20

Triggering for Improved Sensitivity

> Be sure you positively identify your target before you pull the trigger
> *– Tom Flynn*

Why you care: *Triggering provides experimenters with a way to improve sensitivity (statistical power) by filtering out noise created by users who could not have been impacted by the experiment. As organizational experimentation maturity improves, we see more triggered experiments being run.*

Users are triggered into the analysis of an experiment if there is (potentially) some difference in the system or user behavior between the variant they are in and any other variant (counterfactual). Triggering is a valuable tool for your arsenal, but there are several pitfalls that can lead to incorrect results. It is important that you perform the analysis step, at least for all triggered users. It is easier to identify the triggered population of users if you ensure that triggering events are logged at runtime.

Examples of Triggering

If you make a change that only impacts some users, the Treatment effect of those who are not impacted is zero. This simple observation of analyzing only users who could have been impacted by your change has profound implications for experiment analysis and can significantly improve sensitivity or statistical power. Let's look at several examples of triggering in increasing complexity.

Example 1: Intentional Partial Exposure

Suppose you are making a change and running the experiment on a segment of the population: only users from the US. You should only analyze users from the US. Users from other countries were not exposed to the change, so the Treatment effect for them is zero and adding them to the analysis just adds noise and reduces the statistical power. Note that you must include "mixed" users, those from both the United States and other countries, in the analysis if they could have seen the change. Be sure to include all their activities after seeing the change even activities performed outside the United States, because they were exposed and there could be residual effects on the non-US visit.

This observation applies to other partial exposures, such as making a change that only applies to users of the Edge browser, or one that only exposes users whose shipping address is in a given zip code, or making changes to heavy users, users who visited your website at least three times in the last month (note that it's critical that the definition be well-defined based on data prior to the experiment start and not one that could be impacted by the Treatment).

Example 2: Conditional Exposure

Suppose the change is to users who reach a portion of your website, such as checkout, or users who use a feature, like plotting a graph in Excel, then only analyze those users. In these examples, as soon as the user was exposed to a change, they *triggered* into the experiment because there was some difference. Conditional exposure is a very common triggering scenario; here are some additional examples:

1. A change to checkout: only trigger users who started checkout.
2. A change to collaboration, such as co-editing a document in Microsoft Word or Google Docs: only trigger users participating in collaboration.
3. A change to the unsubscribe screen(s): only trigger users that see these changes.
4. A change to the way the weather answer displays on a search engine results page: only trigger users who issue a query resulting in a weather answer.

Example 3: Coverage Increase

Suppose that your site is offering free shipping to users with more than $35 in their shopping cart and you are testing a lower threshold of $25. A key observation is that the change only impacts users who at some point started checkout with a shopping cart *between* $25 and $35. Users with shopping carts over $35 and those with shopping carts under $25 have the same behavior in Treatment as Control. Only trigger users who see the free shipping offer when they have $25 to $35 in their shopping cart. For this example, we assume that no “advertisement” of the free shipping promotion is on the site; if at some point free shipping displays for the user and it is different between Control and Treatment, that immediately becomes a trigger point.

Figure 20.1 shows this example as a Venn diagram: Control represents offering some users free shipping and Treatment increases coverage to a broader user population. You don’t need to trigger users outside of these groups (as in Example 2), but you also don’t need to trigger users meeting the criteria in both Control AND Treatment because the offer is the same.

Example 4: Coverage Change

Things become a bit more complicated when the coverage isn’t increased, but is changed, as shown in the Venn diagram in Figure 20.2. For example, suppose Control offers free shipping to shoppers with at least $35 in their cart

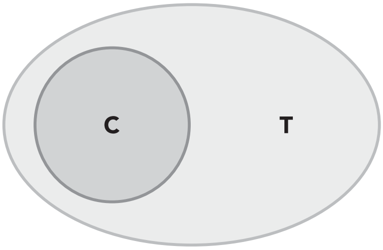

Figure 20.1 Treatment enlarges coverage for a feature. Only users in T\C are triggered. Those in C (and in T) see the same offer, so the Treatment effect is zero

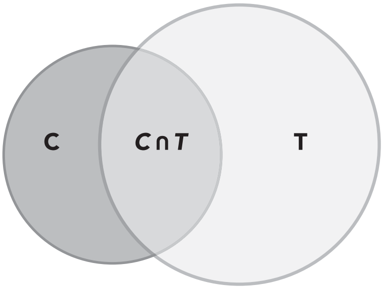

Figure 20.2 Treatment changes the coverage. If users in the intersection see the exact same thing, then only trigger the remaining users

but Treatment offers free shipping to users with at least $25 in their cart except if they returned an item within 60 days before the experiment started.

Both Control and Treatment must evaluate the "other" condition, that is, the counterfactual, and mark users as triggered only if there is a difference between the two variants.

Example 5: Counterfactual Triggering for Machine Learning Models

Suppose you have a machine learning classifier that classifies users into one of three promotions or a recommender model that recommends related products to the one shown on the page. You trained the new classifier or recommender model, and V2 did well in your offline tests. Now you expose to see if it improves your OEC (see Chapter 7).

