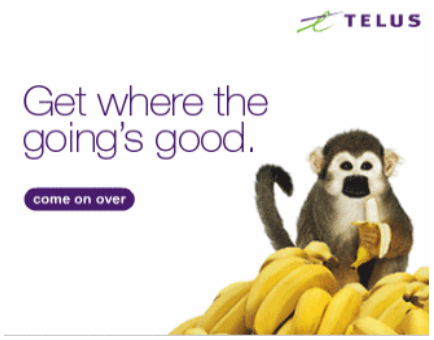
Evaluating Online Ad Campaigns in a Pipeline

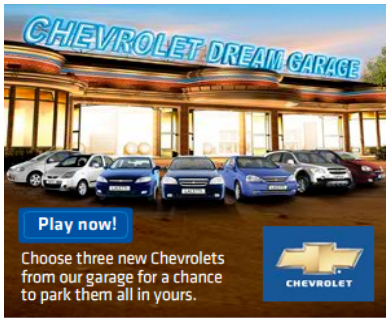
Causal Models At Scale

David Chan, Rong Ge, Ori Gershony,
Tim Hesterberg, Diane Lambert

Google

A display campaign runs





What difference does it make?

How many *MORE* people searched for the brand or visited the brand website *because* of the campaign

Most advertisers want to know this

but they don’t want to experiment

- randomize who does and doesn’t see an ad
  - makes estimating ‘easy’
  - but, they would have to forego showing ads

they want natural experiments

- no restrictions on how a campaign is run
- estimating “how many more” is then hard
  - there is no obvious baseline

Asks

- Don’t require advertisers to run real (aka randomized) experiments
- Still provide a valid estimate of ‘how many more’ whenever possible
- estimate ‘how many more’ over time
- Protect against hidden bias
- Know when estimates are unsafe

Baselines in Natural Experiments

Everyone shown an ad has two potential outcomes

The campaign effect is

$$\Delta = N_E^{-1} \sum_{exposed} (Y_1 - Y_0)$$

The challenge:

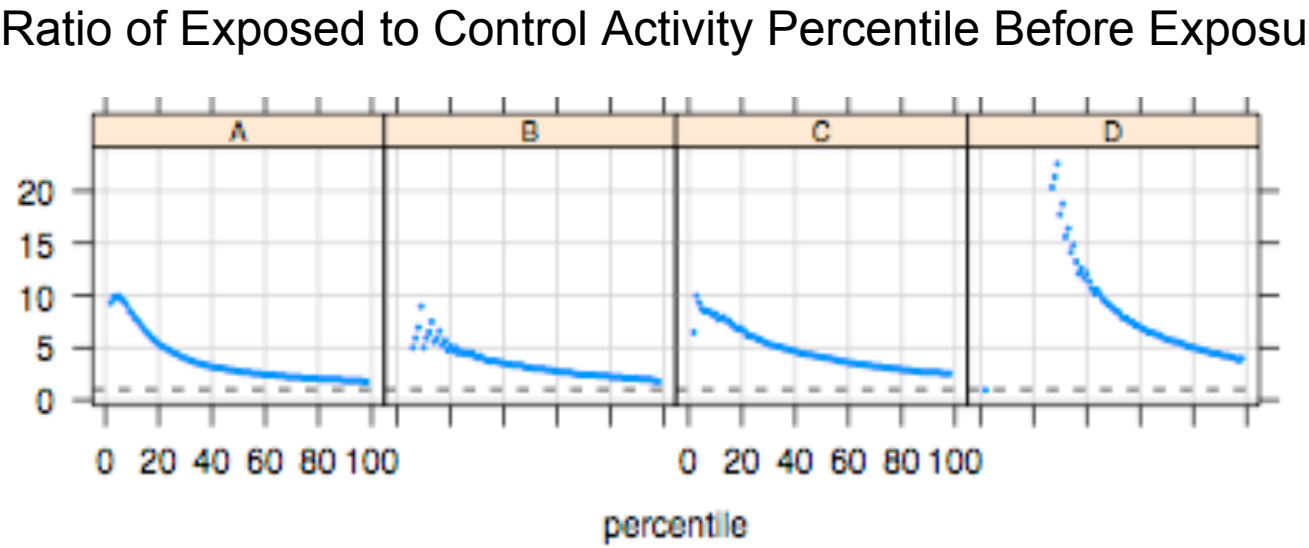
estimate the unobservable baseline $N_E^{-1} \sum_{exposed} Y_0$

