Supplementary Online Appendix

eAppendix A. Derivation of likelihood estimation for causal diagrams presented in Section I.

Figure 1.
Let $\{M_m: m\}, \{N_n: n\}, \{O_o: o\}$ denote the finite support of M, N, and O, respectively. One observes that

$$
\begin{aligned}
P[Y|A,L,S=1] &= \frac{P[Y,S=1|A,L]}{P[S=1|A,L]} \\
&= \frac{\sum_{m,n,o} P[Y,M_m,N_n,O_o,S=1|A,L]}{\sum_{y,m,n,o} P\left[S=1,Y_y,M_m,N_n|A,L\right]} \\
&= \frac{\sum_{m,n,o} P[S=1|Y,M_m,N_n,O_o,A,L] f(M_m,N_n,O_o|Y,A,L) P[Y|A,L]}{\sum_{y,m,n,o} P\left[S=1|Y_y,M_m,N_n,O_o,A,L\right] f\left(M_m,N_n,O_o|Y_y,A,L\right) P\left[Y_y|A,L\right]}
\end{aligned}
$$

Let $\tau(Y,A,L) = \sum_{m,n,o} P[S=1|Y,M_m,N_n,O_o,A,L] f(M_m,N_n,O_o|Y,A,L)$. Therefore

$$
\begin{aligned}
P[Y|A,L,S=1] &= \frac{\tau(Y,A,L)P[Y|A,L]}{\sum_y \tau\left(Y_y,A,L\right) P[Y=y|A,L]} \\
&= \frac{\tau(Y,A,L)\left[\frac{e^{\{y(\alpha+\beta_1 A+\beta_2 L)\}}}{1+e^{\{\alpha+\beta_1 A+\beta_2 L\}}}\right]}{\sum_{y=0}^{1} \tau(y,A,L)\left[\frac{e^{\{y(\alpha+\beta_1 A+\beta_2 L)\}}}{1+e^{\{\alpha+\beta_1 A+\beta_2 L\}}}\right]} \\
&= \frac{\tau(Y,A,L)e^{\{y(\alpha+\beta_1 A+\beta_2 L)\}}}{\sum_{y=0}^{1} \tau(y,A,L)e^{\{y(\alpha+\beta_1 A+\beta_2 L)\}}} \\
&= \frac{\tau(Y,A,L)e^{\{y(\alpha+\beta_1 A+\beta_2 L)\}}}{\tau(y=0,A,L)e^{\{1(\alpha+\beta_1 A+\beta_2 L)\}} + \tau(y=1,A,L)e^{\{0(\alpha+\beta_1 A+\beta_2 L)\}}} \\
&= \frac{\left[\frac{\tau(Y,A,L)}{\tau(y=0,A,L)}\right] e^{\{y(\alpha+\beta_1 A+\beta_2 L)\}}}{\left[\frac{\tau(y=0,A,L)}{\tau(y=0,A,L)}\right] + \left[\frac{\tau(y=1,A,L)}{\tau(y=0,A,L)}\right] e^{\{(\alpha+\beta_1 A+\beta_2 L)\}}} \\
&= \frac{\left[\frac{\tau(y=1,A,L)}{\tau(y=0,A,L)}\right] e^{\{y(\alpha+\beta_1 A+\beta_2 L)\}}}{1 + \left[\frac{\tau(y=1,A,L)}{\tau(y=0,A,L)}\right] e^{\{(\alpha+\beta_1 A+\beta_2 L)\}}}
\end{aligned}
$$

Thus, we conclude $logit\left[P[Y|A,L,S=1]\right] = log\left[\frac{\tau(y=1,A,L)}{\tau(y=0,A,L)}\right] + \alpha + \beta_1 A + \beta_2 L$, indicating that selection bias induces an association between (A,L) and Y if τ depends on A and L, in addition to its dependence on Y.

Below we consider a number of special cases of Figure 1, to illustrate settings where τ induces bias, as well as settings where it does not.