The key observation is that if the new model overlaps the old model for most users, as when making the same classifications or recommendations for the same inputs, then the Treatment effect is zero for those users. How would you know? You must generate the counterfactual. The Control would run both the model for Control and Treatment and expose users to the Control while logging the Control and Treatment (counterfactual) output; the Treatment would run both the model for Control and Treatment and expose users to the Treatment while logging the output of both models. Users are triggered if the actual and counterfactual differ.

Note that the computational cost in this scenario rises (e.g., the model inference cost doubles with one Treatment) as both machine learning models must be executed. Latency could also be impacted if the two models are not run concurrently and the controlled experiment cannot expose differences in the model's execution (e.g., if one is faster or takes less memory) as both executed.

A Numerical Example (Kohavi, Longbotham et al. 2009)

Given an OEC metric with standard deviation σ and a desired level of sensitivity, Δ, that is, the amount of change you want to detect, the minimum sample size for a confidence level of 95% and power of 80% (van Belle 2008, 31) is as shown in Equation 20.1:

$$n = \frac{16\,\sigma^2}{\Delta^2} \qquad (20.1)$$

Let's take an e-commerce site with 5% of users who visit during the experiment period ending up making a purchase. The conversion event is a Bernoulli trial with $p = 0.05$. The standard deviation, σ, of a Bernoulli is $\sqrt{p(1-p)}$ and thus $\sigma^2 = 0.05(1 - 0.05) = 0.0475$. According to the above formula, you need at least $16 * 0.0475/(0.05 \cdot 0.05)^2 = 121{,}600$ users.

If you made a change to the checkout process, as in Example 2, only analyze triggered users who started the checkout process. Assume that 10% of users initiate checkout, so that given the 5% purchase rate, half of them complete checkout, or $p = 0.5$. The variance $\sigma^2 = 0.5(1 - 0.5) = 0.25$. You therefore need at least $16 * 0.25/(0.5 \cdot 0.05)^2 = 6{,}400$ users to go through checkout. Because 90% of users do not initiate checkout, the number of users in the experiment should be at least 64,000, almost half, thus the experiment could have the same power in about half the time (because there are repeat users, reaching half the users typically takes less than half the time).

Optimal and Conservative Triggering

When comparing two variants, the optimal trigger condition is to trigger into the analysis only users for which there was some difference between the two variants compared, such as between the variant the user was in and the counterfactual for the other variant.

If there are multiple Treatments, ideally information representing all variants is logged, the actual plus all counterfactuals. This then allows for optimal triggering of users who were impacted. However, multiple Treatments can present significant cost as multiple models must be executed to generate the counterfactuals.

In practice, it is sometimes easier to do a non-optimal but conservative triggering, such as including more users than is optimal. This does not invalidate the analysis, but rather loses statistical power. If the conservative trigger does not identify many more users than the ideal trigger, the simplicity tradeoff may be worthwhile. Here are some examples:

1. Multiple Treatments. Any difference between the variants triggers the user into the analysis. Instead of logging the output of each variant, just log a Boolean to indicate that they differed. It is possible that the behavior for Control and Treatment1 was identical for some users but differed for Treatment2. So, when comparing just Control and Treatment1, include users with a known zero Treatment effect.
2. Post-hoc analysis. Suppose the experiment was run and there was something wrong with counterfactual logging, perhaps the recommendation

model used during checkout did not properly log counterfactuals. You can use a trigger condition such as “user-initiated checkout.” While it identifies more users than those for which the recommendation model at checkout differed, it may still remove the 90% of users who never initiated checkout, thus had zero Treatment effect.

Overall Treatment Effect

When computing the Treatment effect on the triggered population, you must dilute the effect to the overall user base, sometimes called diluted impact or side-wide impact (Xu et al. 2015). If you improved the revenue by 3% for 10% of users, did you improve your overall revenue by 10%*3% = 0.3%? NO! (common pitfall). The overall impact could be anywhere from 0% to 3%!

Example 1

If the change was made to the checkout process, the triggered users were those who initiated checkout. If the only way to generate revenue is to initiate checkout, then you improved both triggered and overall revenue by 3% and there is no need to dilute that percentage.

Example 2

If the change was made to very low spenders who spend 10% of the average user, then you improved revenue by 3% for 10% of users who spend 10%, so you improved revenue by 3% of 10% of 10% = 0.03%, a negligible improvement.

- Let $\boldsymbol{\omega}$ denote the overall user universe and let $\boldsymbol{\theta}$ denote the triggered population.
- Let **C** and **T** denote Control and Treatment, respectively.

For a given metric *M*, we have

- $M_{\omega C}$ is the metric value for the untriggered Control.
- $M_{\omega T}$ is the metric value for the untriggered Treatment.
- $M_{\theta C}$ is the metric value for the triggered Control.
- $M_{\theta T}$ is the metric value for the triggered Treatment.

Let N denote the number of users and define $\Delta_\theta = M_{\theta T} - M_{\theta C}$, that is, the absolute effect on the triggered population.

