

Appendix 1 Notation

This table summarizes the main notation used throughout the paper.

Table A-1: Key Notation

Symbol	Meaning
r	Campaign/RCT index
i	User index within campaign
R	Total number of campaigns/RCTs
N_r	Number of users in RCT r
N_{tr}	Number of users in RCT r in the test group
N_{cr}	Number of users in RCT r in the control group
Z_i^r	Random assignment indicator (1 = test, 0 = control)
D_i^r	Exposure indicator (1 = exposed, 0 = not exposed)
C_i^r	Click indicator (1 = clicked, 0 = did not click)
U_i^r	User- and RCT-specific unobservable
α_r	Average baseline outcome (without exposure) in RCT r
τ_r	Average treatment effect of exposure in RCT r
γ_r	Effect of user unobservable on baseline outcomes
δ_r	Causal effect of clicking on outcomes
ϕ_r	Exposure rate in test group of RCT r
$\bar{Y}_{tr}$	Average outcome in test group of RCT r
$\bar{Y}_{cr}$	Average outcome in control group of RCT r
ψ_r	Target causal estimand for PIE prediction
$\mathbf{X}_r^{\text{pre}}$	Pre-determined features for RCT r
$\mathbf{X}_r^{\text{post}}$	Post-determined features for RCT r
ICPD_r	Incremental conversions per dollar for RCT r
$\text{LCC}_r(w)$	Last-click conversions with attribution window w

Appendix 2 Data Details

This section provides additional details on the dataset of Meta ad experiments. Recall that we selected experiments with at least one million users assigned to test and retained outcome metrics with at least 5,000 conversions in the test group. These choices balance representativeness with statistical precision and computational feasibility. Initially, larger thresholds were used during preliminary analysis, after which we relaxed these thresholds to add experiments until the computational time required to extract the raw data became the binding constraint.

In November 2020, Meta (then Facebook) disclosed a bug affecting conversion metrics for about a year (Schiff, 2020). Our data were unaffected because we obtained them from an upstream source in the conversion measurement pipeline.

Figure A-1 presents the empirical distribution of several experiment characteristics. The figure illustrates that we observe significant variation in the mix of experiments in our dataset. Table A-2 displays the distribution of RCTs by the industry vertical of the advertiser.

Figure A-1: Distribution of Experiment Characteristics

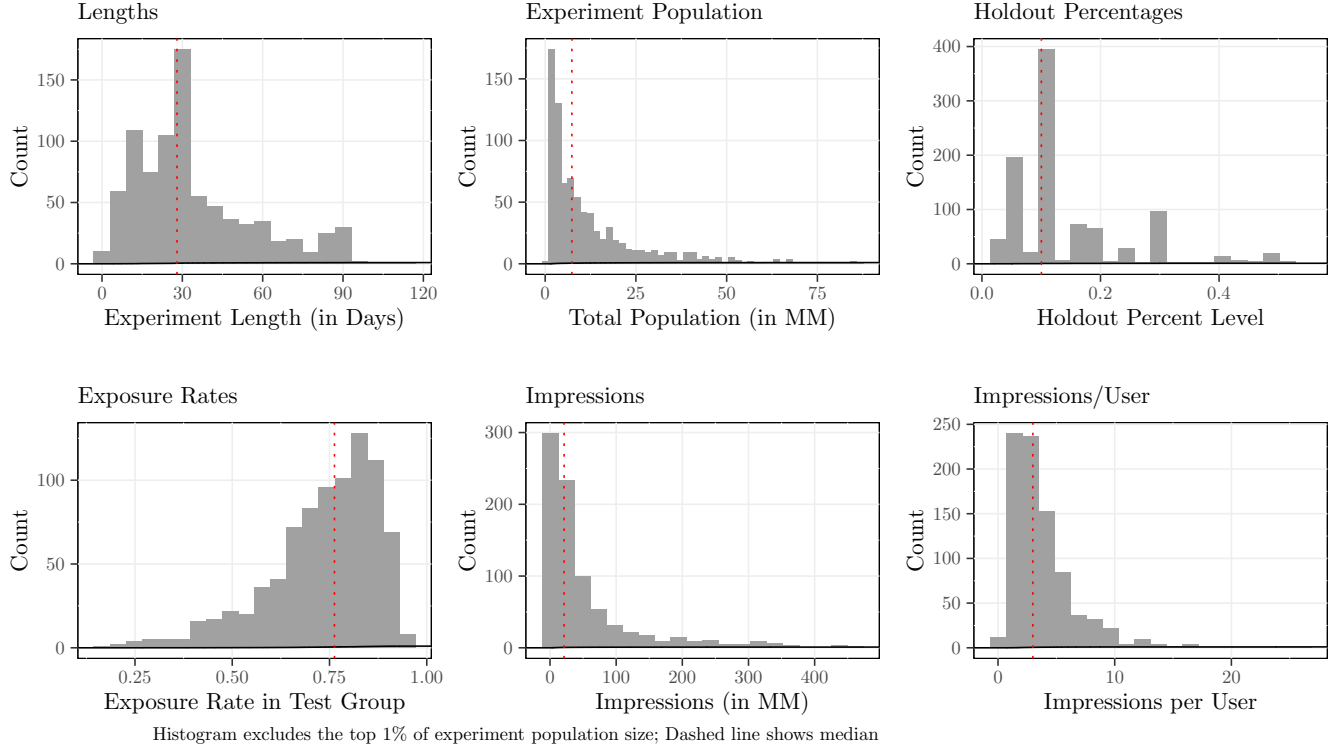


Table A-2: RCT Counts by Vertical

Vertical	N
Ecomm	616
Retail	504
Travel	350
EntMedia	196
CPG	121
Telecom	105
Tech	102
FinServ	83
ProfServ	65
Other	36
Automotive	34
Gaming	14