Figure 2.a.
Recall from the detailed derivation provided for Figure 1 that

$$
\begin{aligned}
\tau(Y,A,L) &= \sum_{m,n,o} P[S=1|Y,M_m,N_n,O_o,A,L] f(M_m,N_n,O_o|Y,A,L) \\
&= \sum_{m,n,o} P[S=1|\mathrm{L},M_m,O_o]\, f(M_m|Y,A,L)\, f(O_o) \\
&= \sum_{m,n,o} P[S=1|\mathrm{L},M_m,O_o,Y]\, f(M_m|Y,A,L)\, f(O_o)
\end{aligned}
$$

By the independencies encoded in the DAG. Therefore,

$$
\begin{aligned}
logit\left[P[Y|A,L,S=1]\right] &= log\left[\frac{\sum_{m,n,o} P[S=1|L,M_m,O_o,Y=1]\, f(M_m|Y=1,A,L)\, f(O_o)}{\sum_{m,n,o} P[S=1|L,M_m,O_o,Y=0]\, f(M_m|Y=0,A,L)\, f(O_o)}\right] + \alpha + \beta_1 A + \beta_2 L \\
&= log\left[\frac{\sum_{m,n,o} P[S=1|L,M_m,O_o,Y=1]\, f(M_m|Y=1,A,L)}{\sum_{m,n,o} P[S=1|L,M_m,O_o,Y=0]\, f(M_m|Y=0,A,L)}\right] + \alpha + \beta_1 A + \beta_2 L
\end{aligned}
$$

which cannot be simplified further, and indicates selection bias in the association of (A,L) on Y.

While $P[S=1|L,M_n,O_o,Y=y]$ can be directly computed using the final sampling weights provided to the analyst, information on $f(M_m|Y=y,A,L)$ may not be available. We note that $f(M_m|Y=y,A,L)$ may be estimated using full maximum likelihood. As an alternative, we present a simple approach which utilizes the final sampling probabilities.

Instead, $f(M_m|Y=1,A,L)$ may be estimated using the following two regression models weighted by sampling probabilities and assuming binary M:

$$logit\big[P[M|Y=0,A,L]\big] = \eta_0 + \eta_1 A + \eta_2 L$$
$$logit\big[P[M|Y=1,A,L]\big] = \gamma_0 + \gamma_1 A + \gamma_2 L$$

The predicted value of M setting Y to 0 or 1 can then be used to construct the offset term under the assumption that the association between A and M is constant across levels of L. Note that if A and L are binary or categorical, a saturated model involving all higher order interactions between A and L may be easily fit and the predicted value of M can be computed without any additional assumptions.

Figure 2.b.
Recall from the detailed derivation provided for Figure 1 that

$$\tau(Y,A,L) = \sum_{m,n,o} P[S=1|Y,M_m,N_n,O_o,A,L]f(M_m,N_n,O_o|Y,A,L)$$
$$= \sum_{m,n,o} P[S=1|M_m,N_n]f(M_m|L)f(N_n|A,L)f(O_o)$$

by the independencies encoded in the DAG. Therefore,

$$logit\big[P[Y|A,L,S=1]\big] = log\left[\frac{\sum_{m,n,o} P[S=1|M_m,N_n]\, f(M_m|L)f(N_n|A,L)\, f(O_o)}{\sum_{m,n,o} P[S=1|M_m,N_n]\, f(M_m|L)f(N_n|A,L)\, f(O_o)}\right] + \alpha + \beta_1 A + \beta_2 L$$
$$= \alpha + \beta_1 A + \beta_2 L$$

which does not depend on $\tau$and therefore indicates no selection bias.

Figure 3.
Recall from the detailed derivation provided for Figure 1 that

$$\tau(Y,A,L) = \sum_{m,n,o} P[S=1|Y,M_m,N_n,O_o,A,L]f(M_m,N_n,O_o|Y,A,L)$$
$$= \sum_{m,n,o} P[S=1|M_m,N_n]f(M_m|L)f(N_n|L)f(O_o)$$

by the independencies encoded in the DAG. Therefore,

$$logit\big[P[Y|A,L,S=1]\big] = log\left[\frac{\sum_{m,n,o} P[S=1|M_m,N_n]\, f(M_m|L)f(N_n|L)\, f(O_o)}{\sum_{m,n,o} P[S=1|M_m,N_n]\, f(M_m|L)f(N_n|L)\, f(O_o)}\right] + \alpha + \beta_1 A + \beta_2 L$$
$$= \alpha + \beta_1 A + \beta_2 L$$

which does not depend on $\tau$and therefore indicates no selection bias.

