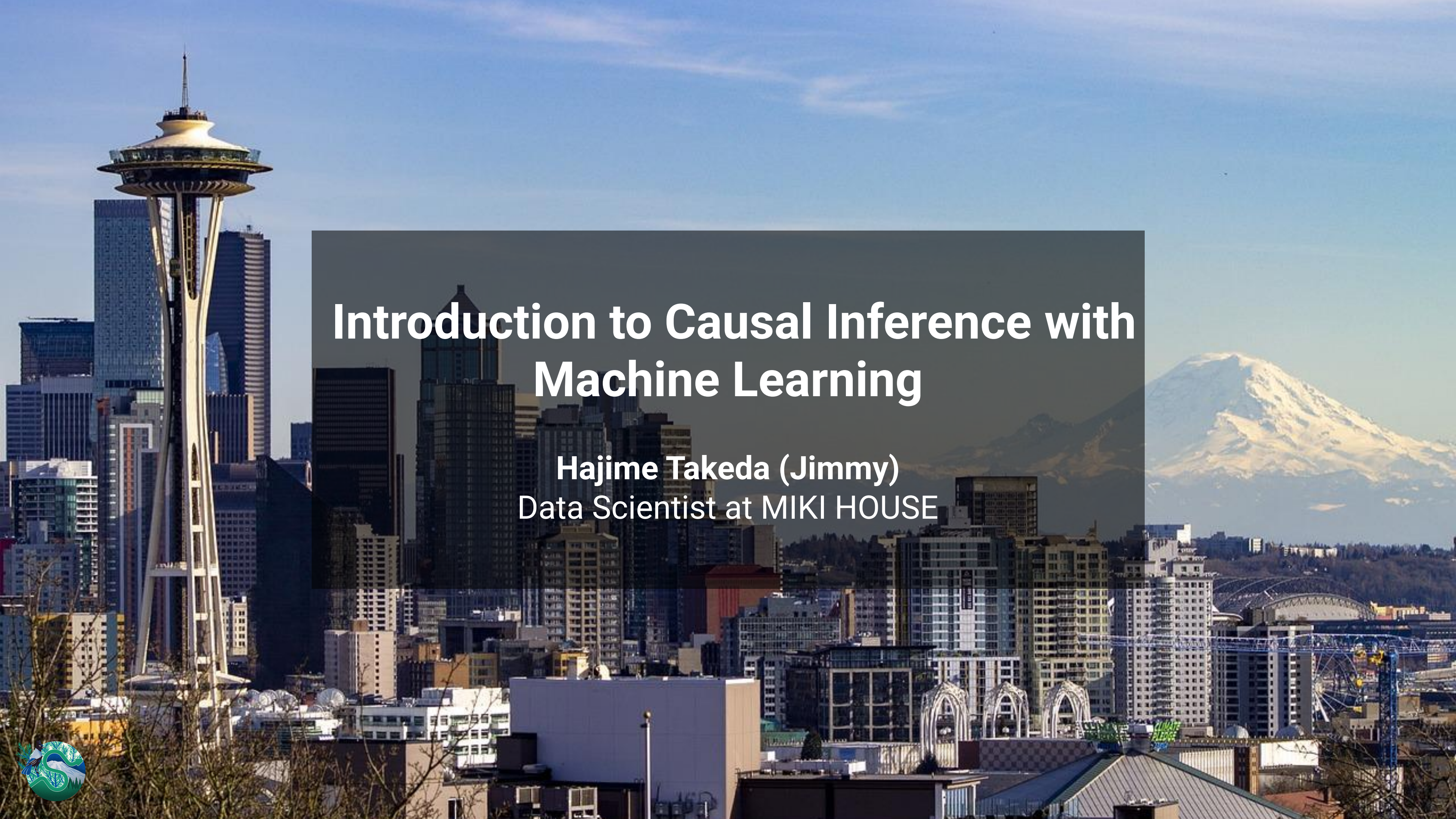


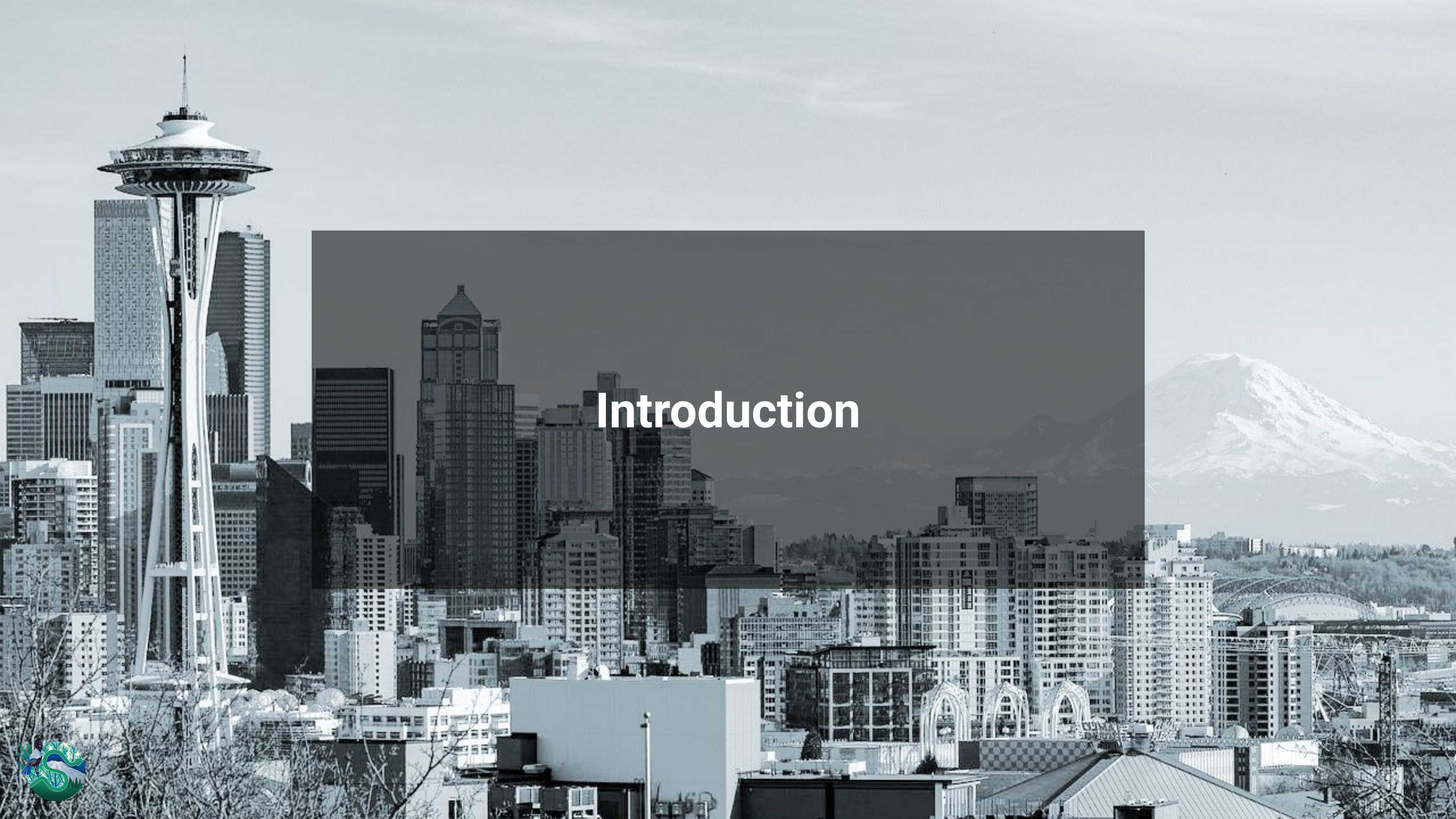
Introduction to Causal Inference with Machine Learning

Hajime Takeda (Jimmy)
Data Scientist at MIKI HOUSE

Agenda

- What is Causal Inference?
- What is Causal Inference with Machine Learning?
- Technique #1 : Meta Learners
  - Case Study in Economics with EconML
- Technique #2 : Uplift Modeling
  - Case Study in Marketing with CausalML
- Summary

For the sake of time, please save your questions until the end!


Introduction

About Me

Hajime Takeda (Jimmy)

Education & Career
- Master’s in CS at Kyoto University 🇯🇵
- Data Analyst at Procter & Gamble 🇯🇵
- Data Scientist at MIKI HOUSE 🇺🇸

Conferences Speaker
- PyData Global (2022) 🌎
- ODSC EAST (2023) 🇺🇸
- PyData NYC (2023) 🇺🇸
- PyData London (2024) 🇬🇧

P&G

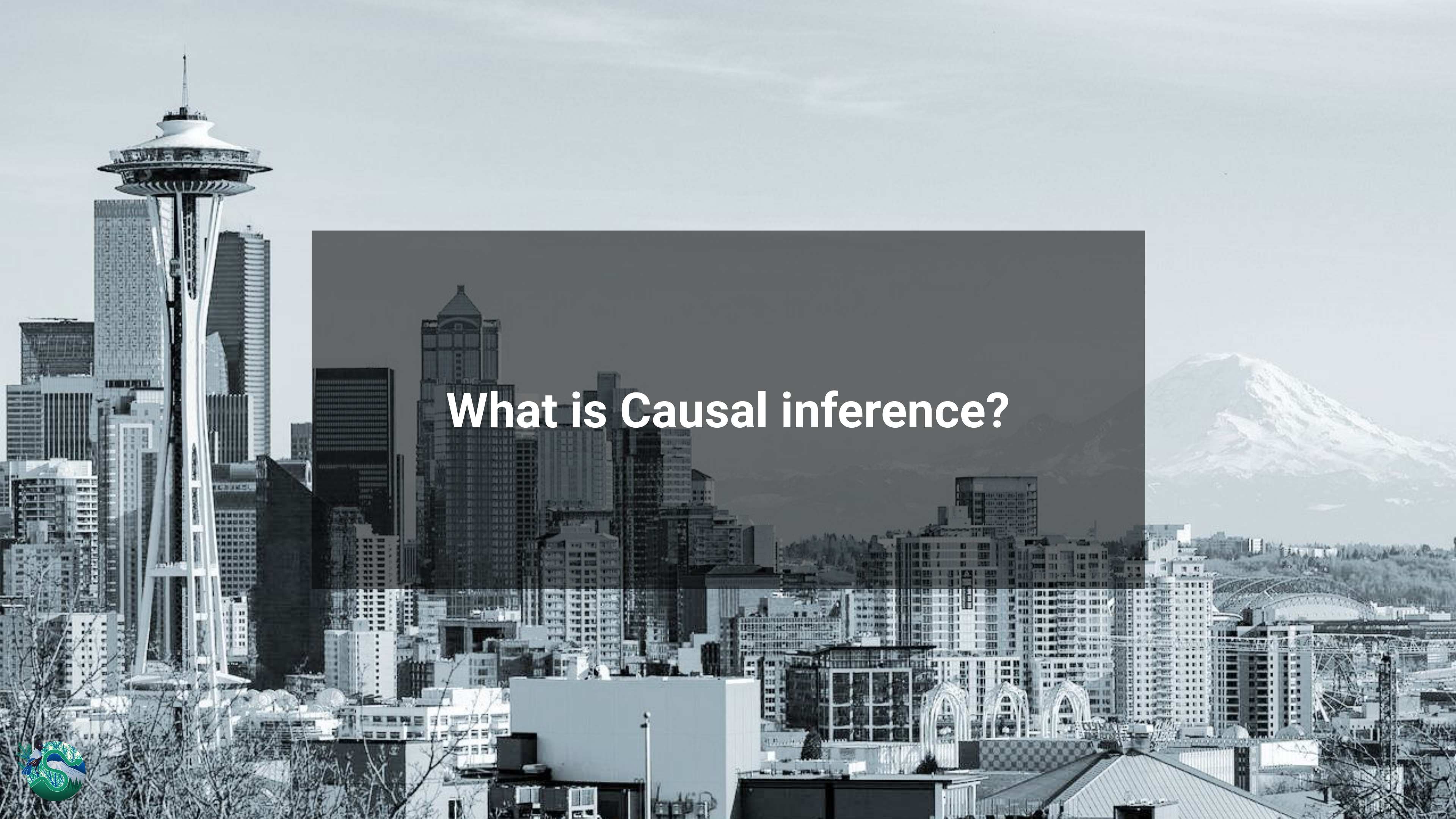
MIKI HOUSE

Expected Takeaways

1 Understand the key concepts and approaches of **Causal Inference with Machine Learning**

2 Learn how to run **Meta Learners and Uplift Modeling** using EconML/CausalML

What is Causal inference?



