agilelab

www.agilelab.it

Causal ML for Smarter Advertising Campaigns with Python

Swiss Python Summit 2025

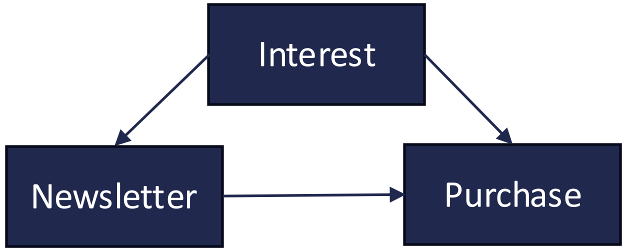
Francesco Conti, Data Scientist @ Agile Lab

Agenda

- Why Does Causality Matter?
- Effect Estimation of Advertising Campaigns Using Causal Inference
- Targeting the Right Customers with Causal ML

Why Does Causality Matter?

Making Decisions



Should I send my newsletter to more customers?

You notice a positive correlation between newsletter readers and those who buy more.

⚠ **Potential Pitfalls:**
Correlation does not imply causation: What if both newsletter reading and buying are driven by a third factor?

Intervention is missing: We are just observing, we need to understand what will happen if we send the newsletter.

How would data change with our intervention?

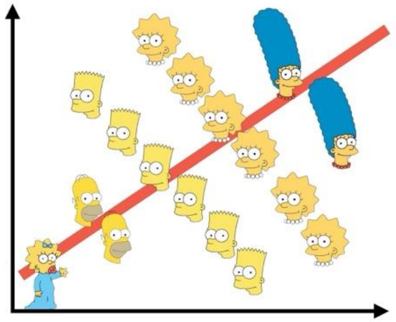
Prescription: Causality matters whenever decisions require intervention, not just observation.

Making Decisions in Digital Marketing

Effect Estimation: *Did my anti-churn marketing promotion have a positive impact on my customers?*

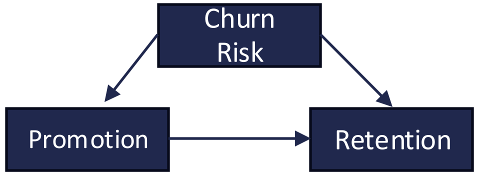
Segment	Total	Low Risk	High Risk
Contacted	26% (260/1000) 😭	10% (20/200) 🥰	30% (240/800) 🥰
Not Contacted	24% (240/1000) 🥰	20% (160/800) 😭	40% (80/200) 😭

Churn rate = (# customers who churned) / (total customers)



Simpson's Paradox

Confounding Bias



Churn Risk

Promotion

Retention

> Causality helps measure the impact of decisions without bias

Targeting: *Which customers should I offer an anti-churn promotion to?*

Predictive Models
Who's going to churn next?

Is prediction enough to make decisions?

What happens after our intervention?

Consider customers with a
high probability of churn.

Still Churn

Don't Churn

> Causality helps in targeting the right customers based on potential effect

How can we produce causal estimates?

Measuring the Treatment Effect on Outcome

$$T \in \begin{cases} 0 & \text{control} \\ 1 & \text{treated} \end{cases}$$

Binary treatment:
Discount, promotion, …

$$y^{(i)}$$

Observed or potential outcome
Retention, purchases, …

$$\tau^{(i)} = y^{(i)}(T=1) - y^{(i)}(T=0)$$

Individual Treatment Effect

Takeaways:
What would happen with the treatment versus without it.
The catch? You'll never observe both.

$$ATT = E[Y_1 - Y_0 | T = 1]$$

Average treatment effect on the treated

Takeaways:
An average effect across treated units.
Still, we can't measure it directly, but it can be estimated.

$$E[Y|T=1] - E[Y|T=0] = \underbrace{E[Y_1 - Y_0|T=1]}_{ATT} + \underbrace{\{E[Y_0|T=1] - E[Y_0|T=0]\}}_{BIAS}$$

Takeaways:
The very essence of causal inference

Measuring the Treatment Effect on Outcome

$$T \in \begin{cases} 0 & \text{control} \\ 1 & \text{treated} \end{cases}$$

outcome

..

$$\tau^{(i)} = y^{(i)}(T=1) - y^{(i)}(T$$

ment versus without it.

h.

