# 02 - Estimation

## Metalearners

Roger Pros & Jordi Vitrià

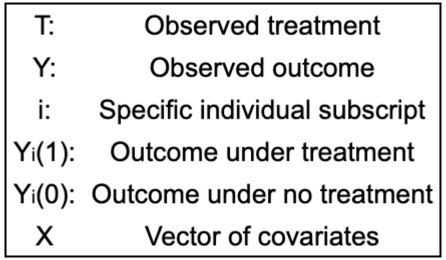
UNIVERSITAT DE BARCELONA

# In this section we’ll see

- → Causal effects estimation under the **backdoor adjustment**
- → Estimation using meta learners
- → Propensity score methods
- → Deep Learning methods
- → A real case application example by Netflix

# Preliminaries and notation

- $ITE = \mathbb{E}[Y_i | do(T_i = 1)] - \mathbb{E}[Y_i | do(T_i = 0)] = Y_i(1) - Y_i(0)$
- $ATE = \mathbb{E}[Y | do(T = 1)] - \mathbb{E}[Y | do(T = 0)] = \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)]$
- $CATE = \mathbb{E}[Y | do(T = 1), S] - \mathbb{E}[Y | do(T = 0), S] = \mathbb{E}[Y(1) | X = x] - \mathbb{E}[Y(0) | X = x]$



| | |
|---|---|
| T: | Observed treatment |
| Y: | Observed outcome |
| i: | Specific individual subscript |
| $Y_i(1)$: | Outcome under treatment |
| $Y_i(0)$: | Outcome under no treatment |
| X | Vector of covariates |

# Backdoor Adjustment

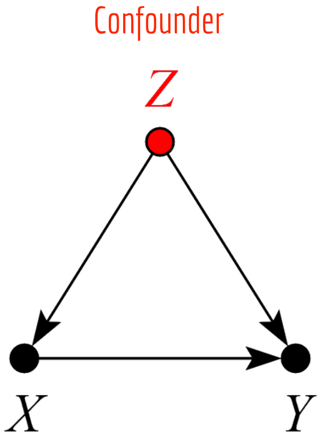
**Backdoor Criterion:** A set of variables $X$ satisfies the backdoor criterion relative to $T$ and $Y$ if the following are True:

1. $X$ blocks all backdoor paths from $T$ to $Y$
2. $X$does not contain any descendants of $T$

If $X$ satisfies the Backdoor Criterion:

$$\boxed{\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \mathbb{E}_X[\mathbb{E}[Y \mid Y = 1, X] - \mathbb{E}_X[\mathbb{E}[Y \mid Y = 0, X]]}$$

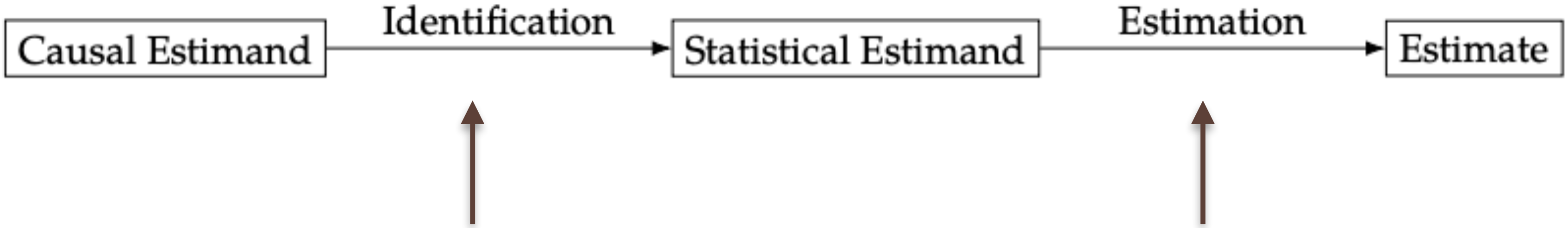
# Backdoor Adjustment http://causality.cs.ucla.edu/blog/wp-content/uploads/2019/08/clear_m_1.png



$$\Pr(Y \mid do(X)) = \sum_z \Pr(Y \mid X, Z)\Pr(Z)$$

We can compute the causal effect of $X$ on $Y$ if we control by $Z$

# Remember - Estimand based Causal Inference Workflow



Assumptions,
Backdoor,
Frontdoor,
[...]