Figure 4.a.
Recall from the detailed derivation provided for Figure 1 that

$$\tau(Y,A,L) = \sum_{m,n,o} P[S=1|Y,M_m,N_n,O_o,A,L]f(M_m,N_n,O_o|Y,A,L)$$

Recall also that in Figure 4.a. we take the exposure to be M. Therefore, $\tau(Y,A,L)$ reduces to

$$\tau(Y,M,L) = \sum_{n,o} P[S=1|N_n,M]f(N_n|L)f(O_o)$$

by the independencies encoded in the DAG. Therefore,

$$logit\big[P[Y|M,L,S=1]\big] = log\left[\frac{\sum_{n,o} P[S=1|N_n,M]\, f(N_n|L)\, f(O_o)}{\sum_{n,o} P[S=1|N_n,M]\, f(N_n|L)\, f(O_o)}\right] + \alpha + \beta_1 A + \beta_2 L$$
$$= \alpha + \beta_1 M + \beta_2 L$$

which does not depend on $\tau$and therefore indicates no selection bias.

Figure 4.b.
Recall from the detailed derivation provided for Figure 1 that

$$\tau(Y,A,L)=\sum_{m,n,o}P[S=1|Y,M_m,N_n,O_o,A,L]f(M_m,N_n,O_o|Y,A,L)$$

Recall also that in Figure 4.b. we take the exposure to be N. Therefore, $\tau(Y,A,L)$ reduces to

$$\tau(Y,N,L)=\sum_{m,o}P[S=1|M_m,N]f(M_m|Y,L)f(O_o)$$

by the independencies encoded in the DAG. Therefore,

$$logit\big[P[Y=1|N,L,S=1]\big]=log\left[\frac{\sum_{m,o}P[S=1|M_m,N]f(M_m|Y=1,L)f(O_o)}{\sum_{m,o}P[S=1|M_m,N]f(M_m|Y=0,L)\,f(O_o)}\right]+\alpha+\beta_1N+\beta_2L$$
$$=log\left[\frac{\sum_m P[S=1|M_m,N]f(M_m|Y=1,L)}{\sum_m P[S=1|M_m,N]f(M_m|Y=0,L)}\right]+\alpha+\beta_1N+\beta_2L$$

which indicates selection bias in the both L and N associations with Y.

While $P[S=1|M_m,N]$, can be directly computed using the final sampling weights provided to the analyst, information on $f(M_m|Y=y,L)$ may not be available. We note that $f(M_m|Y=y,L)$ may be estimated using full maximum likelihood. As an alternative, we present a simple approach which utilizes the final sampling probabilities.

Instead $f(M_m|Y=y,L)$ may be estimated using the following two regression models weighted by sampling probabilities and assuming binary M:

$$logit\big[P[M|Y=0,L]\big]=\eta_0+\eta_1L$$
$$logit\big[P[M|Y=1,L]\big]=\gamma_0+\gamma_1L$$

The predicted values of M can then be used to construct the offset term under the assumption of no model misspecification.

Figure 5.a.
Recall from the detailed derivation provided for Figure 1 that

$$\tau(Y,A,L)=\sum_{m,n,o}P[S=1|Y,M_m,N_n,O_o,A,L]f(M_m,N_n,O_o|Y,A,L)$$

Recall also that in Figure 5.a. we take the outcome to be M. Therefore, $\tau(Y,A,L)$ reduces to

$$\tau(M,A,L)=\sum_{n,o}P[S=1|M,N_n]f(M|A,L)f(N_n|A,L)f(O_o)$$

by the independencies encoded in the DAG. Therefore,

$$logit\big[P[M=1|A,L,S=1]\big]=log\left[\frac{\sum_n P[S=1|M=1,N_n]f(M=1|A,L)f(N_n|A,L)f(O_o)}{\sum_n P[S=1|M=0,N_n](M=0|A,L)f(N_n|A,L)\,f(O_o)}\right]+\alpha+\beta_1A+\beta_2L$$
$$=log\left[\frac{\sum_n P[S=1|M=1,N_n]f(N_n|A,L)}{\sum_n P[S=1|M=0,N_n]f(N_n|A,L)}\right]+\alpha+\beta_1A+\beta_2L$$

which indicates selection bias in both A and L associations with M.

