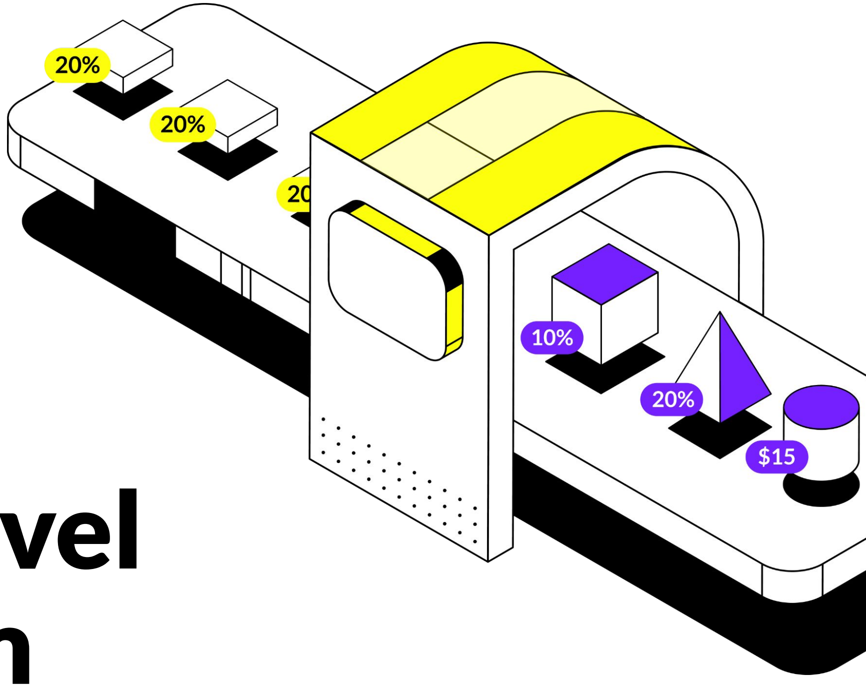
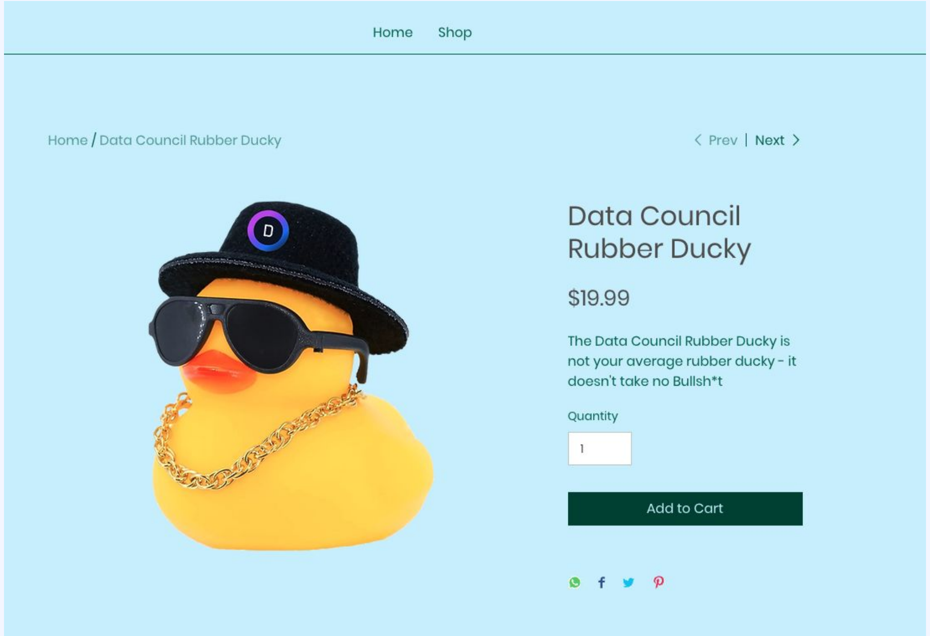


