線形モデル: OLS と Stacking への応用

経済学のための機械学習入門

川田恵介

Table of contents

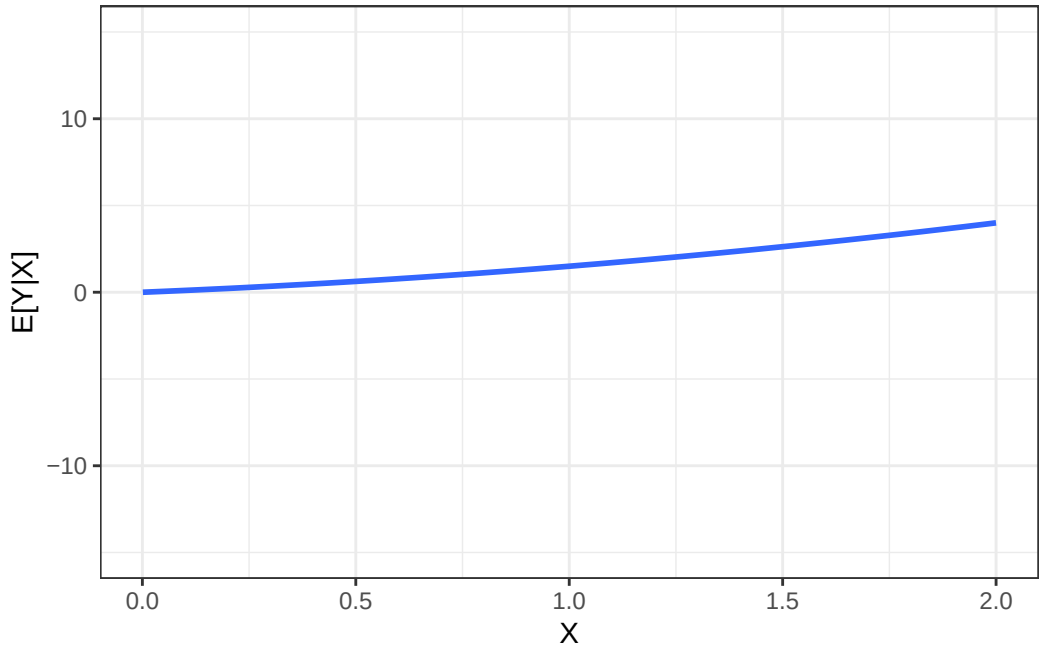
Linear prediction model 2
- 例 . . . 2
- Tree . . . 3
- RandomForest . . . 3
- OLS . . . 4
- Stacking (後述) . . . 4
- サンプル分割による評価 . . . 5
- Empirical Risk Minimization による推定 . . . 5
- Basic function . . . 5
- モデル設定 . . . 5
- 例 . . . 6
- 復習: 過剰適合した決定木 . . . 6

Stacking 7
- 動機 . . . 7
- アイディア . . . 7
- Stacking with linear model . . . 7
- Stacking . . . 8
- 数値例 . . . 8
- 数値例: OLS と決定木 . . . 9
- 数値例: OLS と決定木 . . . 9
- 数値例: OLS と決定木 . . . 10
- 数値例: OLS と決定木 . . . 10
- 他の例: SuperLearner . . . 11
- まとめ . . . 11
- Reference . . . 11