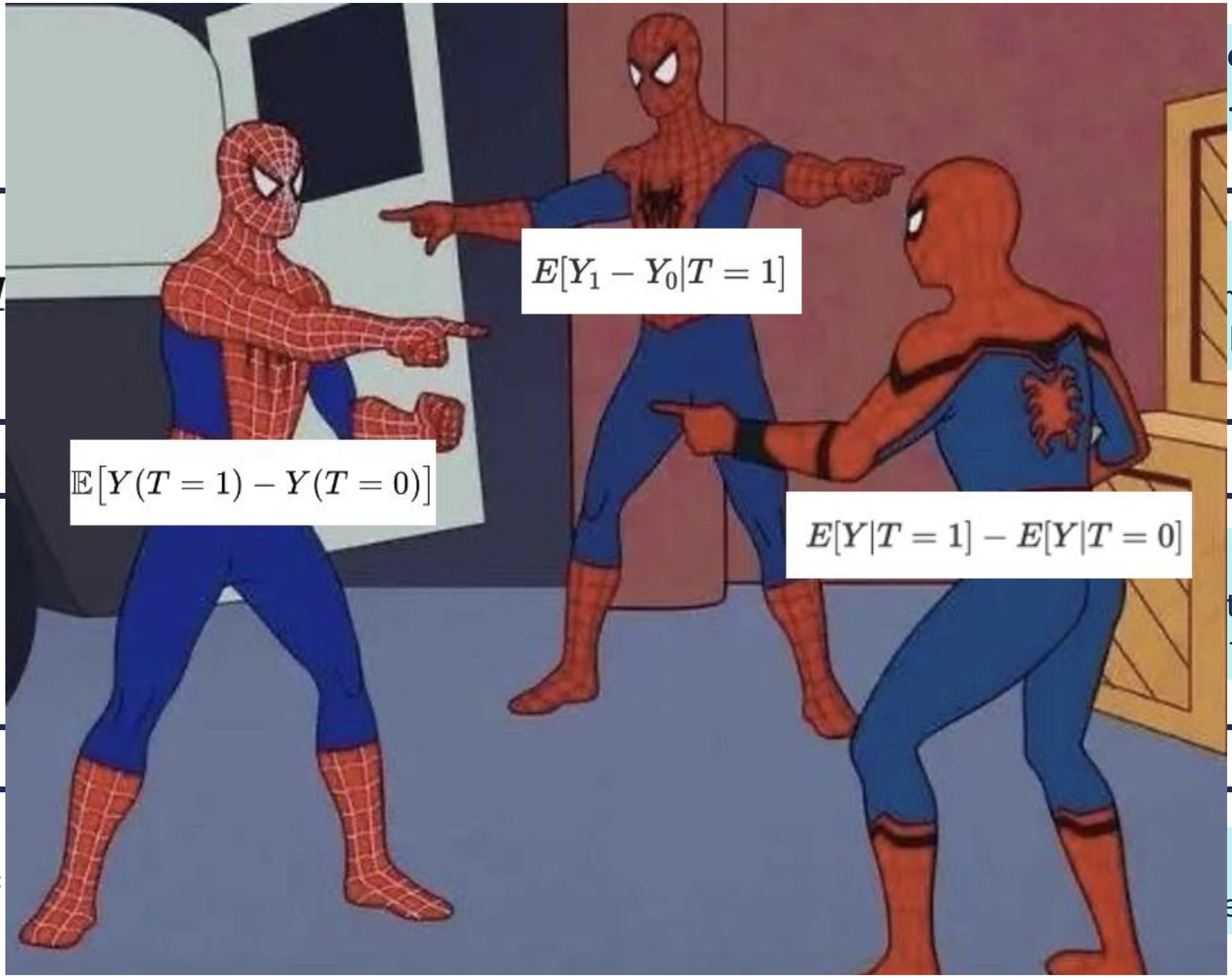
$$ATT = E[Y_1 - Y_0|T=1]$$

ts.

t it can be estimated.

$$E[Y|T=1] - E[Y|T=0] =$$

ence of causal inference



agilelab

From Association to Causation: Accounting for Bias

$$\boxed{E[Y|T=1]-E[Y|T=0]} = \boxed{\underbrace{E[Y_1-Y_0|T=1]}_{ATT}} + \boxed{\underbrace{\{E[Y_0|T=1]-E[Y_0|T=0]\}}_{BIAS}}$$

Observed Association = Causal Effect + Bias

What we actually measure in the data

Average treatment effect on the treated

The intrinsic difference btw treated and control, supposing no treatment.

$$T \in \begin{cases} 0 & \text{control} \\ 1 & \text{treated} \end{cases}$$

Binary treatment: Discount, promotion, …

$$Y$$

Outcome: Retention, purchases, …

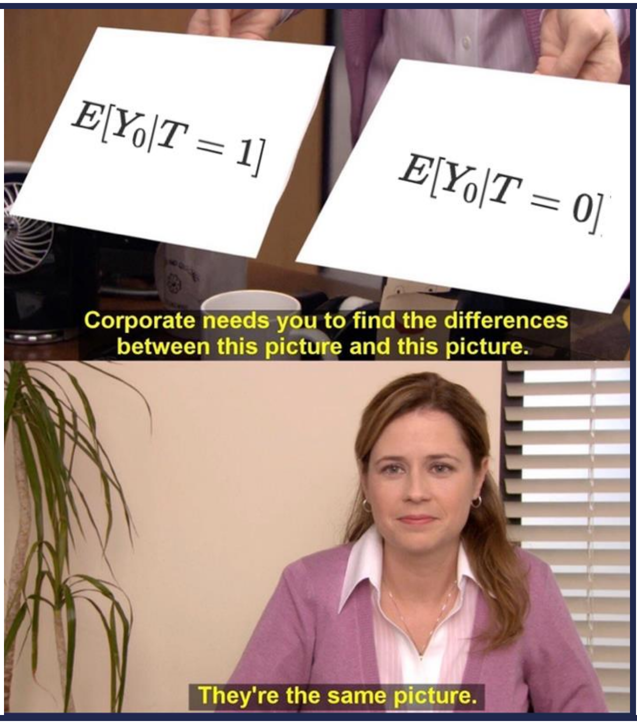
Experimental Framework: Randomized Controlled Trials

Golden Standard

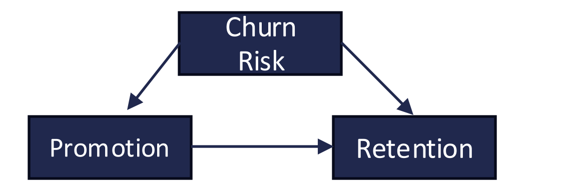
Naturally compensate for bias

Treatment and control groups are statistically identical except for the intervention.