Building a User-Level Targeting Platform

Alex Wood-Doughty
monocle

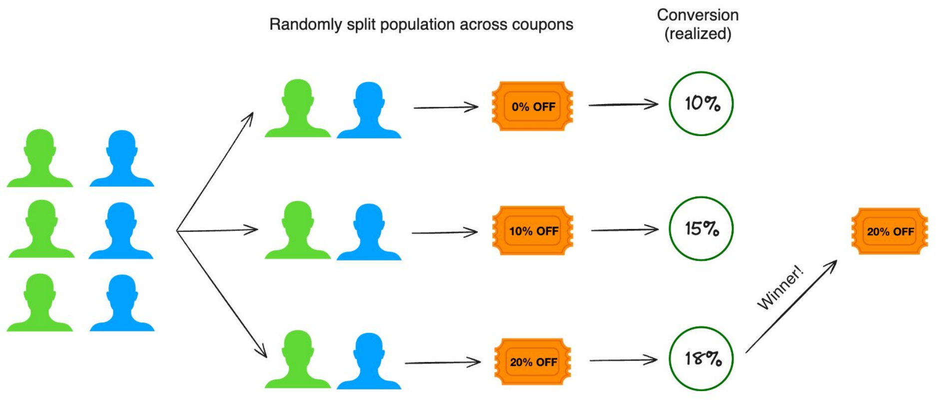
Motivation

- Suppose you run an e-commerce brand selling rubber ducks
- You want to drive growth via coupons but are worried about over-couponing and brand dilution
- How do you choose your coupon strategy?



Option 1: Run an A/B Test

- Run an A/B test where you randomly send coupons to users: (No coupon, 10%-off, 20%-off)
- Analyze the test to see which coupon drives the most impact: E[Y|T]
  - Y: outcome e.g. conversion, revenue, profit
  - T: treatment, e.g. 0%, 10%, 20%
- Ship the best coupon to everyone



Option 1: Run an A/B Test

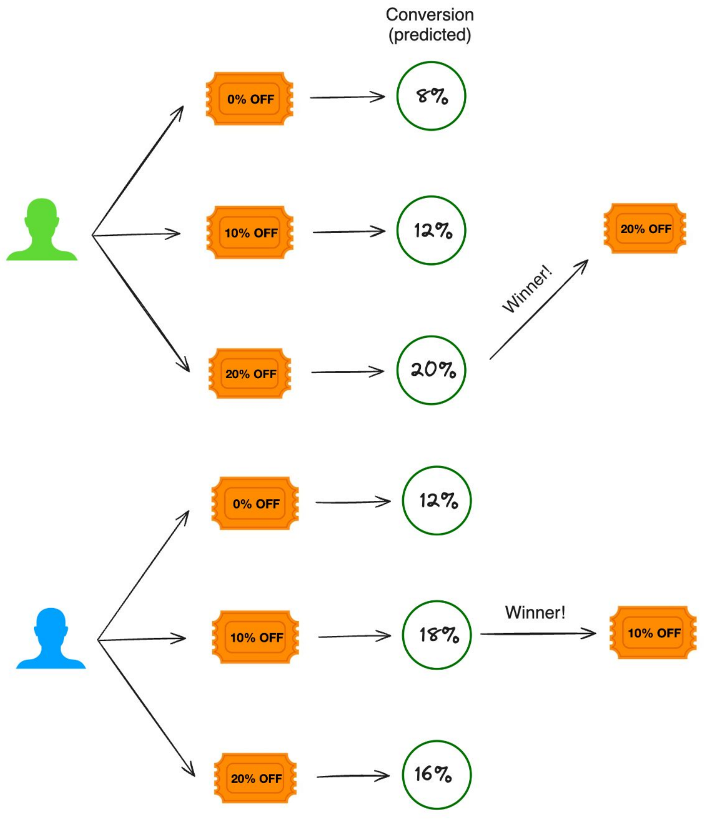
- Run an A/B test where you randomly send coupons to users: (No coupon, 10%-off, 20%-off)
- Analyze the test to see which coupon drives the most impact: E[Y|T]
  - Y: outcome e.g. conversion, revenue, profit
  - T: treatment, e.g. 0%, 10%, 20%
- Ship the best coupon to everyone

What's wrong with this?