Define $\delta_\theta = \Delta_\theta / M_{\theta C}$, that is, the relative effect on the triggered population.

The triggering rate, $\boldsymbol{\tau}$, is the percent of users that were triggered, is $N_{\theta C}/N_{\omega C}$.

The Treatment can be used instead of Control or can be combined as shown in Equation 20.2:

$$(N_{\theta C} + N_{\theta T})/(N_{\omega C} + N_{\omega T}). \tag{20.2}$$

Here are two ways to think of the diluted percent impact:

1. What is the absolute Treatment effect divided by the total (see Equation 20.3):

$$\frac{\Delta_\theta * N_{\theta C}}{M_{\omega C} * N_{\omega C}} \tag{20.3}$$

2. What is the "ratio of the Treatment effect relative to the untriggered metric" times the triggering rate (see Equation 20.4):

$$\frac{\Delta_\theta}{M_{\omega C}} * \tau \tag{20.4}$$

Because τ is $N_{\theta C}/N_{\omega C}$, we can see that this is equivalent to the prior equation.

What is a common pitfall with diluting by the trigger rate directly? The computation is essentially as shown in Equation 20.5:

$$\frac{\Delta_\theta}{M_{\theta C}} * \tau \tag{20.5}$$

The computation holds when the triggered population is a random sample, but if the triggered population is skewed, as is often the case, then this computation is inaccurate by a factor $M_{\omega C}/M_{\theta C}$.

To dilute ratio metrics, more refined formulas need to be used (Deng and Hu 2015). Note that ratio metrics can cause Simpson's paradox (see Chapter 3), where the ratio in the triggered population improves, but the diluted global impact regresses.

Trustworthy Triggering

There are two checks you should do to ensure a trustworthy use of triggering. We have found these to be highly valuable and they regularly point to issues.

1. Sample Ratio Mismatch (SRM; see Chapter 3).

If the overall experiment has no SRM, but the triggered analysis shows an SRM, then there is some bias being introduced. Usually, the counterfactual triggering is not done properly.

2. Complement analysis. Generate a scorecard for *never* triggered users, and you should get an A/A scorecard (see Chapter 19). If more than the expected metrics are statistically significant, then there is a good chance your trigger condition is incorrect; you influenced users not included in the trigger condition.

Common Pitfalls

Triggering is a powerful concept, but there are several pitfalls to be aware of.

Pitfall 1: Experimenting on Tiny Segments That Are Hard to Generalize

If you are trying to improve a metric for the overall population, then it is the diluted value of your experiment that matters. Even if you improve a metric by a massive 5%, if the triggered population is 0.1% of the overall users, then your diluted value will have $\tau = 0.001$ when you compute the diluted value based on Equation 20.6:

$$\frac{\Delta_\theta}{M_{\omega C}} * \tau \tag{20.6}$$

In computer architecture, Amdahl's law is often mentioned as a reason to avoid focusing on speeding up parts of the system that are a small portion of the overall execution time.

There is one important exception to this rule, which is generalizations of a small idea. For example, in Aug 2008, MSN UK ran an experiment whereby the link to Hotmail opened in a new tab (or new window for older browsers), which increased MSN users' engagement, as measured by clicks/user on the homepage, by 8.9% for the triggered users who clicked the Hotmail link (Gupta et al. 2019). This was a massive improvement, but a relatively small segment. However, over several years a series of experiments were run to generalize this idea, which at the time was very controversial. By 2011, MSN US ran a very large experiment, with over 12 million users, which opened the search results in a new tab/window and engagement as measured by clicks-per-user increased by a whopping 5%. This was one of the best features that MSN ever implemented in terms of increasing user engagement (Kohavi et al. 2014, Kohavi and Thomke 2017).

Pitfall 2: A Triggered User Is Not Properly Triggered for the Remaining Experiment Duration

As soon as a user triggers, the analysis must include them going forward. The Treatment might impact their future behavior because of some difference in the experience. Analyses of triggered users by day or session are susceptible to impact from prior experience. For example, assume that the Treatment provides such a terrible experience that users significantly reduce visits. If you analyze users by day or by session, you will underestimate the Treatment effect. If visits-per-user has not significantly changed statistically, you can get statistical power by looking at triggered visits.

Pitfall 3: Performance Impact of Counterfactual Logging

To log the counterfactual, both Control and Treatment will execute each other's code (e.g., model). If the model for one variant is significantly slower than the other, this will not be visible in the controlled experiment. These two things can help:

1. Awareness of this issue. The code can log the timing for each model so that they can be directly compared.
2. Run an A/A'/B experiment, where A is the original system (Control), A' is the original system with counterfactual logging, and B is the new Treatment with counterfactual logging. If A and A' are significantly different, you can raise an alert that counterfactual logging is making an impact.

It is worth noting that counterfactual logging makes it very hard to use shared controls (see Chapter 12 and Chapter 18), as those shared controls are typically running without code changes. In some cases, triggering conditions can be determined through other means, although this can result in suboptimal triggering or erroneous conditions.

Open Questions

The following are issues that we face where we have no clear answer. Awareness is important, even when we have pros and cons and not the "answer."