# Targeted Learning for Data Adaptive Causal Inference in Observational and Randomized Studies

Mark van der Laan[1] and Susan Gruber[2]

[1] Department of Biostatistics, University of California, Berkeley School of Public Health

[2] Department of Population Medicine, Harvard Medical School and Harvard Pilgrim Health Care Institute

# Course Outline

- Part 1
  - Targeted Learning Overview
  - Estimation Roadmap
  - Super Learning
- Part 2
  - Targeted Minimium Loss-Based Estimation (TMLE)
- Part 3
  - TMLE for longitudinal data analysis
  - Concluding Remarks

# Learning from Data

- Requirements for learning from data
  - A clear question
  - Knowledge about the data generating experiment
  - A straightforward, relevant, interpretable result
- Core concepts in Targeted Learning
  - A (statistical) model represents (statistical) knowledge about the data generating experiment
  - Target parameter defined as a feature of the data generating distribution
  - Efficient, data adaptive estimation + statistical inference
    - Super Learning
    - Targeted minimum loss-based estimation (TMLE)

# Traditional Approach to Analyzing Health Care Data

1. Fit several parametric logistic regression models and choose one
2. Report point estimate of coefficient in front of treatment, $p$-value and confidence interval as if this parametric model was pre-specified

- But consider,
  - The parametric model is misspecified
  - The coefficient is interpreted as if the parametric model is correct
  - The model selection procedure is not accounted for in the estimated variance

# Targeted Learning

- Targeted Learning provides a paradigm for transforming data into reliable, actionable knowledge
- Define targeted parameter to address a relevant scientific question, not for convenience
- Avoid reliance on human art and unrealistic parametric models: a priori specified estimator.
- Target the fit of data-generating distribution to the target parameter of interest
- Valid statistical inference in terms of a normal limiting distribution

# Examples of Targeted Learning Toolbox

- Prediction and classification
- Targeted effect estimation
  - Effects of static or dynamic treatments
  - Direct and indirect effects (mediation analysis)
  - Parameters of marginal structural models
  - Variable importance measures
- Types of data
  - Point treatment
  - Longitudinal/Repeated Measures
  - Censoring/Missingness/Time-dependent confounding
  - Case-Control
  - Randomized clinical trials and observational data

# Estimation Roadmap

Step 1. Define a statistical model, $\mathcal{M}$, that contains the true probability distribution of the data, $P_0$.

Step 2. Define the target parameter of interest, $\psi_0^{full}$, as a feature of a full data distribution, $P_0^{full}$.

Step 3. Specify a mapping from the full data to observed data, and $\Psi : M \to \mathbb{R}^d$ such that under explicitly stated identifying assumptions $\psi_0^{full} = \Psi(P_0)$.

Step 4. Estimation and inference of statistical parameter $\psi_0 = \Psi(P_0)$ using super learning and targeted minimum loss based estimation.

Step 5. Provide a considered interpretation of the result.

# Super Learning - Motivation

Both **effect** and **prediction** research questions are inherently *estimation* questions, but they are distinct in their goals.

**Effect:** Interested in estimating the effect of exposure on outcome adjusted for covariates.

**Prediction:** Interested in generating a function to input covariates and predict a value for the outcome.

*Effect parameters where no causal assumptions are made may be referred to as variable importance measures (VIMs).*

# Traditional Approach

Estimation using (misspecified) Parametric Models

- Data $n$ i.i.d. copies of $O = (Y, A, W)$
  - Outcome $Y$, Treatment $A$, Covariates $W$
- Standard practice for prediction and effect estimation
  - assume a parametric statistical model for $E_0(Y \mid A, W)$, the conditional mean of $Y$ given $A$ and $W$
  - use maximum likelihood estimation (MLE) to estimate model parameters
- Parametric regression models
  - varying levels of complexity
  - choice of variables included in model impacts complexity

