UNIVERSITY OF COPENHAGEN

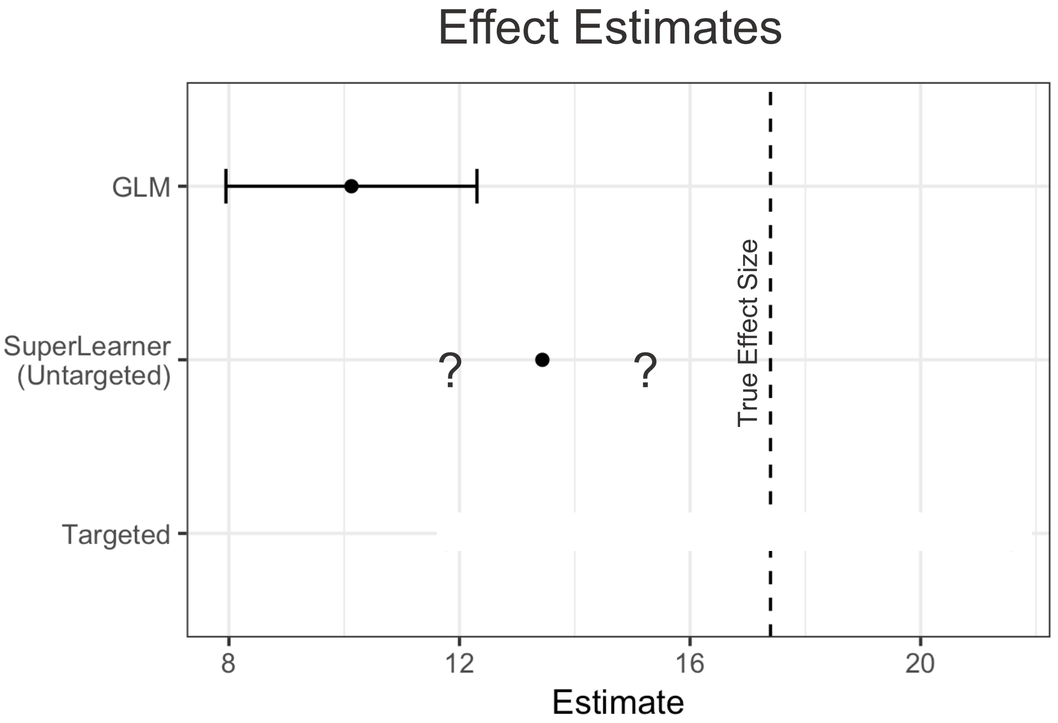
Targeted register analyses
PhD short course

An Introduction to Targeted Maximum Likelihood Estimation

Andrew Mertens
University of California, Berkeley, Division of Biostatistics

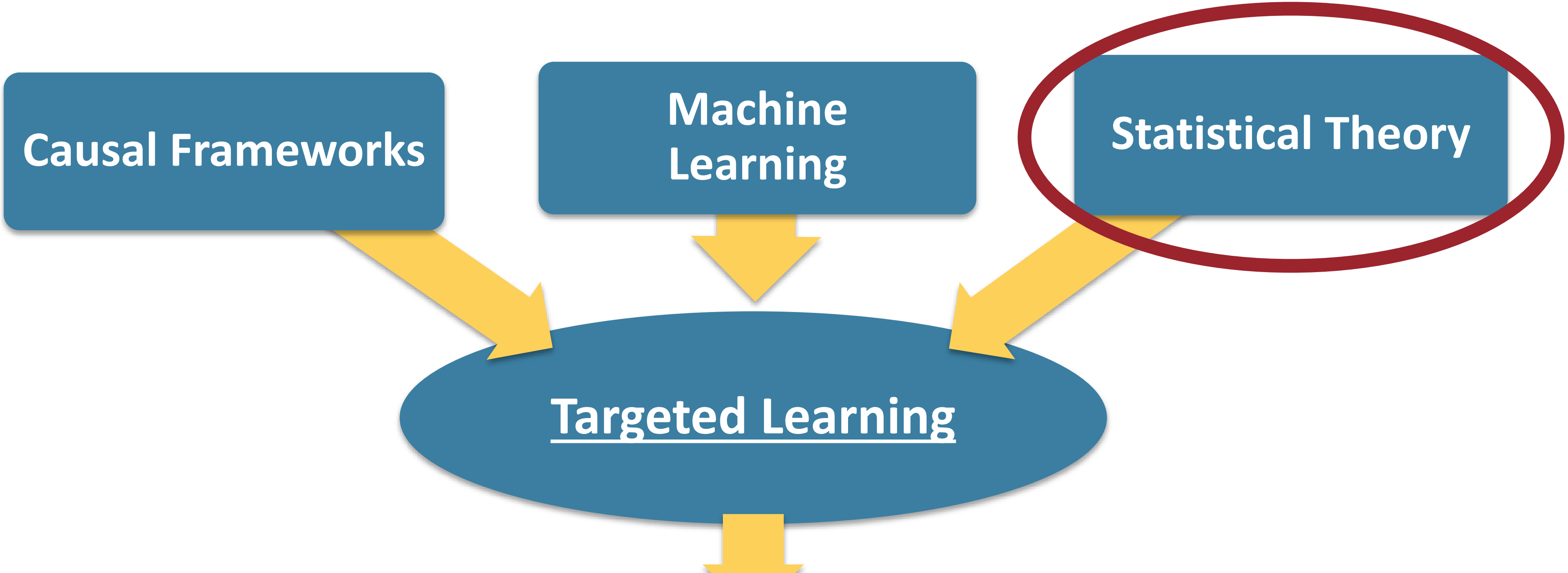
Results: removing bias compared to individual prediction

But what about inference?