Legal, ethical, or operational constraints could prevent random assignment.

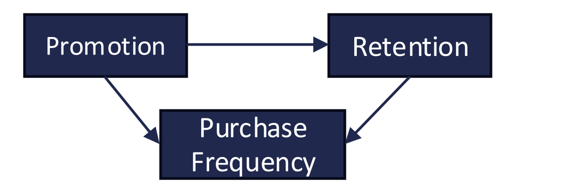

$E[Y_0|T=1]$
$E[Y_0|T=0]$
Corporate needs you to find the differences between this picture and this picture.
They're the same picture.

Observational Data: We must deal with bias



Confounding Bias by **confounding** variables

> You *should* split the groups



Selection Bias by **collider** variables

> You *shouldn't* split the groups

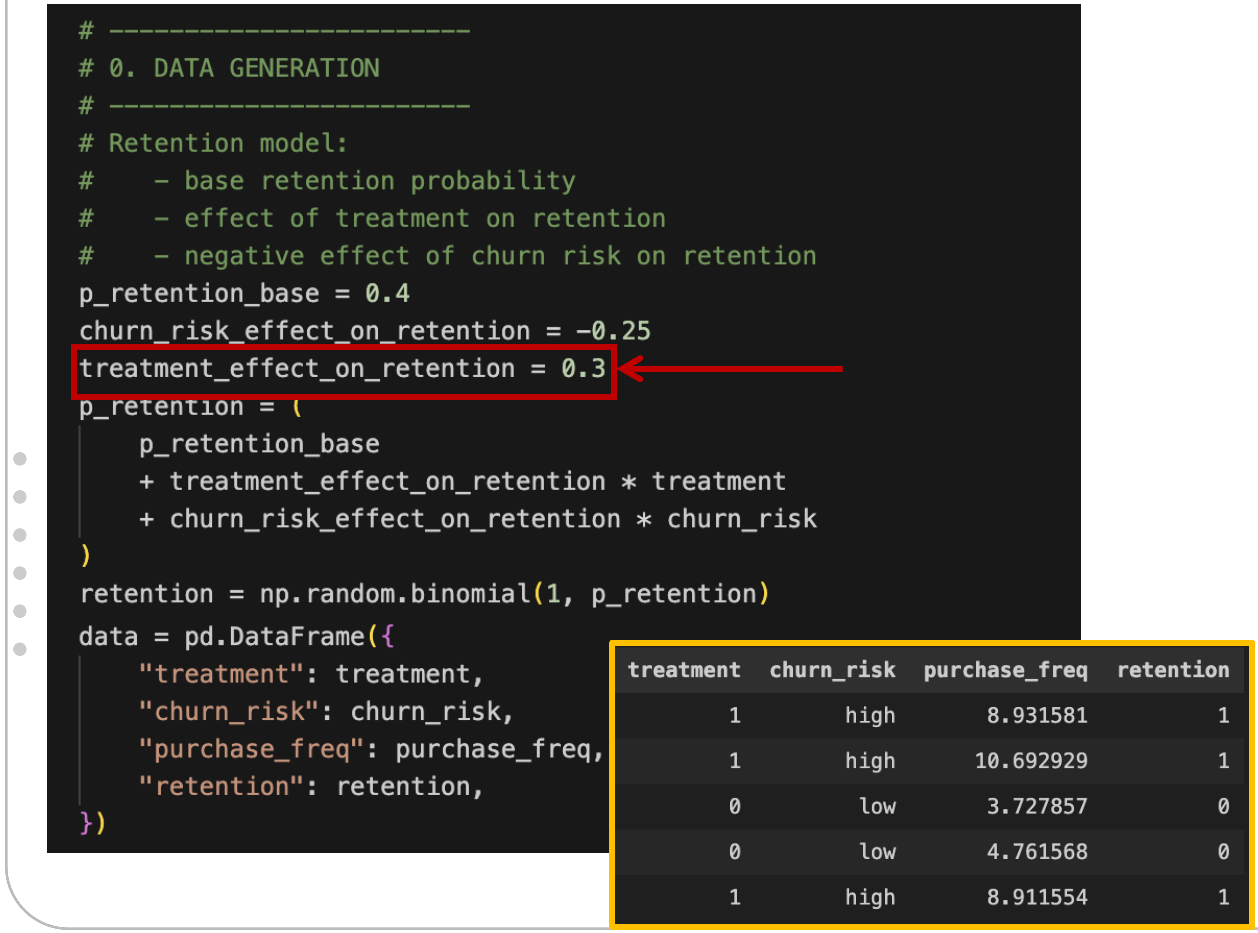
How to compensate bias?