Typical Scenario

Jessy
Marketing

"Awesome! We sent **coupons** to **some users** and their **purchase rate** was **twice** as high as others! That means **the coupons must have doubled the purchase rate**!"

	Without Coupon	**With Coupon**
# of Customers	1,000	1,000
# of Purchasers	100	200
Purchase rate (%)	10%	20%

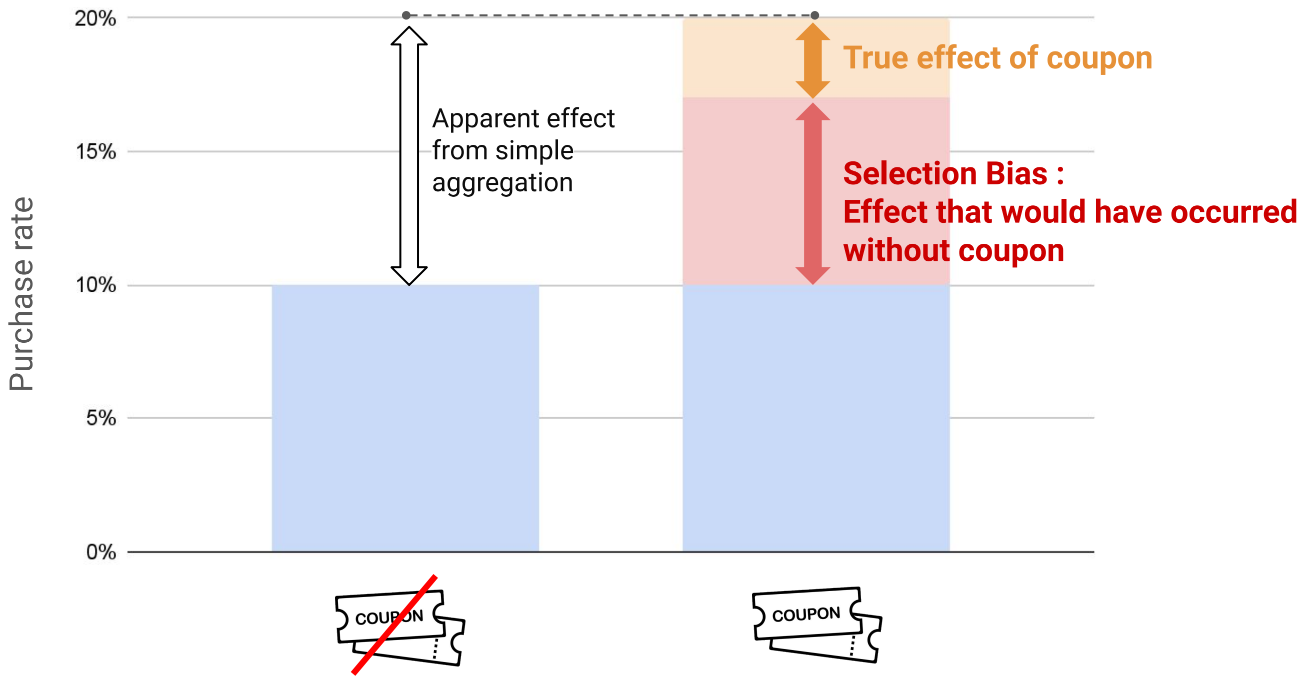
Typical Scenario

Jessy
Marketing

"We sent coupons to **customers who purchased within the last 12 months**"

Pitfall: Selection Bias

Customers who received coupons might have purchased anyway.

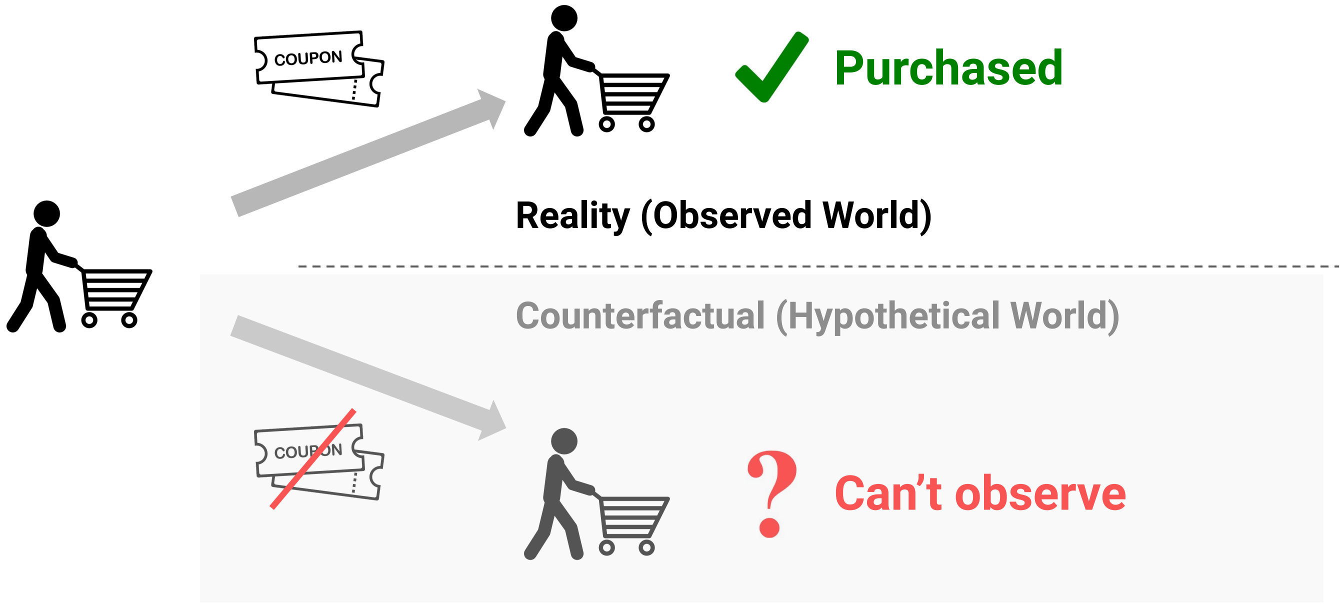


What is Causal Inference?

Causal Inference is the process of determining whether a cause-and-effect relationship exists between **Treatment** and **Outcome**.

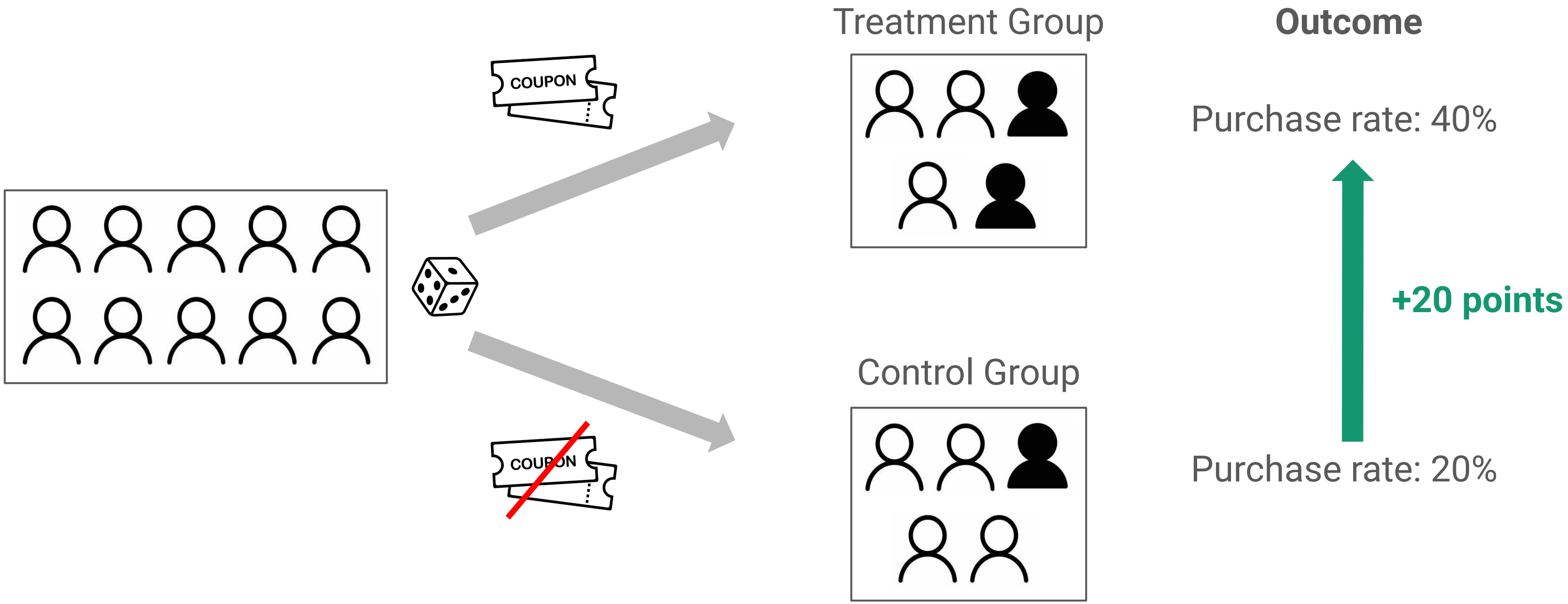
What is the challenge?

Counterfactual : We can only observe one outcome from the same individual.



Randomized Controlled Trial (RCT)

Participants are randomly assigned to either a treatment group or a control group.
RCT = A/B Testing



Limitations of RCT

Business

- RCTs take a lot of time.
- RCTs result in Opportunity losses in sales.

Social and Economic Program

- Assigning citizens randomly to control groups is infeasible.
- Many policies are one-time and irreversible.

Healthcare

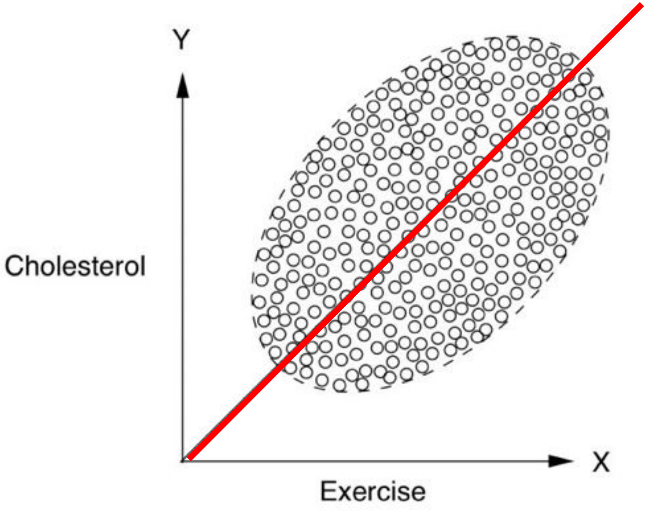
- It is ethically problematic to randomly assign people with life-threatening conditions to control groups.