While $P[S=1|M=m,N_n]$ can be directly computed using the final sampling weights provided to the analyst, information on $f(N_n|A,L)$, may not be directly available. We note that $f(N_n|A,L)$ may be estimated using full maximum likelihood. As an alternative, we present a simple approach which utilizes the final sampling probabilities.

Instead $f(N_n|A,L)$, may be estimated using the following regression model weighted by sampling probabilities and assuming binary N:

$$logit\big[P[N|A,L]\big]=\eta_0+\eta_1A+\eta_2L$$

The predicted value of N can then be used to construct the offset term under the assumption that the association between A and N is constant across levels of L. Note that if A and L are binary or categorical, a saturated model involving all higher order interactions between A and L may be easily fit and the predicted value of N can be computed without any additional assumptions.

Figure 5.b.
Recall from the detailed derivation provided for Figure 1 that

$$\tau(Y,A,L) = \sum_{m,n,o} P[S=1|Y, M_m, N_n, O_o, A, L] f(M_m, N_n, O_o|Y, A, L)$$

Recall also that in Figure 5.b. we take the outcome to be N. Therefore, $\tau(Y,A,L)$ reduces to

$$\tau(N,A,L) = \sum_{m,o} P[S=1|M_m, N] f(M_m|L) f(N|A,L) f(O_o)$$

by the independencies encoded in the DAG. Therefore,

$$logit\big[P[N=1|A,L,S=1]\big] = log\left[\frac{\sum_m P[S=1|M_m, N=1] f(M_m|L) f(N=1|A,L) f(O_o)}{\sum_m P[S=1|M_m, N=0] f(M_m|L) f(N=0|A,L)\, f(O_o)}\right] + \alpha + \beta_1 A + \beta_2$$

$$= log\left[\frac{\sum_m P[S=1|M_m, N=1] f(M_m|L)}{\sum_m P[S=1|M_m, N=0] f(M_m|L)}\right] + \alpha + \beta_1 A + \beta_2 L$$

which indicates selection bias in the L-N association.

While $P[S=1|M_m, N=n]$, can be computed using the final sampling weights provided to the analyst, information on $f(M_m|L)$ may not be directly available. We note that $f(M_m|L)$ may be estimated using full maximum likelihood. As an alternative, we present a simple approach which utilizes the final sampling probabilities.

Instead $f(M_m|L)$ may be estimated using the following regression model assuming binary M:

$$logit\big[P[M|L]\big] = \eta_0 + \eta_1 L$$

The predicted values of M can then be used to construct the offset term under the assumption of no model misspecification.

Figure 6.
Recall from the detailed derivation provided for Figure 1 that

$$\tau(Y,A,L) = \sum_{m,n,o} P[S=1|Y, M_m, N_n, O_o, A, L] f(M_m, N_n, O_o|Y, A, L)$$

Recall also that in Figure 6 we take the exposure to be M and the outcome to be N. Therefore, $\tau(Y,A,L)$ reduces to

$$\tau(N,M,L) = \sum_{o} P[S=1|M, N=1] f(M|L) f(O_o)$$

by the independencies encoded in the DAG. Therefore,

$$\begin{aligned} logit\big[P[N=1|M,L,S=1]\big] &= log\left[\frac{P[S=1|M,N=1] f(M|L) f(O_o)}{P[S=1|M,N=0] f(M|L) f(O_o)}\right] + \alpha + \beta_1 A + \beta_2 L \\ &= log\left[\frac{P[S=1|M,N=1] f(M|L)}{P[S=1|M,N=0] f(M|L)}\right] + \alpha + \beta_1 M + \beta_2 L \\ &= log\left[\frac{P[S=1|M,N=1]}{P[S=1|M,N=0]}\right] + \alpha + \beta_1 M + \beta_2 L \end{aligned}$$

which indicates selection bias in the M-N association.