- **Potential Outcomes**: Propensity Score (Rubin)
- **Graphical approach**: Do-calculus, back-door and front-door criteria (Pearl)

Treatment Effect Estimation with *dowhy*

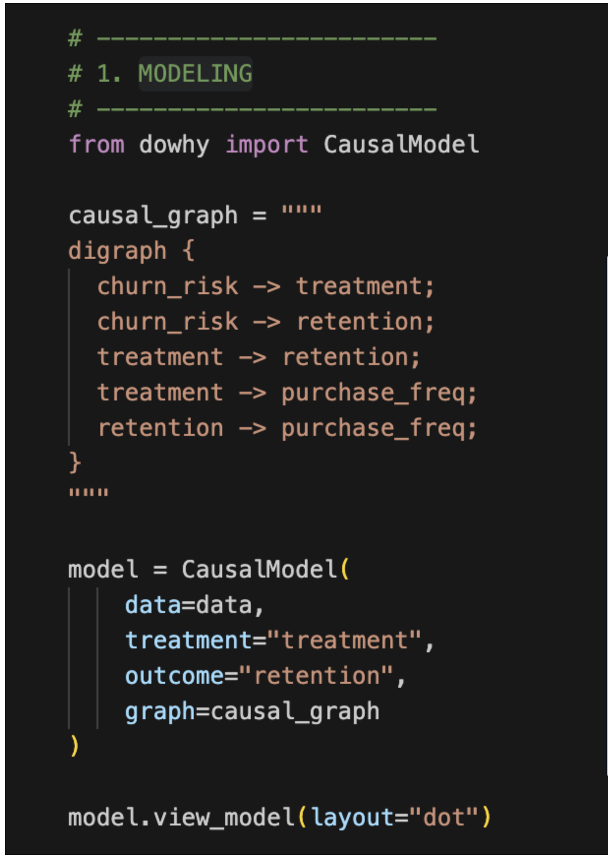
⚠ No ground truth in causality
→ we use synthetic data.

Causal inference is not just data: it’s reasoning!
Build your causal graph using **expert knowledge**, **experiments**, or **discovery algorithms**.

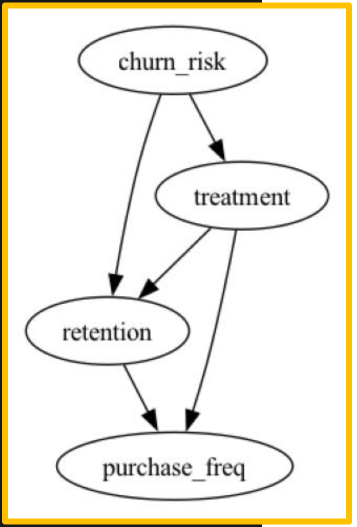


```
# ------------------------
# 0. DATA GENERATION
# ------------------------
# Retention model:
#    - base retention probability
#    - effect of treatment on retention
#    - negative effect of churn risk on retention
p_retention_base = 0.4
churn_risk_effect_on_retention = -0.25
treatment_effect_on_retention = 0.3
p_retention = (
    p_retention_base
    + treatment_effect_on_retention * treatment
    + churn_risk_effect_on_retention * churn_risk
)
retention = np.random.binomial(1, p_retention)
data = pd.DataFrame({
    "treatment": treatment,
    "churn_risk": churn_risk,
    "purchase_freq": purchase_freq,
    "retention": retention,
})
```

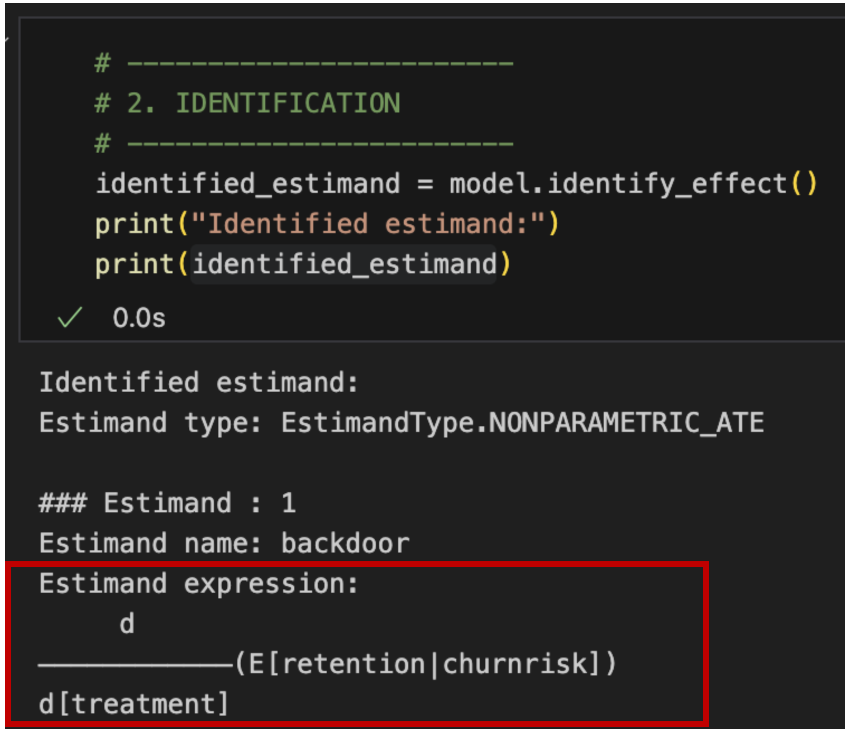
treatment	churn_risk	purchase_freq	retention
1	high	8.931581	1
1	high	10.692929	1
0	low	3.727857	0
0	low	4.761568	0
1	high	8.911554	1



```
# ------------------------
# 1. MODELING
# ------------------------
from dowhy import CausalModel

causal_graph = """
digraph {
    churn_risk -> treatment;
    churn_risk -> retention;
    treatment -> retention;
    treatment -> purchase_freq;
    retention -> purchase_freq;
}
"""

model = CausalModel(
    data=data,
    treatment="treatment",
    outcome="retention",
    graph=causal_graph
)

model.view_model(layout="dot")
```