Appendix 3 Simulation Details

This appendix describes the simulation methodology used to evaluate PIE’s predictive performance. Each simulation constructs potential outcomes for a set of campaigns $r \in \{1, \dots, R\}$, each containing N_r users. We use a linear probability model to generate user-level behaviors, ensuring that

treatment effects are invariant to baseline conversion heterogeneity.

3.1 Campaign-Level Parameter Generation

For each campaign r , we draw the following parameters that govern user behavior and outcomes:

Exposure probability. The probability of exposure p_E^r is drawn from $p_E^r \sim U[p_E^{\text{low}}, p_E^{\text{high}}]$.

Baseline conversion probability and treatment effect. We generate correlated baseline conversion probability α_r and treatment effect τ_r (measured in percentage points) using a Gaussian copula. This method allows us to independently control (1) the marginal distribution of each variable and (2) the correlation between them.

The procedure works as follows. First, we draw (Z_1, Z_2) from a bivariate normal distribution with correlation $\rho_{\alpha\tau}$:

$$\begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} \sim N \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho_{\alpha\tau} \\ \rho_{\alpha\tau} & 1 \end{pmatrix} \right) \quad (1)$$

This gives us two correlated normal random variables. Second, we apply the probability integral transformation to convert each normal variable to a uniform variable on $[0, 1]$:

$$U_1 = \Phi(Z_1), \quad U_2 = \Phi(Z_2) \quad (2)$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function. Because $\Phi(\cdot)$ is a monotonic transformation, U_1 and U_2 remain correlated with approximately the same rank correlation as Z_1 and Z_2 . Moreover, by the probability integral transformation theorem, both U_1 and U_2 are uniformly distributed on $[0, 1]$.

Finally, we transform these uniform variables to our desired marginal distributions using inverse transform sampling:

$$\alpha_r = \alpha^{\text{low}} + (\alpha^{\text{high}} - \alpha^{\text{low}}) \cdot U_1 \quad (3)$$

$$\tau_r = \tau^{\text{low}} + (\tau^{\text{high}} - \tau^{\text{low}}) \cdot U_2 \quad (4)$$

This maps each uniform variable to a uniform distribution on the specified range. These transformations preserve the dependence structure: α_r and τ_r inherit the correlation from Z_1 and Z_2 , while having the desired marginal distributions (both uniform, but with different ranges).

This approach gives us precise control over both the marginal distributions (the ranges from which α_r and τ_r are drawn) and their correlation, which would be difficult to achieve with simple bivariate sampling methods.

Effect of user propensity on outcomes. The parameter γ_r controls how strongly unobservable user propensity U_i^r affects baseline conversion probability. We draw $\gamma_r \sim U[\gamma^{\text{low}}, \gamma^{\text{high}}]$.

Effect of clicks on conversions. The parameter δ_r captures the causal effect of clicking on conversion probability (in percentage points). We draw $\delta_r \sim U[\delta^{\text{low}}, \delta^{\text{high}}]$.

Click probability. The base click probability is drawn as $CP_r \sim U[CP^{\text{low}}, CP^{\text{high}}]$.

3.2 Individual-Level Behavior Generation

For each user i in campaign r , we simulate the following behaviors:

User propensity. We draw unobservable user propensity $U_i^r \sim \text{Uniform}(0, 1)$.

Exposure. Following equation (10), user i is exposed if and only if $U_i^r \geq 1 - p_E^r$:

$$D_i^r = \mathbb{1}\{U_i^r \geq 1 - p_E^r\}$$

Click behavior. The probability that user i clicks (conditional on exposure) depends on user propensity:

$$\Pr(C_i^r = 1 | D_i^r = 1) = CP_r + \beta_{CP} \cdot U_i^r$$

where β_{CP} is a fixed slope parameter we specify in our simulations. Users click only if exposed: $C_i^r = \mathbb{1}\{\text{click}\} \cdot D_i^r$.

Potential outcomes. Using a linear probability model, we generate outcomes for the treatment and control conditions. The underlying probabilities are:

$$\begin{aligned} \text{Prob}_i^{r,T} &= \alpha_r + \gamma_r \cdot U_i^r + \tau_r \cdot D_i^r + \delta_r \cdot C_i^r \\ \text{Prob}_i^{r,C} &= \alpha_r + \gamma_r \cdot U_i^r \end{aligned}$$

We set our simulation parameters such that these probabilities are bounded within $[0, 1]$. Finally, we draw binary outcomes:

$$\begin{aligned} Y_{ti}^r &\sim \text{Bernoulli}(\text{Prob}_i^{r,T}) \\ Y_{ci}^r &\sim \text{Bernoulli}(\text{Prob}_i^{r,C}) \end{aligned}$$

Last-click conversions. We define a last-click conversion as $LCC_i^r = Y_{ti}^r \cdot C_i^r$, indicating that user i both clicked and converted in the treatment condition.