Controls aren’t a baseline

Controls *could* have been exposed but weren’t

visited the publisher site, saw other display ads, met targeting conditions

But controls aren’t as active as the exposed



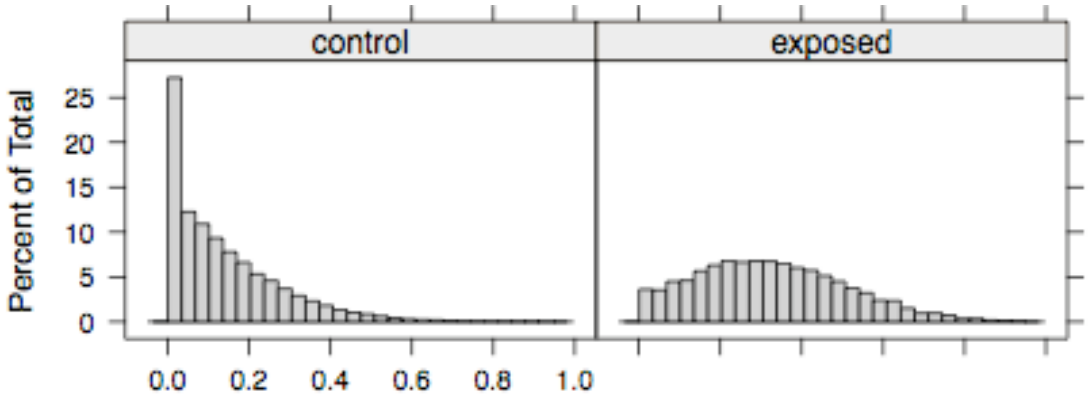
Removing the bias in the controls

1. Match on pre-exposure X

Users alike before exposure would probably be alike later if there is no campaign