# High Dimensional Data

- Potentially thousands of candidate variables to include in the model
- Model complexity can increase exponentially, more unknown parameters than observations
- The true functional for $E_0(Y \mid A, W)$ might be complex, beyond main terms and interaction terms.
- Correct specification is a challenge

# The Complications of Human Art in Traditional Practice

- The moment we use **post-hoc arbitrary criteria** and **human judgment** to select the parametric statistical model after looking at the data, the analysis becomes prone to additional bias.
- Bias manifests in both the effect estimate and the assessment of uncertainty (i.e., standard errors).
- So why not simply use a purely non-parametric model with high dimensional data?
  - $p > n$!
  - data sparsity

# Super Learning - Motivation

- What we want is an automated algorithm to semi-parametrically estimate $E_0(Y \mid A, W)$.
  - Opportunity to reduce bias due to model misspecification
  - Opportunity to reduce variance by improving the fit for the dependent variable
- Many potential algorithms.
  - We cannot bet on a misspecified parametric regression,
  - Many semi-parametric methods that aim to “smooth" the data and estimate this regression function.
  - Yet one particular algorithm is going to do better than the other candidate estimators.
- How to know which one to use?

# The Dangers of Favoritism

- Relative Mean Squared Error (compared to main terms least squares regression) based on the validation sample

| Method | Study 1 | Study 2 | Study 3 | Study 4 |
|---|---|---|---|---|
| Least Squares | 1.00 | 1.00 | 1.00 | 1.00 |
| LARS | 0.91 | 0.95 | 1.00 | 0.91 |
| D/S/A | 0.22 | 0.95 | 1.04 | 0.43 |
| Ridge | 0.96 | 0.9 | 1.02 | 0.98 |
| Random Forest | 0.39 | 0.72 | 1.18 | 0.71 |
| MARS | 0.02 | 0.82 | 0.17 | 0.61 |

# Super Learning in Prediction

| Method | Study 1 | Study 2 | Study 3 | Study 4 | Overall |
|---|---|---|---|---|---|
| Least Squares | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| LARS | 0.91 | 0.95 | 1.00 | 0.91 | 0.95 |
| D/S/A | 0.22 | 0.95 | 1.04 | 0.43 | 0.71 |
| Ridge | 0.96 | 0.9 | 1.02 | 0.98 | 1.00 |
| Random Forest | 0.39 | 0.72 | 1.18 | 0.71 | 0.91 |
| MARS | 0.02 | 0.82 | 0.17 | 0.61 | 0.38 |
| Super Learner | 0.02 | 0.67 | 0.16 | 0.22 | 0.19 |

# Super Learning - Core Concepts

- Loss Based Estimation
  - We will use **loss functions** to define the best estimator of $E_0(Y \mid A, W)$ from a library of algorithms, and then evaluate it.
- Cross Validation
  - Our available data is partitioned to **train** and **validate** our estimators.
- Semi-Parametric Estimation
  - Allow the **data** to drive your **estimates**, but in an honest (cross validated) way.

# Loss-Based Estimation

- Data structure $O = (W, A, Y) \sim P_0$
  - empirical distribution $P_n$ places probability $1/n$ on each observed $O_i$, $i = 1, \ldots, n$.
- Goal is to estimate conditional mean outcome, $Q_0 = E_0(Y \mid A, W)$
- Specify a library of learning algorithms
- "Best" algorithm is with respect to a loss function, $L$.

$$L : (O, Q) \rightarrow L(O, Q) \in \mathbb{R}$$

  - $L$ assigns a measure of performance to a candidate function $Q$ when applied to an observation $O$.
  - $L$ is a function of the random variable $O$ and parameter value $Q$.

