Targeting variants for maximum impact

Egor Kraev and Alexander Polyakov

Suppose you want to send an email to your customers or make a change in your customer-facing UI, and you have several variants to choose from. How do you choose the best option?
The naive way would be to run an A/B/N test, showing each variant to a random subsample of your customers and picking the one that gets the best average response. However, this treats all your customers as having the same preferences, and implicitly regards the differences between the customers as merely noise to be averaged over. Can we do better than that, and choose the best variant to show to each customer, as a function of their observable features?
When it comes to evaluating the results of an experiment, the real challenge lies in measuring the comparative impact of each variant based on observable customer features. This is not as simple as it sounds. We're not just interested in the outcome of a customer with specific features receiving a particular variant, but in the impact of that variant, which is the difference in outcome compared to another variant.
Unlike the outcome itself, the impact is not directly observable. For instance, we can't both send and not send the exact same email to the exact same customer. This presents a significant challenge. How can we possibly solve this?
The answer comes at two levels: firstly, how can we assign variants for maximum impact? And secondly, once we've chosen an assignment, how can we best measure its performance compared to purely random assignment?

Comparing performance of different assignments

The answer to the second question turns out to be easier than the first. The naive way to do that would be to split your customer group into two, one with purely random variant assignment, and another with your best shot at assigning for maximum impact - and to compare the results. Yet this is wasteful: each of the groups is only half the total sample size, so your average outcomes are more noisy; and the benefits of a more targeted assignment are enjoyed by only half of the customers in the sample.
Fortunately, there is a better way: firstly, you should make your targeted assignment somewhat random as well, just biased towards what you think the best option is in each case. This is only reasonable as you can never be sure what's best for each particular customer; and it allows you to keep learning while reaping the benefits of what you already know.
Secondly, as you gather the results of that experiment, which used a particular variant assignment policy, you can use a statistical technique called ERUPT or policy value to get an unbiased estimate of the average outcome of any other assignment policy, in particular of randomly assigning variants. Sounds like magic? No, just math. Check out the notebook at ERUPT basics for a simple example.

Finding optimal assignments

Being able to compare the impact of different assignments based on data from a single experiment is great, but how do we find out which assignment policy is the best one? Here again, CausalTune comes to the rescue.

How do we solve the challenge we mentioned above, of estimating the difference in outcome from showing different variants to the same customer - which we can never directly observe? Such estimates are called uplift modeling, by the way, which is a particular kind of causal modeling.

The naive way would be to treat the variant shown to each customer as just another feature of the customer, and fit your favorite regression model, such as XGBoost, on the resulting set of features and outcomes. Then you could look at how much the fitted model's forecast for a given customer changes if we change just the value of the variant "feature", and use that as the impact estimate. This approach is known as the S-Learner. It is simple, intuitive, and in our experience consistently performs horribly.

You may wonder, how do we know that it performs horribly if we can't observe the impact directly? One way is to look at synthetic data, where we know the right answer.

But is there a way of evaluating the quality of an impact estimate on real-world data, where the true value is not knowable in any given case? It turns out there is, and we believe our approach to be an original contribution in that area. Let's consider a simple case when there's only two variants - control (no treatment) and treatment. Then for a given set of treatment impact estimates (coming from a particular model we wish to evaluate), if we subtract that estimate from the actual outcomes of the treated sample, we'd expect to have the exact same distribution of (features, outcome) combinations for the treated and untreated samples. After all, they were randomly sampled from the same population! Now all we need to do is to quantify the similarity of the two distributions, and we have a score for our impact estimate.

Now that you can score different uplift models, you can do a search over their kinds and hyperparameters (which is exactly what CausalTune is for), and select the best impact estimator.

CausalTune supports two such scores at the moment, ERUPT and energy distance. For details, please refer to the original CausalTune paper.

Optimizing assignments for impact with CausalTune

How do you make use of that in practice, to maximize your desired outcome, such as clickthrough rates?

You first select your total addressable customer population, and split it into two parts. You begin by running an experiment with either a fully random variant assignment, or some heuristic based on your prior beliefs. Here it's crucial that no matter how strong those beliefs, you always leave some randomness in each given assignment - you should only tweak the assignment probabilities as a function of customer features, but never let those collapse to deterministic assignments - otherwise you won't be able to learn as much from the experiment!

Once the results of those first experiments are in, you can, firstly, use ERUPT as described above, to estimate the improvement in the average outcome that your heuristic assignment produced compared to fully random. But more importantly, you can now fit CausalTune on

the experiment outcomes, to produce actual impact estimates as a function of customer features!
You then use these estimates to create a new, better assignment policy (either by picking for each customer the variant with the highest impact estimate, or, better still, by using Thompson sampling to keep learning at the same time as using what you already know), and use that for a second experiment, on the rest of your addressable population.
Finally, you can use ERUPT on the results of that second experiment to determine the outperformance of your new policy against random, as well as against your earlier heuristic policy.

Wise case study: optimizing clickthrough rates

Here is a story of one early application in Wise, where we did pretty much that. The objective of the email campaign was to recommend to existing Wise clients the next product of ours that they should try. The first wave of emails used a simple model, where for existing customers we looked at the sequence of the first uses of each product they use, and trained a gradient boosting model to predict the last element in that sequence given the previous elements, and no other data.
In the ensuing email campaign we used that model's prediction to bias the assignments, and got a clickthrough rate of 1.90% - as compared to 1.74% that a random assignment would have given us, according to the ERUPT estimate on the same experiment's results.
We then trained CausalTune on that data, and the out-of-sample result ERUPT forecast was 2.18% (and 2.22% using the Thompson Sampling) - an improvement of 25% compared to random assignment!
We are now preparing the second wave of that experiment to see if the gains forecast by ERUPT will materialize in the real clickthrough rates.

Conclusion

CausalTune gives you a unique, innovative toolkit for optimal targeting of individual customers to maximize the desired outcome, such as clickthrough rates. Our AutoML for causal estimators allows you to reliably estimate the impact of different variants on the customers' behavior, and the ERUPT estimator allows you to compare the average outcome of the actual experiment to that of other assignment options, giving you performance measurement without any loss in sample size.