# Machine Learning for Causal Inference (Illustrated by Outcome Modeling)

Ian Lundberg

# Learning goals for today

At the end of class, you will be able to:

1. Use machine learning methods to estimate causal effects
2. Select an estimator using predictive performance

# Causal inference by outcome modeling

1. Assume a DAG


2. By consistency, exchangeability, and positivity,

$$\underbrace{\text{E}(Y^a \mid \vec{L} = \vec{\ell})}_{\text{Causal}} = \underbrace{\text{E}(Y \mid A = a, \vec{L} = \vec{\ell})}_{\text{Statistical}}$$

3. Using regression, estimate $\hat{\text{E}}(Y \mid A, \vec{L})$
4. Predict unknown potential outcomes and average

$$\hat{\text{E}}(Y^a) = \frac{1}{n}\sum_{i=1}^{n} \hat{\text{E}}\left(Y \mid A = a, \vec{L} = \vec{\ell}_i\right)$$

**Big idea:** Why constrain ourselves to regression for $\hat{\text{E}}(Y \mid A, \vec{L})$?

# Causal inference by outcome modeling with machine learning[1]

| Inputs | → | Prediction Algorithm | → | Estimated Average Effect |
|---|---|---|---|---|
| $\{\vec{L}, A, Y\}$ | | $\hat{Y} = \hat{f}(\vec{L}, A)$ | | $\hat{\mathrm{E}}(Y^a) = \frac{1}{n}\sum_{i=1}^n \hat{f}(a, \vec{\ell}_i)$ |

(all relies on assumed DAG)

[1] Caveat: There are ways to do even better. This is just a start.
See Van der Laan, M. J., & Rose, S. (2018). Targeted learning in data science. Springer International Publishing.

Hill, Jennifer L. 2011.
"Bayesian nonparametric modeling for causal inference."
Journal of Computational and Graphical Statistics 20.1:217-240.

- Binary treatment (simulated)
- Continuous confounder $X$ (simulated)

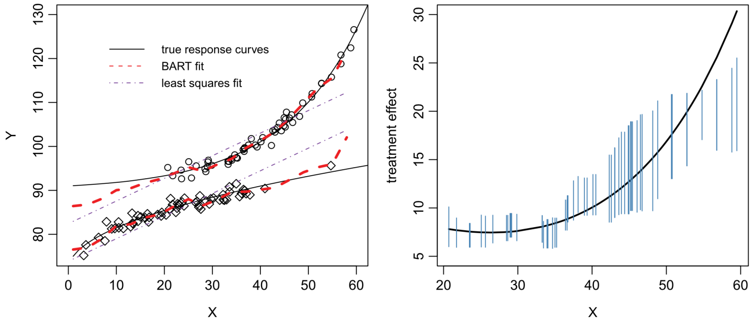


Figure 1. Left panel: simulated data with linear regression and BART fits. Right panel: BART inference for treatment effect on the treated. A color version of this figure is available in the electronic version of this article.

# Hill (2011) prediction algorithm[2]

1) Learn an automated partitioning of the data (aka a "tree")

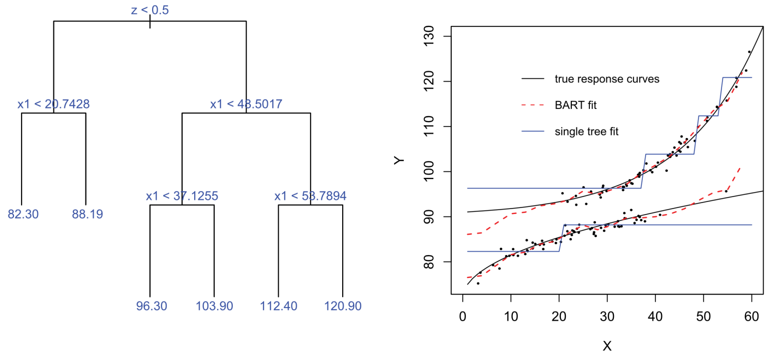


Figure 2. Left panel: the binary tree fit to the data from Figure 1. Right panel: single-tree fits (solid lines) and BART fits (dashed lines). A color version of this figure is available in the electronic version of this article.