3.3 Aggregation and Sample Construction

We aggregate individual-level data at the campaign level, yielding for each campaign r the number of conversions in each group (Y_{tr}, Y_{cr}), clicks ($C_{tr} \geq 0, C_{cr} = 0$), last-click conversions ($LCC_{tr} \geq 0, LCC_{cr} = 0$), and the ad-exposure rate ($\phi_{tr} \geq 0, \phi_{cr} = 0$).

Sample splitting. To avoid overfitting in small samples, we implement sample splitting within each campaign.²⁶ For each campaign r , we randomly assign the N_r users to two equally-sized samples. Sample 1 is used to calculate the observed incremental conversions $Y_{tr} - Y_{cr}$, which we use

²⁶We thank Ryan Shyu for suggesting this.

to calculate $\hat{\psi}_r$, the ATTs that are the dependent variable in the PIE regression. Sample 2 is used to calculate the predictors (Y_{tr} , ϕ_{tr} , C_{tr} , LCC_{tr}). This ensures that predictors and outcomes are calculated from independent samples, eliminating mechanical correlation due to sampling error.

Train-test split. After generating data for all campaigns and calculating both outcomes and predictors, we split campaigns into equally-sized training and test sets. We estimate PIE specifications on the training sample using OLS regression:

$$\hat{\psi}_r = \theta_0 + \theta_1 \cdot Y_{tr} + \theta_2 \cdot \phi_{tr} + \theta_3 \cdot C_{tr} + \theta_4 \cdot LCC_{tr} + \nu_r \quad (5)$$

where all right-hand side variables are observable from the treatment group alone. Using estimated coefficients from the training sample, we generate predictions for the test sample:

$$\hat{\psi}_r^{\text{PIE}} = \hat{\theta}_0 + \hat{\theta}_1 \cdot Y_{tr} + \hat{\theta}_2 \cdot \phi_{tr} + \hat{\theta}_3 \cdot C_{tr} + \hat{\theta}_4 \cdot LCC_{tr} \quad (6)$$

3.4 Evaluation Metrics

We evaluate prediction accuracy using R^2 , which measures the proportional reduction in mean squared error relative to a naive baseline that predicts the mean treatment effect for every campaign:

$$R^2 = 1 - \frac{\sum_r (\hat{\psi}_r - \hat{\psi}_r^{\text{PIE}})^2}{\sum_r (\hat{\psi}_r - \bar{\psi})^2} \quad (7)$$

where $\bar{\psi}$ is the mean ATT across test campaigns. Higher R^2 indicates that the model substantially outperforms the naive baseline; $R^2 = 0$ indicates no improvement over predicting the mean.

We report R^2 for models with increasingly rich feature sets: total conversions only ($R_{y_t}^2$), adding exposure fraction ($R_{y_t, \phi}^2$), adding clicks ($R_{y_t, \phi, C}^2$), and the full model including last-click conversions ($R_{y_t, \phi, C, LCC}^2$). Each simulation is repeated 1,000 times with different random seeds. For each repetition, we calculate R^2 over the test sample, and we report the mean across repetitions, with the standard deviation across repetitions in parentheses below each mean in the appendix tables.

3.5 Detailed Simulation Results

Number and Size of RCTs Table A-3 shows how predictive performance improves with the number of RCTs available for training. With 50 RCTs, the full model is overfit and achieves $R^2 = -0.296$ (with a standard deviation of 0.25 across repetitions), improving to 0.301 with 1,000 RCTs and 0.319 with 10,000 RCTs. This occurs because larger training samples allow the model to better distinguish signal from noise and avoid overfitting. The gains diminish at larger sample sizes, with the increment from 1,000 to 10,000 RCTs adding little to R^2 . Note that the number of RCTs at which gains diminish is highly dependent on other parameters of the simulation. For example, the larger the treatment effect, the fewer RCTs are needed.

The second panel examines RCT size, holding the number of RCTs fixed at 1,000. Increasing the sample size per RCT from 1,000 to 10,000 users improves R^2 from 0.037 to 0.305. This occurs because larger RCTs produce more precise estimates of ψ_r and the predictor variables in the PIE models.

Table A-3: PIE Performance: Varying Number and Size of RCTs

Scenario	R^2 for PIE model using			
	y_t	y_t, ϕ	y_t, ϕ, C	y_t, ϕ, C, LCC
<i>Panel A: Varying Number of RCTs (size = 10,000)</i>				
50 RCTs	0.222 (0.20)	0.222 (0.20)	0.184 (0.22)	-0.296 (0.25)
100 RCTs	0.270 (0.11)	0.270 (0.11)	0.254 (0.12)	0.130 (0.13)
1,000 RCTs	0.311 (0.03)	0.311 (0.03)	0.309 (0.03)	0.301 (0.03)
10,000 RCTs	0.315 (0.01)	0.315 (0.01)	0.315 (0.01)	0.319 (0.01)
<i>Panel B: Varying Size of RCTs ($N = 1,000$)</i>				
1,000 users	0.040 (0.02)	0.040 (0.02)	0.038 (0.02)	0.037 (0.02)
5,000 users	0.215 (0.03)	0.215 (0.03)	0.214 (0.03)	0.197 (0.03)
10,000 users	0.311 (0.03)	0.311 (0.03)	0.309 (0.03)	0.305 (0.03)

Notes: Panel A varies the number of RCTs while holding RCT size constant at 10,000 users. Panel B varies RCT size while holding the number of RCTs constant at 1,000. All scenarios assume full compliance ($p_E = 1.00$), $\alpha_r \sim U[0.13, 0.17]$, and $\tau_r \sim U[0.02, 0.06]$. Out-of-sample R^2 averaged across 1,000 simulation repetitions; standard deviation across reps in parentheses.

