Selection Bias in Observational Data: Opt-In vs Randomized Experiments

Justin S. Eloriaga

Emory University

ECON 521

Motivation

- A fitness app launches **streak reminders** to increase weekly workouts.
- Two rollout strategies:
    1. **Observational (opt-in):** users choose to enable reminders.
    2. **Randomized (RCT):** app randomly assigns reminders to 50% of users.
- Claim: If highly motivated users are more likely to opt in, the observational comparison *overstates* the causal effect.

Potential Outcomes Notation

For user i:

- $Y(1)_i$: weekly workouts *if* treated (has reminders).
- $Y(0)_i$: weekly workouts *if not* treated.
- $\Delta_i \equiv Y(1)_i - Y(0)_i$: individual causal effect.
- $D_i \in \{0, 1\}$: opt-in indicator ($1 =$ chose reminders).
- $T_i \in \{0, 1\}$: randomized assignment ($1 =$ assigned reminders).
- v_i: **unobserved motivation** (higher $v_i \Rightarrow$ more likely to opt in and to work out more).

Observed outcome under each regime:

$$Y_i^{\text{obs}} = \begin{cases} D_i Y(1)_i + (1 - D_i) Y(0)_i & \text{(observational)} \\ T_i Y(1)_i + (1 - T_i) Y(0)_i & \text{(randomized)} \end{cases}$$

Estimands: What We Compare

Selection-Distorted Observational Difference (SDO):

$$\text{SDO} \equiv \mathbb{E}[Y^{\text{obs}} \mid D = 1] - \mathbb{E}[Y^{\text{obs}} \mid D = 0] = \mathbb{E}[Y(1) \mid D = 1] - \mathbb{E}[Y(0) \mid D = 0].$$

Estimands: What We Compare

Selection-Distorted Observational Difference (SDO):

$$\mathrm{SDO} \equiv \mathbb{E}[Y^{\mathrm{obs}} \mid D = 1] - \mathbb{E}[Y^{\mathrm{obs}} \mid D = 0] = \mathbb{E}[Y(1) \mid D = 1] - \mathbb{E}[Y(0) \mid D = 0].$$

Average Treatment Effect (ATE, via RCT):

$$\mathrm{ATE} \equiv \mathbb{E}[Y^{\mathrm{obs}} \mid T = 1] - \mathbb{E}[Y^{\mathrm{obs}} \mid T = 0] = \mathbb{E}[Y(1) - Y(0)] \quad \text{since } T \perp \{Y(0), Y(1)\}.$$

Why SDO is Biased (Decomposition)

$$\begin{aligned}\mathrm{SDO} &= \mathbb{E}[Y(1) \mid D=1] - \mathbb{E}[Y(0) \mid D=0] \\ &= \underbrace{\mathbb{E}[Y(1) - Y(0) \mid D=1]}_{\text{effect among opt-ins}} + \underbrace{\Big(\mathbb{E}[Y(0) \mid D=1] - \mathbb{E}[Y(0) \mid D=0]\Big)}_{\textbf{selection term}}.\end{aligned}$$

- If motivated users (v high) both opt in ($D=1$) and have higher baselines $Y(0)$, then the **selection term** > 0.
- Result: SDO $>$ ATE on average (upward bias).

Simulation Design

- Heterogeneous effects: $\Delta_i \sim \mathcal{N}(0.5,\, 0.2^2)$.
- Draw (e_{0i}, e_{1i}, v_i) from a trivariate normal with positive correlations.
- Potential outcomes: $Y(0)_i = e_{0i}$, $Y(1)_i = e_{1i} + \Delta_i$.
- **Opt-in rule:** $D_i = \mathbf{1}\{v_i > 0\}$.
- **RCT rule:** $T_i \sim \text{Bernoulli}(0.5)$.

Python: Imports & Seed

```
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(7)
```

Python: One Large Run + Bias Decomposition

```
mu = np.array([0.0, 0.0, 0.0])
Sigma = np.array([[1.0, 0.6, 0.4],
                  [0.6, 1.0, 0.4],
                  [0.4, 0.4, 1.0]])
n = 100_000

delta = np.random.normal(loc=0.5, scale=0.2, size=n)
E = np.random.multivariate_normal(mu, Sigma, size=n)
e0, e1, v = E[:,0], E[:,1], E[:,2]

y0 = e0
y1 = e1 + delta

D = (v > 0).astype(int)
T = np.random.binomial(1, 0.5, size=n)
```