1. Assumes average behavior is the best for everyone
2. Assumes observed behavior is consistent over time

Option 2: Run a Bandit

- Use data from the A/B test to train contextual bandit: E[Y|T, X]
  - X: characteristics of each user
  - Bandit lit usually uses: E[r|a, X]
- Use model to predict which coupon to give each user (largest E[Y|T, X])
  - Some kind of optimization + explore
    - e.g. Epsilon Greedy, Thompson Sampling
  - Re-train model over time



Option 2: Run a Bandit

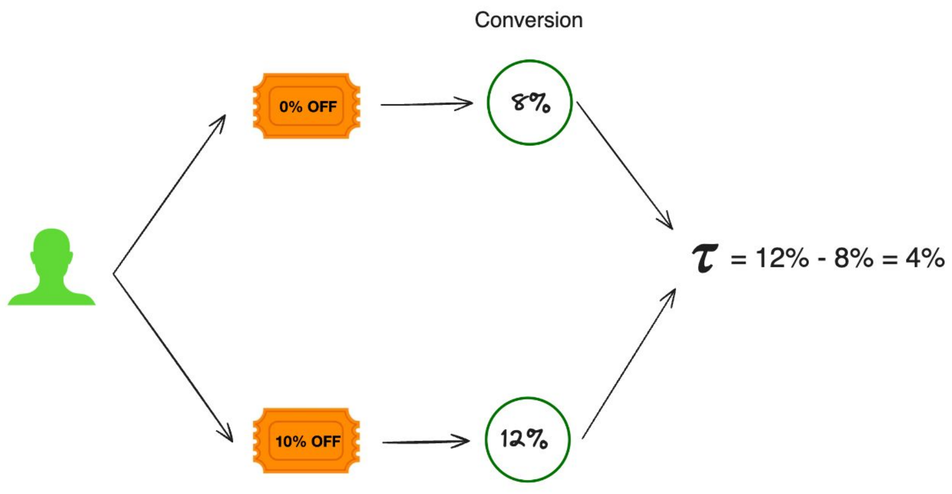
- Use data from the A/B test to train contextual bandit: E[Y|T, X]
  - X: characteristics of each user
  - Bandit lit usually uses: E[r|a, X]
- Use model to predict which coupon to give each user (largest E[Y|T, X])
  - Some kind of optimization + explore
    - e.g. Epsilon Greedy, Thompson Sampling
  - Re-train model over time

What's wrong with this?

- Getting a "good" model of E[Y|T, X] can be tricky
- The effect of X on Y may be much stronger than the effect of T on Y
  - E[Y|0, X] may look very similar to E[Y|10, X]
- What we really care about is the "treatment effect": the difference in Y from 10 vs 0 for a given user (X)

Heterogeneous Treatment Effect Models

- Consider a binary treatment (10% or 0%)
- τ = Y(10%) - Y(0%)
  - For a specific user, how would their outcome change if we gave them a 10%-off coupon instead of 0%
- Interested in estimating E[τ|X]
  - What's the average treatment effect for a user with characteristics X



Heterogeneous Treatment Effect Models

- Consider a binary treatment (10% or 0%)
- τ = Y(10%) - Y(0%)
  - For a specific user, how would their outcome change if we gave them a 10%-off coupon instead of 0%
- Interested in estimating $E[\tau|X]$
  - What's the average treatment effect for a user with characteristics X