Heterogeneity in Baseline Conversion Rates and Treatment Effects Table A-4 examines how heterogeneity in α_r and τ_r affects the performance of PIE. Panel A varies baseline conversion probability (α) heterogeneity while holding $\tau_r \sim U[0.02, 0.06]$. The full model’s R^2 declines from 0.493 (low heterogeneity) to 0.146 (high). This occurs because greater heterogeneity in α_r increases the variance of Y_{tr} without proportionally increasing our ability to predict ψ_r , making the naive mean-based baseline harder to beat, consistent with what we found in Section 3.4.

Panel B varies treatment effect (τ_r) heterogeneity with $\alpha_r \sim U[0.13, 0.17]$. The full model’s R^2 increases from 0.055 (low heterogeneity) to 0.552 (high). When treatment effects vary substantially, the error of the naive baseline (mean prediction) increases, while the PIE model’s predictions exploit the fact that τ is on both sides of equations 17 and 19.

Table A-4: PIE Performance: Varying Heterogeneity in α_r and τ_r

Scenario	R^2 for PIE model using			
	y_t	y_t, ϕ	y_t, ϕ, C	y_t, ϕ, C, LCC
<i>Panel A: Varying Heterogeneity of Baseline Conversion Rate (α_r)</i>				
Low: $U[0.14, 0.16]$	0.470 (0.03)	0.470 (0.03)	0.469 (0.03)	0.493 (0.03)
Moderate: $U[0.13, 0.17]$	0.310 (0.03)	0.310 (0.03)	0.309 (0.03)	0.317 (0.03)
High: $U[0.12, 0.18]$	0.197 (0.03)	0.197 (0.03)	0.196 (0.03)	0.146 (0.03)
<i>Panel B: Varying Heterogeneity of Treatment Effects (τ_r)</i>				
Low: $U[0.03, 0.05]$	0.057 (0.02)	0.057 (0.02)	0.055 (0.02)	0.055 (0.02)
Moderate: $U[0.02, 0.06]$	0.311 (0.03)	0.311 (0.03)	0.309 (0.03)	0.310 (0.03)
High: $U[0.01, 0.07]$	0.542 (0.03)	0.542 (0.03)	0.541 (0.03)	0.552 (0.03)

Notes: Panel A varies baseline conversion probability range while holding $\tau_r \sim U[0.02, 0.06]$ constant. Panel B varies treatment effect heterogeneity while holding $\alpha_r \sim U[0.13, 0.17]$ constant. All scenarios use 1,000 RCTs with 10,000 users each and full compliance ($p_E = 1.00$). Out-of-sample R^2 averaged across 1,000 simulation repetitions; standard deviation across reps in parentheses.

Correlation Between α_r and τ_r Table A-5 varies the correlation from -0.9 to $+0.9$ while holding the marginal distributions at $\alpha_r \sim U[0.13, 0.17]$ and $\tau_r \sim U[0.02, 0.06]$. Positive correlation improves R^2 substantially, increasing from 0.317 (zero correlation) to 0.601 (correlation = 0.90). This occurs because positive correlation amplifies the systematic signal in Y_{tr} : when campaigns with high baseline conversion rates also have large treatment effects, Y_{tr} provides a strong signal about both components.

Conversely, negative correlation severely degrades performance, with R^2 falling to 0.025 at correlation = -0.90 . Strong negative correlation means that in $Y_{tr} = \alpha_r + \tau_r + \varepsilon_{tr}$, α_r and τ_r nearly

cancel each other out, leaving only noise as an explanatory variable.

Table A-5: PIE Performance: Varying Correlation
Between α_r and τ_r

Correlation	R^2 for PIE model using			
	y_t	y_t, ϕ	y_t, ϕ, C	y_t, ϕ, C, LCC
<i>Panel A: Positive Correlation</i>				
$\rho = 0.00$	0.310 (0.03)	0.310 (0.03)	0.308 (0.03)	0.317 (0.03)
$\rho = 0.45$	0.463 (0.03)	0.463 (0.03)	0.462 (0.03)	0.474 (0.03)
$\rho = 0.90$	0.623 (0.02)	0.623 (0.02)	0.622 (0.02)	0.601 (0.02)
<i>Panel B: Negative Correlation</i>				
$\rho = 0.00$	0.311 (0.03)	0.311 (0.03)	0.309 (0.03)	0.309 (0.03)
$\rho = -0.45$	0.160 (0.03)	0.160 (0.03)	0.158 (0.03)	0.144 (0.03)
$\rho = -0.90$	0.013 (0.01)	0.013 (0.01)	0.011 (0.01)	0.025 (0.01)

Notes: Marginal distributions held constant at $\alpha_r \sim U[0.13, 0.17]$ and $\tau_r \sim U[0.02, 0.06]$. Correlation induced using a Gaussian copula. All scenarios use 1,000 RCTs with 10,000 users each and full compliance ($p_E = 1.00$). Out-of-sample R^2 averaged across 1,000 simulation repetitions; standard deviation across reps in parentheses.