GLM did not learn the correct outcome mechanism, so its estimate is very biased

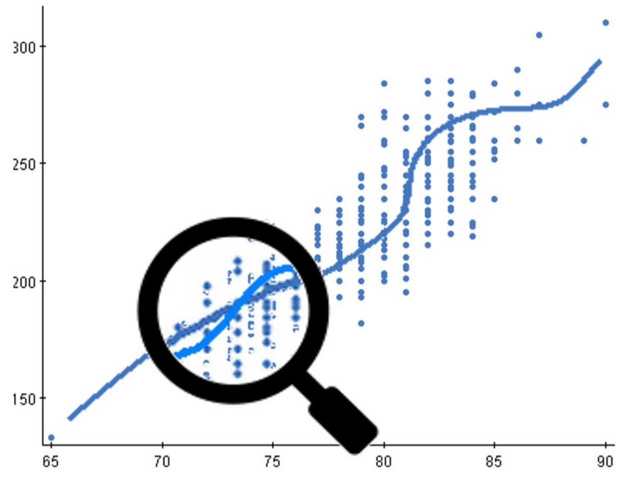
SuperLearner does a better job of estimating the outcome mechanism, but does not allow valid inference



Better (more precise) **answers** to **causal** (actionable) **questions** with **accurate quantification of uncertainty** (signal from noise)

Beyond Machine Learning…

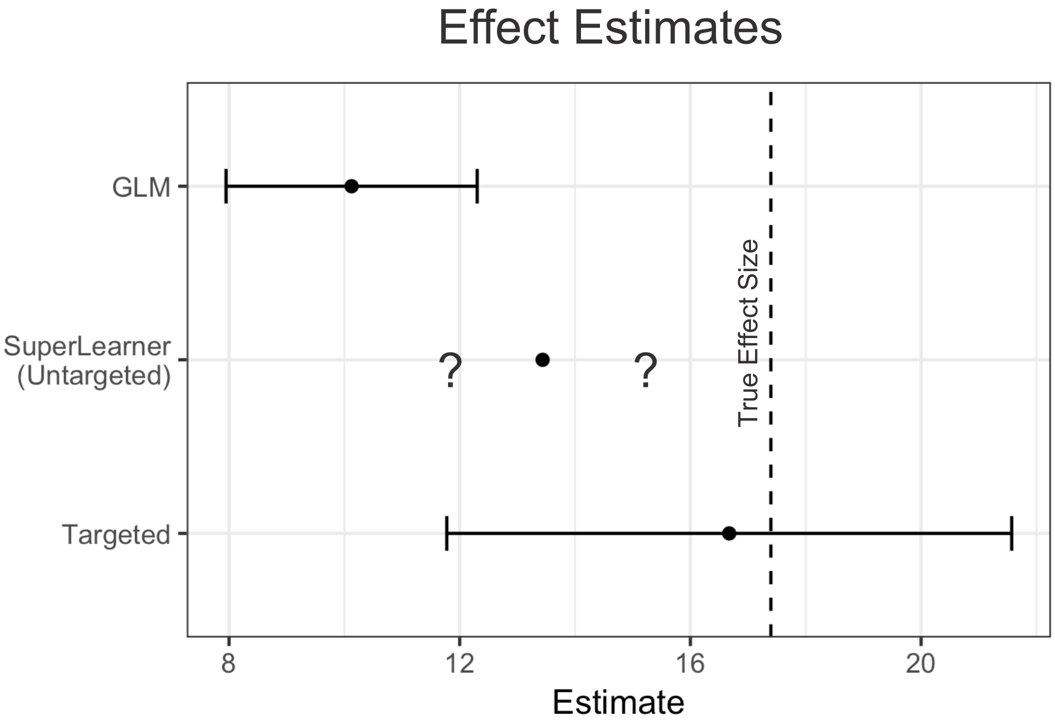
- Could use SuperLearner for more than prediction
  - Fit SuperLearner
  - Evaluate using a “plug in“ estimator for average treatment effect
- Paying a price for too much ambition!
  - Super Learner- trying to do a good job on a harder question
- The costs: Increased bias and variance, no inference



Targeting: Update initial Super learner fit

- Dont try to do a good job for all (causal) questions at once
- Focus estimation where it matters most for the question at hand
- Update depends on other “nuisance parameters“ like propensity score

Results: removing bias AND robust inference



GLM did not learn the correct outcome mechanism, so its estimate is very biased

SuperLearner does a better job of estimating the outcome mechanism, but does not allow valid inference

TMLE combines good outcome mechanism estimation with targeting to get valid inference

Properties of TMLE

Why go through all the effort?

