0.1 simulation studies, TMLE blip variance

n = 1000 (sample size)

1000 replicates

0.1.1 data generating process

True treatment mechanism: $g_0 = expit(.5 * (-0.8 * W_1 + 0.39 * W_2 + 0.08 * W_3 - 0.12 * W_4 - 0.15))$

For the purposes of this study, we will draw from this treatment mechanism and estimate it with a well-specified linear model. Our estimator for blip variance is not doubly robust as the tmle is for average treatment effect so correctly specifying the treatment mechanism does not insure consistency. The second order remainder terms for the tmle of blip variance depend on the doubly robust terms present for average treatment effect as well as a term only dependent on estimating the blip function. Therefore we will focus mainly on cases where we have a misspecified outcome model only. Therefore all estimates for average treatment effect predictably come out fine and we can see the difference in robustness of the estimators.

True outcome model 1: $Q_0 = expit(.14 * (2 * A + 2 * A * W_1 + 20 * cos(W_1) * A - 3 * W_1 * sin(2 * W_2) + cos(W_1) - 3 * W_2 + 4 * A * (W_2^2) + 3 * cos(W_4) * A + A * W_1^2 - 2 * sin(W_2) * W_4 - 6 * A * W_3 * W_4 - 3))$

True outcome model 2: $Q_0 = expit(.14*(2*A+20*cos(W_1)*A+cos(W_1)-4*A*(W_2^2)+3*cos(W_4)*A+A*W_1^2))$

0.1.2 Pitfall of a Narrow Model

To illustrate the pitfall of using a too narrow model, we will form initial predictions using a standard glm with all main terms and interactions with treatment. The TMLE for blip variance, assuming a well-specified treatment mechanism, relies on estimating the blip function to eliminate bias to account for the 2nd order remainder term but, also needs to estimate the outcome well to provide the right inference. The targeting that lessens empirical loss while solving the influence curve equation, will not be worthwhile if the influence curve approximation is off.

Figure 1: For outcome model 1

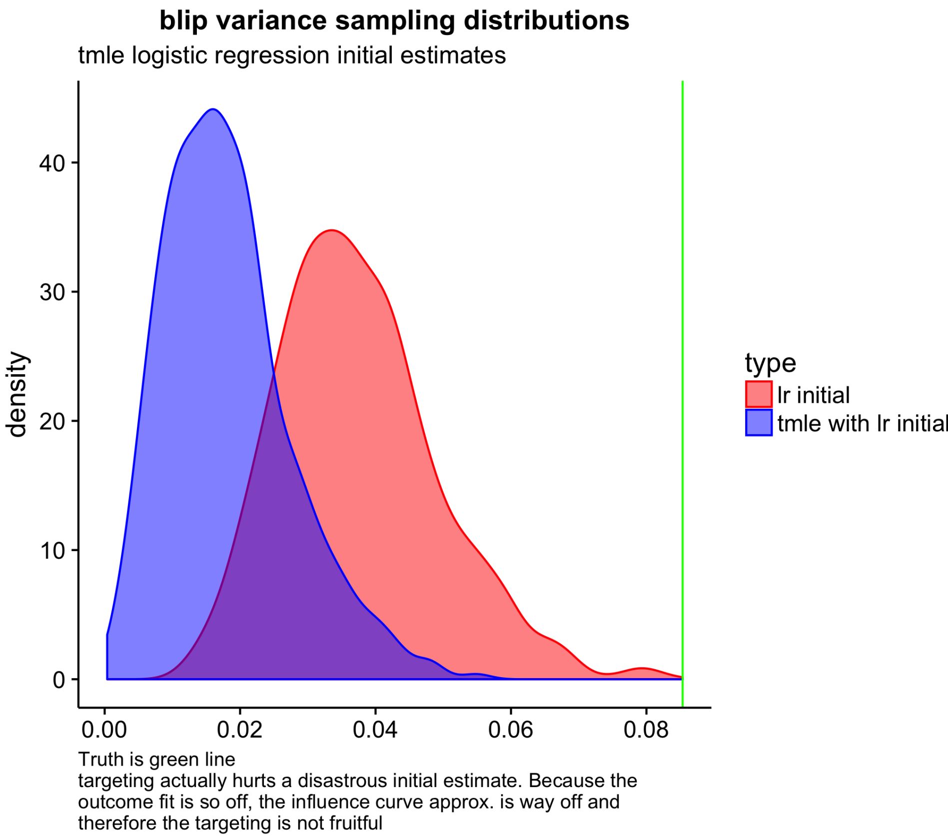
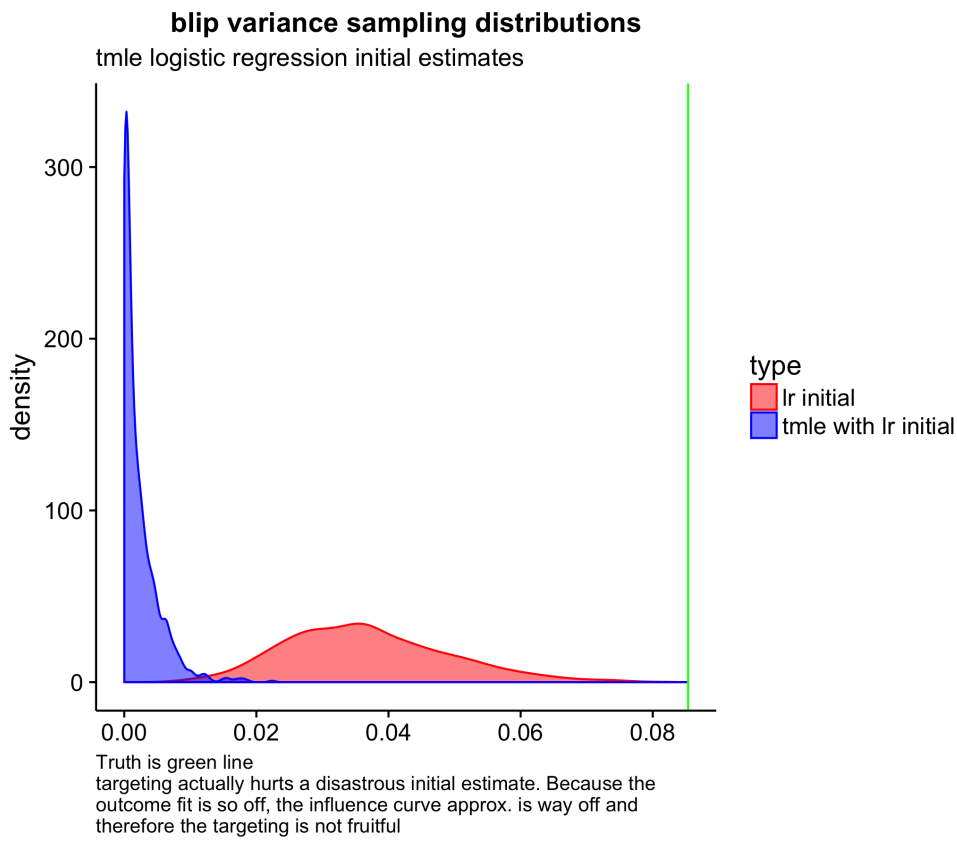


Figure 2: for outcome model 2



0.1.3 Using Highly Adaptive Lasso (HAL) For Initial Estimates

We can see for this data generating process that using glm with interactions cannot be helped by targeting but using the highly adative lasso (HAL), though greatly biased in the initial estimate of the parameter, allowed for targeting to salvage reasonable coverage.