That's why we use Causal Inference.

Simpson's Paradox - Challenges of Observed Data

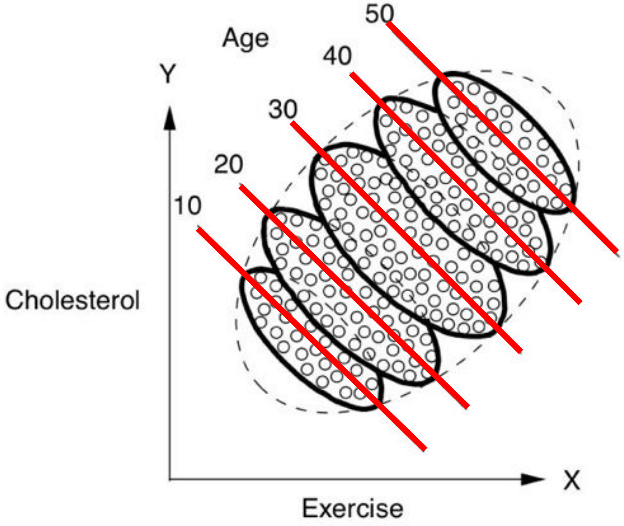
Questions: What's the effect of exercise on cholesterol?

Without segmentation



=

Segmenting by Age Group

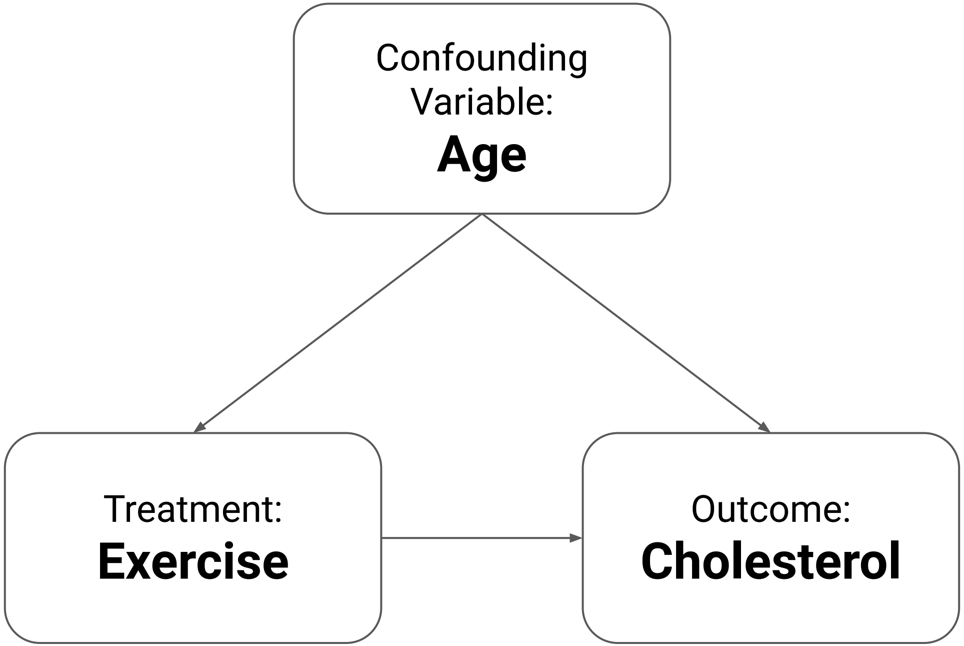


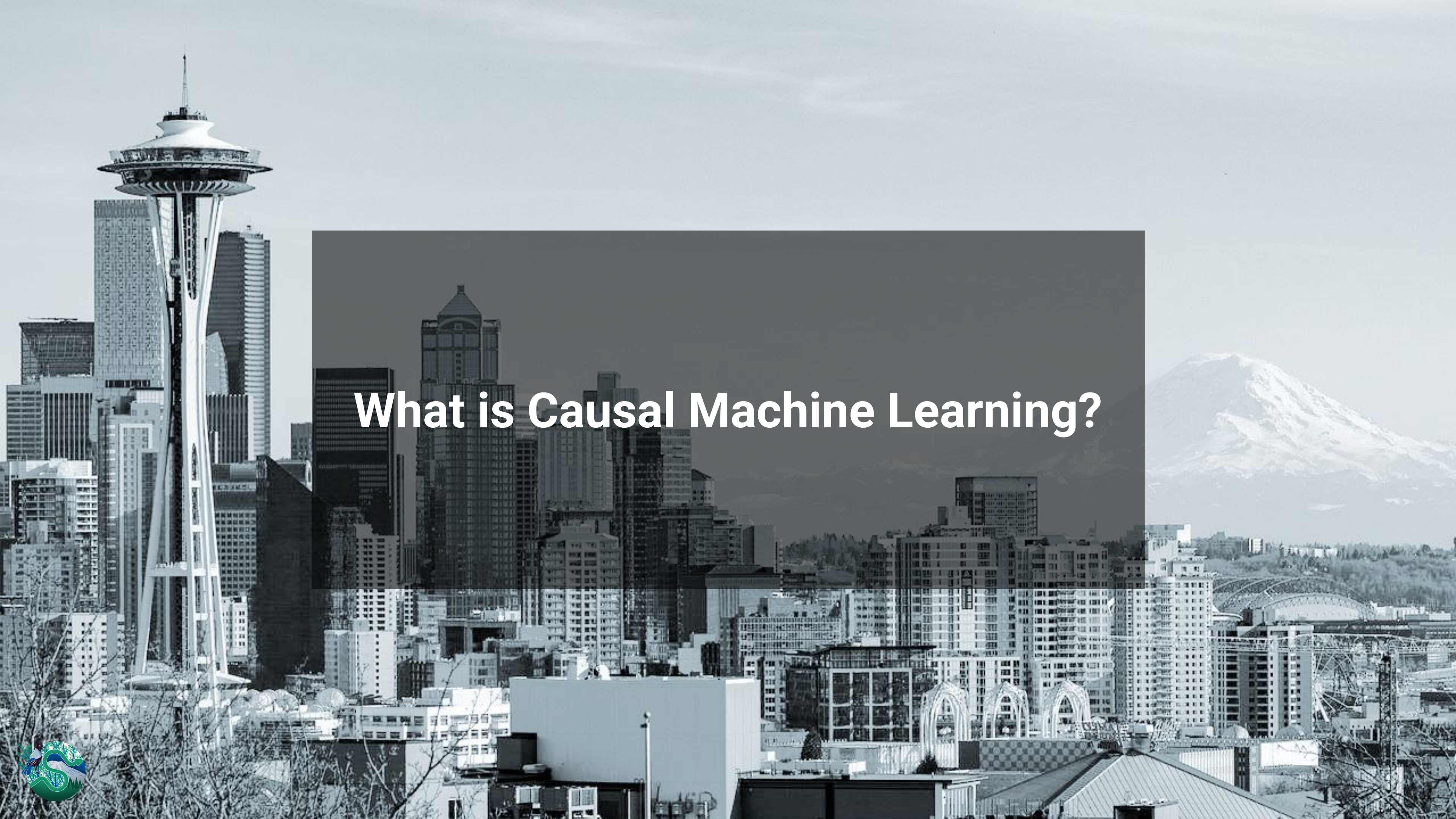
Simpson's Paradox : Overall and subgroup trends differ.

Image : "The book of why" by Judea Pearl and Dana MacKenzie

Confounding Variable

Confounding variable affects both treatment and outcome.





What is Causal Machine Learning?

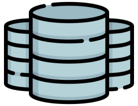
History of Causal Machine Learning

Judea Pearl

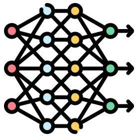
Donald Rubin

Establishment of Traditional Causal Inference
(e.g., causal diagrams, Potential Outcomes Framework)

1970s - 1990s	2000s	2010s	2020s



Big Data



Neural Networks

Machine learning technology evolved rapidly with the increase in data volume.

EconML

CausalML

Fusion of Causal Inference and Machine Learning:
- New methods such as Causal Forest
- **CausalML** and **EconML** (2019)

Machine Learning vs. Causal ML

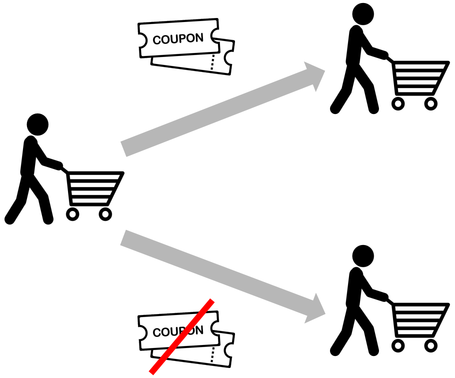
Supervised Machine Learning focuses on “**Prediction**”,
while Causal Machine Learning focuses on “**Causality**”.

	Supervised Machine Learning	**Causal Machine Learning / Causal Inference**
Purpose	Prediction based on correlation	Estimating the treatment effect
Questions	**What will the customer buy next?** What is the probability?	**Did the coupon increase sales?** How much was its impact?
Variables	X (Features): Age, gender, coupon availability Y (Target Variable): Sales	X (Control Variables): Age, gender **Z (Treatment Variable): Coupon availability** Y (Outcome Variable): Sales uplift