[2]Chipman, Hugh A., Edward I. George, and Robert E. McCulloch. "BART: Bayesian additive regression trees." The Annals of Applied Statistics 4.1 (2010): 266-298.

# Hill (2011) prediction algorithm

2) Repeat many times. Take the average.

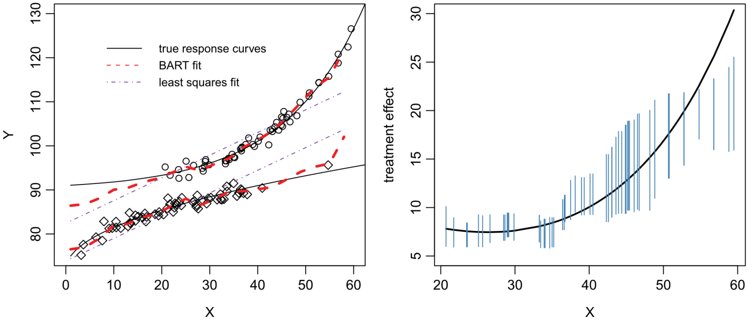


Figure 1. Left panel: simulated data with linear regression and BART fits. Right panel: BART inference for treatment effect on the treated. A color version of this figure is available in the electronic version of this article.

Many candidate prediction algorithms

- OLS

Many candidate prediction algorithms

- OLS
- Penalized linear regression

Many candidate prediction algorithms

- OLS
- Penalized linear regression
- Random forest

Many candidate prediction algorithms

- OLS
- Penalized linear regression
- Random forest

How do you choose?

Many candidate prediction algorithms

- OLS
- Penalized linear regression
- Random forest

How do you choose?

- Try them all

Many candidate prediction algorithms

- OLS
- Penalized linear regression
- Random forest

How do you choose?

- Try them all
- See what predicts best out-of-sample

# Selecting an algorithm: The role of a train-test split

| | | |
|---|---|---|
| Case 1 | $\{\vec{L}_1, A_1\}$ | $Y_1$ |
| Case 2 | $\{\vec{L}_2, A_2\}$ | $Y_2$ |
| Case 3 | $\{\vec{L}_3, A_3\}$ | $Y_3$ |
| Case 4 | $\{\vec{L}_4, A_4\}$ | $Y_4$ |
| Case 5 | $\{\vec{L}_5, A_5\}$ | $Y_5$ |
| Case 6 | $\{\vec{L}_6, A_6\}$ | $Y_6$ |
| Case 7 | $\{\vec{L}_7, A_7\}$ | $Y_7$ |
| Case 8 | $\{\vec{L}_8, A_8\}$ | $Y_8$ |
| Case 9 | $\{\vec{L}_9, A_9\}$ | $Y_9$ |

# Selecting an algorithm: The role of a train-test split

1) Randomly assign cases to a `train` and `test` set

| | | | |
|---|---|---|---|
| `train` | Case 1 | $\{\vec{L}_1, A_1\}$ | $Y_1$ |
| `train` | Case 2 | $\{\vec{L}_2, A_2\}$ | $Y_2$ |
| `test` | Case 3 | $\{\vec{L}_3, A_3\}$ | $Y_3$ |
| `train` | Case 4 | $\{\vec{L}_4, A_4\}$ | $Y_4$ |
| `test` | Case 5 | $\{\vec{L}_5, A_5\}$ | $Y_5$ |
| `test` | Case 6 | $\{\vec{L}_6, A_6\}$ | $Y_6$ |
| `test` | Case 7 | $\{\vec{L}_7, A_7\}$ | $Y_7$ |
| `train` | Case 8 | $\{\vec{L}_8, A_8\}$ | $Y_8$ |
| `train` | Case 9 | $\{\vec{L}_9, A_9\}$ | $Y_9$ |

# Selecting an algorithm: The role of a train-test split

2) First, use only the `train` set.