- Can incorporate Machine Learning
  - G-computation and IPTW approaches also allow for machine learning for inference
    - Methods exist for inference, including bootstrapping
    - But both are susceptible to model misspecification, especially if using parametric methods
- Efficient
  - Lowest (asymptotic) variance among reasonable estimators if both **Q(*Y*|*A,W*)** AND ***g*(*A*|*W*)** estimated consistently at reasonable rates
- Substitution (aka ”plug in”) Estimator
  - Its estimates will always stay within the bounds of the original outcome
  - Improved robustness to sparse data compared to estimating equation alternatives
- Double Robust
  - Consistent if either **Q(*Y*|*A,W*)** or ***g*(*A*|*W*)** estimated consistently

Example data for a walkthrough of TMLE steps

Parameter of interest

- Parameter of interest: Average Treatment Effect Y1-Y0 for a point-treatment (not longitudinal) effect

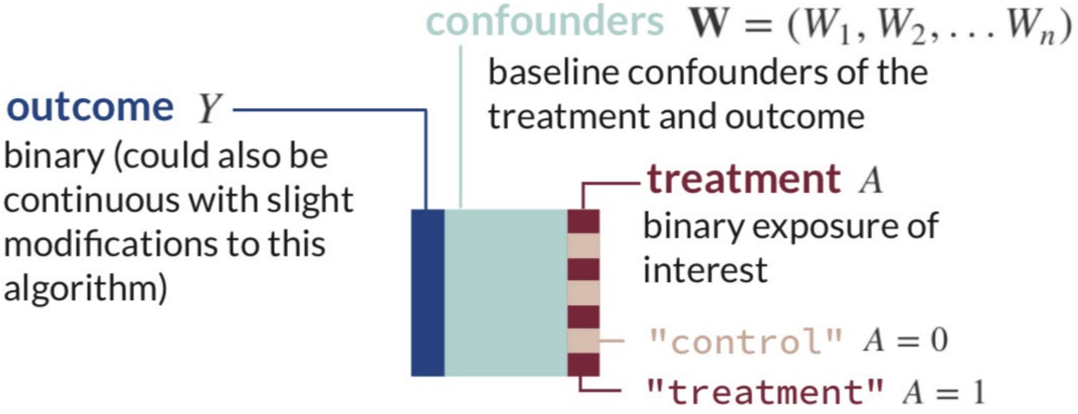
$$ATE = \Psi = E_W[\mathrm{E}[Y|A=1,\mathbf{W}] - \mathrm{E}[Y|A=0,\mathbf{W}]]$$

- Many other parameters possible (causal relative risk, treatment specific mean, more advanced discussed at the end)
- Longitudinal TMLE coming Thursday (Thanks Zeyi!)
- Remember, must meet identifiability assumptions outlined in the causal roadmap for causal inference, otherwise this is an associational analysis

Example data for a walkthrough of TMLE steps

Structure of data

$$ATE = \Psi = E_W[\mathrm{E}[Y|A=1,\mathbf{W}] - \mathrm{E}[Y|A=0,\mathbf{W}]]$$



Example data for a walkthrough of TMLE steps

Example data

Simulated data set.					
Y	W1	W2	W3	W4	A
1	1	0	6	1	1
1	0	1	6	3	0
0	0	0	3	2	0
1	0	1	5	1	1
1	0	0	5	2	0
1	0	1	6	1	1

TMLE: the algorithm steps

1. Estimate the Outcome
2. Estimate the Probability of Treatment
3. Estimate the Fluctuation Parameter
4. Update the Initial Estimates of the Expected Outcome
5. Compute the Statistical Estimand of Interest
6. Calculate the Standard Errors for Confidence Intervals and P-values

TMLE: the algorithm steps

1. Estimate the Outcome
2. Estimate the Probability of Treatment
3. Estimate the Fluctuation Parameter
4. Update the Initial Estimates of the Expected Outcome
5. Compute the Statistical Estimand of Interest
6. Calculate the Standard Errors for Confidence Intervals and P-values

Step 1: Estimate the Outcome

Predict the outcome

$$Q(A, W) = E[Y|A, W]$$

- This could be done with a parametric regression, or any other algorithm of interest
- Recommend using SuperLearner

$Q(A, W)$

Step 1: Estimate the Outcome

Predict the counterfactual

If you could roll back the clock and **treat everyone** in your sample, what would their outcome be?

$$Q(1, W) \qquad = E[Y|A = 1, W]$$

```
<- predict(outcome_fit,newdata=  )
```

What if everyone was **not treated**?

$$Q(0, W) \qquad = E[Y|A = 0, W]$$

```
<- predict(outcome_fit,newdata=  )
```

Step 1: Estimate the Outcome

Dataset after Step 1

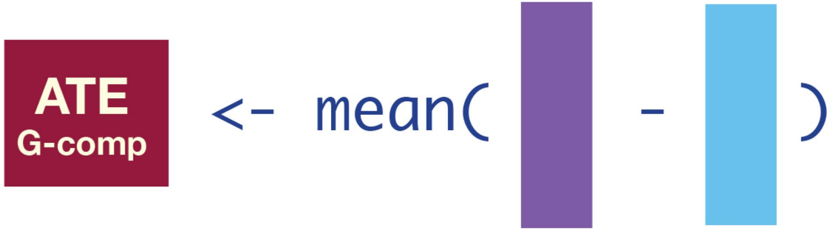
Y	A	Q_A	Q_0	Q_1
1	1	0.8461853	0.6770917	0.8461853
1	0	0.6986440	0.6986440	0.8589257
0	0	0.4932538	0.4932538	0.7188934
1	1	0.8213403	0.6363132	0.8213403
1	0	0.6266258	0.6266258	0.8151742
1	1	0.8578239	0.6966588	0.8578239

- Q_A= predicted probability of outcome under observed value of A
- Q_0= predicted probability of outcome under A=0
- Q_1= predicted probability of outcome under A=1