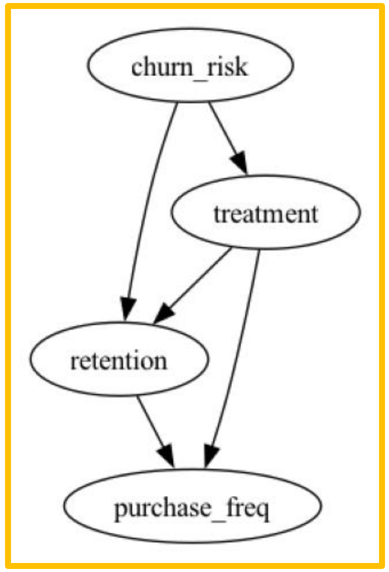
Treatment Effect Estimation with *dowhy*

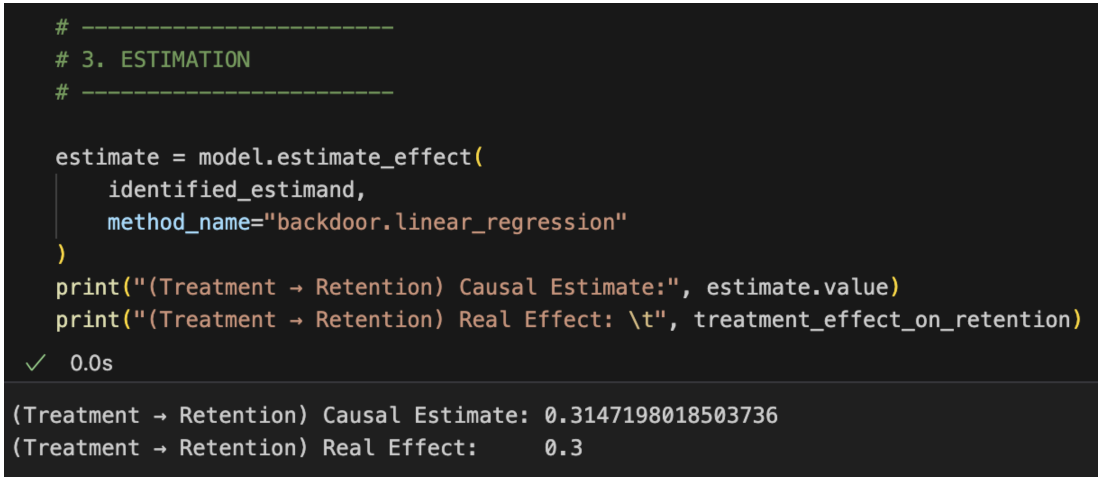


```
# -------------------------
# 2. IDENTIFICATION
# -------------------------
identified_estimand = model.identify_effect()
print("Identified estimand:")
print(identified_estimand)
```

✓ 0.0s

```
Identified estimand:
Estimand type: EstimandType.NONPARAMETRIC_ATE

### Estimand : 1
Estimand name: backdoor
Estimand expression:
     d
────────────(E[retention|churnrisk])
d[treatment]
```





```
# -----------------------
# 3. ESTIMATION
# -----------------------

estimate = model.estimate_effect(
    identified_estimand,
    method_name="backdoor.linear_regression"
)
print("(Treatment → Retention) Causal Estimate:", estimate.value)
print("(Treatment → Retention) Real Effect: \t", treatment_effect_on_retention)
```

✓ 0.0s

```
(Treatment → Retention) Causal Estimate: 0.3147198018503736
(Treatment → Retention) Real Effect:     0.3
```