Two Techniques

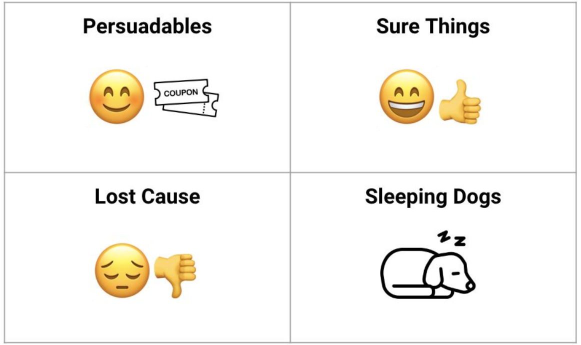
1. Meta Learners

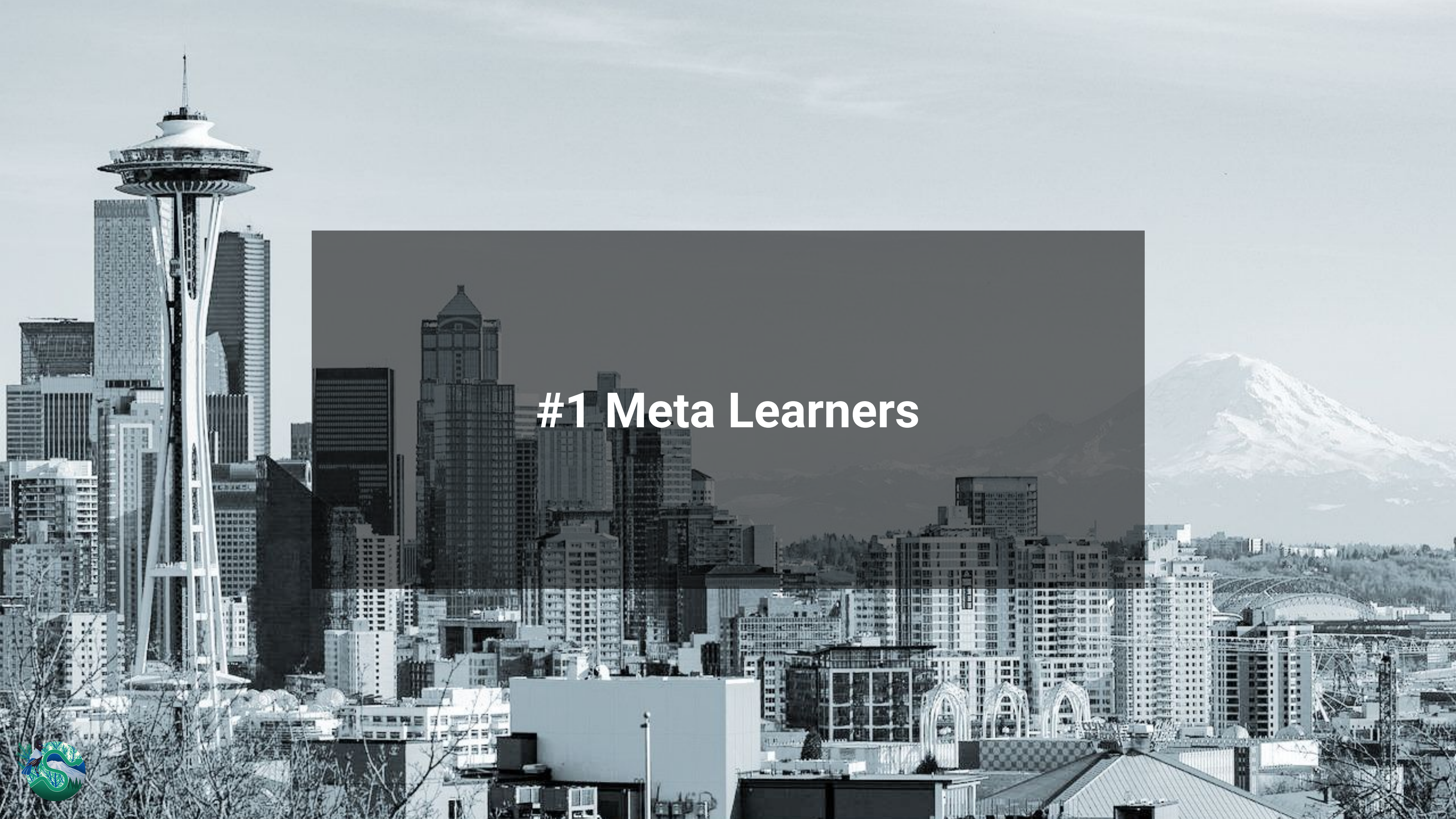
- The goal is to measure the treatment effect
- i.e. S-learner, T-learner, DML etc.



2. Uplift Modeling

- The goal is to select the right users for targeting
- i.e. uplift tree / uplift random forest

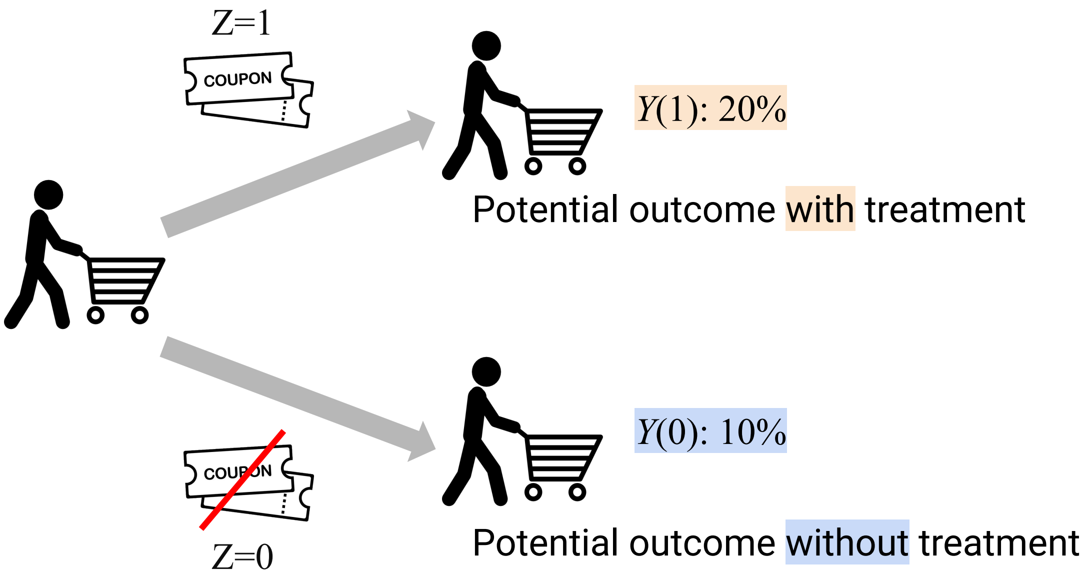




#1 Meta Learners

What is Treatment Effect?

Potential Outcomes Framework, also known as Rubin causal model (RCM)



Treatment Effect $\tau = Y(1) - Y(0)$

$20\% - 10\%$

τ **(Tau) :** Treatment effect

Y **(Outcome)**: Sales, Purchase rate, etc.

Z **(Treatment)**: Campaign, Coupon, etc.

X **(Confounders)**: Age, Gender, Living location, Preference, Past purchase history, etc.

Types of Treatment Effect

There are mainly 3 types of treatment effects

Term	Abbreviation	Formula	Definition
Average **T**reatment **E**ffect	**ATE**	$\tau ATE = E[Y(1) - Y(0)]$	Across the entire customers
Conditional **A**verage **T**reatment **E**ffect	**CATE**	$\tau CATE(X) = E[Y(1) - Y(0) \mid X]$	Segment level (i.e. gender, age groups)
Individual **T**reatment **E**ffect	**ITE**	$\tau i = Yi(1) - Yi(0)$	Individual Level

How to Calculate Treatment Effect

Question: Did a coupon increase sales?

X				Z	Y	$Y(1)$	$Y(0)$	$Y(1)$ - $Y(0)$
Name	**Age**	**Gender**	**Location**	**Treatment**	**Outcome**	**Outcome with treatment**	**Outcome without treatment**	**ITE : Individual Treatment Effect**
Anne	30	Male	Urban	0	0	NA	0	?
Ben	40	Female	Urban	0	0	NA	0	?
Chris	45	Female	Rural	0	1	NA	1	?
Diana	58	Female	Rural	0	0	NA	0	?
Ethan	25	Female	Urban	1	1	1	NA	?
Faye	38	Male	Rural	1	0	0	NA	?
Gary	42	Male	Urban	1	1	1	NA	?
Helen	60	Male	Urban	1	1	1	NA	?

Meta Learners

<u>Meta learners</u> are meta-algorithms designed to estimate treatment effects by using ML models to handle unobserved outcomes

X				Z	Y	$Y(1)$	$Y(0)$	$Y(1)$ - $Y(0)$
Name	**Age**	**Gender**	**Location**	**Treatment**	**Outcome**	**Outcome with treatment**	**Outcome without treatment**	**ITE : Individual Treatment Effect**
Anne	30	Male	Urban	0	0	0.9	0	+0.9
Ben	40	Female	Urban	0	0	0.8	0	+0.8
Chris	45	Female	Rural	0	1	0.9	1	-0.1
Diana	58	Female	Rural	0	0	0.6	0	+0.6
Ethan	25	Female	Urban	1	1	1	0.2	+0.8
Faye	38	Male	Rural	1	0	0	0.1	-0.1
Gary	42	Male	Urban	1	1	1	0.5	+0.5
Helen	60	Male	Urban	1	1	1	0.4	+0.6

Types of Meta Learners

Learner	Approach	Reference	Training Speed (relative to S learner)
S Learner	**S**ingle Model Approach	Künzel et al. 2019	1x
T Learner	**T**wo-Model Approach	Künzel et al. 2019	2x
X Learner	**Cross**-Fitting Approach	Künzel et al. 2019	5x
DR Learner	**D**oubly **R**obust	Kennedy et al.2020	13x
DML	**D**ouble **M**achine **L**earning	Chernozhukov, Victor, et al. 2018	27x

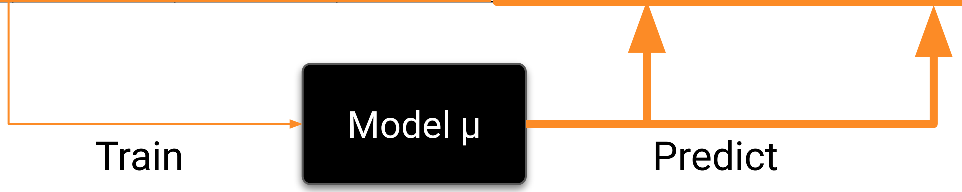
Simple

Complex & Robust

S Learner - Single Model Approach

S Learner uses a single model for training and prediction

X				*Z*	*Y*	*Y(1)*	*Y(0)*	*Y*(1) - *Y*(0)
Name	**Age**	**Gender**	**Location**	**Treatment**	**Outcome**	**Outcome with treatment**	**Outcome without treatment**	**ITE : Individual Treatment Effect**
Anne	30	Male	Urban	0	0	0.4	0.1	0.3
Ben	40	Female	Urban	0	0	0.6	0.2	0.4
Chris	45	Female	Rural	0	1	0.7	0.5	0.2
Diana	58	Female	Rural	0	0	0.9	0.6	0.3
Ethan	25	Female	Urban	1	1	0.6	0.1	0.5
Faye	38	Male	Rural	1	0	0.2	0.0	0.2
Gary	42	Male	Urban	1	1	0.7	0.3	0.4
Helen	60	Male	Urban	1	1	0.8	0.6	0.2



T Learner - Two Model Approach

Uses separate models for treatment and control groups

Model (Untreated)

X				*Z*	*Y*	*Y(1)*	*Y(0)*	*Y*(1) - *Y*(0)
Name	**Age**	**Gender**	**Location**	**Treatment**	**Outcome**	**Outcome with treatment**	**Outcome without treatment**	**ITE : Individual Treatment Effect**
Anne	30	Male	Urban	0	0	0.4	0.1	0.3
Ben	40	Female	Urban	0	0	0.6	0.2	0.4
Chris	45	Female	Rural	0	1	0.7	0.5	0.2
Diana	58	Female	Rural	0	0	0.9	0.6	0.3
Ethan	25	Female	Urban	1	1	0.6	0.1	0.5
Faye	38	Male	Rural	1	0	0.2	0.0	0.2
Gary	42	Male	Urban	1	1	0.7	0.3	0.4
Helen	60	Male	Urban	1	1	0.8	0.6	0.2

Model (Treated)

PyData

How to Select the Appropriate Meta Learner

Step 1: Start with simple methods to get a baseline
Step 2: Compare other methods by measuring accuracy such as MAPE or RMSE, and performing refutation for robustness.

Learner	Approach	Training Speed (relative to S learner)
S Learner	**S**ingle Model Approach	1x
T Learner	**T**wo-Model Approach	2x
X Learner	**Cross**-Fitting Approach	5x
DR Learner	**D**oubly **R**obust	13x
DML	**D**ouble **M**achine **L**earning	27x

Simple

Complex & Robust

Case Study in Economics Using EconML

Useful Libraries

I will use EconML for the case study of meta learners
, and CausalML for the case study of uplift modeling.