eAppendix B. SAS code used in the simulation study

```
%macro sim_select(q,c_strength,alpha,beta1,beta2,beta3);

data simulate&q.;

*** specify number and distribution of categories for m ***;
num_mcat = 3;
c1 = 1/num_mcat; c2 = 1/num_mcat; c3 = 1/num_mcat;

        *** create m, n, and o variables (i.e. determinants of selection) ***;
        do i = 1 to 40000;
                m_cont = round(uniform(0)*100);

                *** distribute continuous m evenly into 3 categories ***;
                m_cat = (m_cont ge 0)
                        + (m_cont gt c1*100)
                        + (m_cont gt (c1 + c2)*100)
                        + (m_cont gt (c1 + c2 + c3)*100);

                        m_cat1 = (m_cat = 1);
                        m_cat2 = (m_cat = 2);
                        m_cat3 = (m_cat = 3);

                *** create o variable and confounder ***;
                o_binary = rand('bernoulli',0.25);
                conf = (log(&c_strength.)*m_cat1 + rand('normal'));

                *** create n variable based on the following individual risk model:
                        logit(P[N=1|M,L])=α+β1*M1+β2*M2+β3*L ***;
                linpred = &alpha. + &beta1.*m_cat1 + &beta2.*m_cat2 + &beta3.*conf;
                prob = exp(linpred)/ (1 + exp(linpred));
                n_binary = rand('bernoulli',prob);

                *** create m-n-o indicator ***;
                mno_cat = m_cat*100 + n_binary*10 + o_binary;
        output;
        end;
run;

*** create allocation proportions for input into proc surveyselect ***;
data samp_prob_mno; set simulate&q.;
        _alloc_ = .;
                if mno_cat = 100 then _alloc_ = 0.2;
                if mno_cat = 200 then _alloc_ = 0.01;
                if mno_cat = 300 then _alloc_ = 0.19;
                if mno_cat = 110 then _alloc_ = 0.1;
                if mno_cat = 210 then _alloc_ = 0.05;
                if mno_cat = 310 then _alloc_ = 0.05;
                if mno_cat = 101 then _alloc_ = 0.05;
                if mno_cat = 201 then _alloc_ = 0.0025;
                if mno_cat = 301 then _alloc_ = 0.0475;
                if mno_cat = 111 then _alloc_ = 0.15;
                if mno_cat = 211 then _alloc_ = 0.075;
                if mno_cat = 311 then _alloc_ = 0.075;
        keep m_cat n_binary o_binary mno_cat conf _alloc_;
run;

*** select 1% sub-sample (n=400) according to m and n ***;
proc sort data = samp_prob_mno; by mno_cat; run;
proc sort data = simulate&q.; by mno_cat; run;
proc surveyselect data = simulate&q.
        out = simulate_svy_mno&q.
        sampsize = 400;
        strata mno_cat /alloc = samp_prob_mno;
run;

*** obtain selection probability (i.e. tau) ***;
proc sort data = simulate_svy_mno&q.; by mno_cat;
```

```
proc means data = simulate_svy_mno&q.;
      by mno_cat;
      var selectionprob;
      ods output summary = offsets&q. (keep = mno_cat SelectionProb_Mean);
run;

*** add m, n, and o variables to selection probabilities file ***;
data offsets&q.; set offsets&q.;
      if mno_cat = 100 or mno_cat = 200 or mno_cat = 300 or
         mno_cat = 101 or mno_cat = 201 or mno_cat = 301 then n_binary = 0;
      if mno_cat = 110 or mno_cat = 210 or mno_cat = 310 or
         mno_cat = 111 or mno_cat = 211 or mno_cat = 311 then n_binary = 1;
      if mno_cat = 100 or mno_cat = 200 or mno_cat = 300 or
         mno_cat = 110 or mno_cat = 210 or mno_cat = 310 then o_binary = 0;
      if mno_cat = 111 or mno_cat = 211 or mno_cat = 311 or
         mno_cat = 101 or mno_cat = 201 or mno_cat = 301 then o_binary = 1;
      m_cat = (mno_cat - n_binary*10 - o_binary)/100;
run;

proc sort data = offsets&q.; by m_cat o_binary; run;
