Web-based Supplementary Materials for "A Generalized Estimator of the Attributable Benefit of an Optimal Treatment Regime,"

Jason Brinkley, Anastasios Tsiatis, and Kevin J. Anstrom

March 4, 2009

1 Web Appendix A

1.1 Sensitivity Analysis

In the main paper we discuss the issue of no unmeasured confounders in our discussion Section 7. The database used for our analysis has been refined over the years as it has been used extensively for treatment related research and for which a great deal of effort has been made in capturing the most relevant data that clinicians use in treatment decision making. Therefore, we are cautiously optimistic that the assumption of no unmeasured confounders may be reasonable for our data analysis. However, except for a randomized study, this assumption is nonidentifiable.

As an example of a sensitivity analysis, we may consider the approach of Brumback *et al.* (2004). They define the sensitivity parameter

$$c(a,x) = E\{Y^*(a)|T=a, X=x\} - E\{Y^*(a)|T=1-a, X=x\}, a=0,1, \tag{1}$$

as a sensitivity function whose range of plausible values may be elicited from subject matter scientists. Under the assumption of no unmeasured confounders $c(a,x)=0$ for all a and x. If we consider any treatment regime $g(X)$ and the corresponding potential response

$$Y^*(g) = Y^*(1)g(X) + Y^*(0)\{1-g(X)\},$$

we have shown that our estimator $\hat{P}\{Y^*(g)=1\}$ converges to

$$E_X[E(Y|T=1,X)g(X) + E(Y|T=0,X)\{1-g(X)\}].$$

It is then straightforward to show that for the sensitivity function (1), the bias of our estimator $\hat{P}\{Y^*(g)=1\}$ is given by

$$E_X[c(1,X)P(T=0|X)g(X) + c(0,X)P(T=1|X)\{1-g(X)\}],$$

where $P(T=1|X)$ is the propensity score. If we consider a model for the propensity score (usually a logistic regression model is used) where we assume that $P(T=1|X)=\theta(X,\gamma)$, where the parameter γ is estimated using maximum likelihood and denoted by $\hat{\gamma}$, then the bias can be estimated by

$$\hat{b} = n^{-1}\sum_{i=1}^{n}\big[c(1,X_i)\{1-\theta(X_i,\hat{\gamma})g(X_i)\} + c(0,X_i)\theta(X_i,\hat{\gamma})\{1-g(X_i)\}\big].$$

A bias corrected estimator for $P\{Y^*(g)=1\}$ could then be obtained as $\hat{P}\{Y^*(g)=1\} - \hat{b}$. The asymptotic properties of this estimator as well as the corrected estimator for $AB(g)$ could then be derived using methods as described in our paper with, of course, accounting for the estimation of γ in the propensity score model.

2 Web Appendix B

2.1 Additional Simulations

In the main text we report results of several simulations, each based on 10000 Monte Carlo data sets. Here we put forth some two covariate simulations to illustrate that the current estimator for $AB_{opt}(0)$ has good coverage probabilities. For simplicity we consider data where Y_i is binary disease status, T_i is treatment, and X_{1i}, X_{2i} will be two independent potentially confounding covariate generated from a $N(0,1)$ distribution. For each given X_{1i}, X_{2i} we generated a Bernoulli treatment indicator T_i from the logistic regression model:

$$\text{logit}\{P(T_i = 1|X_{1i}, X_{2i})\} = \alpha_0 + \alpha_1 X_{1i} + \alpha_2 X_{2i}, \tag{2}$$

and a Bernoulli disease indicator Y_i from the logistic regression model:

$$\text{logit}\{P(Y_i = 1|T_i, X_{1i}, X_{2i})\} = \beta_0 + \beta_1 T_i + \beta_2 X_{1i} + \beta_3 X_{2i} + \beta_4 T_i X_{1i} + \beta_5 T_i X_{2i}. \tag{3}$$

From model (3), the optimal treatment regime is given by $g_{opt}(x_1, x_2) = I(\beta_1 + \beta_4 x_1 + \beta_5 x_2 < 0)$. Just like in the main paper, generated data and logit model estimates were then input into SAS IML where $\widehat{AB}_{opt}$, and confidence intervals were calculated for each Monte Carlo dataset. Web table 1 illustrates seven different examples with different combinations of model parameters. We depict the true parameter values along with estimates of $\widehat{AB}_{opt}$, Monte Carlo standard error, delta-theorem standard error and confidence intervals for both back-transformed and delta-theorem methods averaged over all 10000 Monte Carlo samples. For comparison we also considered the treatment regime where everyone gets the better of the two treatments, AB_{dom}.