In this section we will deal with the estimation phase of the Causal Inference Workflow when the identification method is the **Backdoor Criterion**

# Why backdoor: 1 - Its a usual scenario

- → Most real life Causal Inference problems fall into a scenario that can be identified using the backdoor criterion
- → After applying the backdoor adjustment, the statistical estimand we obtain can be estimated using all the classical methods in machine learning and statistics

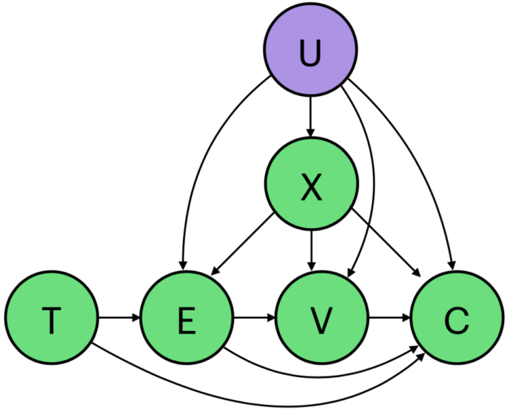


Figure 2: Causal graph of the online advertising system

Example:

Proposed Causal Graph in the Criteo benchmark dataset. X are user attributes, E are clicks, V are visits and C are conversions. U are potential unobserved confounders

https://ailab.criteo.com/criteo-uplift-prediction-dataset/

# Backdoor criterion and ML: Estimation using meta learners

# Meta Learners

→ Meta learners are discrete treatment **CATE estimators** that that can take advantage of any supervised learning or regression method in machine learning and statistics

# Meta Learners

- Meta learners are discrete treatment **CATE estimators** that that can take advantage of any supervised learning or regression method in machine learning and statistics
- They build on base algorithms such as Random Forests or Gradient Boosted Trees to estimate CATE, thus being able to leverage their strengths

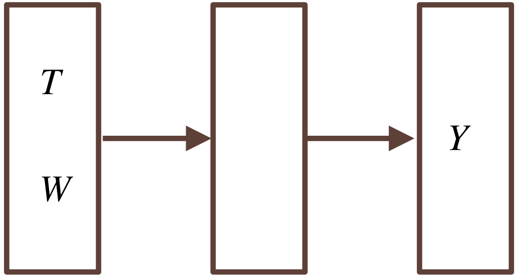
# Meta Learners

- → Meta learners are discrete treatment **CATE estimators** that that can take advantage of any supervised learning or regression method in machine learning and statistics
- → They build on base algorithms such as Random Forests or Gradient Boosted Trees to estimate CATE, thus being able to leverage their strengths
- → Meta learners assume that $X$ is a sufficient adjustment set. In other words, assuming it satisfies the **backdoor criterion**

# Estimation: SLearner

$$\Pr(Y \,|\, do(T)) = \sum_{z} \Pr(Y \,|\, T, Z)\Pr(Z) \longrightarrow y = \mathbb{E}(Y \,|\, T, Z)$$

ML model (Random Forest, MLP, etc.)

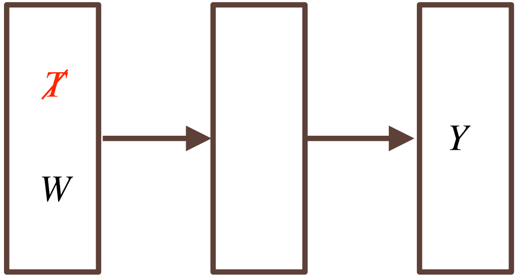


$$\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \mathbb{E}(Y \,|\, T = 1, Z) - \mathbb{E}(Y \,|\, T = 0, Z)$$

# Estimation: SLearner

$$\Pr(Y \mid do(T)) = \sum_{z} \Pr(Y \mid T, Z) \Pr(Z) \longrightarrow y = \mathbb{E}(Y \mid T, Z)$$

ML model (Random Forest, MLP, etc.)