Effect of Clicks on Conversions (δ_r) Table A-6 examines scenarios where clicking has a direct causal effect on conversion probability, parameterized by δ_r .

Under low click probability ($CP_r \sim U[0, 0.1]$, $\beta_{CP} = 0.1$), introducing δ_r heterogeneity has minimal impact, with R^2 remaining around 0.34–0.35 across all scenarios. When few users click, these features contain little information regardless of click effect variation.

Under high click probability ($CP_r \sim U[0, 0.3]$, $\beta_{CP} = 0.5$), R^2 increases from 0.361 to 0.561 as δ_r heterogeneity increases. The incremental contribution of last-click conversions grows substantially with δ heterogeneity, adding effectively zero (in fact, slightly negative, -0.010) under low heterogeneity but a substantial 0.078 under high heterogeneity, reflecting that LCC features are informative only when click effects vary across campaigns.

Selective Exposure and Unobserved Heterogeneity We now examine how PIE performance depends on unobserved heterogeneity in baseline outcomes and selection into exposure. Recall from equation (10) that exposure is determined by $D_i^r = \mathbb{1}\{U_i^r \geq 1 - p_r(Z_i^r)\}$, where $U_i^r \sim \text{Uniform}(0, 1)$ governs selection into exposure. The parameter γ_r controls how strongly U_i^r affects baseline conversion probability. When $\gamma_r = 0$, user propensity determines exposure but has no

Table A-6: PIE Performance: Varying Click Effects on Conversions (δ_r)

Scenario	R^2 for PIE model using			
	y_t	y_t, ϕ	y_t, ϕ, C	y_t, ϕ, C, LCC
<i>Panel A: Low Click Probability, $CP_r \sim U[0.00, 0.10]$, $\beta_{CP} = 0.10$</i>				
Low Het: $\delta_r \sim U[0.04, 0.06]$	0.314 (0.03)	0.314 (0.03)	0.316 (0.03)	0.337 (0.03)
Mod Het: $\delta_r \sim U[0.03, 0.07]$	0.317 (0.03)	0.317 (0.03)	0.318 (0.03)	0.335 (0.03)
High Het: $\delta_r \sim U[0.01, 0.09]$	0.325 (0.03)	0.325 (0.03)	0.327 (0.03)	0.354 (0.03)
<i>Panel B: High Click Probability, $CP_r \sim U[0.00, 0.30]$, $\beta_{CP} = 0.50$</i>				
Low Het: $\delta_r \sim U[0.04, 0.06]$	0.348 (0.03)	0.348 (0.03)	0.371 (0.03)	0.361 (0.03)
Mod Het: $\delta_r \sim U[0.03, 0.07]$	0.380 (0.03)	0.380 (0.03)	0.399 (0.03)	0.449 (0.03)
High Het: $\delta_r \sim U[0.01, 0.09]$	0.473 (0.03)	0.473 (0.03)	0.483 (0.03)	0.561 (0.03)

Notes: Panel A examines low baseline click probability; Panel B examines high click probability. Within each panel, we vary δ_r heterogeneity. Click probability (conditional on exposure) depends on user propensity: $\Pr(C_i^r = 1 | D_i^r = 1) = CP_r + \beta_{CP} \cdot U_i^r$. All scenarios use 1,000 RCTs with 10,000 users each, $\alpha_r \sim U[0.13, 0.17]$, $\tau_r \sim U[0.02, 0.06]$, and full compliance ($p_E = 1.00$). Out-of-sample R^2 averaged across 1,000 simulation repetitions; standard deviation across reps in parentheses.

effect on outcomes. When $\gamma_r > 0$, the same unobservable that determines exposure also affects baseline conversion rates.

i. Effect of γ_r Heterogeneity.

Panel A in Table A-7 shows results with partial selection into exposure ($p_E \sim U[0.4, 0.6]$). The full model's R^2 declines from 0.308 (low γ_r heterogeneity) to 0.260 (high heterogeneity). This is analogous to increasing heterogeneity in the baseline probability α_r : greater heterogeneity in γ_r increases the variance of Y_{tr} without proportionally increasing our ability to predict ψ_r because γ_r is only on the right-hand side of equation 17.

ii. Effect of Exposure Selection Under Different γ_r Scenarios.

Panel B in Table A-7 varies exposure from high ($p_E \sim U[0.7, 0.9]$) to highly selective ($p_E \sim U[0.1, 0.3]$) for the moderate $\gamma_r \sim U[0.10, 0.20]$ case. The full model's R^2 increases from 0.205 for high exposure probability, to 0.277 for medium exposure probability, and 0.397 for low exposure probability.