3 Web Appendix C

3.1 SAS Macro to calculate $AB_{opt}(0)$

The following SAS code can also be found online at https://www.ecu.edu/cs-dhs/bios/ABopt.cfm.

```
/*************************************************************************
|
| Program Name:  AB opt covariate simulation
|
| Program Version: 1.0
|
| Program Purpose:  Calculates AB for the optimal treatment regime from a
| prespecified model.  User must preload data into SAS work folder.
|
| SAS Version: 8 or 9
|
| Created By:   Jason Brinkley
| Date:            19-Feb-2009
|
|*************************************************************************
| Change Log
|
| Modified By:
| Date of Modification:
|
| Modification ID:
| Reason For Modification:
|
*************************************************************************/

/*Instructions:
Load data into SAS work folder. Data must be of form: Binary Disease, Binary
Exposure/Treatment, Continuous or Binary Covariates.  For discrete variable input,
analyst needs to make binary indicators for each level of covariate and input those
binary indicators instead of discrete variables.  User must create interaction
variables in the dataset before running macro.  Analyst needs to use other
```

Table 1: Additional Simulation Results for $\widehat{AB}_{opt}(0)$

Simulation	1	2	3	4	5	6	7
n	1000	1000	1000	1000	1000	1000	1000
Reps	10000	10000	10000	10000	10000	10000	10000
α_0	1	0.5	-1.25	-1.25	-1.25	-1.25	-1.25
α_1	-1	1.5	1	1	1	0.5	1
α_2	1	2	1	1	1	0.5	1
β_0	-1	0.5	-3	-2.5	-1	-1.25	0.8
β_1	-1	1	1.25	1.25	1.25	1.5	0.5
β_2	1	1	1.1	1.1	1.1	0.5	1
β_3	0.5	1	1.1	1.1	1.1	0.5	1
β_4	-0.5	1.5	1	-1.5	2	0	-1.5
β_5	0.5	0	-1	-1.5	-2.5	0	-2.5
$P(T=1)$	0.6755	0.5637	0.2843	0.6755	0.2843	0.2432	0.2843
$P(Y=1)$	0.2423	0.6393	0.1595	0.2423	0.3797	0.3203	0.6075
$P\{Y^*(0)=1\}$	0.3087	0.5785	0.0953	0.3087	0.3621	0.2335	0.6263
$P\{Y^*(1)=1\}$	0.1889	0.6874	0.2644	0.1889	0.5337	0.5505	0.6064
$P\{Y^*(g_{opt})=1\}$	0.1839	0.5682	0.0886	0.184	0.277	0.2335	0.5554
AB_{dom}	0.2204	0.0951	0.4025	0.2204	0.0464	0.2710	0.0018
AB_{opt}	0.2410	0.1112	0.4445	0.2406	0.2705	0.2710	0.0858
$\widehat{AB}_{opt}(0)$	0.2475	0.1155	0.4545	0.2475	0.2741	0.2729	0.0898
$\widehat{SE}\{AB_{opt}(0)\}mc$	0.0464	0.0265	0.0605	0.0464	0.027	0.0303	0.0194
$\widehat{SE}\{AB_{opt}(0)\}dt$	0.0477	0.0266	0.0596	0.0477	0.0269	0.0304	0.0190
BT Lower 95%	0.1479	0.0618	0.3238	0.1479	0.2194	0.2109	0.0518
BT Upper 95%	0.3353	0.166	0.5593	0.3353	0.3249	0.3301	0.1262
BT Coverage	0.9562	0.9557	0.9477	0.9565	0.9502	0.9509	0.9413
DT Lower 95%	0.154	0.0634	0.3377	0.154	0.2214	0.2134	0.0526
DT Upper 95%	0.341	0.1676	0.5712	0.341	0.3249	0.3325	0.1270
DT Coverage	0.9509	0.9542	0.9381	0.9518	0.9482	0.9503	0.9395

```
techniques to find "best" model before running macro.

Macro inputs are as follows:

alpha - Confidence level for intervals (i.e. 95% intervals mean alpha=0.05)
Data - name of dataset in work folder
D -  Binary outcome/disease variable
E -  Binary exposure/treatment variable
X -  one or many covariates.  Continuous or Binary only.
Interactions between covariates go here.
X2 -  Which covariates/exposure interactions are included in the model
(i.e. if E*X is significant then put X here)
Int -  User created interaction variables in model
(i.e. if E*X is significant, analyst creates variable EX=E*X
EX goes here)
Out -  Name of output dataset for further manipulation