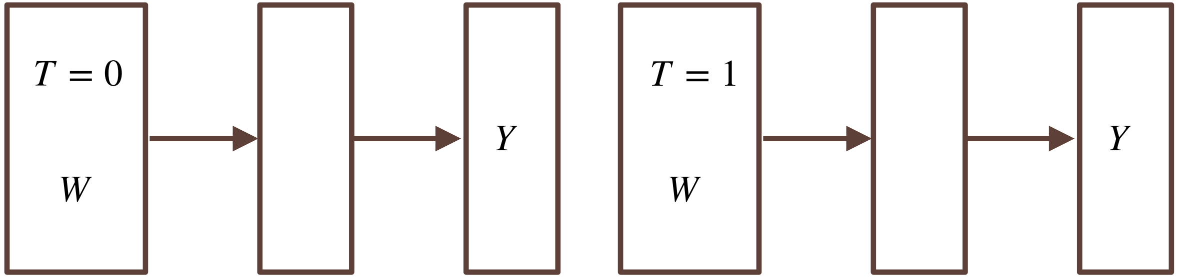
In high dimensions, the model can ignore $T$ and the estimate can be biased toward 0.

$$\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \mathbb{E}(Y \mid T = 1, Z) - \mathbb{E}(Y \mid T = 0, Z)$$

# Estimation: TLearner

$$\Pr(Y\,|\,do(T)) = \sum_{z} \Pr(Y\,|\,T, Z)\Pr(Z) \longrightarrow y = \mathbb{E}(Y\,|\,T, Z)$$

ML model (Random Forest, MLP, etc.)

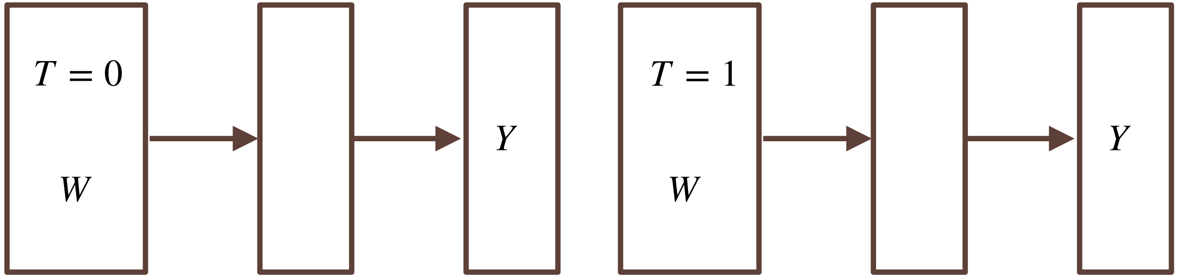


$$\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \mathbb{E}_{T=0}(Y\,|\,Z) - \mathbb{E}_{T=1}(Y\,|\,,Z)$$

# Estimation: TLearner

$$\Pr(Y|do(T)) = \sum_{z} \Pr(Y|T,Z)\Pr(Z) \longrightarrow y = \mathbb{E}(Y|T,Z)$$

ML model (Random Forest, MLP, etc.)



$$\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \mathbb{E}_{T=0}(Y|Z) - \mathbb{E}_{T=1}(Y|,Z)$$

