, and these will be faster, but will have high error rates. As mentioned, 90% of experiments fail to move the metrics they were designed to improve. What percentage of these would be identified by the HiPPO or by a lab test?

For our car analogy, you can reduce structural reinforcements, airbags, the ABS systems, and improve the car efficiency, but most people think these tradeoffs are useful. The largest companies in the world use A/B testing because they see the value. See https://bit.ly/OCESummit1 for an example.

I'd love to hear more issues and commentary. Feel free to add comments to the doc and I'll integrate things.