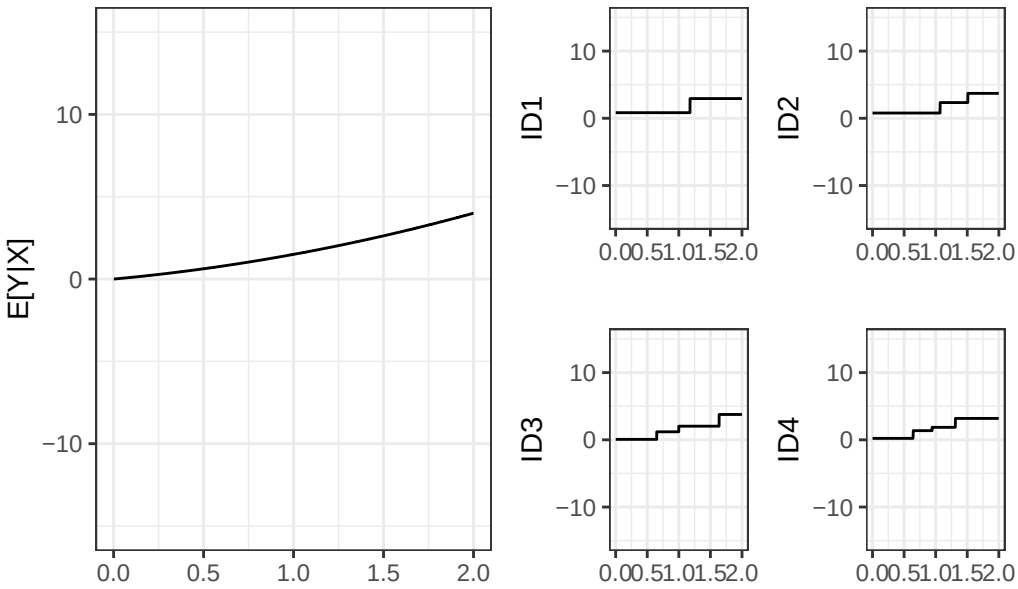
Linear prediction model

- $g(X_1, .., X_L) = \beta_0 + \beta_1 X_1 + .. + \beta_L X_L$
  - “Smooth” な母平均関数に対する、有力な手法
  - 大量の推定方法: **OLS**, Maximum liklehood, Bays, Penalized Regression