2. Match on selection probabilities

Theorem: Matching on P(exposed | **X**) is as good as matching on **X**

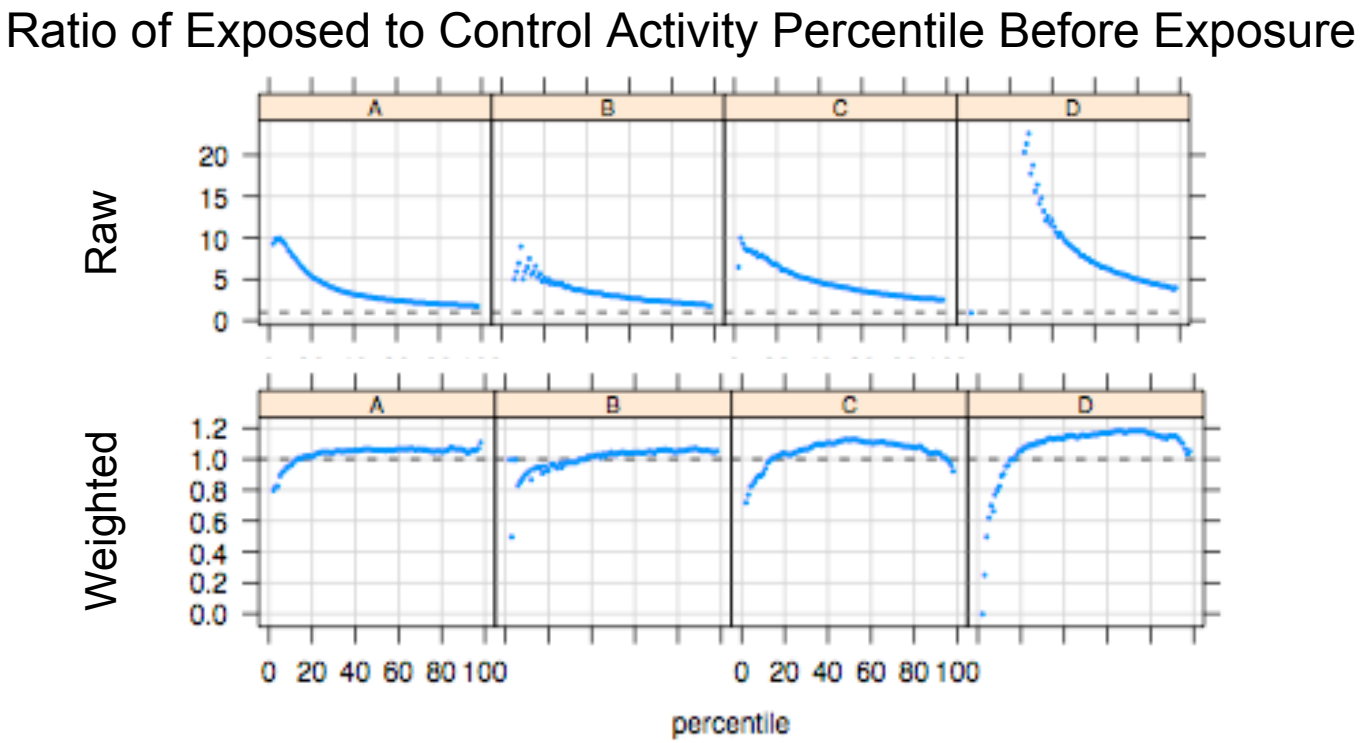


P(exposed | **X**) = selection probability

Instead of matching, re-weight

each exposed gets weight one

each control gets weight p(x) / (1-p(x))



Now there is an obvious estimate of Δ

Use a weighted mean of the control outcomes for the baseline

$$\Delta_{ipw} = \bar{Y}_{exposed} - \sum_{controls} w_i Y_i, \quad with \quad w_i = \frac{p(x_i)/(1-p(x_i))}{\sum_{exposed} p(x_i)/(1-p(x_i))}$$

But Δ_{ipw} is not safe to use in a pipeline

too often has high bias and variance

Improving the weighted estimate

Theorem:

Any consistent estimate of Δ is equivalent to one using weights p(x)/[1-p(x)]

Surprising theorem:

There is an optimal way to use the weights

Loosely, it corrects for the errors in p_i

Doubly Robust Estimate

m_{1i} is a prediction of Y_{1i}

m_{0i} is a prediction of the baseline Y_{0i}

$$\Delta_{DR} = \sum_{i=1}^{n} p_i \left(\frac{Z_i Y_i - (Z_i - p_i) m_{1i}}{p_i} - \frac{(1 - Z_i) Y_i + (Z_i - p_i) m_{0i})}{1 - p_i} \right) \Big/ \sum_{i=1}^{n} p_i$$

The DR estimate is

- optimal if p, m_1 and m_0 have the right form
- consistent if p has the right form, even if m_1 and m_0 do not
- consistent if m_1 and m_0 have the right form, even if p does not
- has an easy standard error estimate

Estimates for The Example Campaign

Δ for	ipw	reg	DR	se(DR)
P(navigates to brand)	.025	.027	.009	.002
P(searches)	.048	.037	.016	.002
P(searches for competitor)	.294	.091	.022	.004

