eAppendix1

Definition of sufficient-cause related and attributable-fraction indices:

1) The completion potential (CP) index for a class of sufficient causes is defined as the ratio of the completion rate of that particular class and that of the all-unknown class. (CP for the all-unknown class is 1.0 by definition.)
2) The individual-based causal-pie weight (CPW) index is the probability that a diseased individual with a particular risk factor profile had developed the disease because of the completion of a specific class of sufficient causes.
3) The population-wide CPW is the average of the individual-based CPWs of all the diseased individuals in a population to present the probability that a randomly chosen diseased individual with a particular risk factor profile had developed the disease because of the completion of a specific class of sufficient causes.
4) Attributable fraction among the exposed (AFE) or population attributable fraction (PAF) is the proportion of diseased individuals that would not have occurred, if an intervention conducted among the exposed or in the study population.

eAppendix2

1. SAS code for sufficient cause modeling with a 1:M matched case-control data

****A. data preparation: transforming original data (cscn) to a matching structure (fornlp)**

<table>
<tr><td>Original data (observation): cscn</td><td>Transformed data (matching set): fornlp</td></tr>
<tr><td>For example,
<table>
<tr><th>ID</th><th>case</th><th>set</th><th>X</th></tr>
<tr><td>001</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>002</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>003</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>004</td><td>0</td><td>2</td><td>1</td></tr>
<tr><td>005</td><td>0</td><td>2</td><td>0</td></tr>
</table>
</td><td>
<table>
<tr><th>set</th><th>_ncases:
(no of cases)</th><th>_ntot:
(total subjects per set)</th><th>z1:
(X of case)</th><th>z2:
(X of control1)</th><th>z3:
(X of control2)</th></tr>
<tr><td>1</td><td>1</td><td>2</td><td>0</td><td>1</td><td>.</td></tr>
<tr><td>2</td><td>1</td><td>3</td><td>0</td><td>1</td><td>0</td></tr>
</table>
</td></tr>
<tr><td colspan="2">The SAS code for automatic transformation can be downloaded as ‘make_case_control.sas’ (see ref 8 and http://aje.oxfordjournals.org/content/171/3/377.full.pdf+html)

%include 'D:\ make_case_control.sas';
%make_case_control(indsn_rs=cscn,setno=set,cc=case, vlist=X,outdsn=fornlp);

/*indsn_rs: original data; setno: matching set variable; cc: case status variable; vlist: exposure variable; outdsn: transformed data*/</td></tr>
</table>

****B. model fitting: using dataset 'fornlp' to obtain point estimates for regression coefficient, CP, iCPW, pCPW, PAF**

i. Definition of variables (bold-type words)

Risk-factor profiles: **X**	No of cases: **count**	Individual-based CPW (%)		PAF
		All-unknown	X	
0		**iCPW11**	**iCPW12**	**PAF1**
1		**iCPW21**	**iCPW22**	**PAF2**
Completion potential for X: **CP** Completion potential for All-unknown is set as 1.0. Population-wide CPW (%) for All-unknown: **pCPWU** Population-wide CPW (%) for X: **pCPWX** Intervention for the elimination of X (targeting to X=0): **PAF1** No intervention (targeting to X=1): **PAF2**				