Step 1: Estimate the Outcome

We could estimate the ATE using G-computation at this point

- Can get 95% CI's via bootstrap
    - Computationally expensive with SuperLearner

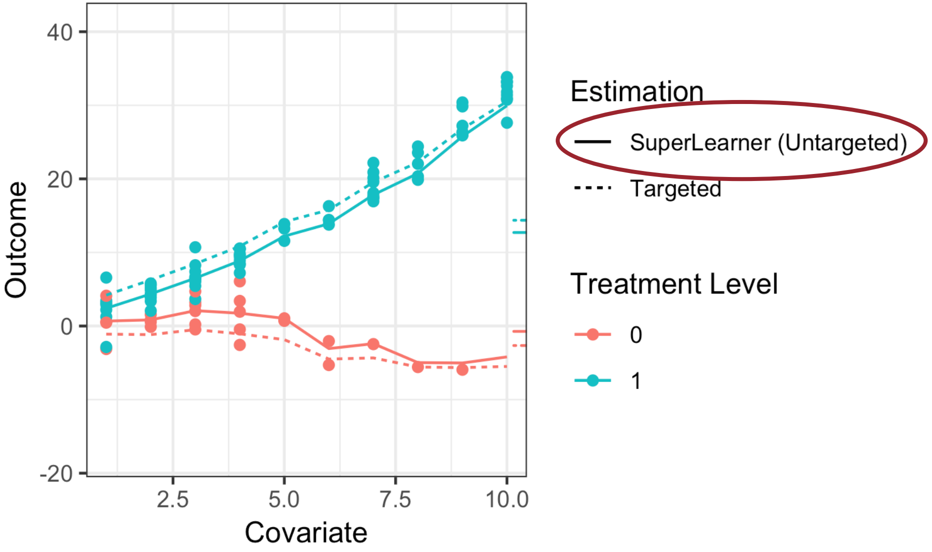


$$\mathrm{ATE} = \mathrm{E}[\mathrm{Y}_1] - \mathrm{E}[\mathrm{Y}_0] = \frac{\sum_{i=1}^{N}(E[Y|A=1,W] - E[Y|A=0,W])}{N}$$

So why not stop here?

- Difference in solid lines on right margin is the G-computation based ATE
- This does not have the more optimal bias/variance tradeoff for the ATE that TMLE
  - Optimized for the outcome prediction
- Does not have the double-robust properties of TMLE

Difference in solid lines on right margin is the G-computation based ATE



TMLE: the algorithm steps

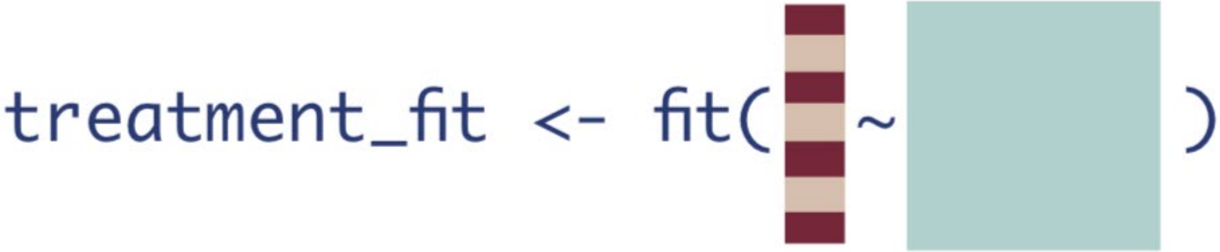
1. Estimate the Outcome
2. Estimate the Probability of Treatment
3. Estimate the Fluctuation Parameter
4. Update the Initial Estimates of the Expected Outcome
5. Compute the Statistical Estimand of Interest
6. Calculate the Standard Errors for Confidence Intervals and P-values

Step 2: Estimate the Probability of Treatment

Predict the probability of treatment (the **propensity score**)

$$g(A) = \Pr(A|W)$$

Estimated with new SuperLearner model



Step 2: Estimate the Probability of Treatment

Now we need to compute three different quantities from this model fit:

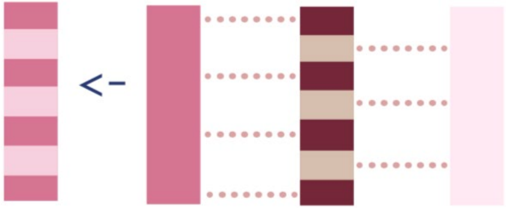
1. The inverse **probability of receiving treatment**

```
<- 1/predict(treatment_fit)
```

2. The negative inverse **probability of not receiving treatment**


```
<- -1/(1-predict(treatment_fit))
```

3. Combine these:
   - If the observation was **treated**, the inverse **probability of receiving treatment**
   - If they were **not treated**, the negative inverse **probability of not receiving treatment**
   - In the TMLE literature this is called the **clever covariate**



Step 2: Estimate the Probability of Treatment

Dataset after Step 2

Y	A	Q_A	Q_0	Q_1	H_1	H_0	H_A
1.00	1.00	0.85	0.68	0.85	2.18	−1.85	2.18
1.00	0.00	0.70	0.70	0.86	1.60	−2.67	−2.67
0.00	0.00	0.49	0.49	0.72	3.39	−1.42	−1.42
1.00	1.00	0.82	0.64	0.82	2.39	−1.72	2.39
1.00	0.00	0.63	0.63	0.82	2.31	−1.76	−1.76
1.00	1.00	0.86	0.70	0.86	2.14	−1.88	2.14

TMLE: the algorithm steps

1. Estimate the Outcome
2. Estimate the Probability of Treatment
3. Estimate the Fluctuation Parameter
4. Update the Initial Estimates of the Expected Outcome
5. Compute the Statistical Estimand of Interest
6. Calculate the Standard Errors for Confidence Intervals and P-values

Step 3: Estimate the Fluctuation Parameter

We will now use information about the treatment mechanism (from Step 2) to **optimize the bias-variance trade-off for the ATE** (rather than the outcome) so we can obtain more valid inference.