Interaction -  Indicator of whether there is covariate/exposure interactions
in the model (0 = no interactions, 1 = interactions).
IF THERE ARE COVARIATE/COVARIATE INTERACTIONS ONLY THEN PUT 0 HERE.

*/

%Macro AB_opt_reg(alpha, Data, D, E, X, X2, Int, Out, Interaction);

*Different logit models whether there is interactions;
%IF &Interaction = 0 %Then %Do;
proc logistic descending data=&Data;
model &D = &E &X;
ods output ParameterEstimates = ParameterEstimates;
run;
%End;

%IF &Interaction = 1 %Then %Do;
proc logistic descending data=&Data;
model &D = &E &X &Int ;
ods output ParameterEstimates = ParameterEstimates;
run;
%End;
        quit;

Data Betas;
            set ParameterEstimates;
            Keep Data Variable Estimate StdErr;
        run;

        proc iml;

                *Output files step, initializing variables;
                P_Dopt=0;
                P_D=0;
                Lower_BT=0;
                Upper_BT=0;
                AB_opt_hat = 0;
                ln_ratio = 0;
```

```
                lower_DT =0;
                upper_DT = 0;
                se_ab=0;

                        *Load data;
                        use &Data;
                        read all var {&X} into X;
                        read all var {&E} into E;
                        read all var {&D} into D;
read all var {&X2} into X2;

if &Interaction = 1 then read all var {&Int} into int;
*Calculate estimates for P(D=1);
                        P_D=(sum(D))/(nrow(D));

                        IF3=(D-P_D)/P_D;

                        Use Betas;
                        read all var{estimate} into B;
read all var{stderr} into StdErrB;

*bound needed for numeric derivatives;
bound = .01 * StdErrB;
bound2= bound;
r = nrow(B);
n=nrow(D);

Outvar = n||P_Dopt || P_D || AB_opt_hat ||ln_ratio || Lower_BT ||Upper_BT
||Lower_DT || Upper_DT ||SE_AB  ;
                cname = {"Sample Size" "Prob D optimal" "Prob Disease" "AB opt hat"
                "ln(ratio)" "Lower BT" "Upper BT" "Lower DT" "Upper DT" "Delta SE" };
                create out from Outvar [ colname=cname ];

*f, f_0, and f_1 are different whether there are interactions or not;
I=j(n,1);
if &Interaction=0 then f=T(I||E||X);
if &Interaction=1 then f=T(I||E||X||Int);
I1=j(n,1);
*I2 is a vector of zeros;
I2=I1-I1;
if &Interaction=0 then f_0=T(I1||I2||X);
if &Interaction=1 then f_0=T(I1||I2||X||I2);

if &Interaction=0 then f_1 = T(I1||I1||X);
if &Interaction=1 then f_1 = T(I1||I1||X||X2);

E_opt = E;

*create E_opt, E_opt chooses level with lowest chance of poor outcome.
If the chance of a poor outcome is the same then macro defaults to treatment 1;
do k = 1 to n;
E_opt[k,] = 0;
M1 = exp(T(B)*f_0[,k]);
M2 = 1/(1+exp(T(B)*f_0[,k]));
```

```
M=M1*M2;

W1 = exp(T(B)*f_1[,k]);
W2 = 1/(1+exp(T(B)*f_1[,k]));
W=W1*W2;

if W < M then E_opt[k,]=1;
end;


*X2 may be a subset of X variables;
if &Interaction=1 then EoptX = E_opt#X2;

if &Interaction=0 then f_opt = T(I1||E_opt||X);
if &Interaction=1 then f_opt = T(I1||E_opt||X||EoptX);

avg1 =0;
                        *P_Dopt calculation and IF1 calculation;
                        M1 = exp(T(B)*f_opt);
                        M2 = 1/(1+exp(T(B)*f_opt));
                        M=M1#M2;
                        avg1=M[,:];
                        IF1=M-avg1;
                        IF1=T(IF1);

P_Dopt=avg1;