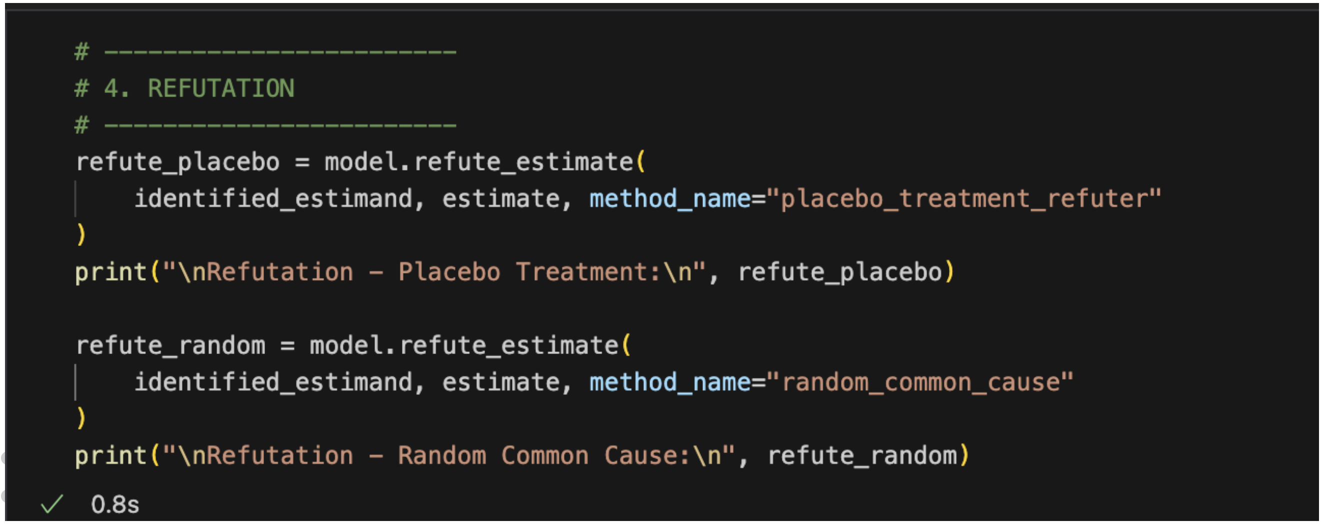
Identification is the key causal problem.
It gives us the **strategy to address bias**.

Here, the strategy is: group your data by churn-risk

We can use the identified strategy to perform estimation.

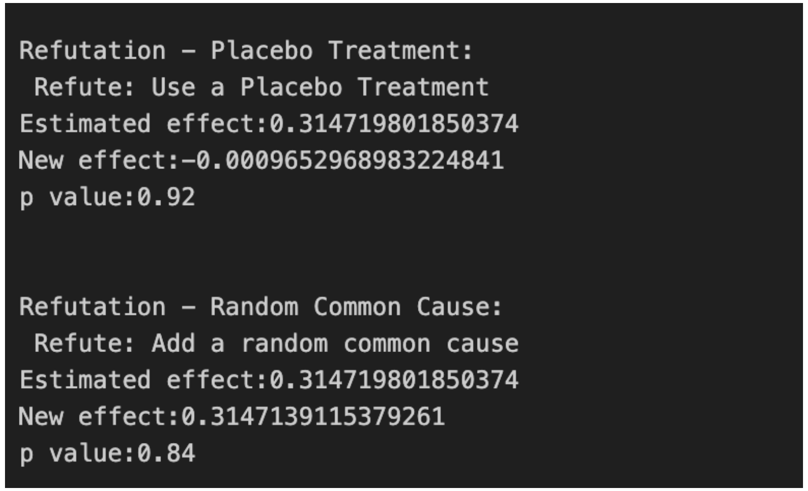
This would be the same:
smf.ols("retention~treatment+churnrisk")

Treatment Effect Estimation with *dowhy*



```
# -------------------------
# 4. REFUTATION
# -------------------------
refute_placebo = model.refute_estimate(
    identified_estimand, estimate, method_name="placebo_treatment_refuter"
)
print("\nRefutation - Placebo Treatment:\n", refute_placebo)

refute_random = model.refute_estimate(
    identified_estimand, estimate, method_name="random_common_cause"
)
print("\nRefutation - Random Common Cause:\n", refute_random)
```

✓ 0.8s



```
Refutation - Placebo Treatment:
 Refute: Use a Placebo Treatment
Estimated effect:0.314719801850374
New effect:-0.0009652968983224841
p value:0.92


Refutation - Random Common Cause:
 Refute: Add a random common cause
Estimated effect:0.314719801850374
New effect:0.3147139115379261
p value:0.84
```

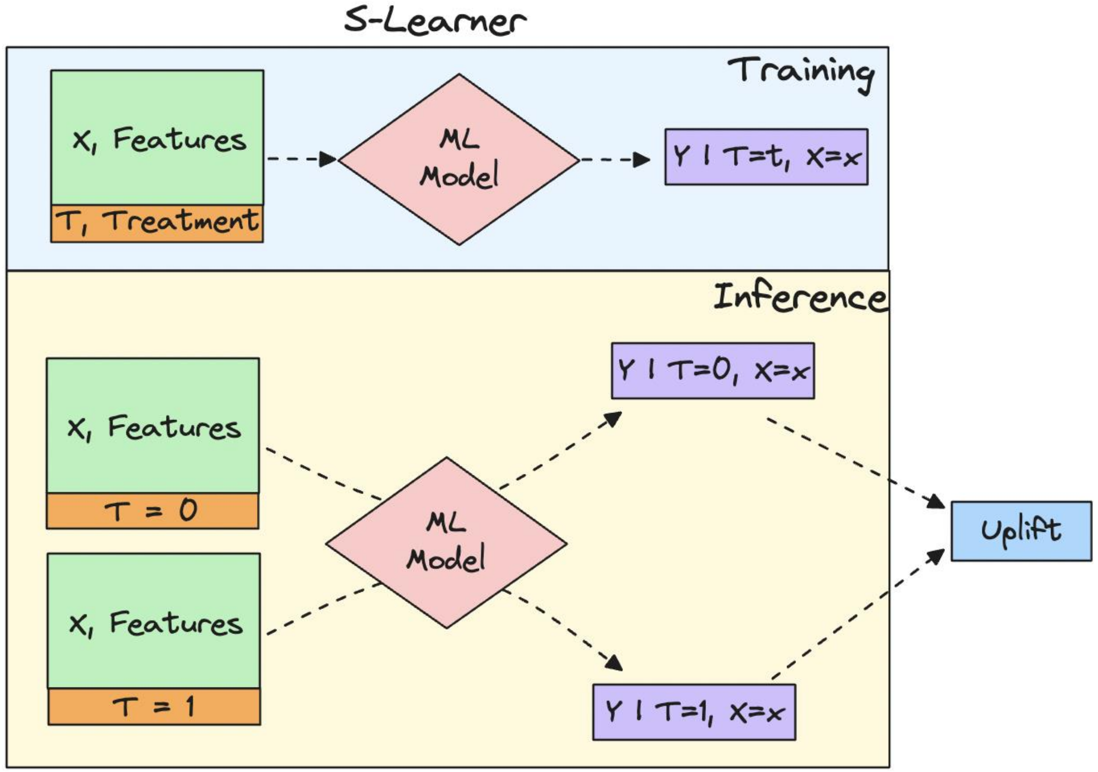
Automated validation of assumptions:
Graph + Estimate refutation
Validate your causal and **check for robustness.**