Table 1: under outcome model 1

	var	bias	mse
est	0.000494	0.014832	0.000714
est	0.000488	0.015359	0.000723
	0.000066	-0.046406	0.002219

We see below that coverage improved with use of the true variance, suggesting that targeting the variance of the influence curve might help coverage here.

Table 2

1 step	iterative	1 step using true var
0.825	0.816	0.891

Figure 3: Under outcome model 1

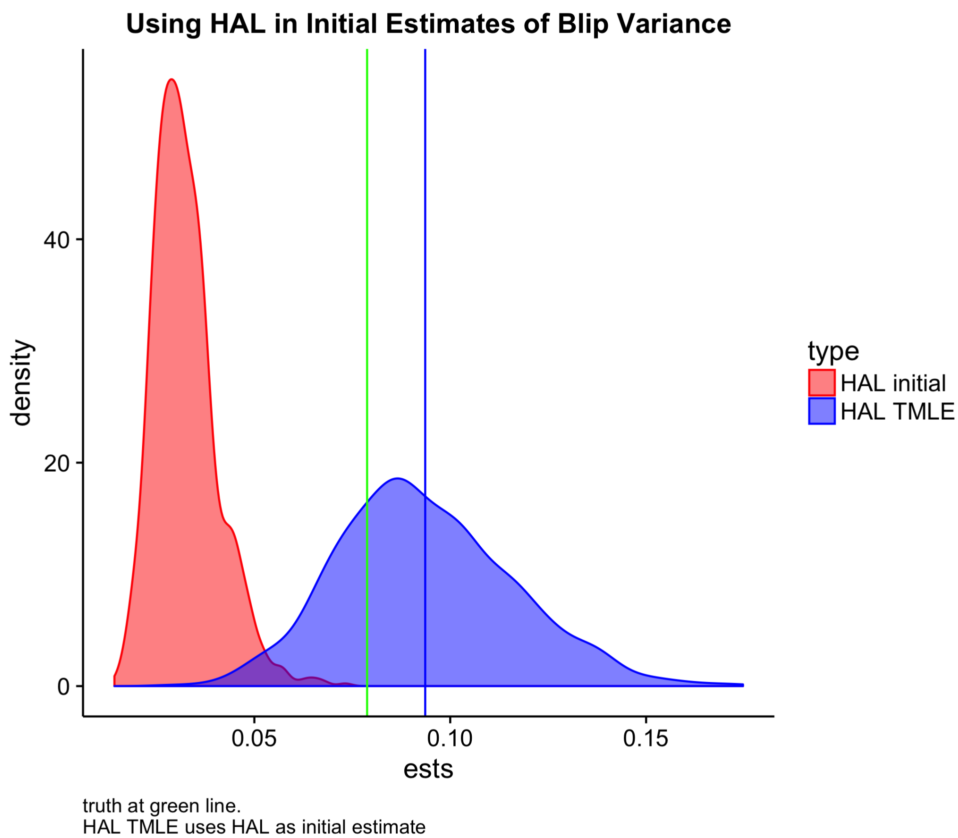


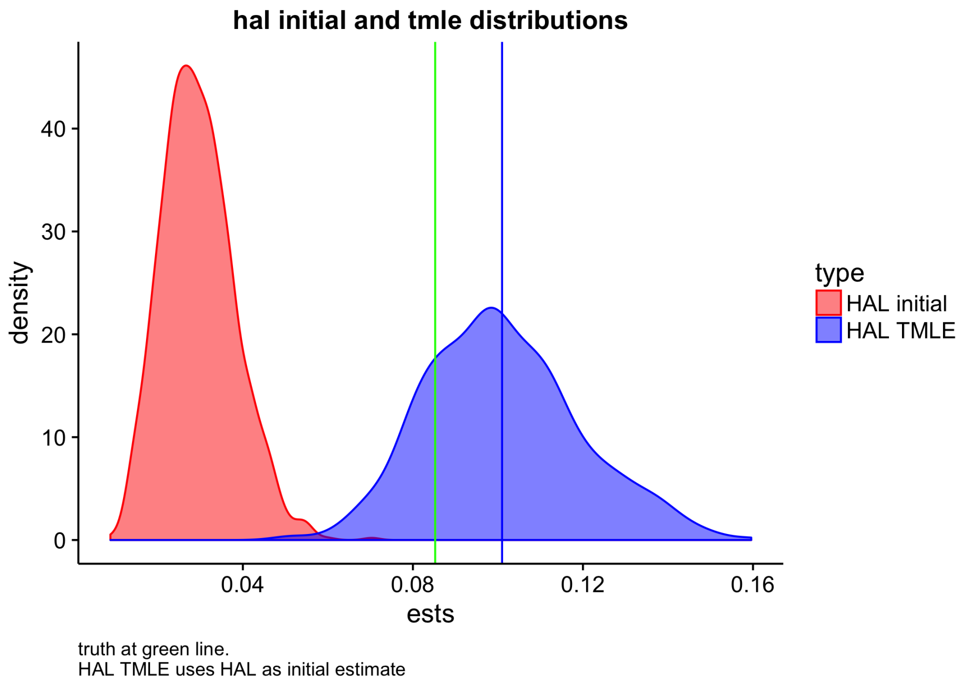
Table 3: under outcome model 2

	var	bias	mse
est	0.000326	0.015737	0.000574
est	0.000318	0.015937	0.000572
	0.000072	-0.055659	0.003170

Table 4

1 step	iterative	1 step using true var
0.859	0.858	0.857

Figure 4: under outcome model 2



0.2 Simulations Involving Superlearner

0.2.1 model 1

Below we keep the same data generating process but use SuperLearner to obtain the outcome prediction, staying with a correctly specified model for the treatment mechanism.

Table 5

	var	bias	mse
ci_simulate.est	0.000972	-0.000630	0.000972
ci_simulate_jer.est	0.000971	-0.000523	0.000971
ci_ATE.est	0.000965	-0.000463	0.000966
ci_simulate_jl.est	0.000975	-0.000579	0.000976
initests1	0.000889	-0.003329	0.000900

Table 6

	var	bias	mse
ci_sigmait.est	0.000302	0.005029	0.000327
ci_sigma.est	0.000301	0.004460	0.000321
ci_simulsig.est	0.000301	0.004519	0.000322
cci_simulsig_jer.est	0.000302	0.005086	0.000328
ci_simulsig_jl.est	0.000303	0.004869	0.000327
initests2	0.000210	-0.020031	0.000611