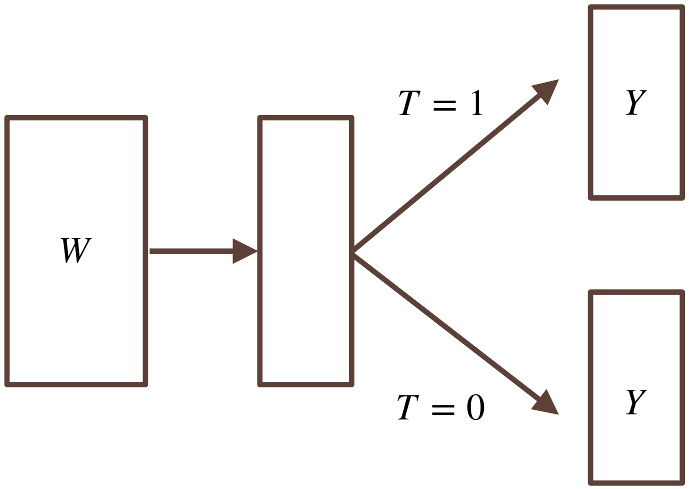
Problem: networks have higher variance than they would if they were trained with all the data (not efficient)

# Model properties intuitions

- → SLearner uses the treatment $T$ as a covariate, so in cases where the number of variables is high, it's possible the model isn't making any use of it.
  - → Due to this, **SLearner** has a **bias**.
- → **TLearner** models treatment and control group separately, so we have **less data to train each model**.
  - → Due to this, **SLearner** has a **variance**.

# Improving data efficiency: TARNet

**Intuition:** The goal of TARNet is to estimate the treatment and no treatment separately, like the TLearner, but making a more efficient use of the data.

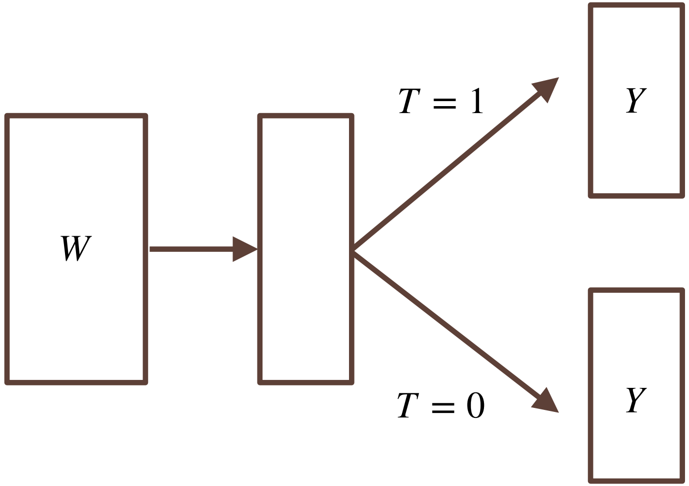


$$\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \mathbb{E}(Y \mid T = 1, Z) - \mathbb{E}(Y \mid T = 0, Z)$$

# Improving data efficiency: XLearner (TARNet)

→ This model **makes use of all the datapoints** and is forced to take into account $T$

→ Each subnetwork is still only trained with treatment group data



Neural Nets at the rescue of CI

$$\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \mathbb{E}(Y \mid T = 1, Z) - \mathbb{E}(Y \mid T = 0, Z)$$