Using logistic regression with variable selection

Simulations based on this dataset show Δ_{DR} 'works'

as claimed whether there is an effect or not

less bias, smaller RMSE

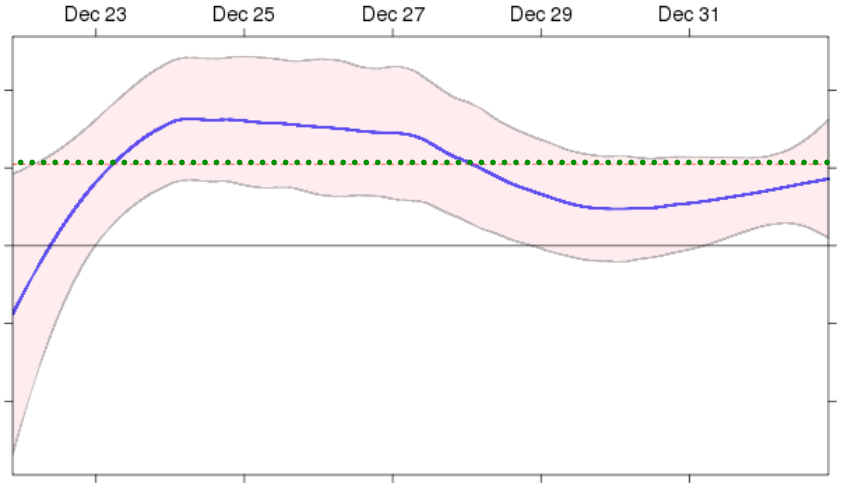
And, DR can estimate how Δ evolves

Weight users first seen near t more heavily

DR theory holds since weights don't depend on Z

$$\Delta_{DR}(w) = \sum_{i=1}^{n} w_i p_i \left(\frac{Z_i Y_i - (Z_i - p_i) m_{1i}}{p_i} - \frac{(1 - Z_i) Y_i + (Z_i - p_i) m_{0i})}{1 - p_i} \right) \Big/ \sum_{i=1}^{n} w_i p_i$$

Incremental Effect Over Time $\Delta_{DR}(t)$



---Δ_{DR} averaged over the entire period

But even Δ_{DR} is not good enough

Hidden bias

An innate difference in the controls and exposed that affects the outcome isn't known

- it can't be included in the models
- it affects any estimate of Δ

Usual diagnostics don't detect hidden bias

- e.g., model fit, effective sample size

$$n_{controls} = \left(\sum_{controls} w_i \right)^2 \Big/ \sum_{controls} w_i^2, \quad w_i = p_i/(1-p_i)$$

Back to the example

Declare a campaign effective if statistically significant at $\alpha = .10$

What is the simulated chance of a false claim if 2 important features omitted from all models?

False alarm rate for Δ_{dr}:

about 20% for 2 outcomes

Protecting Against Hidden Bias

Use the fact that hidden bias affects all outcomes

- outcomes specific to the campaign
- outcomes irrelevant to the campaign
  - recipes is an irrelevant outcome for a campaign for the Chrome web browser