Warning: this step is easy to code, but difficult to understand.

- Requires knowledge of semi-parametric estimation theory.
- We will first work through **how** to do this
- Later we will seek to understand **why** this helps

Step 3: Estimate the Fluctuation Parameter

The point of this step is to solve an estimating equation for the efficient influence function (EIF) of our estimand of interest.

Skipping details of EIF or estimating equations, know that the help us:

- Update our initial outcome estimates so that our estimate of the ATE is asymptotically unbiased (under certain conditions)
- Calculate the variance (allowing for non-bootstrapped 95% CI’s)

Step 3: Estimate the Fluctuation Parameter

- The model that will help us solve an EIF estimating equation and then update our estimates:

$$\mathbf{logit}(\mathbf{E}[\mathbf{Y}|\mathbf{A},\mathbf{W}]) = \mathbf{logit}\left(\hat{\boldsymbol{E}}[\mathbf{Y}|\mathbf{A},\mathbf{W}]\right) + \in \mathbf{H}(\mathbf{A},\mathbf{W})$$

Our estimating equation looks *a lot* like a simple logistic regression:

logit(E[Y|X])=β0+β1X.

- With the clever covariate **H(A,W)** as an added offset (or fixed intercept) with coefficient **ϵ**

Step 3: Estimate the Fluctuation Parameter

- Save the coefficient ϵ from this model (called the fluctuation parameter) for the next step:

TMLE: the algorithm steps

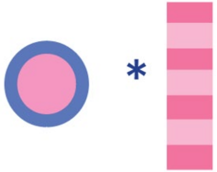
1. Estimate the Outcome
2. Estimate the Probability of Treatment
3. Estimate the Fluctuation Parameter
4. Update the Initial Estimates of the Expected Outcome
5. Compute the Statistical Estimand of Interest
6. Calculate the Standard Errors for Confidence Intervals and P-values

Step 4: Update the Initial Estimates of the Expected Outcome

Calculate how much we need to fluctuate our initial predictions of E[Y|A,W] to optimize the estimation of the ATE rather than the outcome

1. Put the initial outcome predictions on the logit scale because that was the scale we solved the EIF on (qlogis() in R).

2. Multiple the outputs of steps 2 and 3 (the **clever covariate** and **fluctuation parameter**)

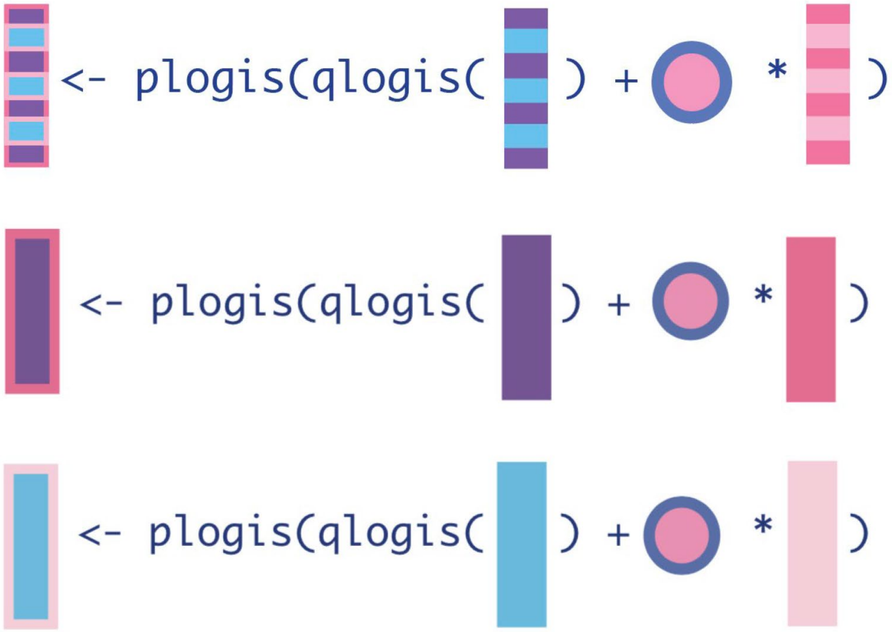


3. Sum these 2 quantiles and then transform back to the true outcome scale with the expit function (plogis() in R):

Step 4: Update the Initial Estimates of the Expected Outcome

Update all 3 predictions

1. Update the expected outcomes of all observations, given the treatment they actually received and their baseline confounders.
2. Update the expected outcomes, conditional on baseline confounders and everyone receiving the treatment.
3. Update the expected outcomes, conditional on baseline confounders and no one receiving the treatment.