| | | | |
|---|---|---|---|
| `train` | Case 1 | $\{\vec{L}_1, A_1\}$ | $Y_1$ |
| `train` | Case 2 | $\{\vec{L}_2, A_2\}$ | $Y_2$ |
| `test` | Case 3 | $\{\vec{L}_3, A_3\}$ | $Y_3$ |
| `train` | Case 4 | $\{\vec{L}_4, A_4\}$ | $Y_4$ |
| `test` | Case 5 | $\{\vec{L}_5, A_5\}$ | $Y_5$ |
| `test` | Case 6 | $\{\vec{L}_6, A_6\}$ | $Y_6$ |
| `test` | Case 7 | $\{\vec{L}_7, A_7\}$ | $Y_7$ |
| `train` | Case 8 | $\{\vec{L}_8, A_8\}$ | $Y_8$ |
| `train` | Case 9 | $\{\vec{L}_9, A_9\}$ | $Y_9$ |

# Selecting an algorithm: The role of a train-test split

2) First, use only the `train` set. Learn a prediction function.

$$f() : \{\vec{L}, A\} \rightarrow Y$$

| | | | | |
|---|---|---|---|---|
| `train` | Case 1 | $\{\vec{L}_1, A_1\}$ | $\longrightarrow$ | $Y_1$ |
| `train` | Case 2 | $\{\vec{L}_2, A_2\}$ | $\longrightarrow$ | $Y_2$ |
| `test` | Case 3 | $\{\vec{L}_3, A_3\}$ | | $Y_3$ |
| `train` | Case 4 | $\{\vec{L}_4, A_4\}$ | $\longrightarrow$ | $Y_4$ |
| `test` | Case 5 | $\{\vec{L}_5, A_5\}$ | | $Y_5$ |
| `test` | Case 6 | $\{\vec{L}_6, A_6\}$ | | $Y_6$ |
| `test` | Case 7 | $\{\vec{L}_7, A_7\}$ | | $Y_7$ |
| `train` | Case 8 | $\{\vec{L}_8, A_8\}$ | $\longrightarrow$ | $Y_8$ |
| `train` | Case 9 | $\{\vec{L}_9, A_9\}$ | $\longrightarrow$ | $Y_9$ |

# Selecting an algorithm: The role of a train-test split

3) Open the `test` set.

| | | | | |
|---|---|---|---|---|
| train | Case 1 | $\{\vec{L}_1, A_1\}$ | $\longrightarrow$ | $Y_1$ |
| train | Case 2 | $\{\vec{L}_2, A_2\}$ | $\longrightarrow$ | $Y_2$ |
| test | Case 3 | $\{\vec{L}_3, A_3\}$ | | $Y_3$ |
| train | Case 4 | $\{\vec{L}_4, A_4\}$ | $\longrightarrow$ | $Y_4$ |
| test | Case 5 | $\{\vec{L}_5, A_5\}$ | | $Y_5$ |
| test | Case 6 | $\{\vec{L}_6, A_6\}$ | | $Y_6$ |
| test | Case 7 | $\{\vec{L}_7, A_7\}$ | | $Y_7$ |
| train | Case 8 | $\{\vec{L}_8, A_8\}$ | $\longrightarrow$ | $Y_8$ |
| train | Case 9 | $\{\vec{L}_9, A_9\}$ | $\longrightarrow$ | $Y_9$ |

# Selecting an algorithm: The role of a train-test split

3) Open the `test` set. Predict.

| | | | | | |
|---|---|---|---|---|---|
| `train` | Case 1 | $\{\vec{L}_1, A_1\}$ | $\longrightarrow$ | | $Y_1$ |
| `train` | Case 2 | $\{\vec{L}_2, A_2\}$ | $\longrightarrow$ | | $Y_2$ |
| `test` | Case 3 | $\{\vec{L}_3, A_3\}$ | $\longrightarrow$ | $\hat{Y}_3$ | $Y_3$ |
| `train` | Case 4 | $\{\vec{L}_4, A_4\}$ | $\longrightarrow$ | | $Y_4$ |
| `test` | Case 5 | $\{\vec{L}_5, A_5\}$ | $\longrightarrow$ | $\hat{Y}_5$ | $Y_5$ |
| `test` | Case 6 | $\{\vec{L}_6, A_6\}$ | $\longrightarrow$ | $\hat{Y}_6$ | $Y_6$ |
| `test` | Case 7 | $\{\vec{L}_7, A_7\}$ | $\longrightarrow$ | $\hat{Y}_7$ | $Y_7$ |
| `train` | Case 8 | $\{\vec{L}_8, A_8\}$ | $\longrightarrow$ | | $Y_8$ |
| `train` | Case 9 | $\{\vec{L}_9, A_9\}$ | $\longrightarrow$ | | $Y_9$ |