proc transpose data = offsets&q. out = offsets&q. prefix = SelectionProbN;
      by m_cat o_binary;
      id n_binary;
      var SelectionProb_Mean;
run;

*** merge selection probability file with simulated survey data ***;
proc sort data = offsets&q.; by m_cat o_binary; run;
proc sort data = simulate_svy_mno&q.; by m_cat o_binary; run;
data simulate_svy_mno&q.;
      merge offsets&q. simulate_svy_mno&q.;
            by m_cat o_binary;

      *** create offset term from ratio of selection probabilities (i.e. tau) ***;
      offset = log(SelectionProbN1/SelectionProbN0);
      m_cat1 = (m_cat = 1);
      m_cat2 = (m_cat = 2);
      m_cat3 = (m_cat = 3);

      id = _n_;

      keep id n_binary m_cont m_cat o_binary samplingweight offset m_cat1 m_cat2 m_cat3 conf;
run;

*******************************************************************************;
*** estimate alpha and beta coeffients from logistic regression models
            with and without adjustment for selection ***;
*******************************************************************************;
*** no adjustment ***;
proc logistic descending data = simulate_svy_mno&q.;
      model n_binary = m_cat1 m_cat2 conf;
      ods output ParameterEstimates = noadjust (keep = Variable Estimate StdErr);
run;
data noadjust_betas&t.; set noadjust_betas&t.noadjust; run;

*** adjust via unweighted conditional regression ***;
proc logistic descending data = simulate_svy_mno&q.;
      model n_binary = m_cat1 m_cat2 conf o_binary;
      ods output ParameterEstimates = condition (keep = Variable Estimate StdErr);
run;
data condition_betas&t.; set condition_betas&t.condition; run;

*** adjust via weighted unconditional regression ***;
proc genmod descending data = simulate_svy_mno&q.;
      class id;
      weight samplingweight;
      model n_binary = m_cat1 m_cat2 conf/ link=logit dist=binomial;
      repeated subject=id / type=ind;
      ods output GEEEmpPEst = ipw (keep = Parm Estimate UpperCL LowerCL);
run;
```

```
data ipw_betas&t.; set ipw_betas&t.ipw; run;

*** adjust via maximum likelihood ***;
proc logistic descending data = simulate_svy_mno&q.;
	model n_binary = m_cat1 m_cat2 conf /offset = offset;
	ods output ParameterEstimates = like (keep = Variable Estimate StdErr);
run;
data like_betas&t.; set like_betas&t.like; run;

%mend sim_select;

%macro looper(t,c_strength,alpha,beta1,beta2,beta3);

*** create empty data sets to store alpha and beta coefficients ***;
data noadjust_betas&t.; set _null_; run;
data condition_betas&t.; set _null_; run;
data ipw_betas&t.; set _null_; run;
data like_betas&t.; set _null_; run;

*** conduct 1,000 simulations ***;
%do q = 1 %to 1000;
	%sim_select(&q.,&c_strength.,&alpha.,&beta1.,&beta2.,&beta3.);
	proc datasets library = work;
		delete
			simulate&q.
			simulate_svy_mno&q.
			offsets&q.;
	run;
%end;

%mend looper;

*********************************************************************************;
*** run simulation under 8 model specifications defined by:
		a) weak and strong exposure effects
			(β1=0.1,β2=0.4 and β1=0.4,β2=0.8, respectively),
		b) weak and strong confounding effects
			(β3=0.4 and β3=0.8, respectively), and
		c) weak and strong associations between exposure and confounder
			(OR(E-C)=1.1 and OR(E-C)=1.6, respectively)
	with all assuming a 2% marginal disease prevalence(α=-3.9) ***;
*********************************************************************************;
%looper(1,1.1,-3.9,0.1,0.4,0.4);
%looper(2,1.1,-3.9,0.1,0.4,0.8);
%looper(3,1.1,-3.9,0.2,0.8,0.4);
%looper(4,1.1,-3.9,0.2,0.8,0.8);
%looper(5,1.6,-3.9,0.1,0.4,0.4);
%looper(6,1.6,-3.9,0.1,0.4,0.8);
%looper(7,1.6,-3.9,0.2,0.8,0.4);
%looper(8,1.6,-3.9,0.2,0.8,0.8);
```