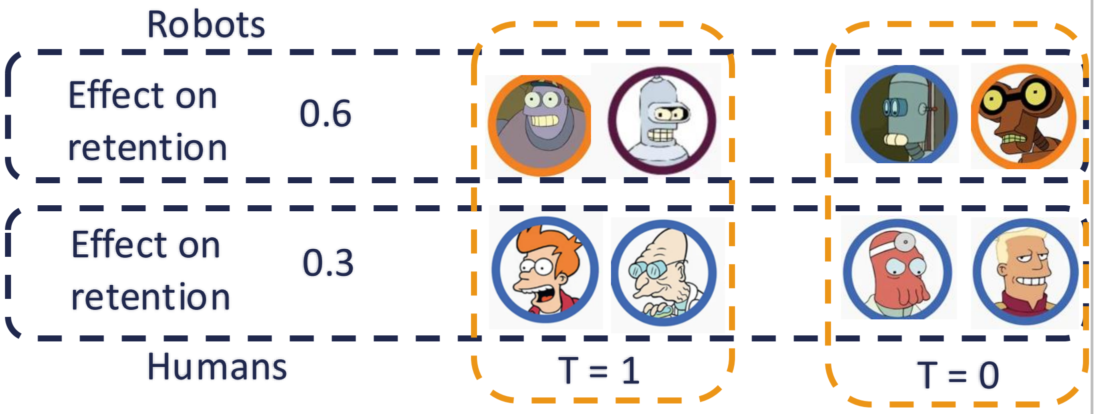
Targeting the Right Customers with Causal ML

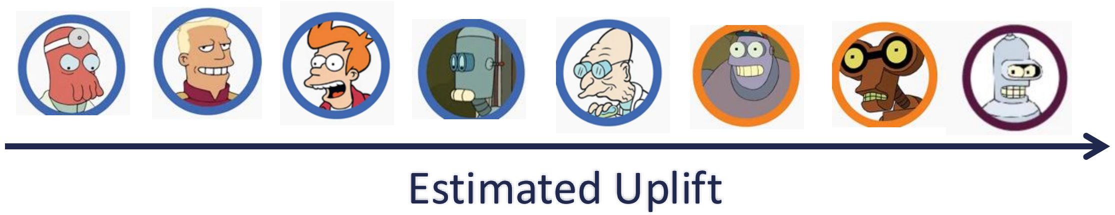
Heterogeneous Effects: Uplift Modeling

$$\mathrm{CATE}(x) = \mathbb{E}[\,Y(T=1) - Y(T=0) \mid X = x\,]$$

Estimated Uplift = Estimated outcome with treatment - Estimated Outcome without treatment







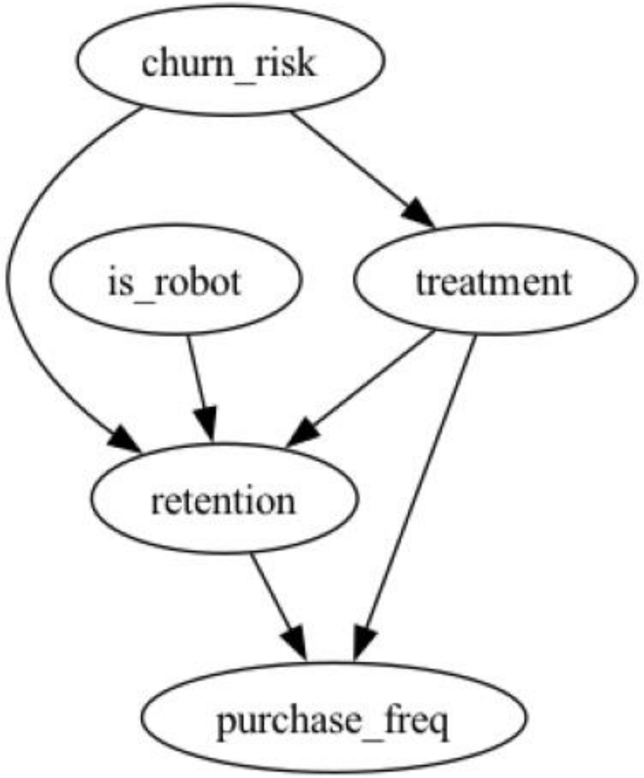
⚠ Pitfall: What features X should we include?