# Existence of other estimand based methods

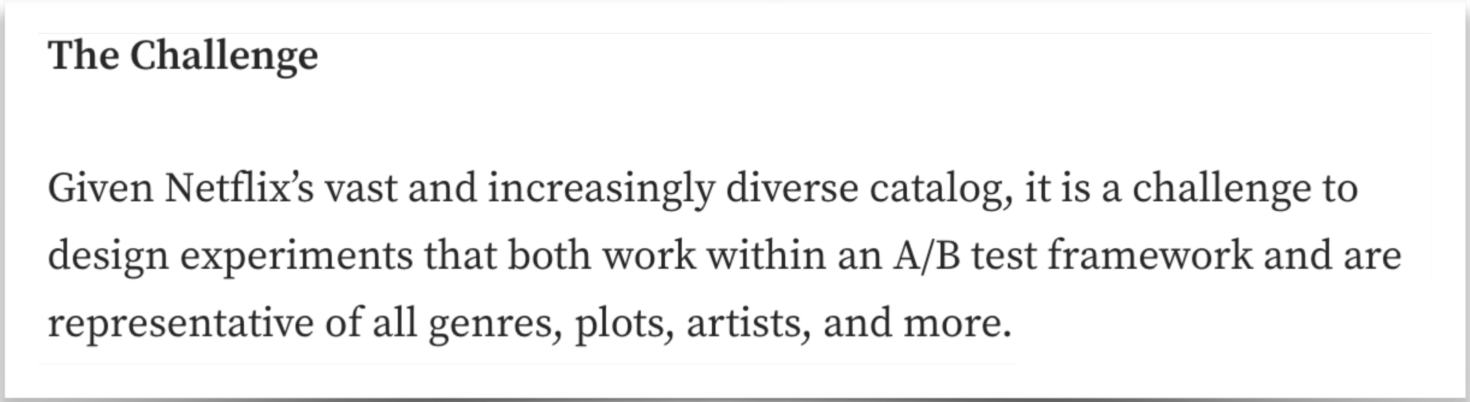
- There exists other estimand based methods when the backdoor criterion doesn’t hold:
  - Frontdoor adjustment methods
  - Instrumental Variables (IV)
- These cases are not as common as the backdoor cases and usually require a more customised approach

⚠ Out of scope for this course of this talk!

# Real Case by Netflix

# Netflix Case

https://netflixtechblog.medium.com/causal-machine-learning-for-creative-insights-4b0ce22a8a96

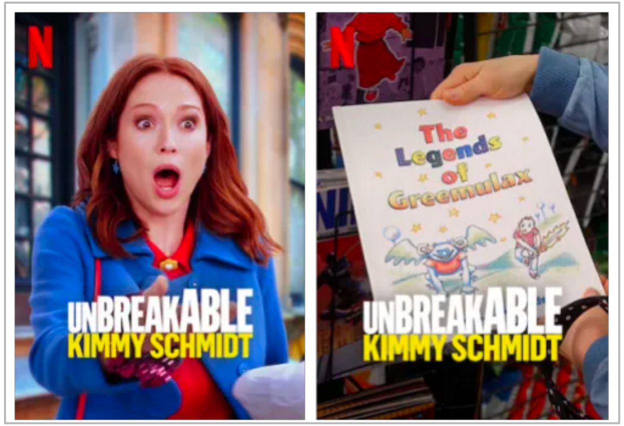


### The Challenge

Given Netflix's vast and increasingly diverse catalog, it is a challenge to design experiments that both work within an A/B test framework and are representative of all genres, plots, artists, and more.

# Netflix Case

https://netflixtechblog.medium.com/causal-machine-learning-for-creative-insights-4b0ce22a8a96



They know that the image on the left performed better than the image on the right. However, the difference between them is not only the presence of a face. There are many other variances, like the difference in background, text placement, font size, face size, etc.

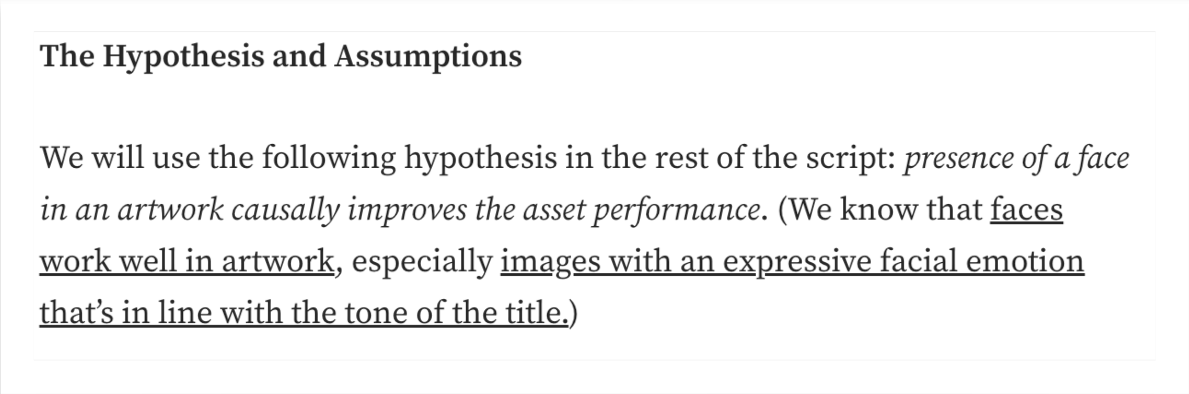
# Netflix Case

https://netflixtechblog.medium.com/causal-machine-learning-for-creative-insights-4b0ce22a8a96

Two many combinations to perform AB Testing!