ii. SAS macro: substitute the codes in ***bold italic*** word type when implementing.

```
%macro cpindex (obsdata=, setdata=,csdata=,EB=,CP=, estimate1=, estimate2=, estimate3=);
/*obsdata: original data; setdata: transformed data; csdata: original data with only case subjects; EB: regression coefficient estimates; CP: completion potential estimates; estimate1: iCPW estimates; estimate2: pCPW estimates; estimate3: PAF estimates*/
```

```
/*point estimate of regression coefficient beta1 for X: EB*/
ods trace on;
  ods output ParameterEstimates=pp1;
     proc nlp data=&setdata cov=2 pcov;          /*pcov: present the variance-covariance matrix*/
       parms beta1=0;                            /*define initial values for iteration for all regression coefficients fitted in the following model*/
       array z[1,3] z1-z3;                       /* z[a,b]: a= no. of X, b: max(_ntot); a=1, b=3 in the example*/
       sum = 0; _ntot = _ntot;
    do j = _ntot to 1 by -1;
       phi = 1+z{1,j}*beta1;                      /*modeling with only main effect of X: z{1,j}*/
       sum=sum + phi;
      end;
  L = log(phi / sum);                            /*conditional likelihood formula*/
  max L;
  run;
ods trace off;

data out1;                                       /* saved beta1 estimate in dataset out1*/
  set pp1 (KEEP=estimate appstderr);
    if appstderr='.' then delete;
run;

/*create dataset with case subjects only (csdata) and calculate frequency of X status (count)*/
  data &csdata;
```

```
    set &obsdata;
      if case=0 then delete;
        X1=put (X, 9.);
    proc freq ;
      tables X1/nopercent norow out=FreqCount1 sparse;
  run;

/*point estimate of CP, iCPW, pCPW, PAF*/
    proc iml;
      reset print;
    use out1;
    read all into para [colname=varNames];
      CP=para[, 'estimate'];
      EB=para[ , 'estimate'];
    use FreqCount1;
    read all into cscount [colname=varNames];
      count=cscount[, 'count'];
      iCPW11=100;
      iCPW12=0;
      iCPW21=100/ (CP[1,1]+1);
      iCPW22=100-iCPW21;
      pCPWU=(count[1,1]#iCPW11+count[2,1]#iCPW21)/ (count[+ ,]);
      pCPWX=(count[1,1]#iCPW12+count[2,1]#iCPW22)/ (count[+ ,]);
```

```
    PAF1=pCPWX;
    PAF2=0;
  iCPW=iCPW11 || iCPW12 || iCPW21 || iCPW22;
  pCPW=pCPWU || pCPWX;
  PAF=PAF1 || PAF2;

  nameEB={'EB'};
    create &EB from EB [colname=nameEB];
    append from EB;
  nameCP={'CP'};
    create &CP from CP [colname=nameCP];
    append from CP;
  nameicpw={'iCPW11' 'iCPW12' 'iCPW21' 'iCPW22'};
    create & estimate1 from iCPW [colname=nameicpw];
    append from iCPW;
  namepcpw={'pCPWU' 'pCPWX'};
    create &estimate2 from pCPW [colname=namepcpw];
    append from pCPW;
  namePAF={'PAF1' 'PAF2'};
    create &estimate3 from PAF [colname=namePAF];
    append from PAF;
  quit;
%mend;
```

%*cpindex* (obsdata=cscn, setdata=fornlp, csdata=csdata,EB=EB, CP=CP, estimate1=iCPW,estimate2=pCPW,estimate3=PAF);

****C. model fitting: bootstrapped 95% CI**

/*create dataset with X status*/

data category;
input X $ @@;
cards;
0 1
;
run;

/*bootstrapped 95% CI*/

%macro ci (setdata=,csdata=, N=, NumTrials=, bciEB=, bciCP=, bestimate1=, bestimate2=, bestimate3=, bciicpw=, bcipcpw=,bciPAF=);
/*setdata: transformed data; csdata: original data with only case subjects; N: number of bootstrapping; NumTrials: number of cases; bciEB: 95% CIs for regression coefficients; bciCP: 95% CIs for CPs; bestimate1: bootstrapped values for iCPWs; bestimate2: bootstrapped values for pCPWs; bestimate3: bootstrapped values for PAFs; bciicpw: 95% CIs for iCPWs; bcipcpw: 95% CIs for pCPWs; bciPAF: 95% CIs for PAFs)