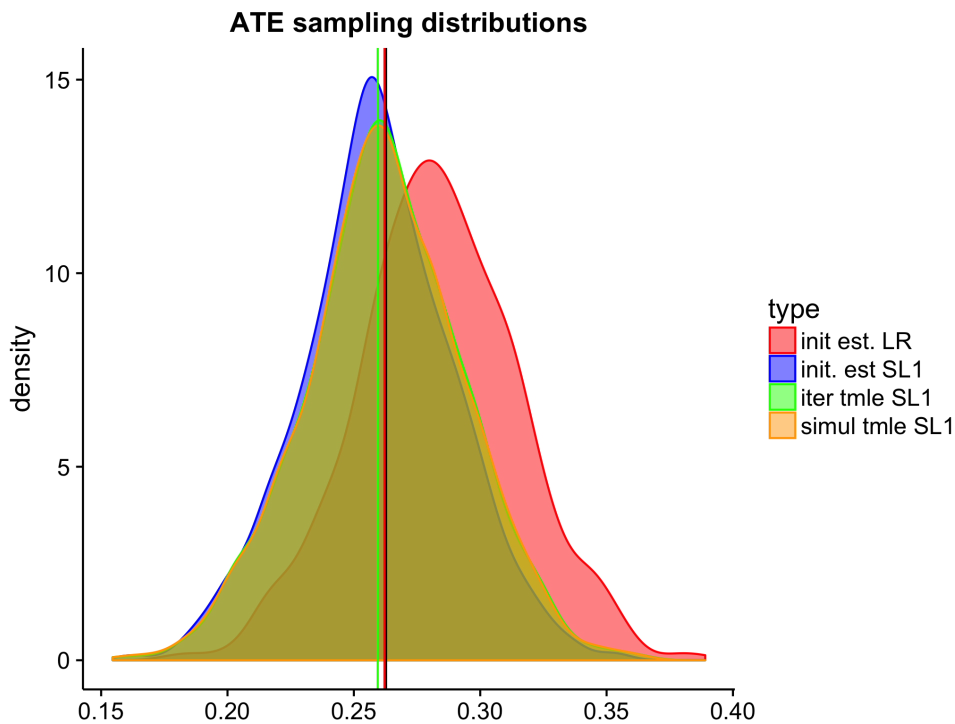
Table 7

	coverage
cov.sig.it	0.945
cov.sig.1step	0.946
cov.simul.1step	0.933
cov.simul.it1	0.931
cov.simul.it	0.933
cov.ate	0.927
blipvar using true var	0.940
ATE using true var	0.949

Table 8

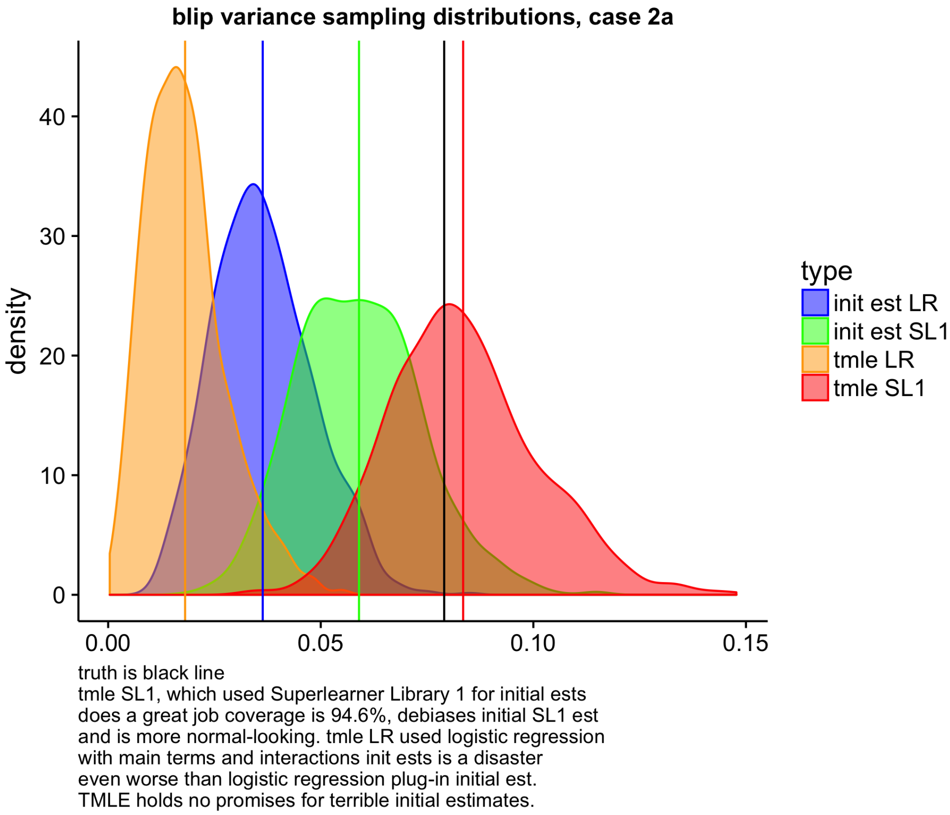
	colMeans.results...41.57..
SL.gam3_screen.Main	0.00030
SL.gam3_screen10	0.06662
SL.gam3_screen6	0.02327
SL.glmnet_1_All	0.00507
SL.glmnet_2_All	0.01476
SL.glmnet_3_All	0.01382
xgbFull_All	0
xgbMain_screen.Main	0
nnetMain_screen.Main	0.23125
earthMain_screen.Main	0.35669
rangerFull_screen.Main	0
SL.glm_screen.Main	0.00007
SL.glm_screen6	0.00363
SL.glm_screen10	0.03033
SL.stepAIC_All	0.14267
SL.hal_screen.Main	0.11138
SL.mean_All	0.00014

Figure 5



Both TMLE's are virtually the same, initial is excellent
Truth is at black vline. TMLE for ATE using true variance covers at 94.9%
as opposed to 92.7% using IC approximation for variance, suggesting
targeting the variance of IC would be a little fruitful or running cv-tmle. 1step simul TMLE
estimated blip variance with ATE

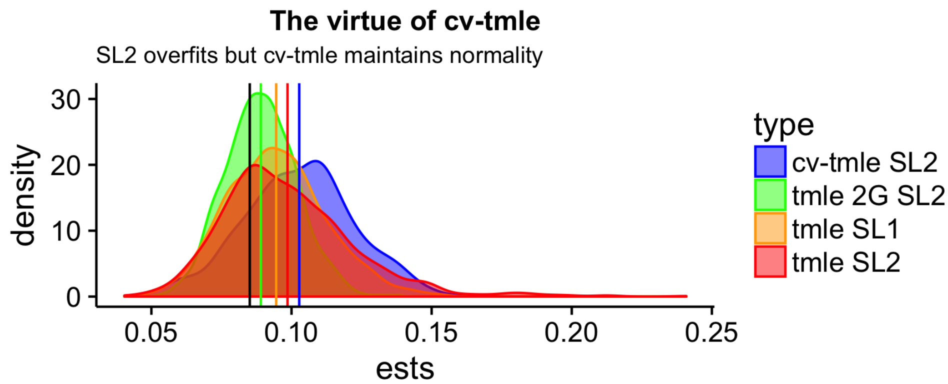
Figure 6



0.2.2 model 2: A Case for Using CV-TMLE as a precaution

We performed 1000 simulations for each of the four cases below. Sample size was 1000 except for 1 step tmel 2G SL2 which had sample size 2000. We can see that though the slightly overfitting SuperLearner library (Library 2) caused some bias, cv-tmle still covers nearly nominally and has a nice normal-shaped sampling distribution where as the regular one-step tmle using this same library, suffers from some skewing and severe outliers even though the library only had 1 out of the 13 algorithms sometimes overfitting. These overfitting learners got 0 SuperLearner coefficient once sample size was bumped up to 2000 instead of 1000 and hence the excellent performance. The regular 1 step tmle using library 1, which was designed not to overfit, had a normal-looking sampling distribution and covered very near nominally.