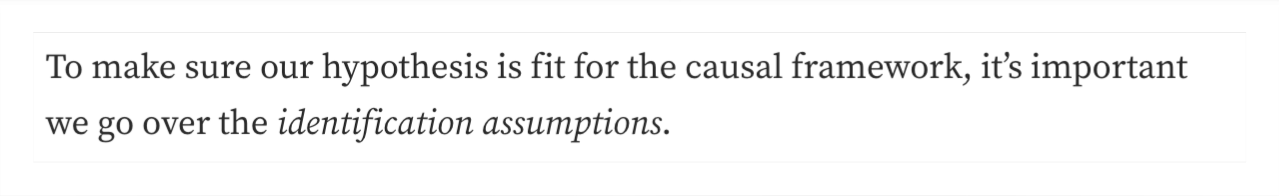
# Netflix Case

https://netflixtechblog.medium.com/causal-machine-learning-for-creative-insights-4b0ce22a8a96



### The Hypothesis and Assumptions

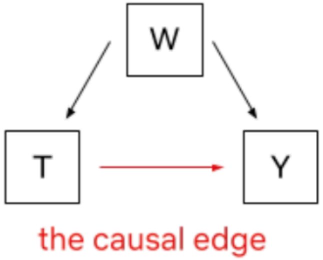
We will use the following hypothesis in the rest of the script: *presence of a face in an artwork causally improves the asset performance*. (We know that faces work well in artwork, especially images with an expressive facial emotion that's in line with the tone of the title.)

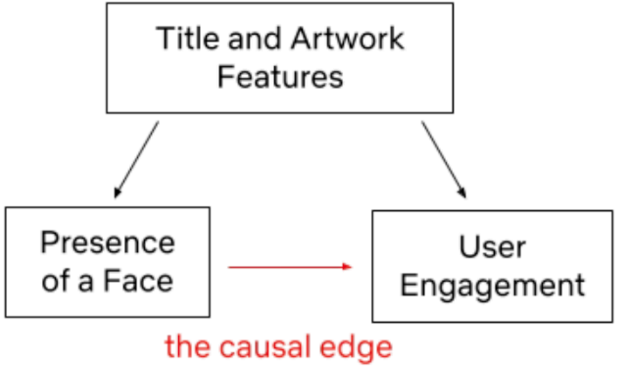


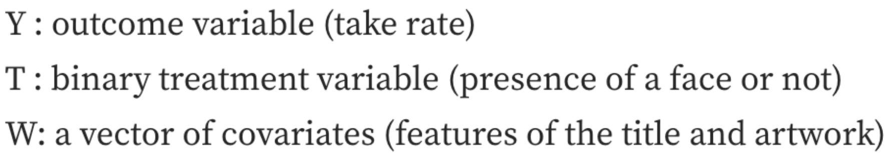
To make sure our hypothesis is fit for the causal framework, it's important we go over the *identification assumptions*.

# Netflix Case

https://netflixtechblog.medium.com/causal-machine-learning-for-creative-insights-4b0ce22a8a96







Y : outcome variable (take rate)

T : binary treatment variable (presence of a face or not)

W: a vector of covariates (features of the title and artwork)

# Netflix Case

https://netflixtechblog.medium.com/causal-machine-learning-for-creative-insights-4b0ce22a8a96

This a backdoor scenario with a rich covariate set!
Let’s train Metalearners to estimate the causal effect of a face!