This substantial improvement occurs because as selection increases, only users with increasingly high U_i^r are exposed, and ϕ_r becomes increasingly informative about user composition, which PIE can exploit relative to a prediction of the mean. The table shows that PIE's use of the exposure fraction (ϕ_r) as a post-determined feature is helpful when the selection mechanism correlates with outcomes. When $\gamma_r > 0$, the unobservable that determines exposure also affects baseline outcomes, and ϕ_r becomes increasingly valuable as a proxy for user composition.

iii. Click Features as Proxies for User Composition.

Panel C demonstrates an important property of click-based features: they improve predictions even when clicks have no direct causal effect on conversions ($\delta_r = 0$). Under moderate unobserved heterogeneity ($\gamma_r \sim U[0.10, 0.20]$) and partial exposure ($p_E \sim U[0.4, 0.6]$), adding click features increases R^2 from 0.040 (using only y_t) to 0.093 (adding ϕ), and further to 0.155 (adding clicks and last-click conversions).

This occurs because under partial exposure, clicks reveal information about user composition. Users with high U_i^r are both more likely to be exposed and, through the γ_r parameter, more likely to convert. Since click probability also depends on U_i^r (via $\Pr(C_i^r = 1 | D_i^r = 1) = CP_r + \beta_{CP} \cdot U_i^r$), click features serve as a proxy for the unobserved user characteristics that drive both exposure and outcomes. This demonstrates that click features can be valuable predictors of treatment effects even in the absence of any causal click-to-conversion pathway.

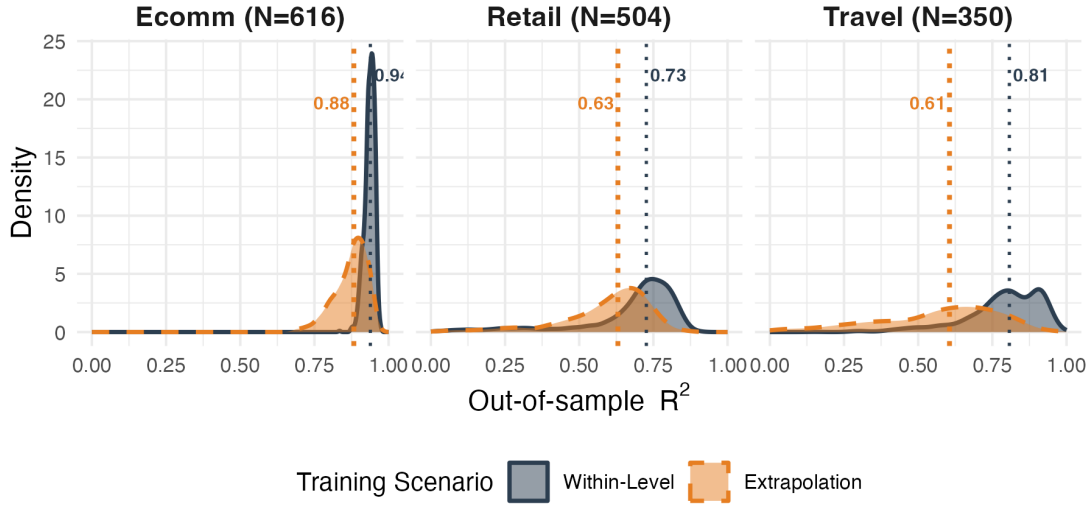
Appendix 4 Additional Extrapolation by Segment Figures

Table A-7: PIE Performance: Heterogeneity γ_r and Exposure Selection

Exposure Range	R^2 for PIE model using			
	y_t	y_t, ϕ	y_t, ϕ, C	y_t, ϕ, C, LCC
<i>Panel A: Varying Heterogeneity in the effect of U_i</i>				
Low Heterogeneity: $\gamma_r \sim U[0.14, 0.16]$	0.144 (0.03)	0.211 (0.03)	0.238 (0.04)	0.308 (0.04)
Moderate Heterogeneity: $\gamma_r \sim U[0.10, 0.20]$	0.074 (0.02)	0.160 (0.03)	0.196 (0.03)	0.265 (0.04)
High Heterogeneity: $\gamma_r \sim U[0.00, 0.30]$	0.011 (0.01)	0.121 (0.03)	0.163 (0.03)	0.260 (0.03)
<i>Panel B: Varying Exposure Selection</i>				
$p_E \sim U[0.70, 0.90]$	0.143 (0.03)	0.173 (0.03)	0.175 (0.03)	0.205 (0.04)
$p_E \sim U[0.40, 0.60]$	0.067 (0.02)	0.161 (0.03)	0.162 (0.03)	0.277 (0.03)
$p_E \sim U[0.10, 0.30]$	0.028 (0.02)	0.370 (0.03)	0.370 (0.03)	0.397 (0.03)
<i>Panel C: Click Features Without Click Treatment Effect ($\delta_r = 0$)</i>				
$\gamma_r \sim U[0.10, 0.20], p_E \sim U[0.40, 0.60]$	0.040 (0.02)	0.093 (0.03)	0.091 (0.03)	0.155 (0.03)