# Loss-Based Estimation

Examples of loss functions

- $L_1$ absolute error loss function:

$$L(O, Q) = |Y - Q(A, W)|,$$

- $L_2$ squared error (or quadratic) loss function:

$$L(O, Q) = (Y - Q(A, W))^2,$$

- Negative log loss function:

$$L(O, Q) = -\log(Q(A, W)^Y (1 - Q(A, W))^{1-Y}).$$

# Loss-Based Estimation

- Squared error loss: $L(O, \bar{Q}) = (Y - \bar{Q}(A, W))^2$
- Expected squared error loss $E_0 L(O, \bar{Q})$ is also known as *risk*
- Risk evaluates candidate $\bar{Q}$
  - Small risk is better
  - Risk is minimized at the optimal choice of $\bar{Q}_0$
- Define our parameter of interest, $\bar{Q}_0 = E_0(Y \mid A, W)$, as the minimizer of the risk:

$$\bar{Q}_0 = \arg\min_{\bar{Q}} E_0 L(O, \bar{Q}).$$

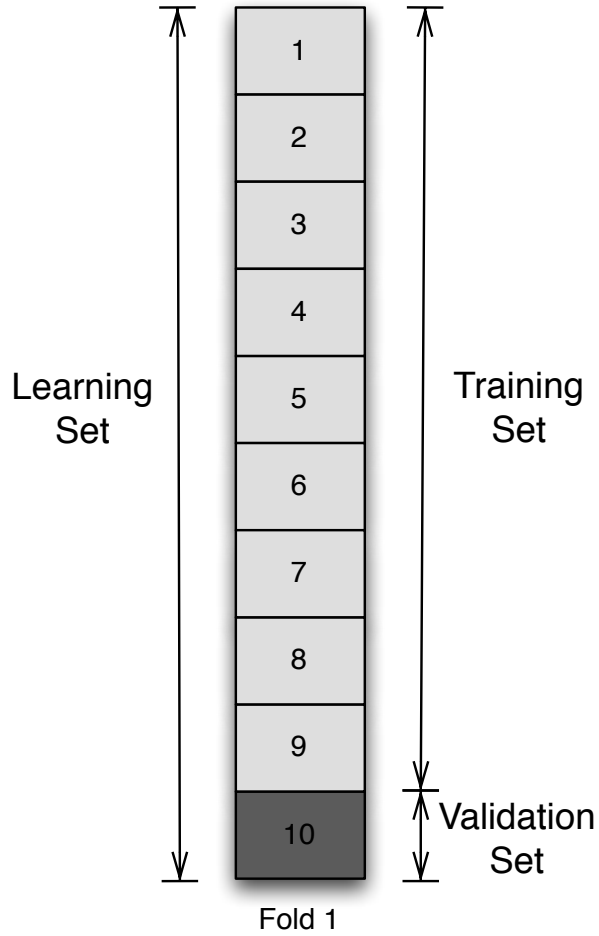
# Cross-Validation

**Cross-validation** to obtain an accurate estimate of risk

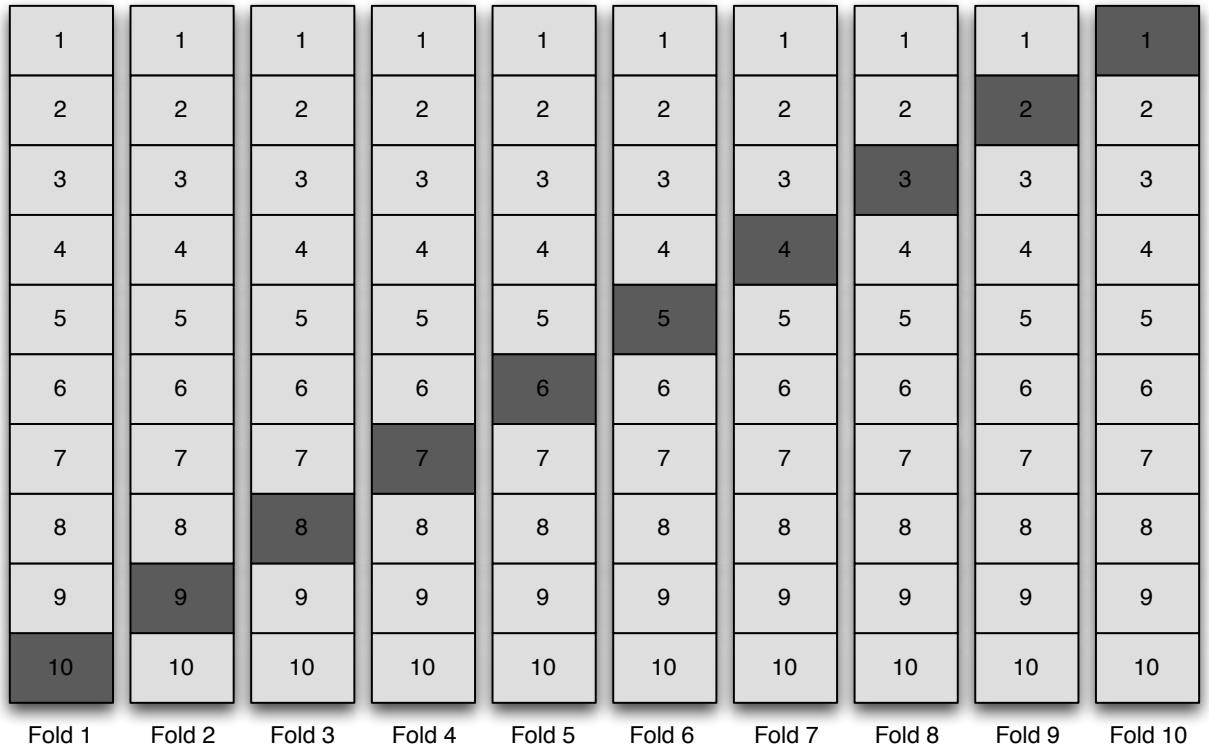
- Partitions the sample of $n$ observations $O_1, \ldots, O_n$ into training and corresponding validation sets.
- Produces an accurate estimate of risk
- Discrete super learner: selects "best" algorithm with smallest risk among a library of algorithms
- We can also use cross-validation to evaluate the overall performance of the super learner itself.

# V-fold Cross-Validation

- Observed data $O_1, \ldots, O_n$ is referred to as the learning set.
- Learning set is partitioned into $V$ sets of size $\approx \frac{n}{V}$.