Library	Features	GitHub
EconML	• Covers a wide range of algorithms, strong in economics • Part of a bigger DoWhy ecosystem • Developed by Microsoft Research	py-why/EconML (3.6k star)
CausalML	• Focus on Uplift modeling and Meta Learners • Designed as a standalone tool • Developed by Uber	uber/causalml (4.8k star)

Get the Full Code Here!

https://bit.ly/causalml

Or
https://github.com/takechanman1228/Effective-Uplif-Modeling

Lalonde Dataset : National Supported Work Demonstration

- An employment program led by Lalonde in the 1970s.

```python
# Load the data
data = dowhy.datasets.lalonde_dataset()
data.head()
```

	treat	age	educ	black	hisp	married	nodegr	re74	re75	re78	u74	u75
0	False	23.0	10.0	1.0	0.0	0.0	1.0	0.0	0.0	0.00	1.0	1.0
1	False	26.0	12.0	0.0	0.0	0.0	0.0	0.0	0.0	12383.68	1.0	1.0
2	False	22.0	9.0	1.0	0.0	0.0	1.0	0.0	0.0	0.00	1.0	1.0
3	False	18.0	9.0	1.0	0.0	0.0	1.0	0.0	0.0	10740.08	1.0	1.0
4	False	45.0	11.0	1.0	0.0	0.0	1.0	0.0	0.0	11796.47	1.0	1.0

Treatment or control label

Demographics

Salary before the program

Salary in the following year

Define the Causal Model

```
# Set features and target
features = ['age', 'educ', 'black', 'hisp', 'married', 'nodegr', 're74', 're75']
X = data[features]
y = data['re78']
T = data['treat']

# Scale the features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Define the causal model using DoWhy
model = CausalModel(
    data=data,
    treatment='treat',
    outcome='re78',
    common_causes=features
)
# Identify the causal effect
estimand = model.identify_effect(proceed_when_unidentifiable=True)
```

Estimate ATE

```
# Estimate the causal effect using DML
estimate = model.estimate_effect(
    identified_estimand=estimand,
    method_name='backdoor.econml.dml.LinearDML',
    target_units='ate',
    method_params={
        'init_params': {
            'model_y': LGBMRegressor(n_estimators=100, max_depth=3, verbose=-1),
            'model_t': LogisticRegression(max_iter=1000),
            'discrete_treatment': True,
        },
        'fit_params': {}
    })

# Display the estimated causal effect
print(f"Estimated Average Treatment Effect (ATE): {estimate.value}")
```

← Select Meta Learner method

← Select Base Learner algorithm

```
Estimated Average Treatment Effect (ATE): 1684.7821381418985
```

Evaluation : Refutation

```
refutation_methods = [
    "random_common_cause",
    "placebo_treatment_refuter"
]

for method in refutation_methods:
    result = model.refute_estimate(estimand, estimate, method_name=method)
    print(result)
```

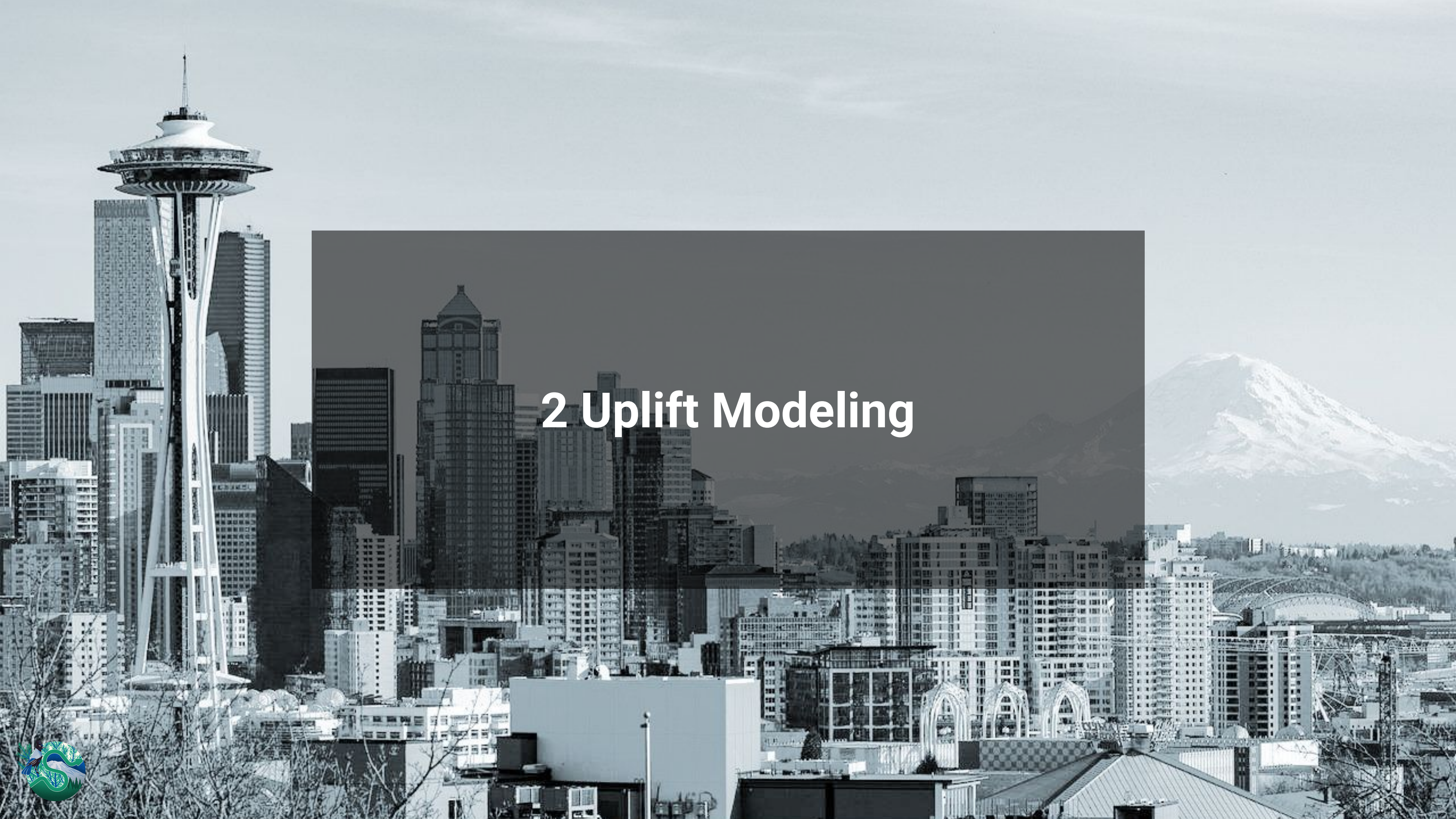
```
Refute: Add a random common cause
Estimated effect:1684.7821381418985
New effect:1714.3310110163663
p value:0.9199999999999999
```

← OK if the new effect remains similar to the original and the p-value exceeds 0.05

```
Refute: Use a Placebo Treatment
Estimated effect:1684.7821381418985
New effect:13.757303277165368
p value:1.0
```

← OK if the new effect is close to zero

2 Uplift Modeling

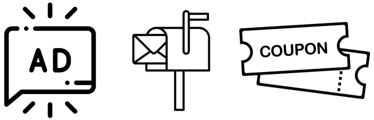


What is Uplift Modeling?

Uplift modeling identifies customers who are influenced positively by marketing offers.

Treatment effect

$$\tau = Y(1) - Y(0)$$



Segmentation of Customers

Focus marketing efforts on the **Persuadables**

If treated \ if NOT treated	Won't Buy	Buy
Buy	**Persuadables** Will buy if receives an incentive	**Sure Things** Will buy no matter what
Won't Buy	**Lost Cause** Won't buy regardless of the campaign	**Sleeping Dogs** Won't buy if receives an incentive

Two Methods for Uplift Modeling

While uplift modeling can also be implemented with Meta Learners, **decision-tree based method** is a common approach

Meta Lerners

- Predict the outcome with treatment and the outcome without treatment separately, and then calculate the uplift
- i.e. T Learner (Two model approach)

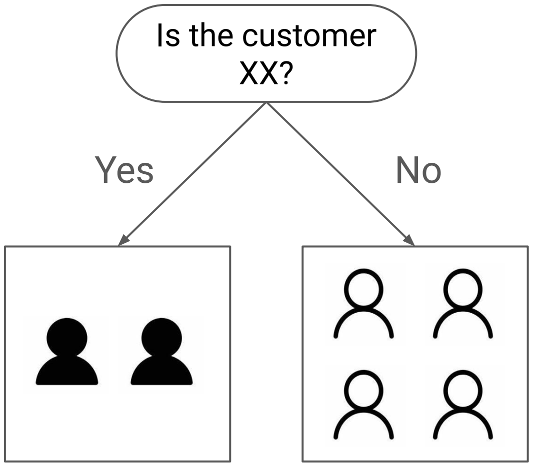
Decision Tree Based Method

- Directly estimate the uplift
- i.e. Uplift Trees, Uplift Random Forest
- Some algorithms support multiple treatment groups (5% coupon vs 10% coupon, 15% coupon)
- Feature importance

Traditional Decision Tree to Uplift Tree

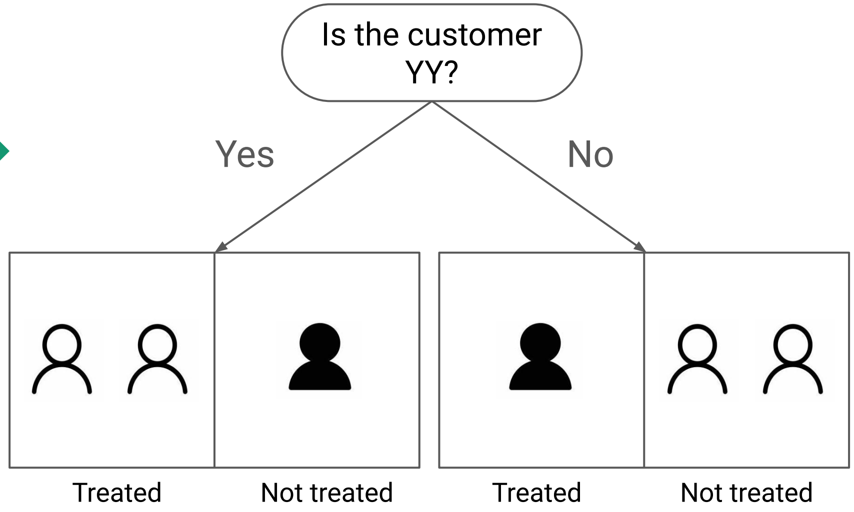
Traditional Decision Tree

Will the customer make a purchase?
(**Prediction**)