Figure 7



truth at the black line
'SL2' means we used SuperLearner library 2, slightly overfiting
SL1 was a library that did not overfit
2G had samples size 2000 instead of 1000 like the others
For 2G, the overfitters in library 2 got 0 coefficient.
All have nice symmetry except '1 step tmle SL2' which had some bad outliers
due to SuperLearner library overfitting. Also MSE and coverage for
'1 step tmle SL2' is 86.8% as opposed to 90%, 93.3% and 93.4% for 1 step tmle
which used SuperLearner Library 1, cv-tmle and tmle 2G, respectively.

0.2.3 model 2

Table 9

	var	bias	mse
sig	0.0003	0.009	0.0004
sigit	0.0003	0.010	0.0004
simul	0.0003	0.009	0.0004
simul_line	0.0003	0.010	0.0004
simul_full	0.0003	0.010	0.0004
sig_glm	0.00001	-0.083	0.007
sigit_glm	0.00001	-0.082	0.007
simul_glm	0.00001	-0.080	0.006
initest	0.0003	-0.008	0.0003
initest_lr	0.0001	-0.071	0.005

Table 10

	var	bias	mse
simulATE	0.001	0.001	0.001
ATE	0.001	0.001	0.001
simulATE_glm	0.001	0.001	0.001
ATE_glm	0.001	0.0004	0.001
initest_ATE	0.001	0.001	0.001
initest_lr_ATE	0.001	0.028	0.002

Table 11

	coverage
1 step tmle SL	0.933
iter. tmle SL	0.932
1 step simul tmle SL	0.951
1 step tmle LR	0
iter. tmle LR	0

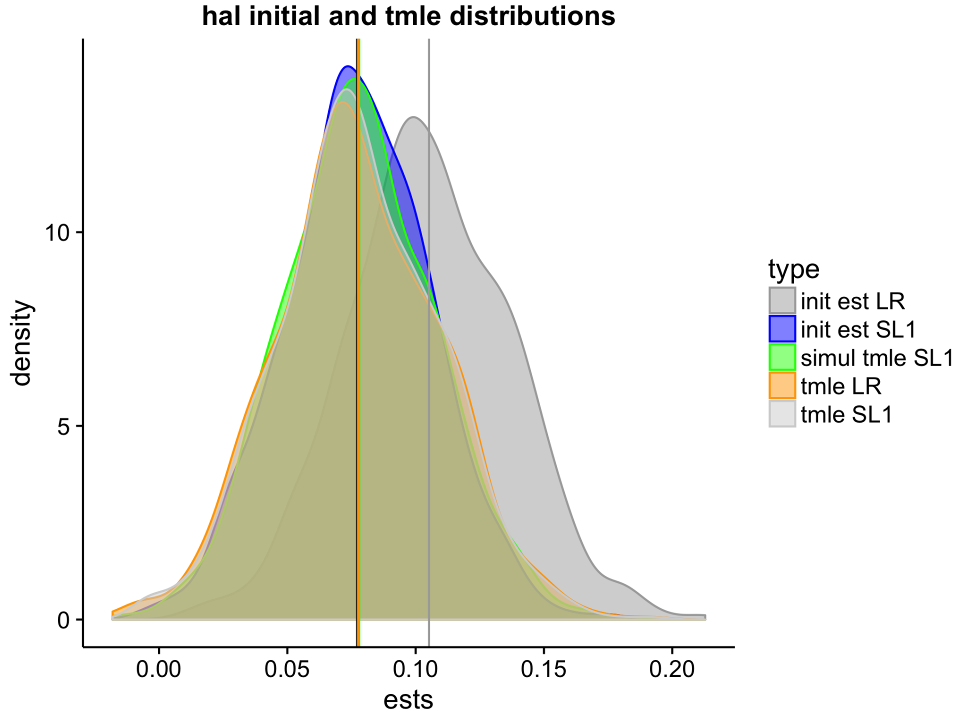
Table 12

	coverage_ate
1 step tmle SL	0.984
iter tmle SL	0.974
1 step tmle LR	0.986
iter tmle LR	0.978

Table 13

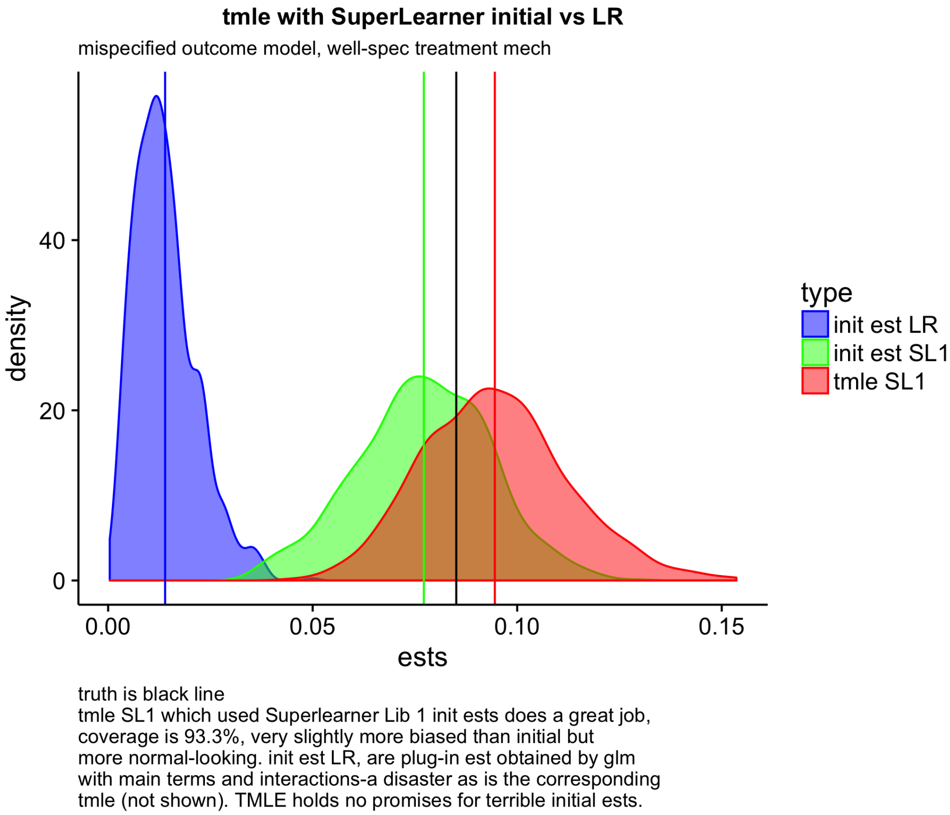
	coef
SL.gam3_screen.Main	0.008
SL.gam3_screen10	0.083
SL.gam3_screen6	0.153
SL.glmnet_1_All	0.006
SL.glmnet_2_All	0.010
SL.glmnet_3_All	0.013
rpartPrune_All	0.025
nnetFull_All	0.064
nnetMain_screen.Main	0.048
earthFull_All	0.046
earthFull_screen10	0.067
earthFull_screen6	0.081
earthMain_screen.Main	0.058
SL.glm_screen6	0.025
SL.glm_screen10	0.018
SL.stepAIC_All	0.252
SL.hal_screen.Main	0.041
glm.mainint_screen.Main	0.003
SL.mean_All	0.0002

Figure 8: ATE Sampling Dists



truth at black line
Since tmle is doubly robust as ATE estimator
a well-specified treatment mechanism results in
excellent results regardless of bias in estimating
outcome model. All tmles do very well.