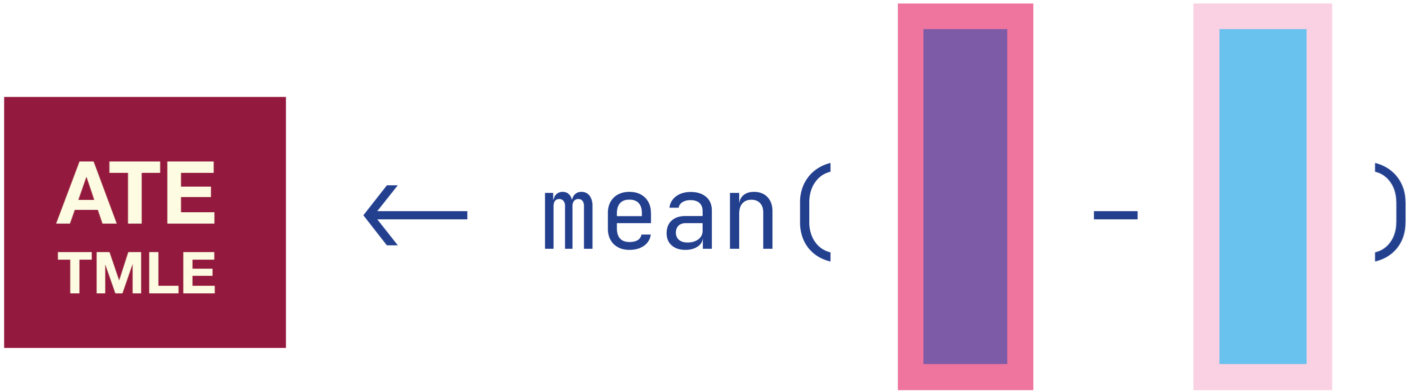


TMLE: the algorithm steps

1. Estimate the Outcome
2. Estimate the Probability of Treatment
3. Estimate the Fluctuation Parameter
4. Update the Initial Estimates of the Expected Outcome
5. Compute the Statistical Estimand of Interest
6. Calculate the Standard Errors for Confidence Intervals and P-values

Step 5: Compute the Statistical Estimand of Interest

Calculate the more optimized ATE



TMLE: the algorithm steps

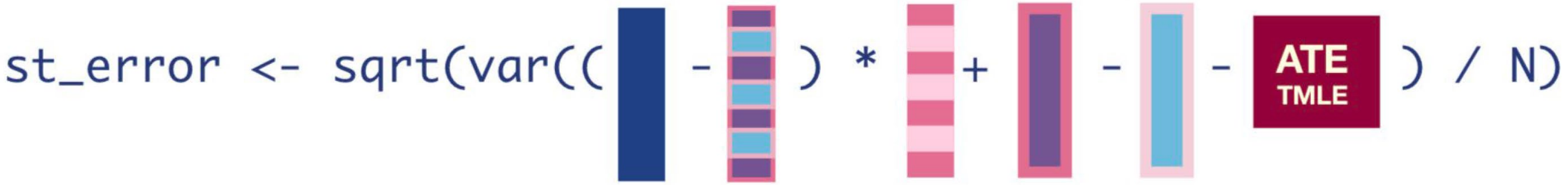
1. Estimate the Outcome
2. Estimate the Probability of Treatment
3. Estimate the Fluctuation Parameter
4. Update the Initial Estimates of the Expected Outcome
5. Compute the Statistical Estimand of Interest
6. Calculate the Standard Errors for Confidence Intervals and P-values

Step 6: Calculate the Standard Errors for Confidence Intervals and P-values

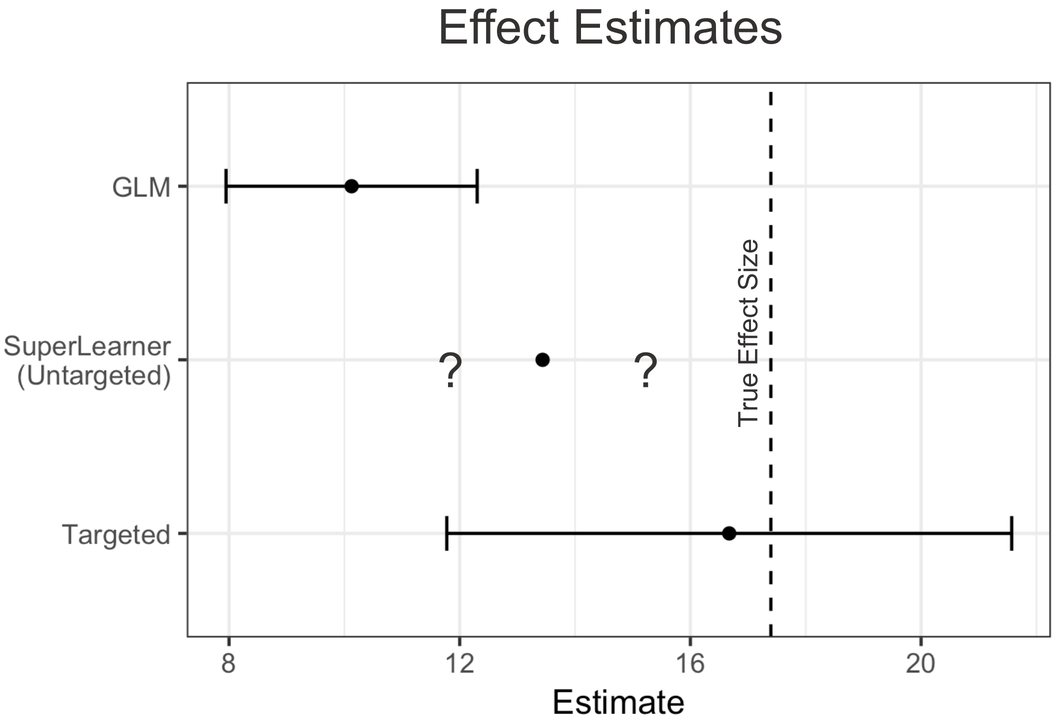
Compute the Influence Function (IF)