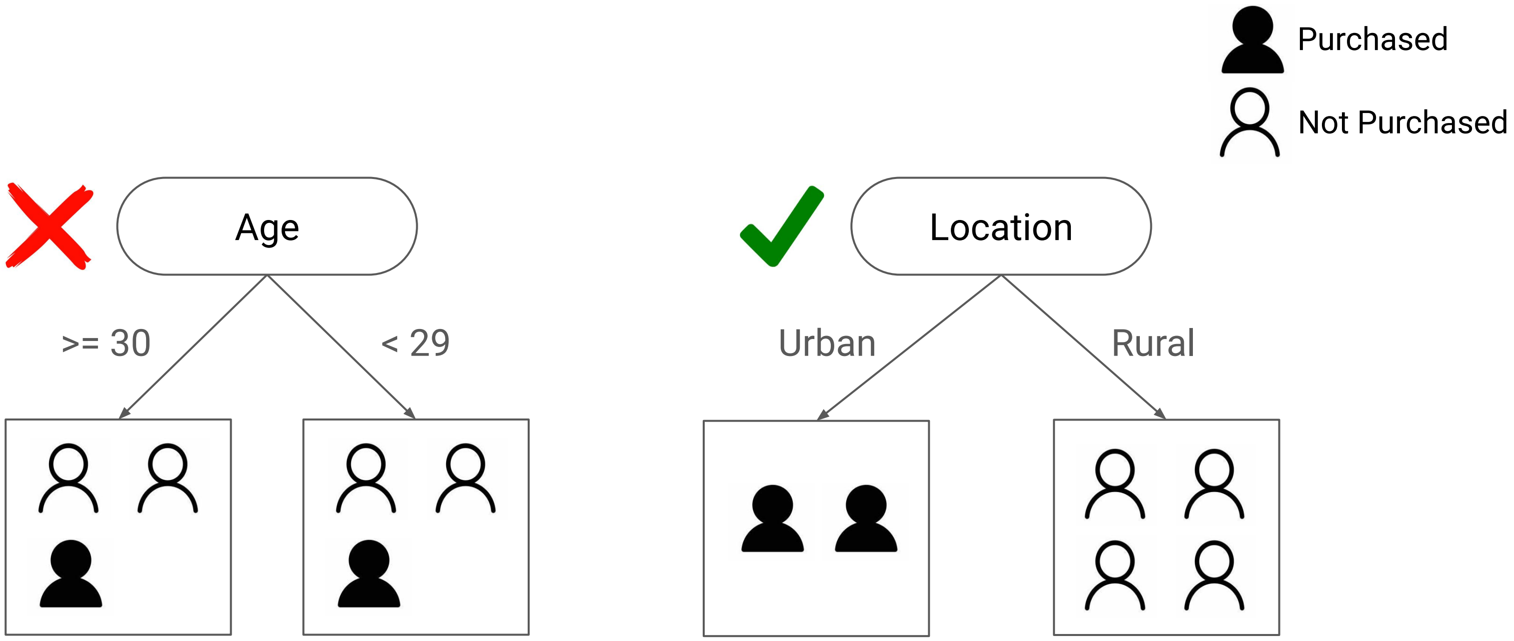
Uplift Tree

Who should we give coupons to?
(**Causality**)



Traditional Decision Tree

These tree tries to identify: “Will the customer make a purchase?”
How can we construct a better decision tree?



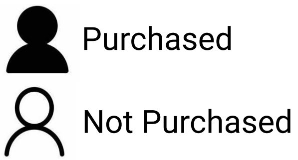
Purchasers are mixed between the two clusters

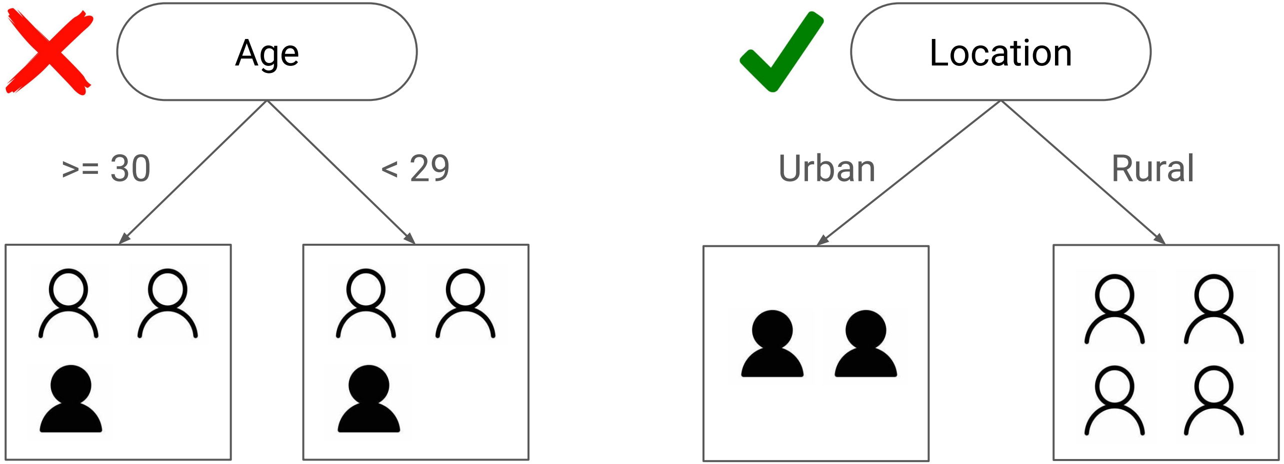
Purchasers are grouped into one cluster

For split criteria, “Gini Impurity” is used

$$Gini\ impurity = 1 - (p_1^2 + p_0^2)$$

Here, p1 is the probability of purchase and p0 is the probability of no purchase.



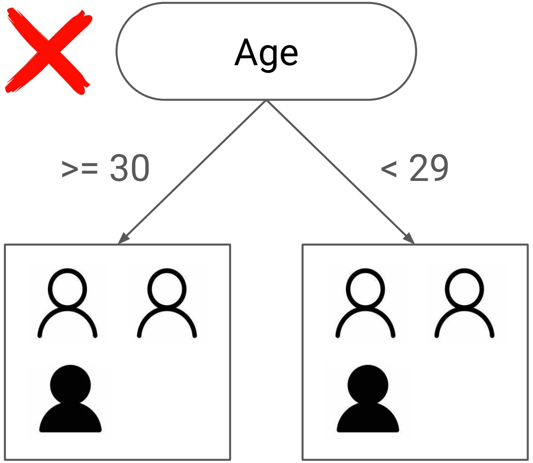


Gini impurity = 0.44
High Impurity

Gini impurity = 0
Low Impurity

Reference: Calculation of Gini Impurity for the Left Tree

$$Gini\ impurity = 1 - (p_1^2 + p_0^2)$$



Gini impurity = 0.44
High Impurity

Weighted Gini Impurity for the split:

$$= \frac{3}{6} \cdot Gini_{\text{left}} + \frac{3}{6} \cdot Gini_{\text{right}}$$

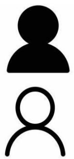
$$= \frac{3}{6} \cdot \left(1 - \left(\left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2\right)\right) + \frac{3}{6} \cdot \left(1 - \left(\left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2\right)\right)$$

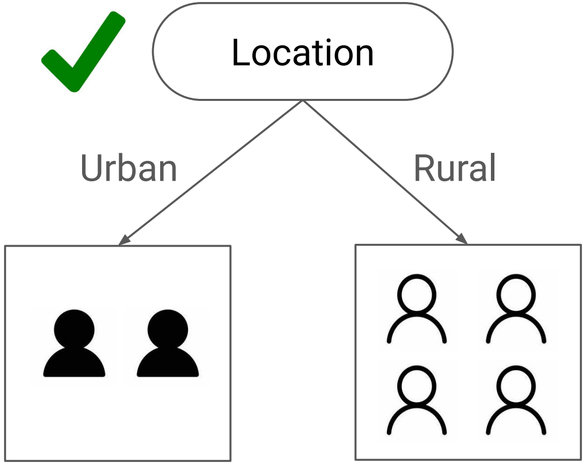
$$= \frac{4}{9}$$

$$\approx 0.44$$

Reference: Calculation of Gini Impurity for the Right Tree

$$Gini\ impurity = 1 - (p_1^2 + p_0^2)$$





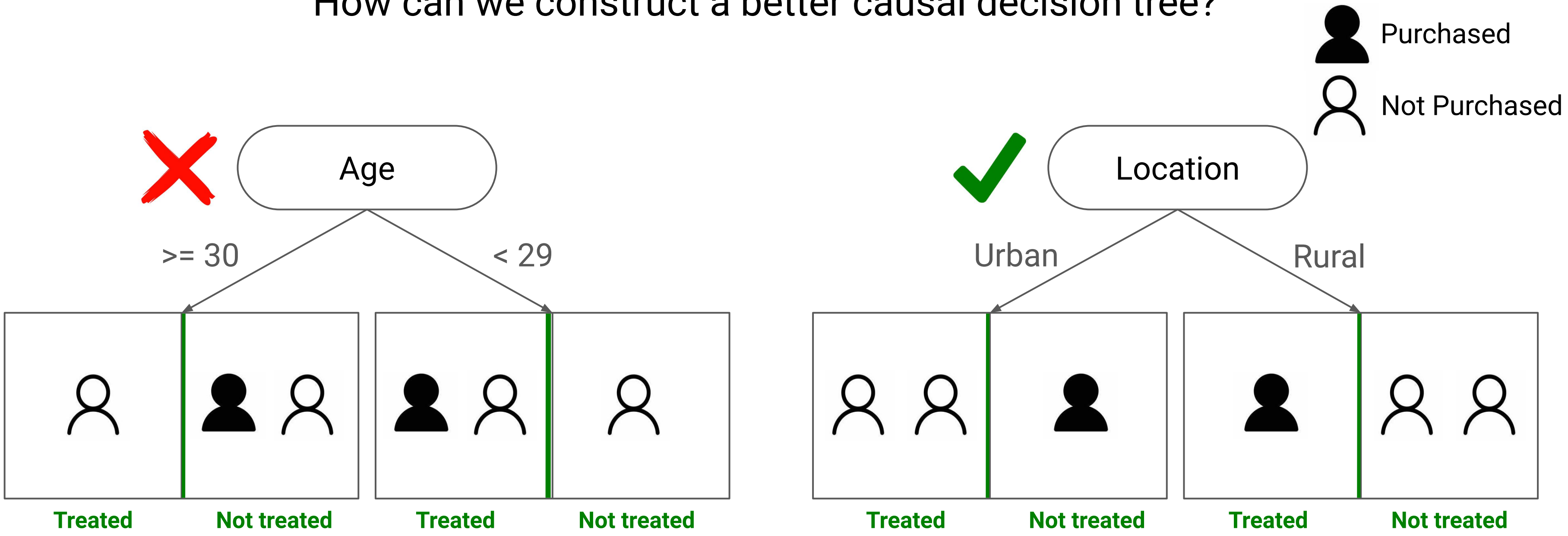
Gini impurity = 0
Low Impurity

Weighted Gini Impurity for the split:

$$
\begin{aligned}
&= \frac{2}{6} \cdot \text{Gini}_{\text{left}} + \frac{4}{6} \cdot \text{Gini}_{\text{right}} \\
&= \frac{2}{6} \cdot \left(1 - \left(\left(\frac{2}{2}\right)^2 + \left(\frac{0}{2}\right)^2\right)\right) + \frac{4}{6} \cdot \left(1 - \left(\left(\frac{0}{4}\right)^2 + \left(\frac{4}{4}\right)^2\right)\right) \\
&= \frac{2}{6} \cdot 0 + \frac{4}{6} \cdot 0 \\
&= 0
\end{aligned}
$$

Uplift Tree

This uplift tree tries to identify: “Who should we give coupons to?”
How can we construct a better causal decision tree?