Figure 9: Blip Variance Sampling Dists



0.2.4 under outcome model 2 but sample size, n =2000

Here we see for n = 2000, the TMLE backed by the power of a modest superlearner does a miraculous job.

Table 14

	var	bias	mse
ci_simulate.est	0.000461	-0.007225	0.000514
ci_simulate_jer.est	0.000462	-0.007195	0.000513
ci_ATE.est	0.000461	-0.007202	0.000513
ci_simulate_jl.est	0.000462	-0.007246	0.000514
initests1	0.000429	-0.005846	0.000463

Table 15

	var	bias	mse
ci_sigmait.est	0.000158	0.003857	0.000173
ci_sigma.est	0.000157	0.003605	0.000170
ci_simulsig.est	0.000157	0.003613	0.000170
cci_simulsig_jer.est	0.000158	0.003863	0.000173
ci_simulsig_jl.est	0.000158	0.003730	0.000172
initests2	0.000138	-0.004965	0.000163

Table 16

	coverage
cov.sig.it	0.933
cov.sig.1step	0.937
cov.simul.1step	0.955
cov.simul.it1	0.953
cov.simul.it	0.953
cov.ate	0.965
blipvar using true var	0.944
ATE using true var	0.968

Table 17

	colMeans.results...41.57..
SL.gam3_screen.Main	0.00409
SL.gam3_screen10	0.16856
SL.gam3_screen6	0.37134
SL.glmnet_1_All	0.00429
SL.glmnet_2_All	0.01467
SL.glmnet_3_All	0.02295
xgbFull_All	0
xgbMain_screen.Main	0
nnetMain_screen.Main	0.05432
earthMain_screen.Main	0.08363
rangerFull_screen.Main	0
SL.glm_screen.Main	0.00013
SL.glm_screen6	0.01699
SL.glm_screen10	0.01277
SL.stepAIC_All	0.21296
SL.hal_screen.Main	0.03209
SL.mean_All	0.00122

Figure 10

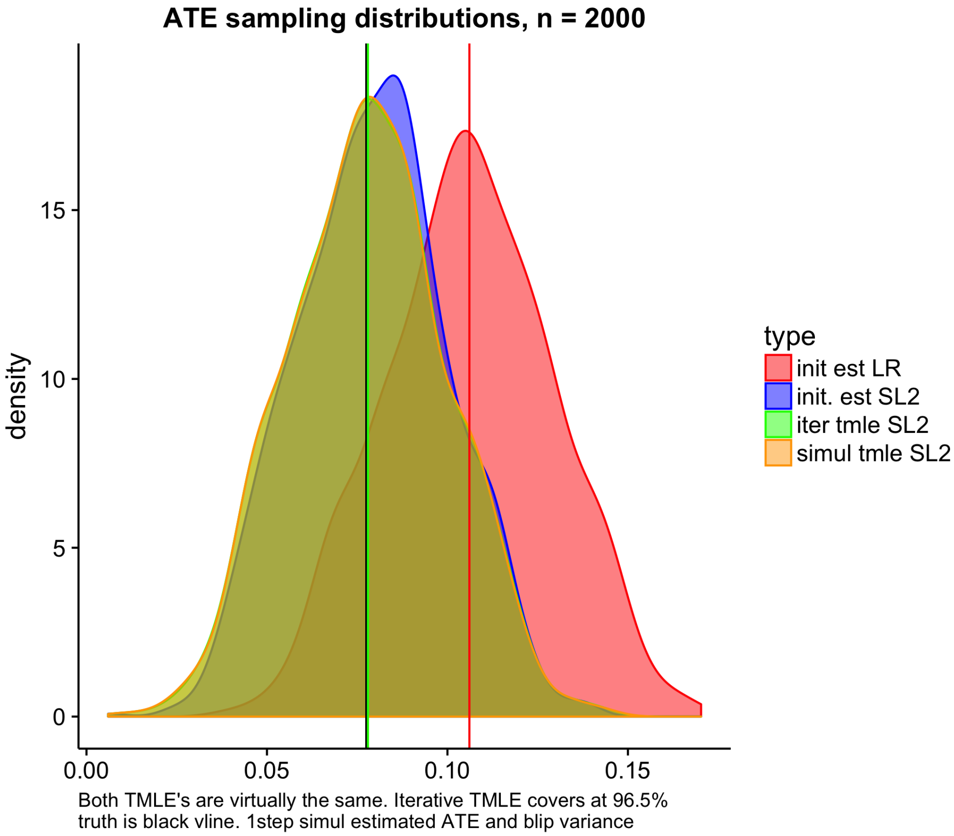

ATE sampling distributions, n = 2000
density
15
10
5
0
0.00
0.05
0.10
0.15
type
init est LR
init. est SL2
iter tmle SL2
simul tmle SL2
Both TMLE's are virtually the same. Iterative TMLE covers at 96.5%
truth is black vline. 1step simul estimated ATE and blip variance

Figure 11

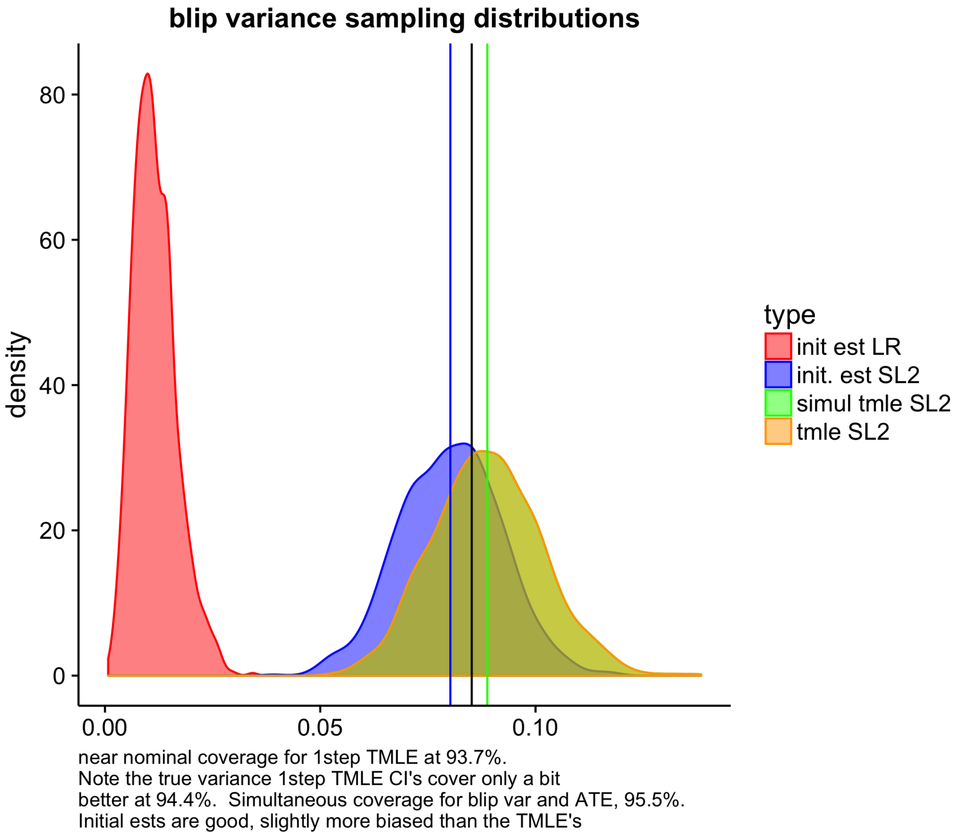


Table 18

	var	bias	mse
ci_simulate.est	0.000993	0.001915	0.000997
ci_simulate_jer.est	0.000993	0.001987	0.000997
ci_ATE.est	0.000992	0.002067	0.000997
ci_simulate_jl.est	0.000994	0.001887	0.000998
initests1	0.000914	0.002240	0.000919