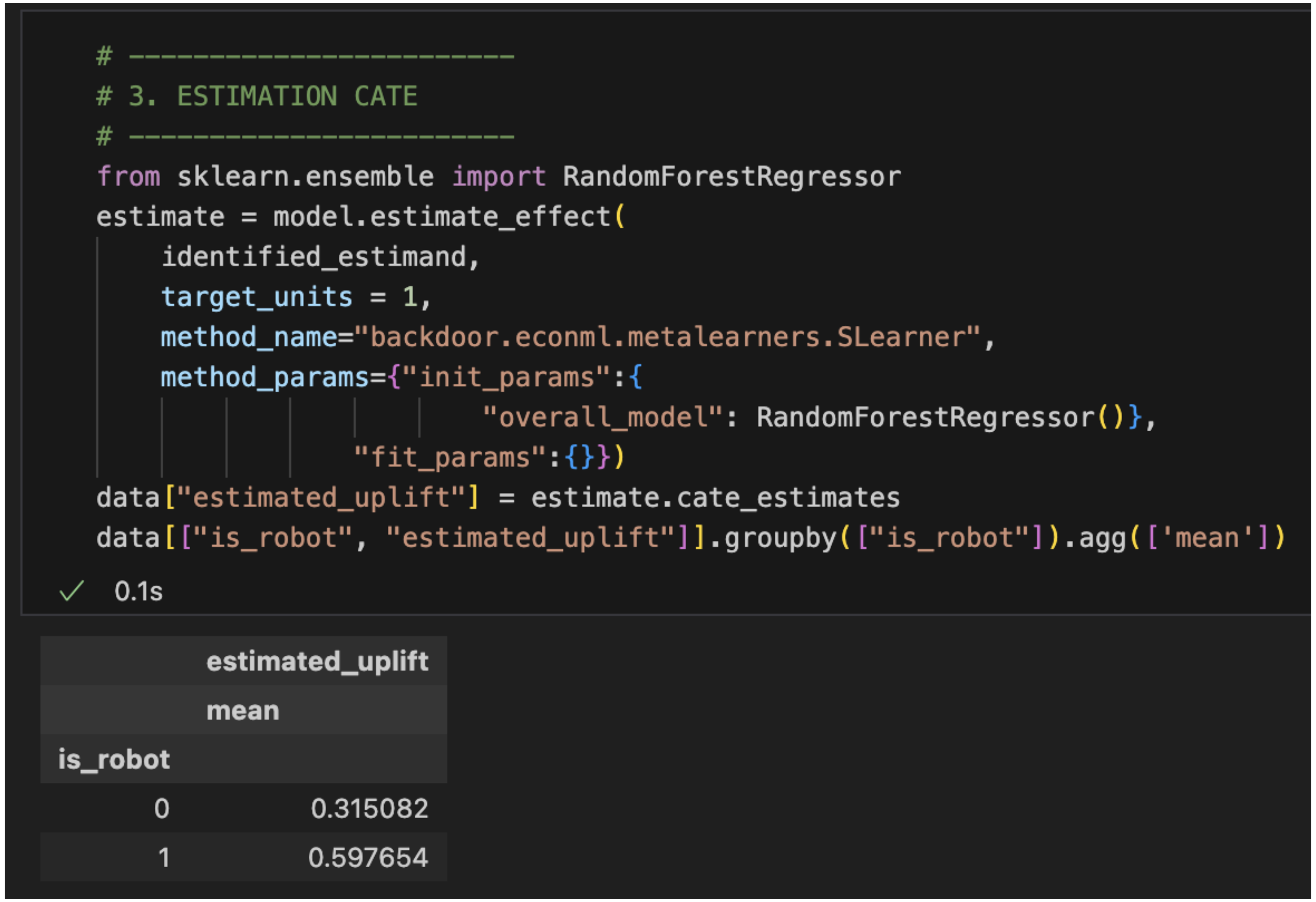
- ✓ Confounders
- ✓ Predictors of Y only, Predictors of T only
- ✗ Colliders, Mediators

Uplift modeling in Python

DoWhy

EconML





```
# ------------------------
# 3. ESTIMATION CATE
# ------------------------
from sklearn.ensemble import RandomForestRegressor
estimate = model.estimate_effect(
    identified_estimand,
    target_units = 1,
    method_name="backdoor.econml.metalearners.SLearner",
    method_params={"init_params":{
                        "overall_model": RandomForestRegressor()},
                "fit_params":{}})
data["estimated_uplift"] = estimate.cate_estimates
data[["is_robot", "estimated_uplift"]].groupby(["is_robot"]).agg(['mean'])
```

✓ 0.1s

	estimated_uplift
	mean
is_robot	
0	0.315082
1	0.597654

agilelab

Final Takeaways

- Today we just scratched the surface.
- There’s huge room for growth, especially from the Python community.
- Causality it’s a training ground for asking the right questions.

Thank you!

www.agilelab.it @agilelab_official in linkedin.com/company/agile-lab @agilelabsrl