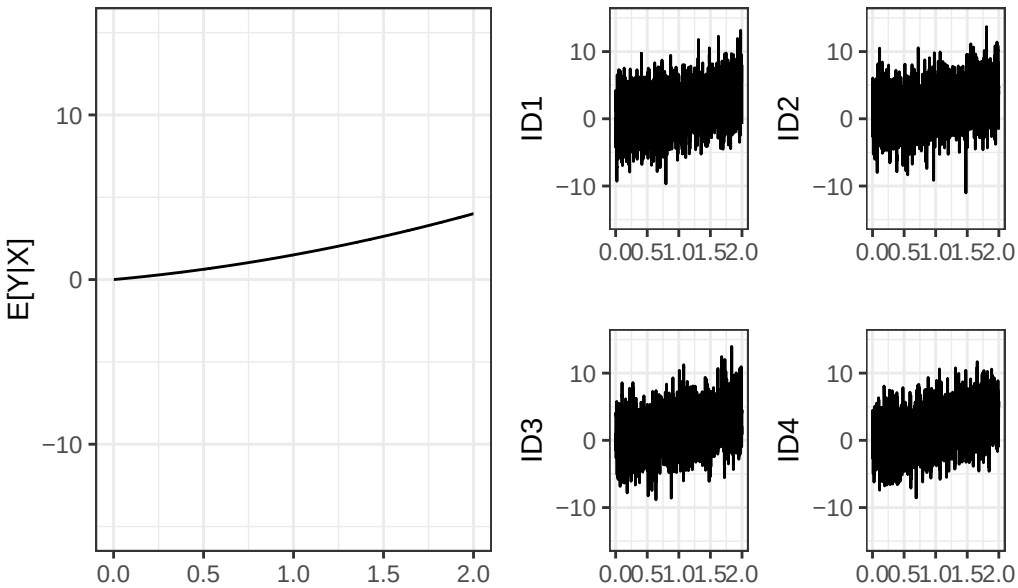
例



Tree

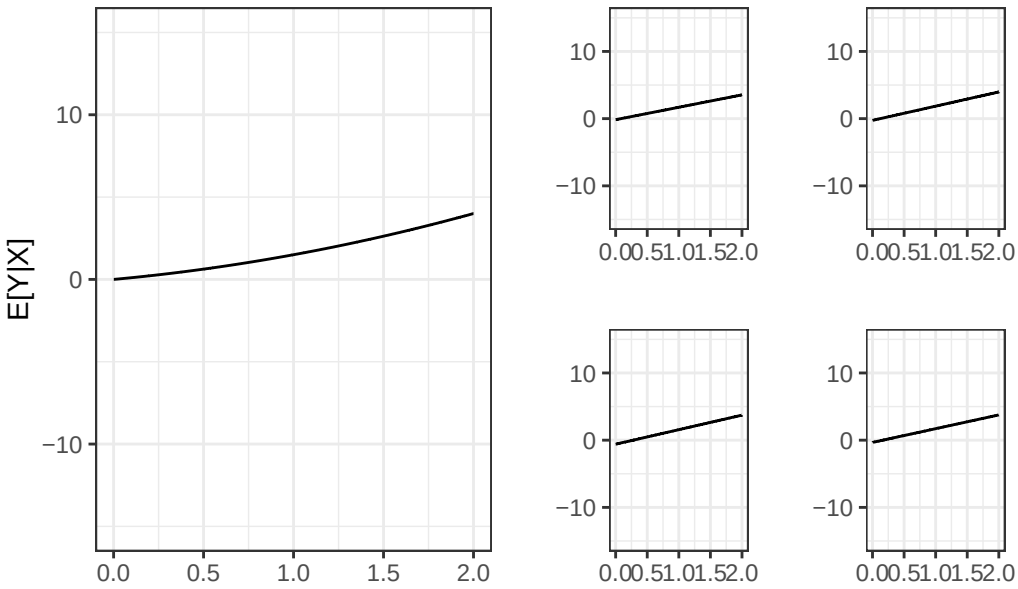


RandomForest

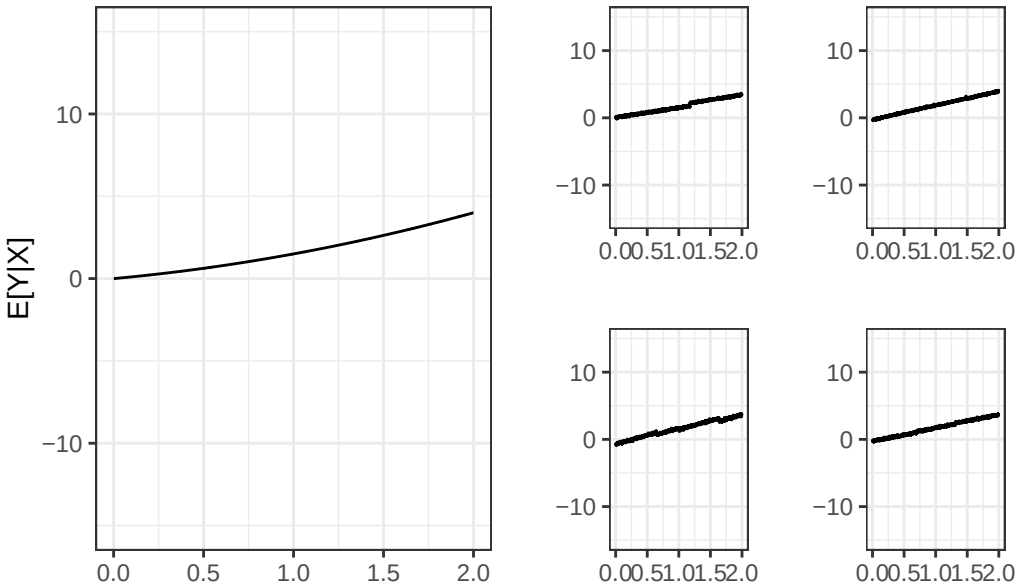


- 参考: Smooth な母平均への対応 (Friedberg et al. 2020) : grf

OLS



Stacking (後述)



サンプル分割による評価

```
   nr                  task_id  learner_id resampling_id iteration    regr.rsq
1:  1 Linear Population (5000)     regr.lm       holdout         1  0.04573460
2:  2 Linear Population (5000) OptimalTree       holdout         1  0.03353547
3:  3 Linear Population (5000) regr.ranger       holdout         1 -0.24427837
4:  4 Linear Population (5000)       Stack       holdout         1  0.04727011
Hidden columns: uhash, task, learner, resampling, prediction
```

Empirical Risk Minimization による推定

1. 研究者が事前にモデルを指定

$$g(X_i) = \beta_0 + .. + \beta_L X_L$$

2. “OLS” 推定

$$\min_{\beta_0,..,\beta_L} E[(Y_i - g(X_i))^2]$$

Basic function

- Linear model は” 一直線” とは限らない
  - 非常に自由度が高いフレームワーク
- Linear model with Basic function