any campaign has many irrelevant outcomes

A Test That Can Withstand Hidden Bias

The test statistic is $T = \Delta_{DR}/se_{DR}$

Compute T^* for the campaign

Compute $T_1,\ldots,T_K$ for irrelevant outcomes

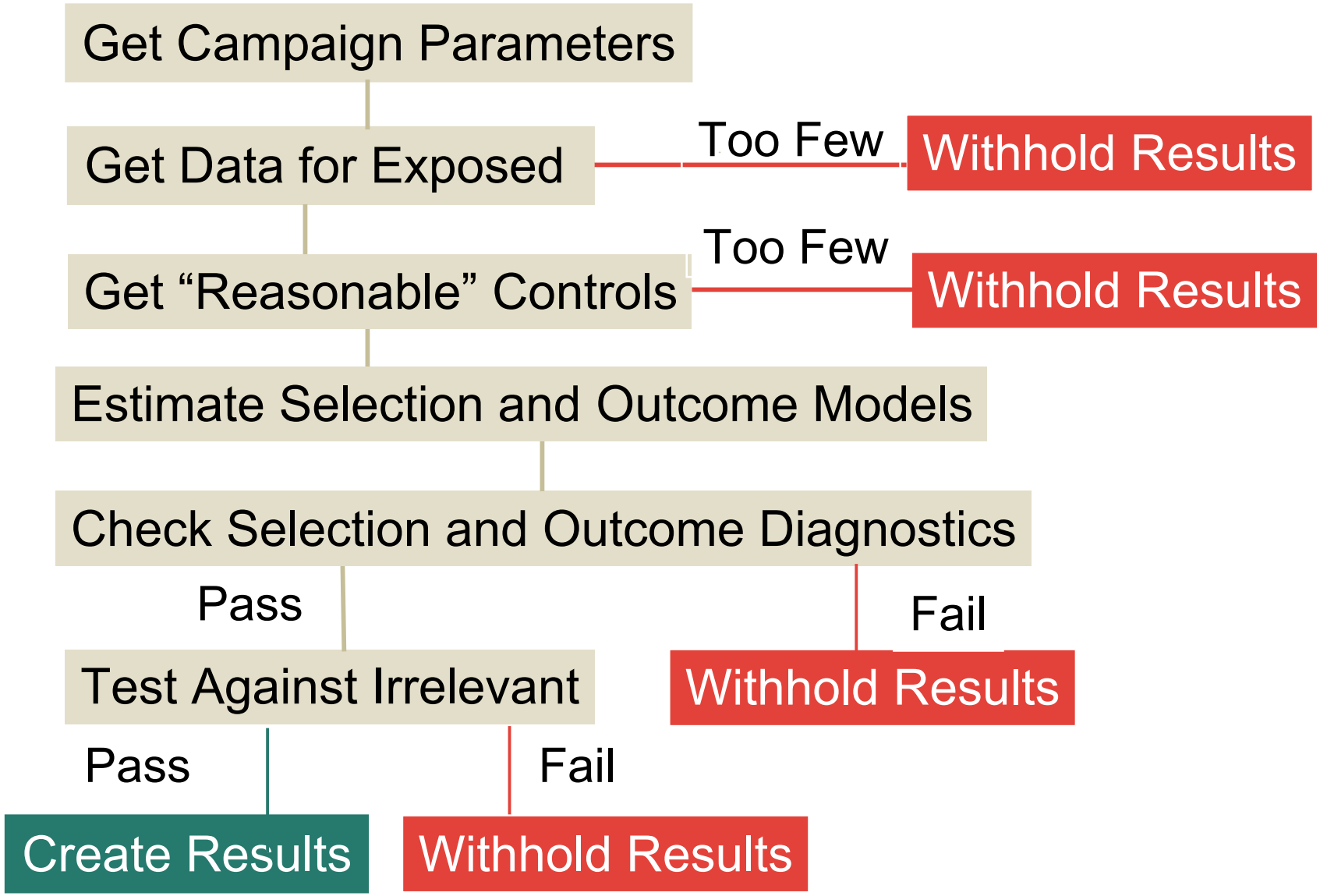
these estimate the null distribution

Nonparametric test of size α

reject if $T^* > (1-\alpha)$ of $(T_1,\ldots,T_K, T^*)$

Gives a conservative, advertiser-friendly test

A Pipeline


Get Campaign Parameters
Get Data for Exposed
Too Few
Withhold Results
Get "Reasonable" Controls
Too Few
Withhold Results
Estimate Selection and Outcome Models
Check Selection and Outcome Diagnostics
Pass
Fail
Test Against Irrelevant
Withhold Results
Pass
Fail
Create Results
Withhold Results

Take aways

Effect estimation from logs/databases is subject to selection bias

Re-weighting ‘unexposed’ or using regression is not enough

- combining them is better, but still not enough

Exploit irrelevant outcomes to protect against hidden selection bias