# Selecting an algorithm: The role of a train-test split

3) Open the `test` set. Predict. Evaluate squared error.

| | | | | |
|---|---|---|---|---|
| `train` | Case 1 | $\{\vec{L}_1, A_1\}$ | $\longrightarrow$ | $Y_1$ |
| `train` | Case 2 | $\{\vec{L}_2, A_2\}$ | $\longrightarrow$ | $Y_2$ |
| `test` | Case 3 | $\{\vec{L}_3, A_3\}$ | $\longrightarrow$ | $(\hat{Y}_3 - Y_3)^2$ |
| `train` | Case 4 | $\{\vec{L}_4, A_4\}$ | $\longrightarrow$ | $Y_4$ |
| `test` | Case 5 | $\{\vec{L}_5, A_5\}$ | $\longrightarrow$ | $(\hat{Y}_5 - Y_5)^2$ |
| `test` | Case 6 | $\{\vec{L}_6, A_6\}$ | $\longrightarrow$ | $(\hat{Y}_6 - Y_6)^2$ |
| `test` | Case 7 | $\{\vec{L}_7, A_7\}$ | $\longrightarrow$ | $(\hat{Y}_7 - Y_7)^2$ |
| `train` | Case 8 | $\{\vec{L}_8, A_8\}$ | $\longrightarrow$ | $Y_8$ |
| `train` | Case 9 | $\{\vec{L}_9, A_9\}$ | $\longrightarrow$ | $Y_9$ |

# Selecting an algorithm: The role of a train-test split

3) Open the `test` set. Predict. Evaluate squared error. Average.

| | | | | |
|---|---|---|---|---|
| train | Case 1 | $\{\vec{L}_1, A_1\}$ | $\longrightarrow$ | $Y_1$ |
| train | Case 2 | $\{\vec{L}_2, A_2\}$ | $\longrightarrow$ | $Y_2$ |
| test | Case 3 | $\{\vec{L}_3, A_3\}$ | $\longrightarrow$ | $(\hat{Y}_3 - Y_3)^2$ |
| train | Case 4 | $\{\vec{L}_4, A_4\}$ | $\longrightarrow$ | $Y_4$ |
| test | Case 5 | $\{\vec{L}_5, A_5\}$ | $\longrightarrow$ | $(\hat{Y}_5 - Y_5)^2$ |
| test | Case 6 | $\{\vec{L}_6, A_6\}$ | $\longrightarrow$ | $(\hat{Y}_6 - Y_6)^2$ |
| test | Case 7 | $\{\vec{L}_7, A_7\}$ | $\longrightarrow$ | $(\hat{Y}_7 - Y_7)^2$ |
| train | Case 8 | $\{\vec{L}_8, A_8\}$ | $\longrightarrow$ | $Y_8$ |
| train | Case 9 | $\{\vec{L}_9, A_9\}$ | $\longrightarrow$ | $Y_9$ |

$$\widehat{\text{MSE}} = \frac{1}{n_{\texttt{test}}} \sum_{i \in \texttt{test}} (\hat{Y}_i - Y_i)^2$$

# Then estimate the average causal effect

| **Inputs** → | **Prediction Algorithm** → | **Estimated Average Effect** |
|---|---|---|
| $\{\vec{L}, A, Y\}$ | $\hat{Y} = \hat{f}(\vec{L}, A)$ | $\hat{\text{E}}(Y^a) = \frac{1}{n}\sum_{i=1}^{n}\hat{f}(a, \vec{\ell}_i)$ |

(all relies on assumed DAG)

# Learning goals for today

At the end of class, you will be able to:

1. Use machine learning methods to estimate causal effects
2. Select an estimator using predictive performance