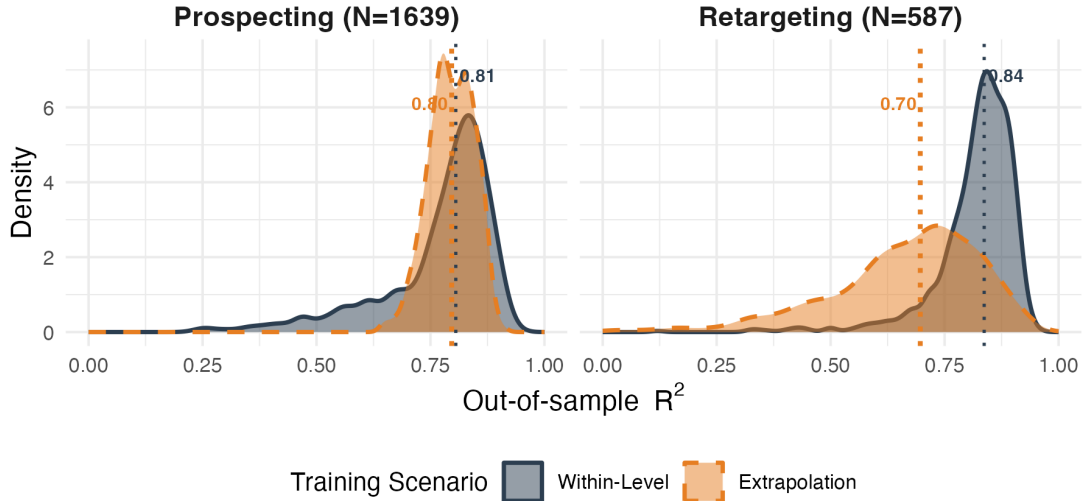
Notes: Panels A and B use $\delta_r \sim U[0.03, 0.07]$. Panel C demonstrates that click features (C and LCC) improve predictions even when clicks have no causal effect on conversions ($\delta_r = 0$), because they proxy for user composition under partial exposure. All scenarios use 1,000 RCTs with 10,000 users each, $\alpha_r \sim U[0.13, 0.17]$, and $\tau_r \sim U[0.02, 0.06]$. Out-of-sample R^2 averaged across 1,000 simulation repetitions; standard deviation across reps in parentheses.

Figure A-2: Extrapolation performance by advertiser vertical



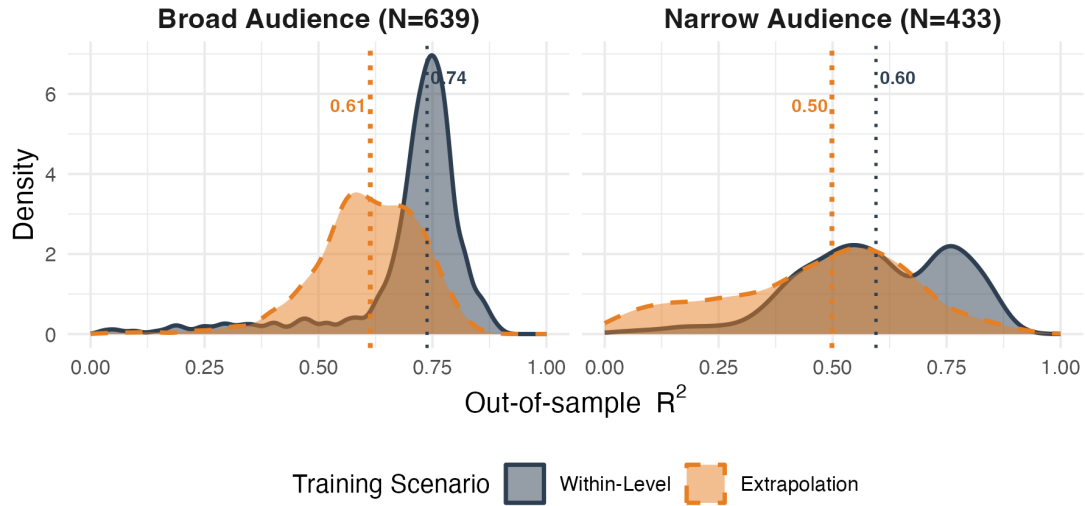
Notes: Empirical distributions of out-of-sample R^2 from the segment-level hold-out experiment, shown here for a subset of advertiser verticals. Each panel corresponds to a vertical (e-commerce, retail, travel), with separate density curves for models trained within the focal group (red) and for models trained by extrapolating from the remaining groups (blue). The distributions are based on 1,000 repeated random partitions of the data, providing a view of how predictive accuracy varies across resamples under the within- and extrapolation-training conditions. Vertical dotted lines indicate the median R^2 within each training scenario.