Table 19

	var	bias	mse
ci_sigmait.est	0.000570	0.013417	0.000750
ci_sigma.est	0.000565	0.012857	0.000730
ci_simulsig.est	0.000567	0.012898	0.000733
cci_simulsig_jer.est	0.000572	0.013454	0.000753
ci_simulsig_jl.est	0.000575	0.013249	0.000751
initests2	0.000233	-0.009207	0.000318

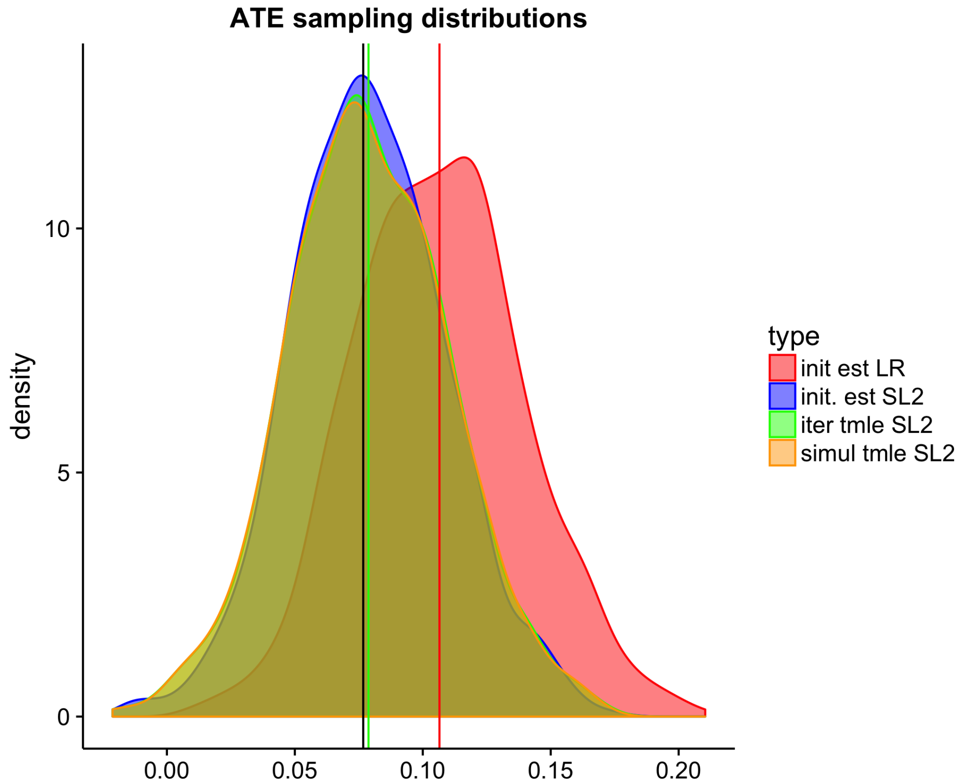
Table 20

	coverage
cov.sig.it	0.863
cov.sig.1step	0.866
cov.simul.1step	0.871
cov.simul.it1	0.868
cov.simul.it	0.869
cov.ate	0.938
blipvar using true variance	0.923
ATE using true variance	0.939

Table 21

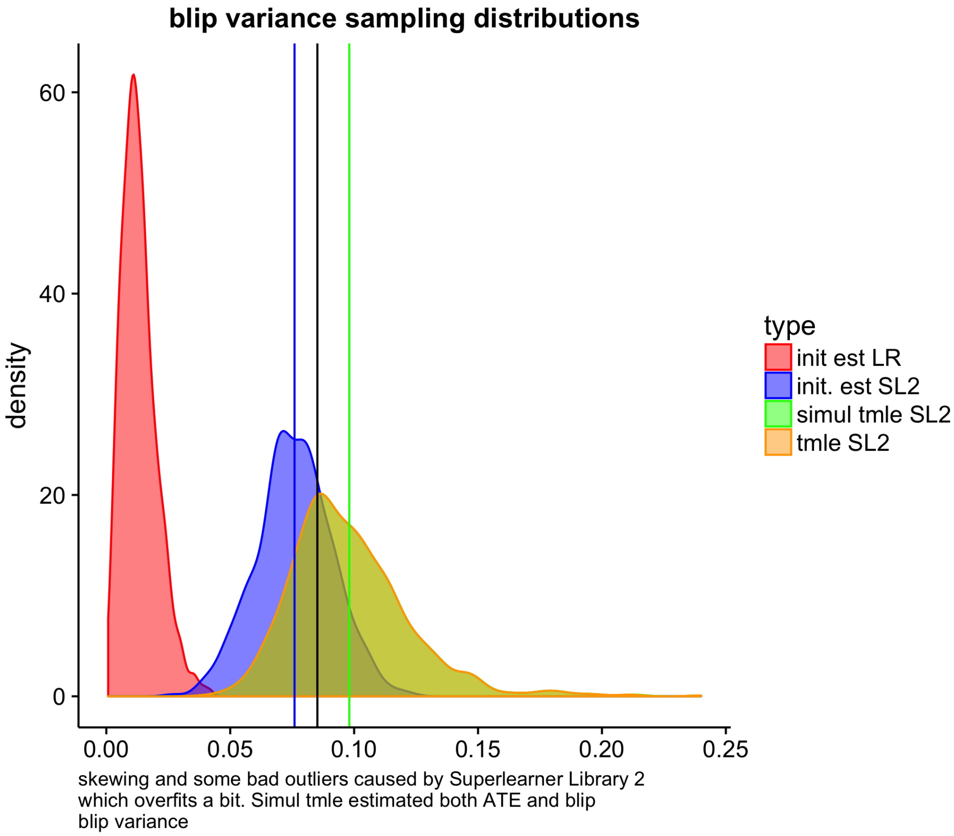
	colMeans.results...41.57..
SL.gam3_screen.Main	0.00691
SL.gam3_screen10	0.12003
SL.gam3_screen6	0.19020
SL.glmnet_1_All	0.00606
SL.glmnet_2_All	0.01267
SL.glmnet_3_All	0.02540
xgbFull_All	0.02310
xgbMain_screen.Main	0.01953
nnetMain_screen.Main	0.04382
earthMain_screen.Main	0.08092
rangerFull_screen.Main	0.03104
SL.glm_screen.Main	0.00014
SL.glm_screen6	0.03132
SL.glm_screen10	0.02691
SL.stepAIC_All	0.35589
SL.hal_screen.Main	0.02415
SL.mean_All	0.00191

Figure 12



Both TMLE's are virtually the same. Iterative TMLE covers at 93.9%
truth is black vline. simul tmle estimated ATE and blip variance

Figure 13



0.3 TMLE with mis-specified treatment mechanism and outcome

Our first example where we will recover both treatment mechanism and outcome model, which are both mis-specified (as opposed to just mispecifying the outcome model), involves a comparison of three different ways of producing initial estimates before the targeting step. First we will perform 1000 simulations using generalized linear models with main terms and interactions for the outcome regression and main terms glm for the treatment mechanism estimation before targeting. We also perform 1000 simulations where we use hal, the highly adaptive lasso (ref Mark) for both outcome regression and treatment mechanism estimates before targeting. Finally we run simulations where we put both the previously described learners in the SuperLearner and use the SuperLearner predictions as the initial estimates for both outcome model and treamtent mechanism.