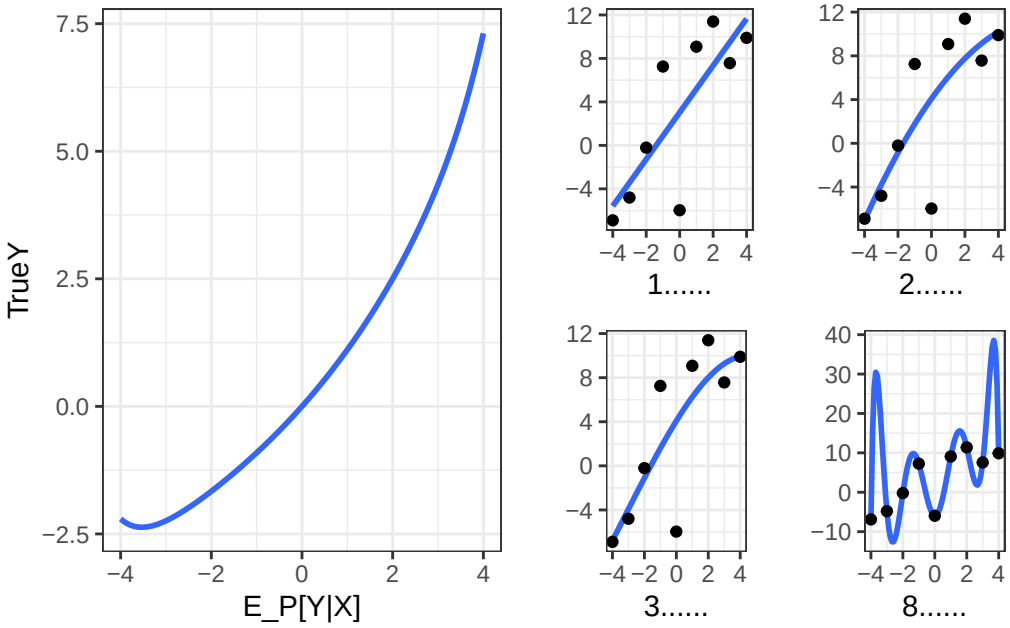
$$g(Y_i) = \beta_0 + .. + \beta_L b_L(X_i)$$

- b : 研究者が指定する既知の関数
  - 例: $b_1(X_i) = X_1^2, b_2(X_i) = X_2^2, b_2(X_i) = X_1 \times X_2$

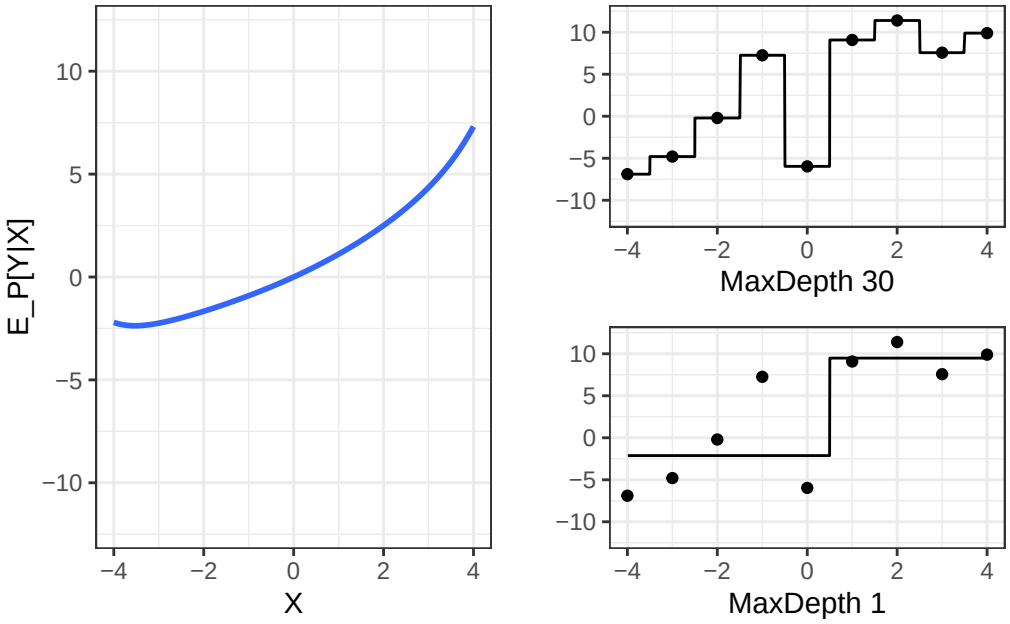
モデル設定

- 事例数に比べて、十分に単純 (推定するパラメタが少ない) なモデルを指定できれば、OLS 推定可能
  - モデルを複雑にしすぎると、過剰適合する
- 極めて難しい課題
  - 実践として、連続変数についての二乗項の優先順位は高い