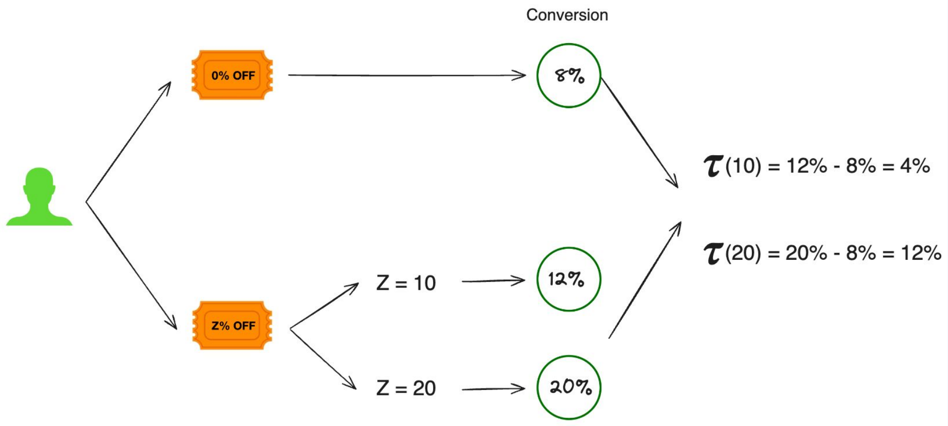
- $E[\tau|X] = E[Y(10) - Y(0)|X]$

  $= E[Y(10)|X] - E[Y(0)|X]$

  $= E[Y|10, X] - E[Y|0, X]$
- Simplest HTE models just model $E[Y|T, X]$
  - S(ingle)-Learner
  - T(wo)-Learner
- But lots of literature showing that we can do better with more sophisticated models
- However, almost all assume a binary treatment, how can we handle multiple coupons?

Extending HTE Models to Multiple Treatments

- E[τ|X] is based on two treatments
- Can frame it as any coupon vs no coupon and then parameterize the coupon details
  - %-off in our example
- Have to be careful about how we incorporate these treatment features into the model
  - Can't just be regular Xs



Extending HTE Models to Multiple Treatments

- $E[\tau|X]$ is based on two treatments
- Can frame it as any coupon vs no coupon and then parameterize the coupon details
  - %-off in our example
- Have to be careful about how we incorporate these treatment features into the model
  - Can’t just be regular Xs

- Neural nets can handle this
  - But they don’t work very well with limited data
- Bayesian trees can do ok with this, but somewhat annoying to implement
- Definitely an area for more research

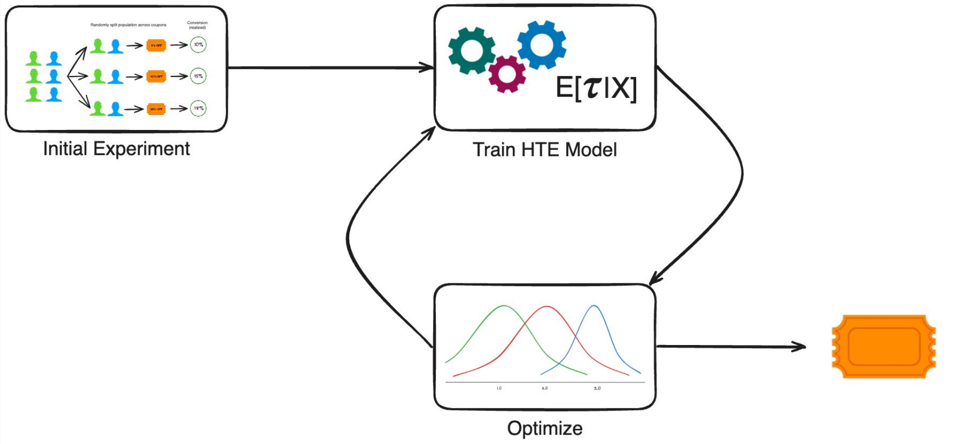
But is this all worth it?

- HTE models require a lot more work, are they actually worth it?
- Important distinctions between incentives and other things (messaging, design)
  - $$$ + trade-offs
  - Have a clear “baseline” treatment that is meaningful to compare against

	Conversion	Net Revenue	Gross Profit	LTV	Brand Dilution
0% OFF	10%	$2	$1	$50	??
10% OFF	15%	$4	$2	$45	??
20% OFF	18%	$3	$0	$40	??

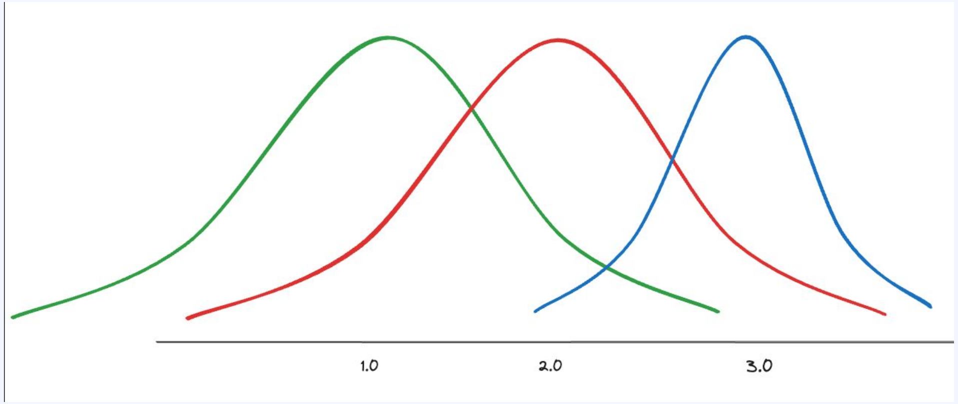
Putting it all together!