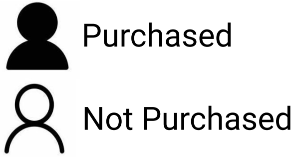


Purchasers and non-purchasers are mixed together within the same clusters

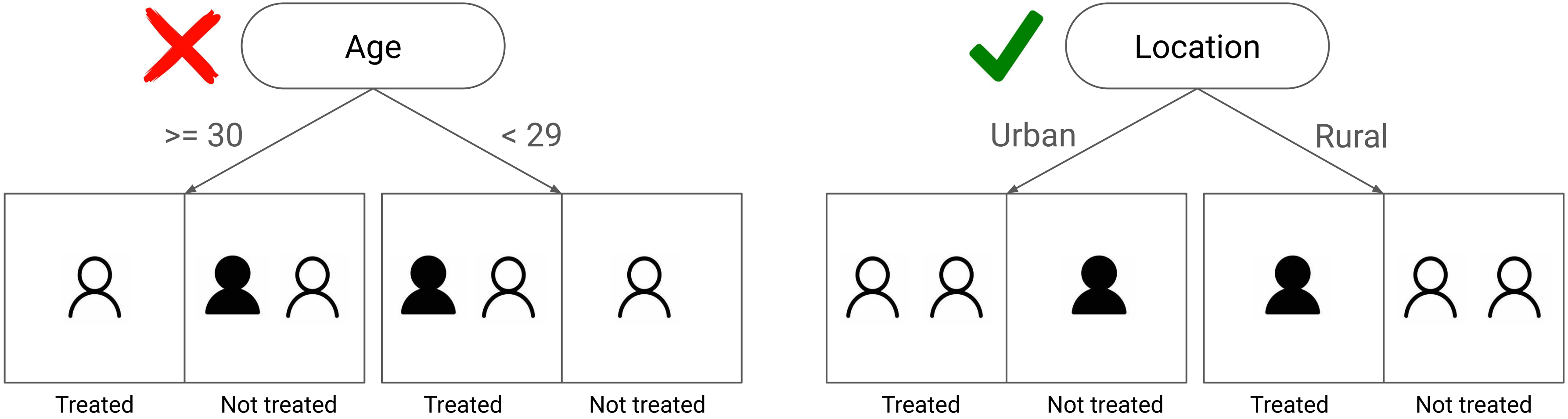
Purchasers and non-purchasers are cleanly separated

For split criteria, “Squared Euclidean Distance” is used

$$Squared\ Euclidean\ Distance = (P(0) - Q(0))^2 + (P(1) - Q(1))^2$$



- $P(1)$ / $P(0)$ is the probability of purchase / no purchase in the treatment group.
- $Q(1)$ / $Q(0)$ is the probability of purchase / no purchase in the control group.

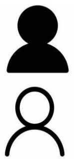


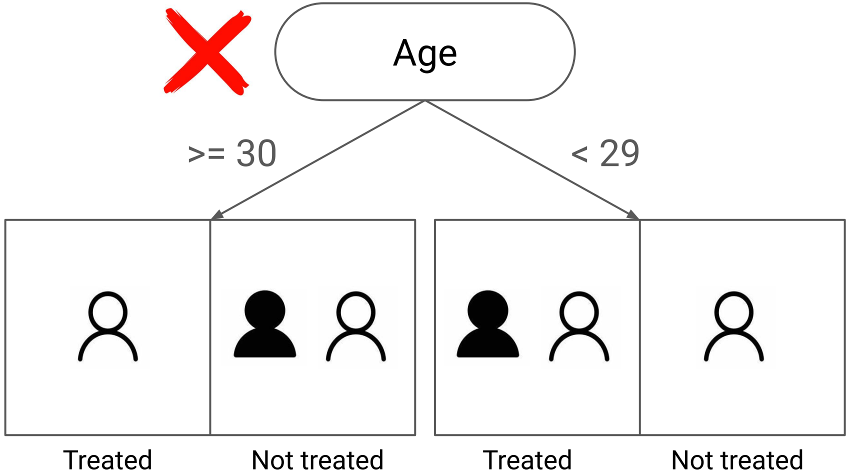
Distance = 1
Low divergence

Distance = 4
High divergence

Reference: Euclidean Distance Calculation for the Left Tree

$$Squared\ Euclidean\ Distance = (P(0) - Q(0))^2 + (P(1) - Q(1))^2$$

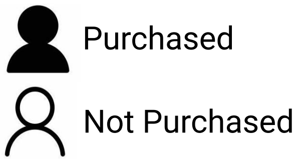


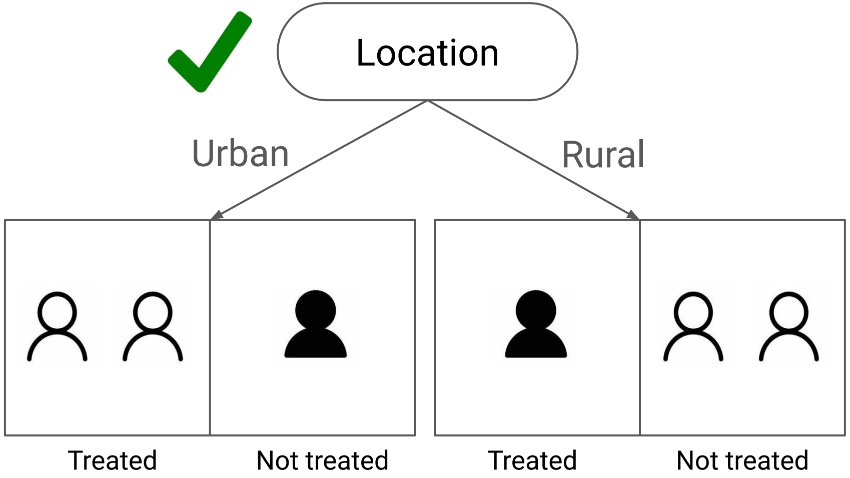


$$\begin{aligned}
&\text{Squared Euclidean distance of split} \\
&= \text{Distance of left} + \text{Distance of right} \\
&= (P_{\text{left}}(1) - Q_{\text{left}}(1))^2 + (P_{\text{left}}(0) - Q_{\text{left}}(0))^2 \\
&\quad + (P_{\text{right}}(1) - Q_{\text{right}}(1))^2 + (P_{\text{right}}(0) - Q_{\text{right}}(0))^2 \\
&= (0 - \frac{1}{2})^2 + (1 - \frac{1}{2})^2 \\
&\quad + (\frac{1}{2} - 0)^2 + (\frac{1}{2} - 1)^2 \\
&= 1
\end{aligned}$$

Reference: Euclidean Distance Calculation for the Right Tree

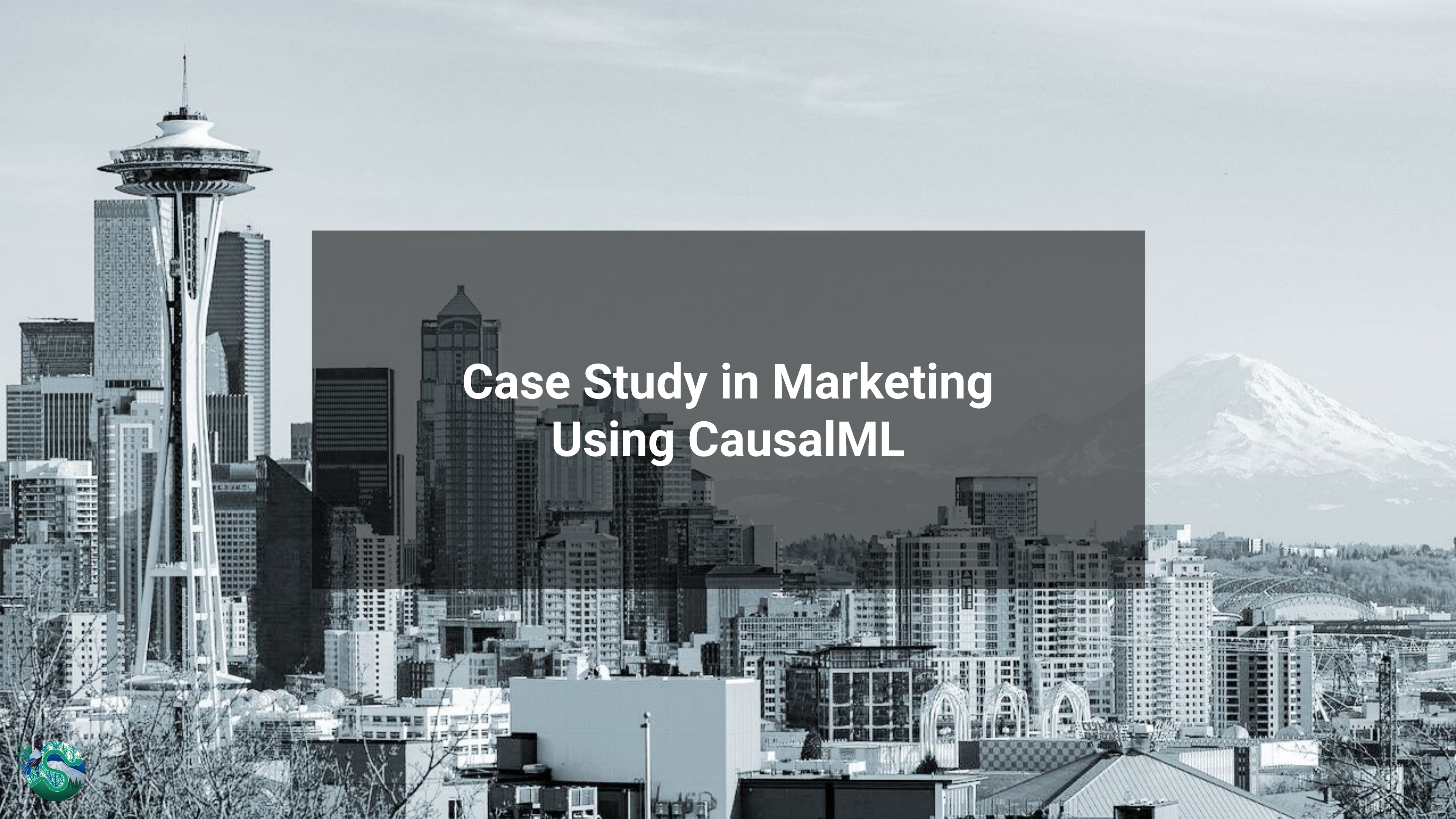
$$Squared\ Euclidean\ Distance = (P(0) - Q(0))^2 + (P(1) - Q(1))^2$$





$$\begin{aligned}
&\text{Squared Euclidean distance of split} \\
&= \text{Distance of left} + \text{Distance of right} \\
&= (P_{\text{left}}(1) - Q_{\text{left}}(1))^2 + (P_{\text{left}}(0) - Q_{\text{left}}(0))^2 \\
&\quad + (P_{\text{right}}(1) - Q_{\text{right}}(1))^2 + (P_{\text{right}}(0) - Q_{\text{right}}(0))^2 \\
&= (0-1)^2 + (1-0)^2 \\
&\quad + (1-0)^2 + (0-1)^2 \\
&= 4
\end{aligned}$$

Case Study in Marketing Using CausalML



Useful Libraries

I will use EconML for the case study #1 (meta learners)
and CausalML for the case study #2 (uplift modeling).