- This is the empirical version of the EIF we used our estimating equation to figure out in Step 3
- The IF shows how much each observation influences the final estimate
- The 95% confidence intervals can be then calculated as normal from the SE
  - These are called influence curve based confidence intervals

$$\hat{IF} = (Y - \hat{E}^*[Y|A,\mathbf{W}])H(A,\mathbf{W}) + \hat{E}^*[Y|A=1,\mathbf{W}] - \hat{E}^*[Y|A=0,\mathbf{W}] - \hat{ATE}$$



Results: removing bias AND robust inference

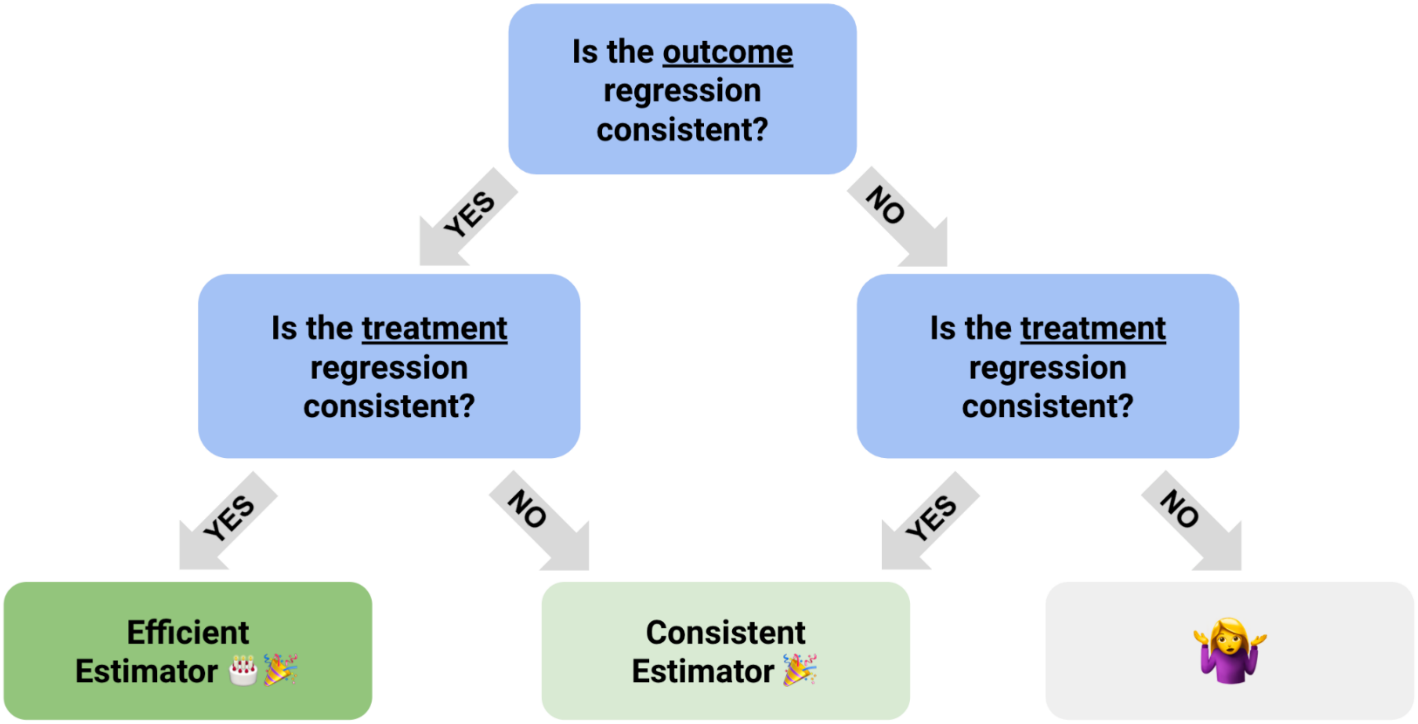


GLM did not learn the correct outcome mechanism, so its estimate is very biased

SuperLearner does a better job of estimating the outcome mechanism, but does not allow valid inference

TMLE combines good outcome mechanism estimation with targeting to get valid inference

