

Bridging the Causal Inference Gap: Using Simulations and Probabilistic Programming to Drive Causal Inference Adoption

Dr. Juan Orduz

Abstract

One of the most significant barriers to implementing causal inference in industry is not technical complexity, but rather organizational resistance rooted in unfamiliarity with causal thinking. While some (not all) data scientists understand the mathematical foundations of causal inference, translating these concepts to non-technical stakeholders—executives, product managers, and domain experts—remains a persistent challenge. This talk explores how simulation-based approaches using probabilistic programming languages can serve as powerful pedagogical tools to demonstrate causal concepts, build organizational understanding, and secure buy-in for causal inference initiatives.

Simulations prove particularly valuable in scenarios where traditional A/B testing falls short. Consider marketing funnel effects, where upper-funnel brand marketing (TV advertising) influences lower-funnel performance marketing (Google Ads), creating complex confounding relationships. Using probabilistic programming, we can simulate these multi-channel attribution problems and demonstrate how naive correlation analysis might incorrectly attribute sales increases to Google Ads when the true driver is TV advertising.

Similarly, in logistics and operations research, switchback experiments—where interventions are applied across time periods or geographic regions rather than individual users—present unique analytical challenges. Simulations can illustrate how temporal correlation and spillover effects complicate causal inference, helping stakeholders understand why specialized methods are needed beyond simple before-and-after comparisons.

The Power of Explicit Data Generation

Traditional approaches to explaining causal inference often rely on abstract theoretical frameworks that can alienate non-technical audiences. By contrast, simulations with explicit data generating processes provide intuitive, visual demonstrations of causal relationships. Using probabilistic programming languages like NumPyro and PyMC, we can construct transparent models where the true causal structure is known by design, enabling stakeholders to witness firsthand how confounders bias naive analyses and how proper causal methods recover true effects.

Leveraging Observe and Do Operators

Modern probabilistic programming frameworks offer unique advantages through their **observe** and **do** operators, which directly map to Pearl's causal hierarchy. The **observe** operator demonstrates passive observation (correlation), while the **do** operator showcases active intervention (causation). This clear distinction helps non-technical stakeholders understand why "correlation is not causation" and why A/B tests and observational studies require different analytical approaches.

Talk Outline

This talk will present practical strategies for using simulation-based education to drive causal inference adoption:

- **Use simulations to create intuition:** Instead of using mathematical notation, use simulations to create intuition.
- **Workshop to define causal DAGS:** Meet with different stakeholders to draft causal DAGS relevant to the domain. It is always interesting to see the different perspectives and how the causal directions differ across different domains.
- **Simulated Case Studies:** Use one simple component of the causal model to illustrate the causal concepts. For example, confounders, selection bias, etc. Emphasize that ignoring these can lead to biased estimates, which ultimately lead to bad decisions with significant financial implications.
- **User Real Data to simulate future scenarios:** This is especially useful to understand change in strategies like pricing or structural product changes.
- For the first practical application do not aim for the best solution but think about strategic incremental wins to get the trust and business value from the causal inference initiatives.