Library	Features	GitHub
EconML	• Covers a wide range of algorithms, strong in economics • Part of a bigger DoWhy ecosystem • Developed by Microsoft Research	py-why/EconML (3.6k star)
CausalML	• Focus on Uplift modeling and Meta Learners • Designed as a standalone tool • Developed by Uber	uber/causalml (4.8k star)

Criteo Uplift Prediction Dataset

- Fetch Criteo Data via sklift library
- Data : https://ailab.criteo.com/criteo-uplift-prediction-dataset

CRITEO

```
from sklift.datasets import fetch_criteo

X, y, treatment = fetch_criteo(target_col='conversion', treatment_col='treatment', return_X_y_t=True)

data = X.copy()
data['conversion'] = y.astype('int64')
data['treatment'] = treatment.replace({0: 'control', 1: 'treatment'})

data.head()
```

	f0	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	conversion	treatment
0	12.616365	10.059654	8.976429	4.679882	10.280525	4.115453	0.294443	4.833815	3.955396	13.190056	5.300375	-0.168679	0	treatment
1	12.616365	10.059654	9.002689	4.679882	10.280525	4.115453	0.294443	4.833815	3.955396	13.190056	5.300375	-0.168679	0	treatment
2	12.616365	10.059654	8.964775	4.679882	10.280525	4.115453	0.294443	4.833815	3.955396	13.190056	5.300375	-0.168679	0	treatment
3	12.616365	10.059654	9.002801	4.679882	10.280525	4.115453	0.294443	4.833815	3.955396	13.190056	5.300375	-0.168679	0	treatment
4	12.616365	10.059654	9.037999	4.679882	10.280525	4.115453	0.294443	4.833815	3.955396	13.190056	5.300375	-0.168679	0	treatment

Features

conversion

Treatment / control

Uplift modeling

- Train the Uplift Random Forest model (uplift_rf) and predict the uplift (y_pred)

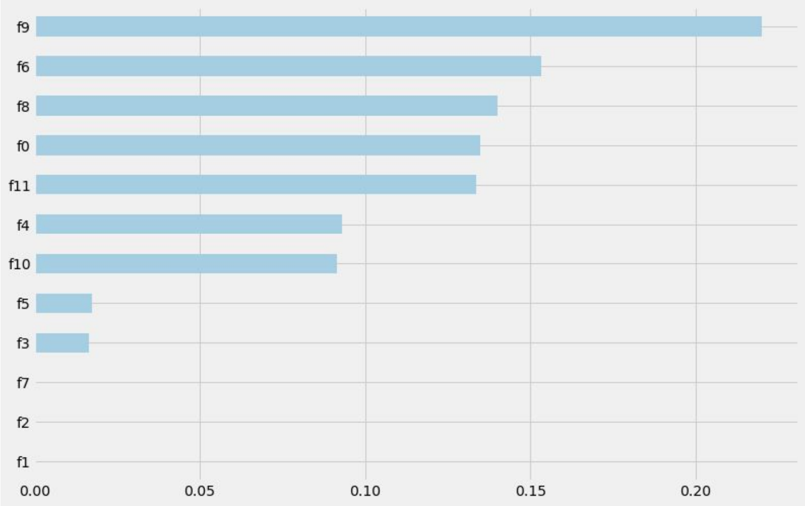
```
# Split data into training and testing sets
X_train, X_test, y_train, y_test, treatment_train, treatment_test = train_test_split(
    df.drop(columns = ['conversion', 'treatment']), df['conversion'], df['treatment'],
    test_size=0.3, random_state=42)

# Train Uplift Random Forest model
uplift_rf = UpliftRandomForestClassifier(control_name='control')
uplift_rf.fit(X_train.values, treatment=treatment_train.values, y=y_train.values)

# Predict using the trained model
y_pred = uplift_rf.predict(X_test)
```

Feature Importance

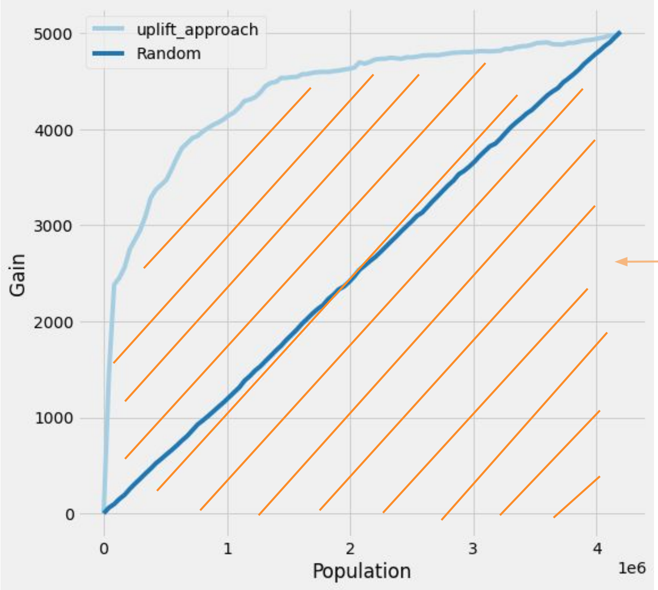
```
# Plotting the feature importance of the uplift tree
pd.Series(uplift_tree.feature_importances_, index=X.columns).sort_values().plot(kind='barh', figsize=(12,8))
```



Uplift Curve : Total Cumulative Gain

- Targeting just 20% of the total users can achieve 80% of the results as if we targeted everyone

```
plot_gain(auuc_metrics, outcome_col='conversion', treatment_col='is_treated')
plt.show()
```



AUUC
(Area Under the Uplift
Curve)

AUUC (Area Under the Uplift Curve)

- Evaluate the modeling using AUUC score.
- The concept is similar to AUC (Area Under the ROC Curve).
- The closer the AUUC is to 1, the better.

```
from causalml.metrics import auuc_score
score = auuc_score(auuc_metrics, outcome_col='conversion', treatment_col='is_treated')
print(score)
```

```
uplift    0.844375
Random    0.506221
dtype: float64
```

Extract User ID to Be Targeted

- Extract the customer IDs who should be targeted.

```python
# Calculate Rank and Decile Label
uplift_results_sorted['rank'] = uplift_results_sorted['uplift'].rank(method='first', ascending=False)
uplift_results_sorted['decile'] = pd.qcut(uplift_results_sorted['rank'], 10, labels=False)

decile_labels = [
    "top 10%", "top 10%-20%", "top 20%-30%", "top 30%-40%", "top 40%-50%",
    "top 50%-60%", "top 60%-70%", "top 70%-80%", "top 80%-90%", "bottom 10%"
]
uplift_results_sorted['decile_label'] = uplift_results_sorted['decile'].map(lambda x: decile_labels[x])
display(uplift_results_sorted.head(10))
```

	user_id	uplift	rank	decile	decile_label
0	4202557	0.132893	1.0	0	top 10%
1	3076434	0.132893	2.0	0	top 10%
2	5407785	0.132893	3.0	0	top 10%
3	1102088	0.132893	4.0	0	top 10%

Summary

Summary

1. When measuring the treatment effect of marketing activities, it is important to be mindful of **selection bias** and to control for **confounding variables**.

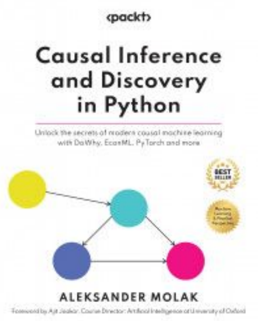
2. **Meta learners** are techniques designed to estimate treatment effects by using ML models to handle unobserved outcomes

3. **Uplift Modeling** enables the identification of customers who are most likely to respond positively to treatments, thereby improving the marketing ROI.

4. **EconML or CausalML** is a good first step.

Reference

- **CausalML** : https://github.com/uber/causalml
- **Criteo Dataset :** https://ailab.criteo.com/criteo-uplift-prediction-dataset
- **Demo Code :** https://github.com/takechanman1228/Effective-Uplif-Modeling
- **You tube :** “Decision Trees are more powerful than you think” by “CodeEmporium”
- **Book :** Causal Inference and Discovery in Python

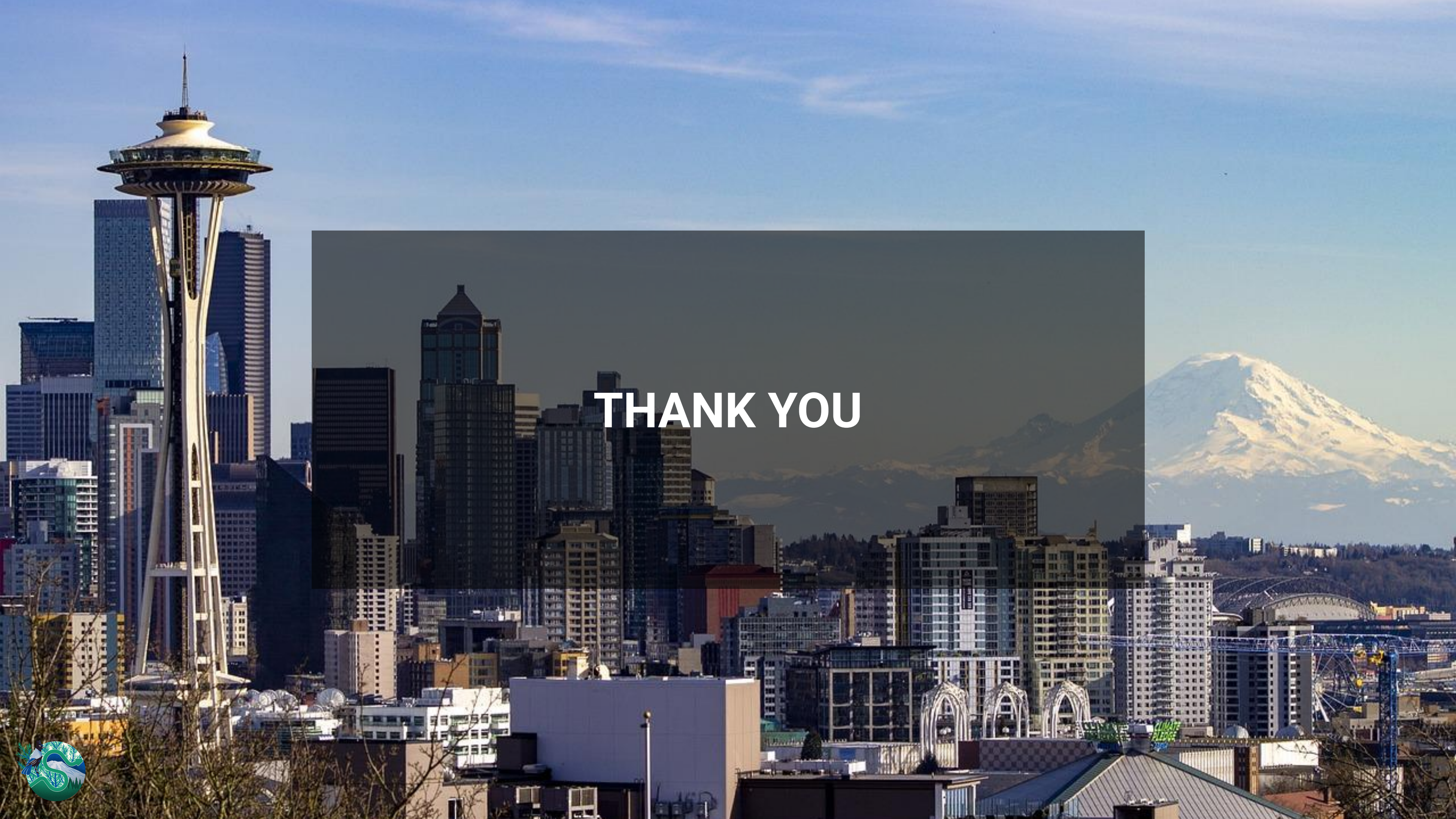


Questions/Collaboration

Feel free to contact me!

takeda.hajime.ja@gmail.com

linkedin.com/in/hajime-takeda


THANK YOU