We can see how glm fails due to both bias and bad bias-variance trade-off which targeting (i.e. tmle) could not help very much. In this case we can see glm will fail to satisfy the tmle conditions. By contrast, for a cv-tmle (ref zheng) any initial estimates obtained by a SuperLearner containing hal in the library will be asymptotically efficient and for a non cv-tmle, we just need to insure our learners do not overfit and therefore violate conditions 3 in section 5.1.1 (donsker condition). In this case, which is finite sampled, SuperLearner with just hal and glm leads to better estimates than just hal alone, which is not surprising because it creates the best convex combination of the two learners. Inference was obtained using the influence curve approximation. Using the true variance obtained from the 1000 simulations only helped coverage for when we used hal only for the initial estimates and made very little difference otherwise.

0.3.1 Data Generating Process

outcome model: $E[Y \mid A, W] = expit(0.14(2A + 2AW_1 + 4AW_3W_4 + W_2W_1 + W_3W_4 + 10Acos(W_4)))$
treatment mechanism: $E[A \mid W] = expit(0.5(-0.08W_1^2W_2 + 0.5W_1 + 0.49cos(W_2)W_3 + 0.18W_3^2 - 0.12sin(W_4) - 0.15))$
True blip variance: 0.0263 True ATE = 0.2294

Table 22

glm tmle	hal tmle	hal tmle true var	hal+glm SL tmle
0.4950	0.8600	0.9060	0.9320

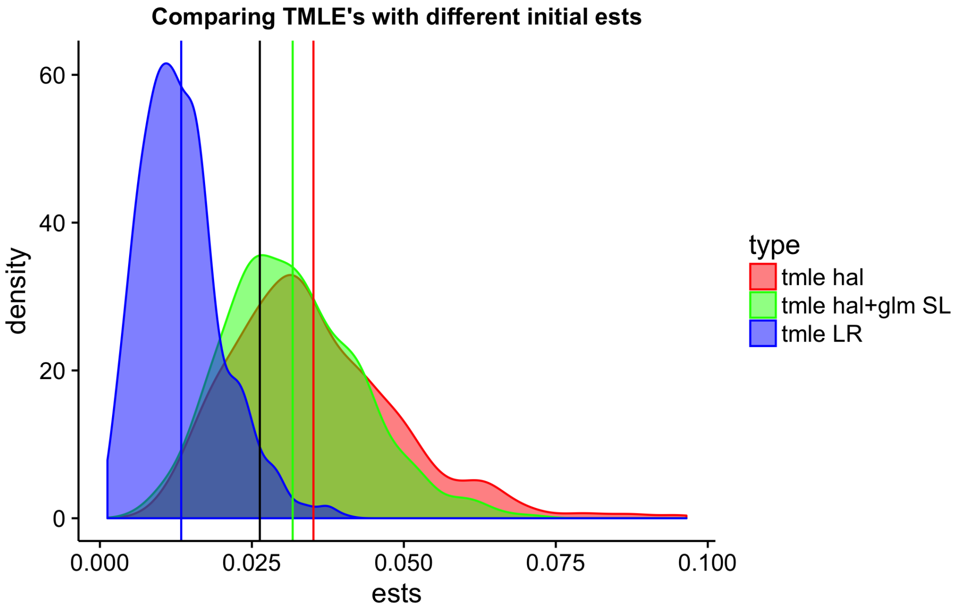
for 95% Confidence Bands
hal tmle underestimated the true var a bit
hal+glm SuperLearner tmle almost nominal.
Other than true var, CI's formed from influence curve approx.
sample standard deviation.

Table 23

	var	bias	mse
glm tmle	0.000045	-0.012936	0.000212
hal tmle	0.000192	0.008820	0.000270
hal+glm SL tmle	0.000124	0.005404	0.000153
glm init	0.000039	-0.014150	0.000239
hal init	0.000010	-0.017333	0.000310
hal+glm SL init	0.000009	-0.017440	0.000313

glm tmle still bad bias var trade-off
glm+hal SL tmle clearly wins

Figure 14



truth at black line.
tmle LR uses glm with interactions for outcome model and glm for treatment mechanism initial estimates. tmle LR CI's cover at 50%
hal tmle uses highly adaptive lasso for initial estimates of both outcome and treatment mechanism initials estimates.
tmle hal CI's cover at 86%. tmle hal+glm SL uses a SuperLearner with hal and glm for outcome and treatment mechanism initial estimates
hal+glm tmle CI's cover at 93.2%. All coverage here is using the infuence curve approximation for inference.

We can see that the initial estimate using HAL is more biased than the initial estimate using glm with main terms and interactions for the outcome. However, we must also estimate the treatment mechanism at $n^{-.25}$ L_2 convergence rate as well as the outcome model to obtain an asymptotically efficient tmle estimator as described in this paper. For doubly robust tmle estimators as with the average treatment effect, the product of the L_2 rates for both the treatment mechanism and the outcome model must be $n^{-0.5}$ so generally, we must do pretty well estimating both nuisance parameters. HAL achieves these requirements asymptotically but, of course, with a linear model, mispecification is guaranteed in reality as well as in this simulation. The glm tmle does pretty well because it does capture interactions in the outcome model but it is still clearly more biased than when using HAL as the initial estimate for the tmle.

0.3.2 Data Generating Process

outcome model: $E[Y \mid A, W] = expit(0.14(2A + 5AW_1 + 4AW_3W_4 + W_2W_1 + W_3W_4 + 10Acos(W_4)))$

treatment mechanism: $E[A \mid W_{}] = expit(0.5(-0.08W_1^2W_2 + 0.5W_1 + 0.49cos(W_2)W_3 + 0.18W_3^2 - 0.12sin(W_4) -$

0.15))