- For each fold, $V - 1$ sets will comprise the *training set*. The remaining set is the *validation set*.
- Observations in the *training set* are used to construct (or train) the candidate estimators.
- Observations in the *validation set* are used to evaluate risk



# V-fold Cross-Validation




The validation set rotates $V$ times such that each set is used as the validation set once.

# Discrete Super Learner

- Suppose a researcher cannot decide between three different statistical methodologies for estimating $E_0(Y \mid A, W)$
- SL library consists of (MLE, Deletion/Substitution/Addition (DSA), Random Forest)
- Discrete SL chooses the one with the smallest (honest) cross-validated risk.

| Method | CV-Risk |
|---|---|
| MLE | 0.30 |
| DSA | 0.04 |
| Random Forest | 0.23 |

Which algorithm does the discrete super learner pick?

# Oracle Properties

- The Oracle selector is the best estimator among the $K$ algorithms in the SL library
  - Chooses the algorithm whose fit on the training samples yields the smallest risk under $P_0$, the true probability distribution of random variable $O$.
  - Unknown, since it depends on both observed data and $P_0$.
- Discrete super learner performs as well as the Oracle selector, up to a second order term.
  - assuming a bounded loss function
  - number of algorithms in the library polynomial in sample size
- That is, ratio of loss-based dissimilarities for oracle selected estimator and cross-validated selected estimator w.r.t. truth converges to 1!