例



復習: 過剰適合した決定木



Stacking

- 大きく異なる予測モデル群を、最適化された加重を用いて集計する
  - Bagging: 決定木を単純集計
- 応用むけに推奨される (Naimi, Mishler, and Kennedy 2021; Díaz 2019)
  - Einav et al. (2018) でも活用

動機

- 大量のアルゴリズムが提案されている
- 最善のアルゴリズム = 母集団の性質に依存
  - 社会分析においては、BlackBox
- 解決策: 交差検証で性能を比較し、最善のアルゴリズムを選択
  - Stacking = 一般化
  - かなり現実的な選択肢

アイディア

- 最終予測モデル:

$$f(X) = \beta_{OLS} \times \underbrace{f_{OLS}(X)}_{OLSの予測} + \beta_{RF} \times \underbrace{f_{RF}(X)}_{RFの予測} + ...$$

- β_a 各予測への重み付け
  - “交差推定で最善のアルゴリズムを探す” のであれば、$\beta_a = \{0, 1\}$
  - $\{0, 1\}$ に限定する理由はない

Stacking with linear model

$$g(X) = \beta_0 + \beta_1 g_1(X) + .. + \beta_A g_A(X)$$

- $g_a(X) :=$ Algorithm a (例: OLS, RandomForest) によって生成される予測モデル

Stacking

- 全訓練データを用いて、$g_a(X)$ などを推定
- β_a を推定
  1. 交差推定を用いて、$\bar{g}_a$ を推定
  2. 以下を解く

$$\min_{\beta_a} E[(Y_i - \beta_0 - .. - \beta_A \times \bar{g}_A)^2]$$

数値例

```
# A tibble: 20 x 3
   Group     X     Y
   <dbl> <dbl> <dbl>
 1     1     1  2.24
 2     1     2  5.67
 3     1     3 11.7
 4     1     4 17.7
 5     1     5 29.8
 6     1     1  2.14
 7     1     2  4.5
 8     1     3 11.0
 9     1     4 19.0
10     1     5 29.5
11     2     1  1.83
12     2     2  5.59
13     2     3 13.8
14     2     4 20.4
15     2     5 30.8
16     2     1  0.43
17     2     2  5.91
18     2     3 11.6
19     2     4 18.8
20     2     5 30.4
```

数値例: OLS と決定木

```
# A tibble: 20 x 5
   Group     X     Y FitOLS FitTree
   <dbl> <dbl> <dbl>  <dbl>   <dbl>
 1     1     1  2.24 -0.318    2.19
 2     1     2  5.67  6.51     5.08
 3     1     3 11.7  13.3     11.3
 4     1     4 17.7  20.2     18.4
 5     1     5 29.8  27.0     29.7
 6     1     1  2.14 -0.318    2.19
 7     1     2  4.5   6.51     5.08
 8     1     3 11.0  13.3     11.3
 9     1     4 19.0  20.2     18.4
10     1     5 29.5  27.0     29.7
11     2     1  1.83 -0.589    3.44
12     2     2  5.59  6.69     3.44
13     2     3 13.8  14.0     12.7
14     2     4 20.4  21.2     19.6
15     2     5 30.8  28.5     30.6
16     2     1  0.43 -0.589    3.44
17     2     2  5.91  6.69     3.44
18     2     3 11.6  14.0     12.7
19     2     4 18.8  21.2     19.6
20     2     5 30.4  28.5     30.6
```