True blip variance: 0.05497 True ATE = 0.1943

Figure 15

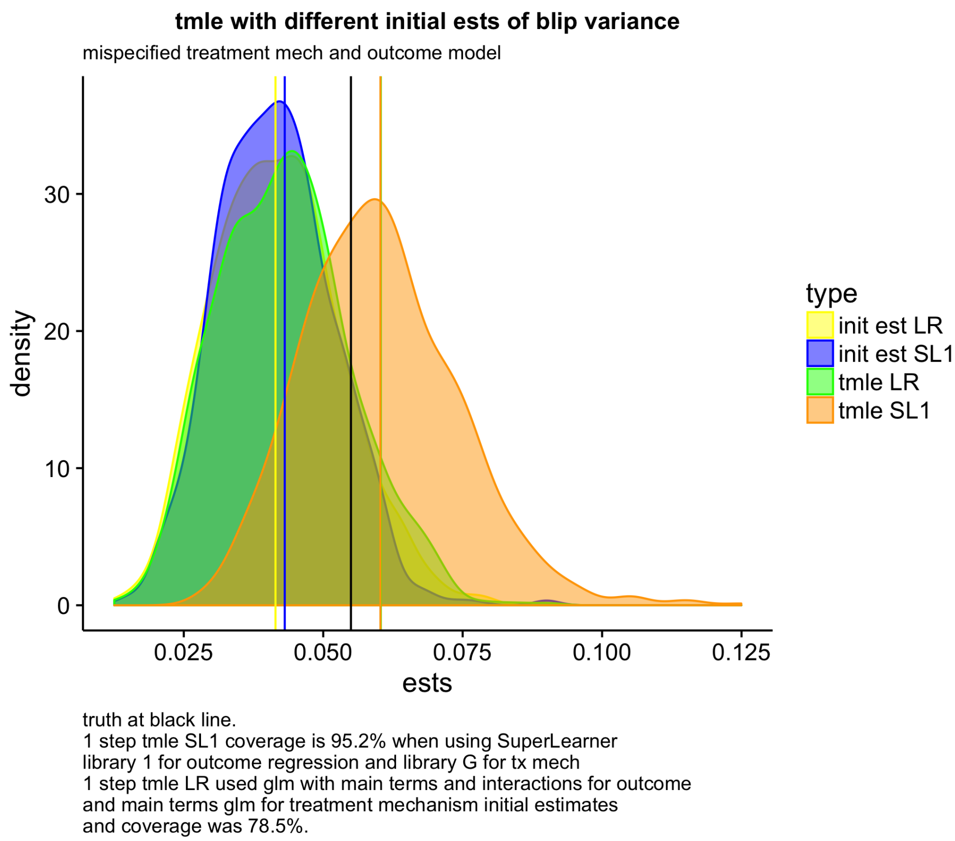


Figure 16

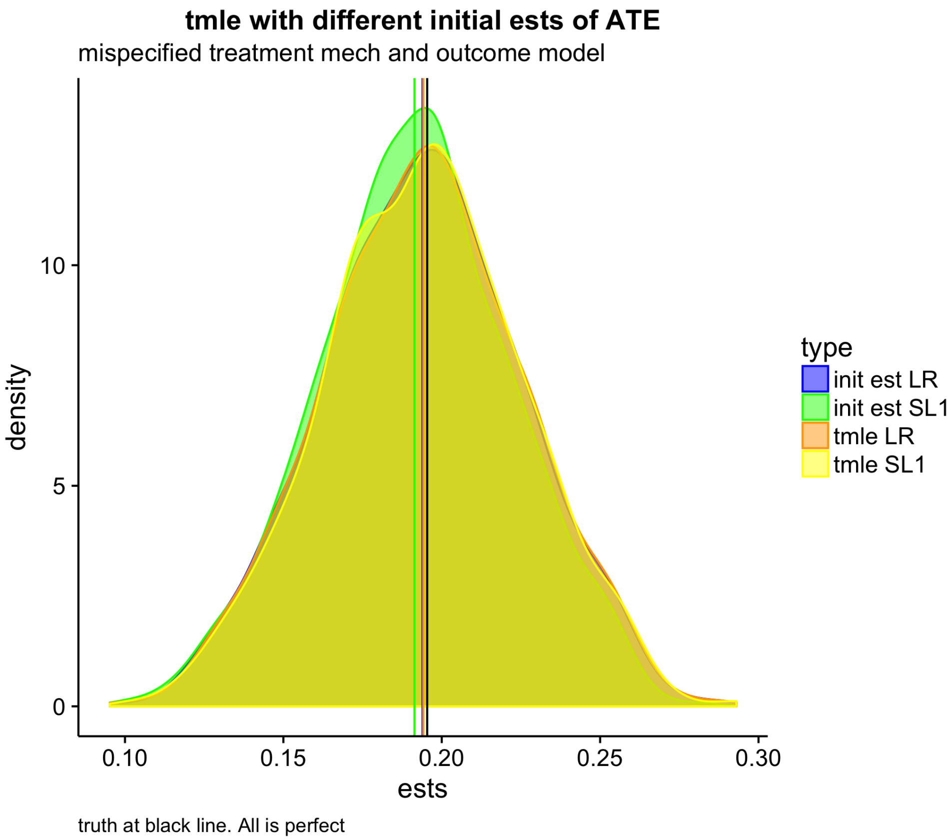


Table 24

	var	bias	mse
sig	0.00019	0.00529	0.00022
sigit	0.00020	0.00570	0.00023
simul	0.00019	0.00535	0.00022
simul_line	0.00020	0.00555	0.00023
simul_full	0.00020	0.00574	0.00023
sig_glm	0.00014	-0.01187	0.00028
sigit_glm	0.00014	-0.01177	0.00028
initest	0.00011	-0.01353	0.00029
initest_lr	0.00013	-0.01314	0.00030

1 totals

Table 25

	coverage
1 step tmle SL	0.954
iterative tmle SL	0.952
1 step simul tmle	0.953
1 step tmle LR	0.785
iterative tmle LR	0.785

Table 26

	SL.coef
SL.gam3_screen.Main	0.004
SL.gam3_screen10	0.052
SL.gam3_screen6	0.009
SL.glmnet_1_All	0.019
SL.glmnet_2_All	0.024
SL.glmnet_3_All	0.054
nnetMain_screen.Main	0.184
earthMain_screen.Main	0.094
SL.glm_screen.Main	0.006
SL.glm_screen6	0.011
SL.glm_screen10	0.014
SL.stepAIC_All	0.426
SL.hal_screen.Main	0.093
SL.mean_All	0.001
glm.mainint_screen.Main	0.009

Table 27

	SLg.coef
nnetMain_All	0.124
SL.mean_All	0.0005
SL.hal_All	0.098
SL.earth_All	0.074
SL.glm_All	0.445
SL.step.interaction_All	0.193
SL.glm.interaction_All	0.065

Table 28

	var	bias	mse	.1
tmle SL case 2a	0.00030	0.00446	0.00032	0.94600
simul tmle SL case 2a	0.00030	0.00452	0.00032	0.93300
init. ests SL case 2a	0.00021	-0.02003	0.00061	
tmle LR case 2a	0.00013	-0.04267	0.00195	0
init. ests LR case 2a	0.00009	-0.06093	0.00380	0
tmle SL case 2b	0.00031	0.00942	0.00040	0.93300
simul tmle SL case 2b	0.00031	0.00945	0.00040	0.95100
init. ests SL case 2b	0.00026	-0.00793	0.00032	
tmle LR case 2b	0.00001	-0.08271	0.00685	0
init. ests LR case 2b	0.00006	-0.07118	0.00512	0
tmle LR case 4	0.00004	-0.01294	0.00021	0.49500
init. ests LR case 4	0.00004	-0.01415	0.00024	0
tmle hal case 4	0.00019	0.00882	0.00027	0.90600
init. ests hal case 4	0.00001	-0.01733	0.00031	
tmle hal+LR SL case 4	0.00012	0.00540	0.00015	0.93200
init ests hal+LR SL case 4	0.00001	-0.01744	0.00031	
tmle SL case 3	0.00019	0.00529	0.00022	0.95400
init ests case 3	0.00011	-0.01353	0.00029	
tmle LR case 3	0.00014	-0.01187	0.00028	0.78500
init ests LR case 3	0.00013	-0.01314	0.00030	0