/*present variance-covariance matrix*/
ods output CovMat=VarB;
proc nlp data=&setdata cov=2 pcov;

```
      parms beta1=0;
      array z[1,3] z1-z3;
      sum = 0; _ntot = _ntot;
      do j = _ntot to 1 by -1;
         phi = 1+z{1,j}*beta1;
        sum=sum + phi;
      end;
     L = log(phi / sum);
     max L;
     run;
ods trace off;

  /*delta method*/
    proc iml;
     use EB;
     read all into EB [colname=varNames];
     use VarB;
     read all into VarB [colname=varNames];
        logEB=log(EB);
        dEB=1/EB;
        VarlogB=dEB # VarB # dEB`;
     create mean from logEB;
     append from logEB;
```

```
    create vcov from VarlogB;
    append from VarlogB;
  quit;

%do i=1 %to &N;

  /*random sampling of regression coefficients from log-normal distribution*/
  %inc 'D:\mvn1.sas';
     %mvn1 (varcov=vcov , means=mean ,n=1 , sample=bootbeta);
  run;
       /* 'mvn1.sas' is modified from 'mvn.sas' (downloaded at http://support.sas.com/kb/25/addl/fusion25008_1_mvn.sas.txt), taking exponential value
          for output samples of mvn.sas*/

  /*random sampling of case subjects: output as rscase*/
 proc surveyselect data=&csdata method=urs sampsize=&NumTrials rep=1 seed=0 out=rscase;
 run;

  data rscase1;
  set rscase;
    do i=1 to numberhits;
      output;
    end;
  drop i;
```

```
run;

data bFreqCount;
set recase1 category;
proc freq;
tables X / out=bFreqCount1 sparse;
run;

 proc iml;
 reset print;
  use bootbeta;
  read all into bootbeta [colname=varNames];
     bEB=bootbeta`;
     bCP=bootbeta`;
  use bFreqCount1;
  read all into bcscount [colname=varNames];
     bcount=bcscount[, 'count'];
     biCPW11=100;
     biCPW12=0;
     biCPW21=100/(bCP[1,1]+1);
     biCPW22=100-biCPW21;
     bpCPWX=(bcount[1,1]#biCPW12+bcount[2,1]#biCPW22)/(bcount[+ ,]);
     bpCPWU=(bcount[1,1]#biCPW11+bcount[2,1]#biCPW21)/(bcount[+ ,]);
```

```
        bPAF1=bpCPWX;
        bPAF2=0;
        biCPW=biCPW11 || biCPW12 || biCPW21 || biCPW22;
        bpCPW=bpCPWU || bpCPWX;
        bPAF=bPAF1 || bPAF2;
     namebEB={'bEB'};
     create booteb from bootbeta [colname=namebEB];
     append from bootbeta;
   namebCP={'bCP'};
     create bootcp from bootbeta [colname=namebCP];
     append from bootbeta;
   namebicpw={'biCPW11' 'biCPW12' 'biCPW21' 'biCPW22'};
   create estimate1 from biCPW [colname=namebicpw];
   append from biCPW;
   namebpcpw={'bpCPWU' 'bpCPWX'};
   create estimate2 from bpCPW [colname=namebpcpw];
   append from bpCPW;
   namebPAF={'bPAF1' 'bPAF2'};
   create estimate3 from bPAF [colname=namebPAF];
   append from bPAF;
quit;

     proc append base=bEB       data=booteb;
```

```
  run;
    proc append base=bCP        data=bootcp;
  run;
    proc append base=&bestimate1        data=estimate1;
  run;
    proc append base=&bestimate2        data=estimate2;
  run;
    proc append base=&bestimate3        data=estimate3;
  run;
%end;

 data kk1;
 set bEB;
  proc univariate;
    var bEB;
    output out=&bciEB pctlpre=p1_ pctlpts=2.5,97.5;        /*95%CI for beta coefficient*/
 run;

 data kk2;
 set bCP;
  proc univariate;
    var bCP;
    output out=&bciCP pctlpre=p1_ pctlpts=2.5,97.5;      /*95%CI for CP index*/
```

```
  run;

data kk3;
set &bestimate1;
  proc univariate;
    var biCPW11 biCPW12 biCPW21 biCPW22;
    output out=&bciicpw pctlpre=p1 p2 p3 p4 _ pctlpts=2.5,97.5;          /*95%CI for iCPW*/
run;

data kk4;
set &bestimate2;
  proc univariate;
    var bpCPWU bpCPWX;
    output out=&bcipcpw pctlpre=p1 p2 _ pctlpts=2.5,97.5;          /*95%CI for pCPW*/
run;

data kk5;
set &bestimate3;
  proc univariate;
  var bPAF1 bPAF2;
  output out=&bciPAF pctlpre=p1 p2_ pctlpts=2.5,97.5;          /*95%CI for PAF*/
run;
%mend;
```