Figure A-3: Extrapolation performance for prospecting vs. retargeting campaigns



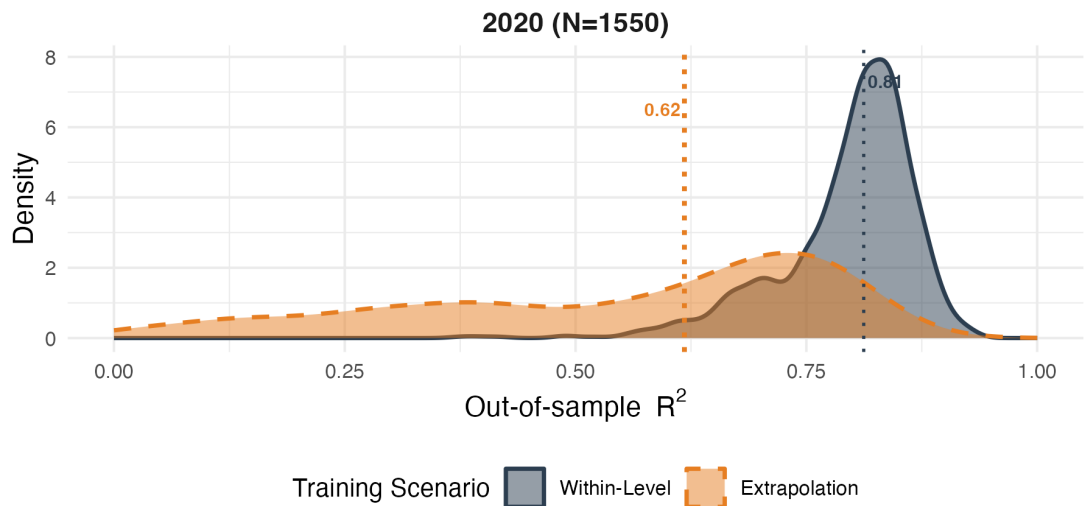
Notes: Empirical distributions of out-of-sample R^2 from the segment-level hold-out experiment, shown here based on whether the campaign is focused on reaching new potential customers (**Prospecting**) or aims to reach existing customers (**Retargeting**). In each panel, there are separate density curves for models trained within the focal group (red) and for models trained by extrapolating from the remaining groups (blue). The distributions are based on 1,000 repeated random partitions of the data, providing a view of how predictive accuracy varies across resamples under the within- and extrapolation-training conditions. Vertical dotted lines indicate the median R^2 within each training scenario.

Figure A-4: Extrapolation performance for custom audience campaigns



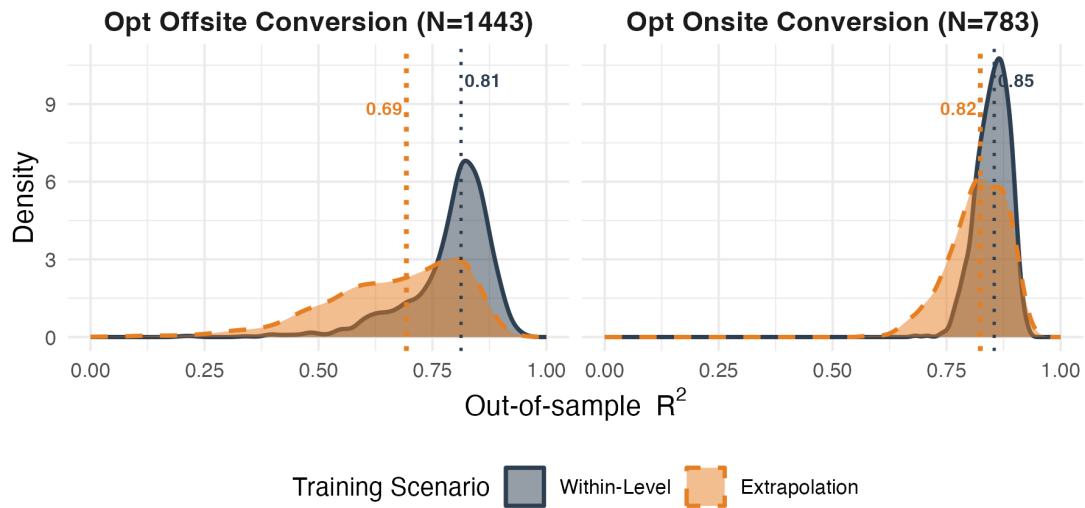
Notes: Empirical distributions of out-of-sample R^2 from the segment-level hold-out experiment, shown here based on whether the campaign targets a **narrow** custom audience or a **broad** custom audience. In each panel, there are separate density curves for models trained within the focal group (red) and for models trained by extrapolating from the remaining groups (blue). The distributions are based on 1,000 repeated random partitions of the data, providing a view of how predictive accuracy varies across resamples under the within- and extrapolation-training conditions. Vertical dotted lines indicate the median R^2 within each training scenario.

Figure A-5: Extrapolation performance training on campaigns in 2019 to extrapolate to campaigns run in 2020



Notes: Empirical distributions of out-of-sample R^2 from the segment-level hold-out experiment. In this case, the within (red) trains a model on campaigns from 2020 and predicts on campaigns from 2020, whereas the extrapolation (blue) trains the model on campaigns from 2019 and predicts on campaigns from 2020. The distributions are based on 1,000 repeated random partitions of the data, providing a view of how predictive accuracy varies across resamples under the within- and extrapolation-training conditions. Vertical dotted lines indicate the median R^2 within each training scenario.

Figure A-6: Extrapolation performance by type of conversion optimization (onsite vs. offsite)



Notes: Empirical distributions of out-of-sample R^2 from the segment-level hold-out experiment, shown here based on whether the campaign is optimizing for offsite conversions or onsite conversions. In each panel, there are separate density curves for models trained within the focal group (red) and for models trained by extrapolating from the remaining groups (blue). The distributions are based on 1,000 repeated random partitions of the data, providing a view of how predictive accuracy varies across resamples under the within- and extrapolation-training conditions. Vertical dotted lines indicate the median R^2 within each training scenario.