- Run initial experiment (randomly allocate three coupons)
- Train initial HTE model on experiment data
- Optimize + send coupons
- Re-train model over time



Optimization

- Can be pretty flexible about how you define an objective function
    - As long as you can model the inputs
- Thompson Sampling to allocate
    - Assume some uncertainty in the model predictions, take draws
    - Efficiently balance explore/exploit
    - Need a model that can handle not-completely-random data

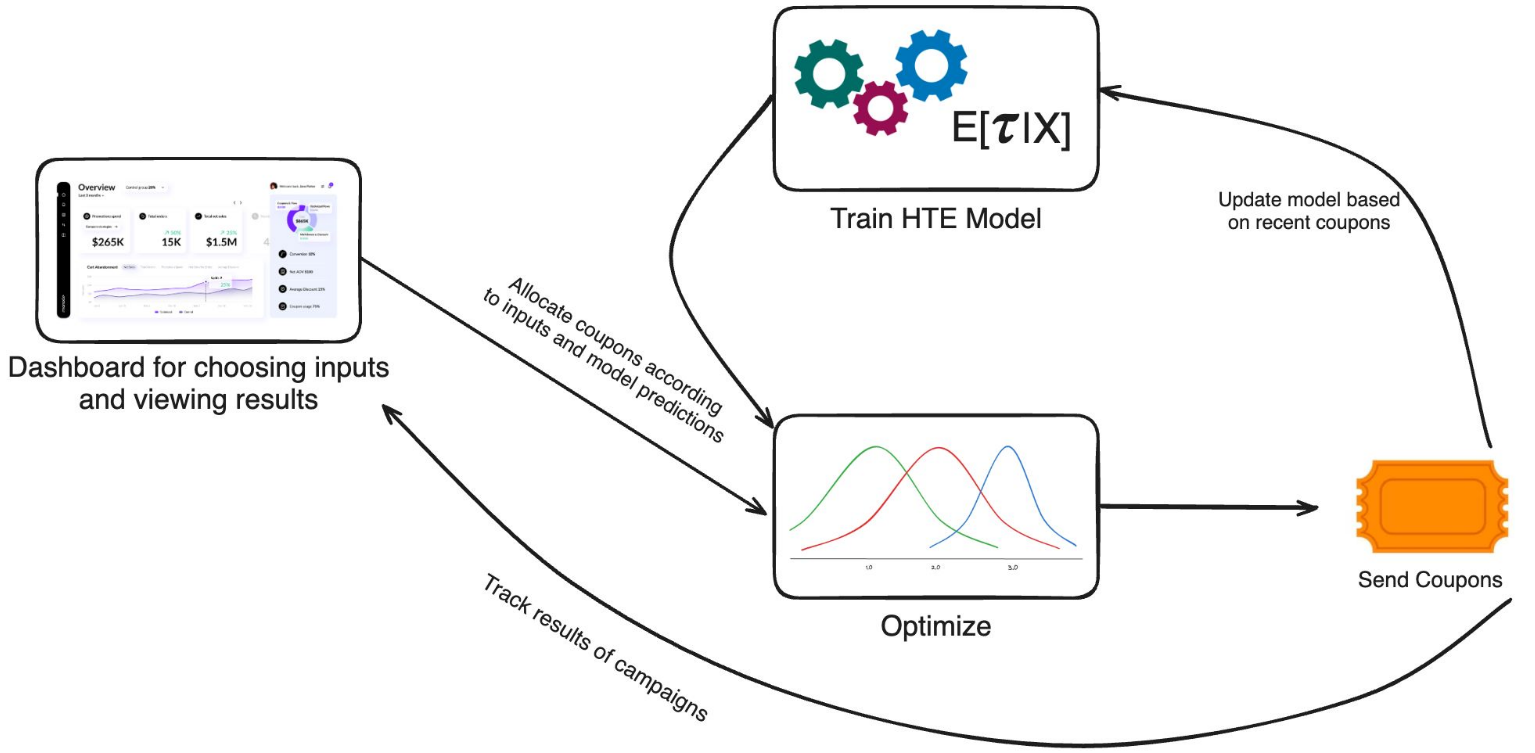


More Fun Stuff

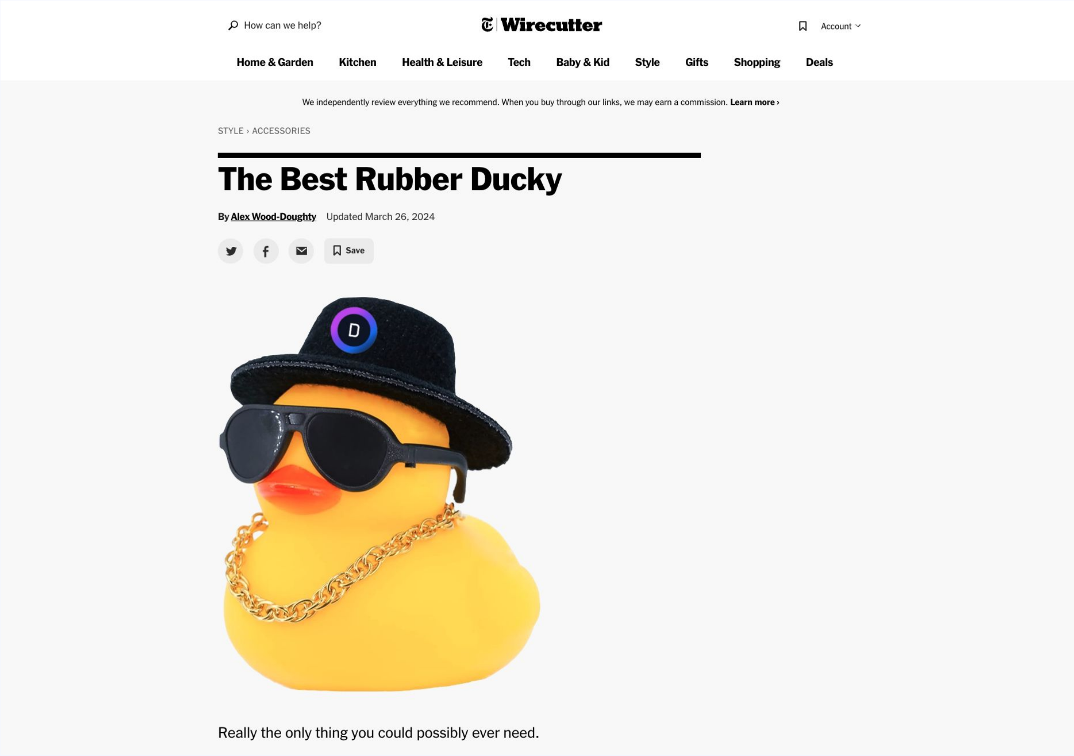
- Adding a new coupon
  - Can cold-start with treatment features
- Model evaluation
  - Hard to evaluate causal models
  - Current methods aren’t great with small-ish data

- Long-term Effects
  - LTV, surrogate models
- Allocating across multiple “campaigns”
  - Here focusing on a single example, but what if we have different ways of engaging with users?
- Cold-starting a new campaign
  - Avoid running initial experiment

Building a Platform



Conclusion



How can we help?

Wirecutter

Account

Home & Garden Kitchen Health & Leisure Tech Baby & Kid Style Gifts Shopping Deals

We independently review everything we recommend. When you buy through our links, we may earn a commission. **Learn more ›**

STYLE › ACCESSORIES

The Best Rubber Ducky

By Alex Wood-Doughty Updated March 26, 2024

Save

Really the only thing you could possibly ever need.

monocle

Platform Verticals (3) Resources (2) Company Contact

Create AI-powered incentives that generate profits

Deploy smart incentives that drive incremental profit without sacrificing brand equity or margins, boosting profits by 35%

TRUSTED BY 50+ HIGH-GROWTH ECOMMERCE EXECUTIVES

chubbies invision Privy melio F2 EVERLANE instacart