%*ci* (setdata=fornlp,csdata=csdata, N=10000, NumTrials=***2***, bciEB=ciEB, bciCP=ciCP, bestimate1=biCPW, bestimate2=bpCPW, bestimate3=bPAF, bciicpw=bciicpw, bcipcpw=bcipcpw, bciPAF=bciPAF);

2. SAS code for sufficient cause modeling with a survival data

****A. data preparation**

i. create risk sets

Follow-up data: **survival**

ID	**death**	**time**	**X1**	**X2**	**X3**
001	1	1	0	0	0
002	1	1	1	0	1
003	1	2	0	0	1
004	0	2	1	1	1
005	0	2	0	1	1

SAS codes for density sampling

/*in the example, number of death (no_death):**3**; input data (data): survival; output data (out): **cscn**

```
%macro riskset (no_death=, data=,out=);
  %do i=1 %to &no_death;
      proc sort data=&data;
      by descending death time;
      run;
    data death_&i;
    set &data;
        if _n_=&i then call symput ('m_time', time);
        run;
    data cscn_&i;    /*using variable ‘case’ for case/control status in risk sets*/
     set death_&i;
```

death (1 for failure, 0 for censor);
time (survival time);
X1, X2, X3 (exposures)
	if time>&m_time then case=0;
	else if death =1 & (time=&m_time) then case=1;
	else if death =0 & (time=&m_time) then case=0;
	else if time<&m_time then delete;
	strata=&i;
	run;
	proc append base=&out data=cscn_&i;
	run;
	%end;
	%mend riskset;
	%riskset (no_death =**3**, data=survival ,out=cscn);

ii. create matching set data

Original data (observation): **cscn**

For example,

ID	**case**	**set**	**X1**	**X2**	**X3**
001	1	1	0	0	0
002	1	1	1	0	1
003	0	1	0	0	1
004	0	1	1	1	1
005	0	1	0	1	1
003	1	2	0	0	1

Transformed data (matching set): **fornlp**

set	**_ncases**	**_ntot**	**z1**	**z2**	**z3**	**z4**	**z5**	**z6**	**z7**	**z8**	**z9**	**z10**	**z11**	**…**	**z15**
1	2	5	0	1	0	1	0	0	0	0	1	1	0	...	1
2	1	3	0	1	0	.	.	0	1	1	.	.	1	...	.

/* z1-z5: X1 status for cases (former) and then controls (latter);

004 0 2 1 1 1 005 0 2 0 1 1	z6-z10: X2 status for cases and then controls; z11-z15: X3 status for cases and then controls.*/
%include 'D:\ make_case_control.sas'; %make_case_control(indsn_rs=**cscn**,setno=**set**,cc=**case**, vlist=**X1 X2 X3**,outdsn=**fornlp**);	

****B. model fitting: point estimate**

i. Definition of variables (bold-type words)

Risk-factor profiles			Category of risk-factor profiles: **a1**	No of cases: **count**	Individual-based CPW (%)			PAF
X1	**X2**	**X3**			All-unknown	X2	X1 and X3	
0	0	0	1		**iCPW11**	**iCPW12**	**iCPW13**	**PAF1**
1	0	0	2		**iCPW21**	**iCPW22**	**iCPW23**	**PAF2**
0	1	0	3		**iCPW31**	**iCPW32**	**iCPW33**	**PAF3**
0	0	1	4		**iCPW41**	**iCPW42**	**iCPW43**	**PAF4**
1	1	0	5		**iCPW51**	**iCPW52**	**iCPW53**	**PAF5**
1	0	1	6		**iCPW61**	**iCPW62**	**iCPW63**	**PAF6**
0	1	1	7		**iCPW71**	**iCPW72**	**iCPW73**	**PAF7**
1	1	1	8		**iCPW81**	**iCPW82**	**iCPW83**	**PAF8**