# Ensemble Super Learner

- Ensemble super learner improves upon discrete super learning by enlarging set of candidate estimators.
  - Define the SL library as all weighted averages of individual algorithms
    - Each weighted average is a unique candidate algorithm in this augmented library.
    - One of these weighted combinations might perform better than any single algorithm
    - Each individual algorithm remains a candidate
  - Cross-validation guides the selection of the optimal weighted combination
  - Ensemble SL is no more computer intensive than discrete SL

# Ensemble Super Learner: How it works

Once the discrete super learner has been completed,

- Propose a family of weighted combinations of library algorithms, indexed by weight vector $\alpha$.
  - consider only $\alpha$-vectors that sum to one, where each weight is non-negative
- Determine which combination minimizes the cross-validated risk

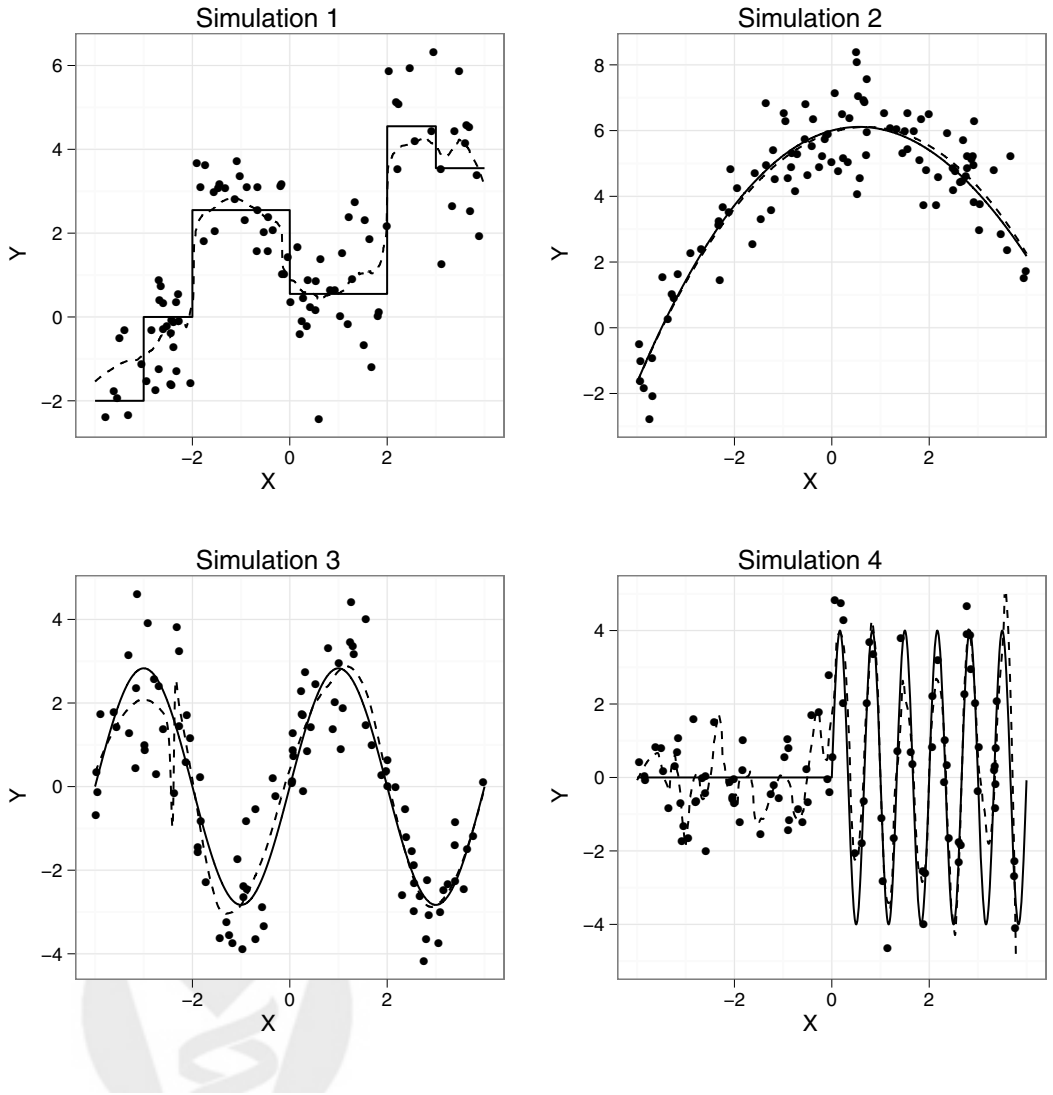
  $$P_n(Y = 1 \mid Z) = \text{expit}\left(\alpha_{1,n}Z_1 + \alpha_{2,n}Z_2 + \ldots + \alpha_{K,n}Z_K\right)$$