数値例: OLS と決定木

```
# A tibble: 20 x 7
   Group     X     Y FitOLS FitTree PredOLS PredTree
   <dbl> <dbl> <dbl>  <dbl>   <dbl>   <dbl>    <dbl>
 1     1     1  2.24 -0.318    2.19  -0.589     3.44
 2     1     2  5.67  6.51     5.08   6.69      3.44
 3     1     3 11.7  13.3     11.3   14.0      12.7
 4     1     4 17.7  20.2     18.4   21.2      19.6
 5     1     5 29.8  27.0     29.7   28.5      30.6
 6     1     1  2.14 -0.318    2.19  -0.589     3.44
 7     1     2  4.5   6.51     5.08   6.69      3.44
```

```
 8     1     3 11.0  13.3     11.3   14.0      12.7
 9     1     4 19.0  20.2     18.4   21.2      19.6
10     1     5 29.5  27.0     29.7   28.5      30.6
11     2     1  1.83 -0.589    3.44  -0.318     2.19
12     2     2  5.59  6.69     3.44   6.51      5.08
13     2     3 13.8  14.0     12.7   13.3      11.3
14     2     4 20.4  21.2     19.6   20.2      18.4
15     2     5 30.8  28.5     30.6   27.0      29.7
16     2     1  0.43 -0.589    3.44  -0.318     2.19
17     2     2  5.91  6.69     3.44   6.51      5.08
18     2     3 11.6  14.0     12.7   13.3      11.3
19     2     4 18.8  21.2     19.6   20.2      18.4
20     2     5 30.4  28.5     30.6   27.0      29.7
```

数値例: OLS と決定木

```
lm(Y ~ PredOLS + PredTree,
   ExampleData)                                                    ①
```

① OLS による最適加重の計算

```
Call:
lm(formula = Y ~ PredOLS + PredTree, data = ExampleData)

Coefficients:
(Intercept)      PredOLS     PredTree
    -0.1176       0.2598       0.7488
```

数値例: OLS と決定木

```
# A tibble: 20 x 6
   Group     X     Y FitOLS FitTree Stacking
   <dbl> <dbl> <dbl>  <dbl>   <dbl>    <dbl>
 1     1     1  2.24 -0.454    1.66     1.26
 2     1     2  5.67  6.60     5.42     5.34
 3     1     3 11.7  13.6     12.0     12.4
 4     1     4 17.7  20.7     19.0     19.5
 5     1     5 29.8  27.7     30.1     29.6
 6     1     1  2.14 -0.454    1.66     1.26
```

7	1	2	4.5	6.60	5.42	5.34
8	1	3	11.0	13.6	12.0	12.4
9	1	4	19.0	20.7	19.0	19.5
10	1	5	29.5	27.7	30.1	29.6
11	2	1	1.83	-0.454	1.66	1.26
12	2	2	5.59	6.60	5.42	5.34
13	2	3	13.8	13.6	12.0	12.4
14	2	4	20.4	20.7	19.0	19.5
15	2	5	30.8	27.7	30.1	29.6
16	2	1	0.43	-0.454	1.66	1.26
17	2	2	5.91	6.60	5.42	5.34
18	2	3	11.6	13.6	12.0	12.4
19	2	4	18.8	20.7	19.0	19.5
20	2	5	30.4	27.7	30.1	29.6

他の例: SuperLearner

- Van der Laan, Polley, and Hubbard (2007) により提案
  - OLS 推定だが、$\beta_0 = 0, \beta_a \geq 0$ と制約
- 細かいチュートリアル (Phillips et al. 2022)
- SuperLearner Pacakge

まとめ

- Stacking の実戦では、大きく異なる予測モデルを生み出すアルゴリズムを使用すべき
  - 少なくとも決定木系統と LinearModel 系統を含めている実践が多い
  - 単純な OLS など、伝統的な推定方法も含める

Reference

Díaz, Iván. 2019. "Machine Learning in the Estimation of Causal Effects: Targeted Minimum Loss-Based Estimation and Double/Debiased Machine Learning." *Biostatistics*.

Einav, Liran, Amy Finkelstein, Sendhil Mullainathan, and Ziad Obermeyer. 2018. "Predictive Modeling of US Health Care Spending in Late Life." *Science* 360 (6396): 1462–65.

Friedberg, Rina, Julie Tibshirani, Susan Athey, and Stefan Wager. 2020. "Local Linear Forests." *Journal of Computational and Graphical Statistics* 30 (2): 503–17.

Naimi, AI, AE Mishler, and EH Kennedy. 2021. "Challenges in Obtaining Valid Causal Effect Estimates

with Machine Learning Algorithms." *American Journal of Epidemiology*, kwab201–1.

Phillips, Rachael V, Mark J van der Laan, Hana Lee, and Susan Gruber. 2022. "Practical Considerations for Specifying a Super Learner." *arXiv Preprint arXiv:2204.06139*.

Van der Laan, Mark J, Eric C Polley, and Alan E Hubbard. 2007. "Super Learner." *Statistical Applications in Genetics and Molecular Biology* 6 (1).