Determining your Determining your Doubly Robust Estimator’s Properties

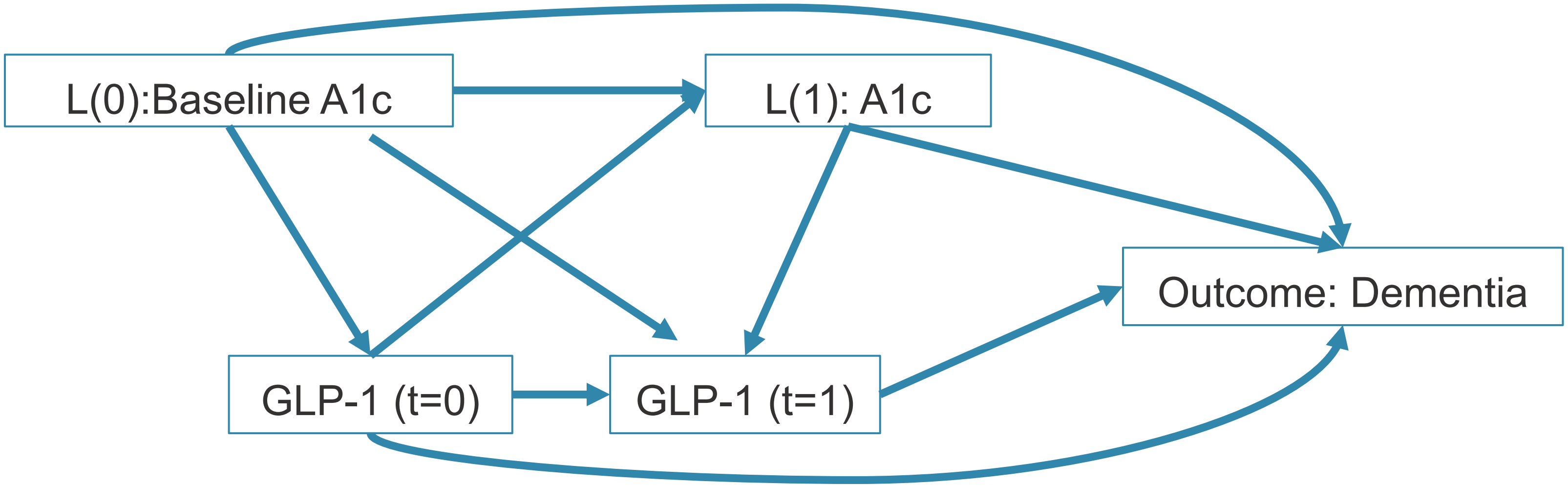


khstats.com/blog/tmle/tutorial

Consistent: bias decreases to zero as sample size grows large
Efficient: bias decreases to zero *and* variance decreases to smallest possible value for a fixed number of observations

Longitudinal TMLE

- Coming Thursday!
- Allows for untangling time-dependent confounding through iterated Q and g models, and retains all the TMLE properties



Resources

- Some lecture material adapted from
  - Introduction to modern causal inference online textbook (https://alejandroschuler.github.io/mci/introduction-to-modern-causal-inference.html)
  - Tlverse handbook (https://tlverse.org/tlverse-handbook/)
  - Illustrated guide to TMLE (https://www.khstats.com/blog/tmle/tutorial-pt1)
  - Slides shared by Maya Petersen and Ben Arnold
- Other good learning references
  - https://achambaz.github.io/tlride/
  - https://multithreaded.stitchfix.com/blog/2021/07/23/double-robust-estimator/

References

H. Bang and J.M. Robins. Doubly-robust estimation in missing data and causal inference models. Biometrics, 61:962 – 972, 2005.

L. Breiman. Bagging predictors. Machine Learning, 24:123 – 140, 1996.

M A Hern´an, E Lanoy, D Costagliola, and J M Robins. Comparison of dynamic treatment regimes via inverse probability weighting. Basic & Clinical Pharmacology & Toxicology, 98:237 – 242, 2006.

Judea Pearl. Causal diagrams for empirical research. Biometrika, 82(4): 669–688, 1995.

Judea Pearl and James M Robins. Probabilistic evaluation of sequential plans from causal models with hidden variables. In UAI, volume 95, pages 444–453. Citeseer, 1995.

Maya L Petersen and Mark J van der Laan. Causal models and learning from data: integrating causal modeling and statistical estimation.

Epidemiology (Cambridge, Mass.), 25(3):418, 2014.

J. Robins and A. Rotnitzky. Recovery of information and adjustment for dependent censoring using surrogate markers. In AIDS Epidemiology, pages 297–331. Springer, 1992.

References

James Robins. The control of confounding by intermediate variables.

Statistics in medicine, 8(6):679–701, 1989.

J.M. Robins. A new approach to causal inference in mortality studies with sustained exposure periods – application to control of the healthy worker survivor effect. Mathematical Modelling, 7:1393 – 1512, 1986.

J.M. Robins. Robust estimation in sequentially ignorable missing data and causal inference models. In Proceedings of the American Statistical Association on Bayesian Statistical Science, 1999, pages 6–10, 2000.

J. Schwab, S. Lendle, M. Petersen, and M. van der Laan. ltmle: Longitudinal Targeted Maximum Likelihood Estimation, 2013. R package version 0.9.3, http://cran.r-project.org/web/packages/ltmle/.

L Tran, Constantin T Yiannoutsos, B Musick, K Wools-Kaloustian, A Siika, S Kimaiyo, M van der Laan, and M Petersen. Evaluating the impact of a hiv low-risk express care task-shifting program: a case study of the targeted learning roadmap. Epidemiologic Methods, 5(1):69–91, 2016.

References

M.J. van der Laan and S. Gruber. Targeted minimum loss based estimation of causal effects of multiple time point interventions. The International Journal of Biostatistics, 8(1):Article 8, 2012.

MJ van der Laan and S Rose. Targeted learning: causal inference for observational and experimental data. Springer Science & Business Media, 2011.

M.J. van der Laan, E. Polley, and A. Hubbard. Super learner. Statistical Applications in Genetics and Molecular Biology, 6(25), 2007.