*IF2 Calculation;

Avg2=0;

ncol = nrow(f);
Temp = I(ncol);
*avg is a matrix of zeros;
Avg3=T(Temp - Temp);

*numeric derivatives;
row_vector = j(1,r,1);
LowB = B*row_vector;
UpB = LowB;

*need these for the numric derivatives later;
do v = 1 to r;
LowB[v,v] = B[v,]-bound[v,];
UpB[v,v] = B[v,]+bound[v,];
bound2[v,]=.5*(1/bound[v,]);
end;

*numeric Derivative loop;
ND1 = B;
ND2 = B;

Do v = 1 to r;
```

```
NM1 = exp(T(LowB[,v])*f_opt);
NM2 = 1/(1+exp(T(LowB[,v])*f_opt));
NM3 = sum(NM1#NM2)/n;
ND1[v,1] = NM3;

NM4 = exp(T(UpB[,v])*f_opt);
                        NM5 = 1/(1+exp(T(UpB[,v])*f_opt));
                        NM6=sum(NM4#NM5)/n;

ND2[v,1]= NM6;
end;


*ND is numeric approximation for partial mu/partial beta;
ND = (ND2 - ND1)#bound2;


Do k = 1 to n;
Q1=f[,k]*T(f[,k]);
                                Q2 = exp(T(B)*f[,k]);
                                Q3 = (1+exp(T(B)*f[,k]))*(1+exp(T(B)*f[,k]));
                                Q3=1/Q3;
                                Q4=(Q2*Q3)*Q1;
                                Avg3 = Avg3 + Q4;
                            end;

                            Avg3 = Avg3/n;
                            Avg3 = inv(Avg3);
                            IF2 = T(ND)*Avg3*f;
                            IF2 = T(IF2);

                            R1 = exp(T(B)*f);
                            R2 = 1/(1+exp(T(B)*f));
                            R=T(R1#R2);
                            IF2 = IF2 # (D-R);

                            *Put all the pieces together to get IF;
                                    IF = ((IF1+IF2)/p_dopt)-IF3;
V = T(IF) * IF;
                                    V = V / ((n-1)*(n-1));
                                    V = sqrt(V);

                    *V is estimate of the standard error of
                    ln(PDstar0=1)-ln(P D=1) now need to put back together;
                    Ratio=P_Dopt/P_D;
                    AB_opt_hat = 1-ratio;
                    ln_ratio =log(ratio);

alpha=&alpha;
z = probit(1-(alpha/2));

                    *Back Transform 95% Confidence Interval;
                    Lower_BT =1-exp(ln_ratio + (z* V));
                    Upper_BT=1-exp(ln_ratio - (z*V));
```

```
                    *Delta Theorem 95% Confidence Interval;
                    Lower_DT = AB_opt_hat - (z*ratio*V);
                    Upper_DT = AB_opt_hat + (z*ratio*V);
                    SE_AB = ratio*V;

            *Output needed values;
Outvar = n || P_Dopt || P_D || AB_opt_hat ||ln_ratio || Lower_BT ||
Upper_BT ||Lower_DT || Upper_DT ||SE_AB ;
append from Outvar;
run;

*******************************************************************************
*
* Start of analysis for output
*
*******************************************************************************;

Data &out;
set out;
                label
Sample_Size = 'Sample Size'
ln_ratio_ = 'Log{P(D)/P(D_opt)}'
                Prob_D_optimal = 'Probability D optimal'
                Prob_Disease = 'Probability of Disease'
                AB_opt_hat = 'AB opt hat'
               Lower_BT = 'Backtransformed Lower 95% C.I.'
               Upper_BT = 'Backtransformed Upper 95% C.I.'
               Lower_DT = 'Delta Thm Lower 95% C.I.'
               Upper_DT = 'Delta Thm Upper 95% C.I.'
Delta_SE = 'Delta Thm Standard Error';

Proc Print data=Out label;
run; quit;
quit;
%MEND;
```