Python: One Large Run + Bias Decomposition

```
y_obs_obs = D*y1 + (1-D)*y0
y_obs_rct = T*y1 + (1-T)*y0

SDO = y_obs_obs[D==1].mean() - y_obs_obs[D==0].mean()
ATE = y_obs_rct[T==1].mean() - y_obs_rct[T==0].mean()
true_ATE = delta.mean()

effect_optins = (y1 - y0)[D==1].mean()
selection_term = y0[D==1].mean() - y0[D==0].mean()

print(f"SDO: {SDO:.3f}, ATE: {ATE:.3f}, true : {true_ATE:.3f}")
print("Decomp:",
      f"E[Y(1)-Y(0)|D=1]={effect_optins:.3f},",
      f"Sel={selection_term:.3f}")
```

Python: Repeat Experiments (10k Sims)

```
nsample = 1000
nsim = 10_000
sdo_vals, ate_vals, true_deltas = [], [], []

for _ in range(nsim):
    delta = np.random.normal(0.5, 0.2, nsample)
    E = np.random.multivariate_normal(mu, Sigma, size=nsample)
    e0, e1, v = E[:,0], E[:,1], E[:,2]
    y0 = e0
    y1 = e1 + delta

    D = (v > 0).astype(int)
    T = np.random.binomial(1, 0.5, size=nsample)

    y_obs_obs = D*y1 + (1-D)*y0
    y_obs_rct = T*y1 + (1-T)*y0
```

Python: Repeat Experiments (10k Sims)

```python

    sdo_vals.append(y_obs_obs[D==1].mean() - y_obs_obs[D==0].mean())
    ate_vals.append(y_obs_rct[T==1].mean() - y_obs_rct[T==0].mean())
    true_deltas.append(delta.mean())

sdo_vals = np.array(sdo_vals)
ate_vals = np.array(ate_vals)
true_deltas = np.array(true_deltas)

print(f"Mean SDO: {sdo_vals.mean():.3f}")
print(f"Mean ATE: {ate_vals.mean():.3f}")
print(f"Mean true : {true_deltas.mean():.3f}")
```

Python: Plot Distributions

```
# SDO histogram
fig, ax = plt.subplots(figsize=(6,4))
ax.hist(sdo_vals, bins=50)
ax.axvline(true_deltas.mean(), linestyle='--')
ax.set_title("Distribution of SDO (observational opt-in)")
ax.set_xlabel("Estimated effect"); ax.set_ylabel("Frequency")
plt.show()

# ATE histogram
fig, ax = plt.subplots(figsize=(6,4))
ax.hist(ate_vals, bins=50)
ax.axvline(true_deltas.mean(), linestyle='--')
ax.set_title("Distribution of ATE (randomized)")
ax.set_xlabel("Estimated effect"); ax.set_ylabel("Frequency")
plt.show()
```

Figure Placeholders

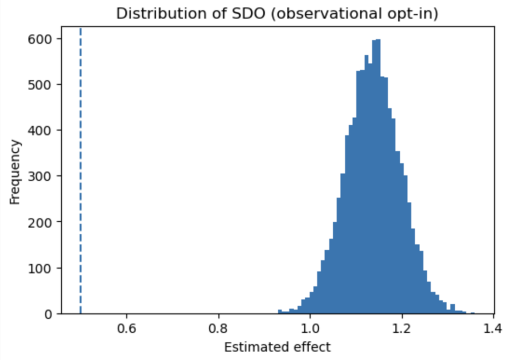

Figure 1: Paste SDO histogram. Dashed line = mean true effect.

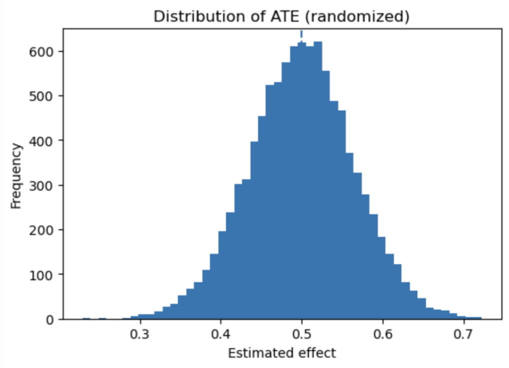

Figure 2: Paste ATE histogram. Dashed line = mean true effect.

Interpretation

- **Observational (opt-in):** D correlates with unobserved v, which also raises $Y(0)$.
- $\Rightarrow \mathbb{E}[Y(0) \mid D = 1] > \mathbb{E}[Y(0) \mid D = 0] \Rightarrow$ positive **selection term**.
- **Randomization:** $T \perp \{Y(0), Y(1)\} \Rightarrow$ treated vs control difference recovers $\mathbb{E}[Y(1) - Y(0)]$.

Takeaways

- $Y(1)$ and $Y(0)$ are the **counterfactual** outcomes; you only ever observe one.
- **SDO** mixes treatment effect with selection on unobservables.
- **RCT ATE** centers on the true causal effect under random assignment.
- If RCT is infeasible, consider strong quasi-experimental designs (IV, RD, DID with good parallel-trends, matching with rich covariates, etc.).