  - Cross-validated predictions ($Z$) for each algorithm are inputs in a working (statistical) model to predict the outcome $Y$.
- SL prediction is a weighted combination of predictions from algorithms fit on the entire dataset. Given $n \times k$ prediction matrix $Z'$,

$$\bar{Q}_n(A, W) = \mathbf{Z}'\alpha_n$$

# SL: Finite sample performance

Four simulated datasets ($n = 100$)




True functional form

Data points

SL predictions

Fig. 2, Polley and van der Laan, 2010

# SL: ICU Mortality Prediction

Cross-validated Area under the Receiver-Operating Curve

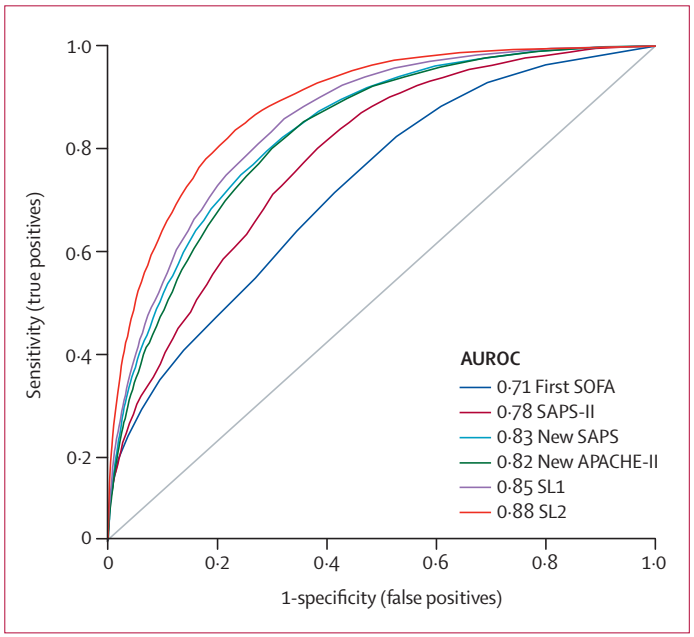



***Figure 1:*** **Receiver-operating characteristics curves**

Pirracchio, et al, *Lancet*, 2014

- Sepsis-related Organ Failure Assessment (SOFA)
- Simplified Acute Physiology Score (SAPS-II)
- Acute Physiology and Chronic Health Evalution (APACHE)
- Super Learner, standard categorized variables (SL1)
- Super Learner, non-transformed variables (SL2)

- SL better distinguishes between high and low risk patients

# The Bottom Line

- There is no point in painstakingly trying to decide what estimators to enter in the collection; **instead add them all.**
- The theory supports this approach and finite sample simulations and data analyses only confirm that **it is very hard to over-fit the super learner by augmenting the collection**, but benefits are obtained.
- Indeed, for large data sets, we simply do not have enough algorithms available to build the desired collection that would fully utilize the power of the super learning.

# Super Learning Demonstration

- *SuperLearner* R package (CRAN and GitHub)
- Using the package
- Practical considerations
  - Algorithms for the SL library
  - Loss function
  - Dimension Reduction
  - How to choose $V$