Completion potential for X2: **CP1** Completion potential for X1 and X3: **CP2** Completion potential for All-unknown: 1.0 Population-wide CPW (%) for All-unknown: **pCPWU** Population-wide CPW (%) for X2: **pCPWX2** Population-wide CPW (%) for X1 and X3: **pCPWX1X3** Intervention for the elimination of X1, X2, X3 (targeting to X1=X2=X3=0): **PAF1**; **i**ntervention for the elimination of X2, X3 (targeting to X2=X3=0): **PAF2**; etc.

ii. SAS macro: only substitute the codes in ***bold italic*** word type when implementing.

```
%macro cpindex (obsdata=, setdata=,csdata=,EB=,CP=, estimate1=, estimate2=, estimate3=);

  /*point estimate of regression coefficient beta2 (for main effect X2) and beta13 (for interaction between X1 and X3): EB*/

   /*modeling for multiple cases*/
     ods trace on;
      ods output    ParameterEstimates=pp1;
         proc nlp data=&setdata cov=2 pcov;
           parms beta2=0, beta13=0;
           profile beta2 beta13 / alpha=0.05;
           array z[3,5] z1-z15;
     * Breslow likelihood;
```

```
            caseprod = 1; _ncases=_ncases;
               do j = 1 to _ncases;
                  caseprod = caseprod*(1+z{2,j}*beta2+z{1,j}*z{3,j}*beta13);       /* z{2,j}: main effect of X2; z{1,j}*z{3,j}:product term*/
               end;
            sum=0; _ntot=_ntot;
              do j = 1 to _ntot;
                 sum = sum + (1+z{2,j}*beta2+z{1,j}*z{3,j}*beta13);
            end;
        L = log(caseprod / sum**_ncases) ;
      max L;
      run;
    ods trace off;

/*create dataset with case subjects and calculate frequency of risk-factor profiles*/
      data out1;
        set pp1 (KEEP=estimate appstderr);
            if appstderr='.' then delete;
        run;

      data &csdata;
        set &obsdata;
        if death=0 then delete;
        proc freq ;
```

```
  tables a1/nopercent norow out=FreqCount1 sparse;
 run;

/*point estimate of CP, iCPW, pCPW, PAF*/
 proc iml;
  reset print;
    use out1;
   read all into para [colname=varNames];
   CP=para[, 'estimate'];
   EB=para[ , 'estimate'];
    use FreqCount1;
    read all into cscount [colname=varNames];
    count=cscount[, 'count'];

     iCPW11=100;   iCPW12=0;   iCPW13=0;
     iCPW21=100;   iCPW22=0;   iCPW23=0;
     iCPW31=100/(CP[1,1]+1);   iCPW32=100-iCPW31;   iCPW33=0;
     iCPW41=100;   iCPW42=0;   iCPW43=0;
     iCPW51=100/(CP[1,1]+1);   iCPW52=100-iCPW51;   iCPW53=0;
     iCPW61=100/(CP[2,1]+1);   iCPW62=0;   iCPW63=100-iCPW61;
     iCPW71=100/(CP[1,1]+1);   iCPW72=100-iCPW71;   iCPW73=0;
     iCPW81=100/(CP[1,1]+CP[2,1]+1);   iCPW82=CP[1,1]#iCPW81;   iCPW83=CP[2,1]#iCPW81;
```

```
pCPWU=(count[1,1]#iCPW11+count[2,1]#iCPW21+count[3,1]#iCPW31+count[4,1]#iCPW41+count[5,1]#iCPW51
        +count[6,1]#iCPW61+count[7,1]#iCPW71+count[8,1]#iCPW81)/ (count[+ ,]);
pCPWX2=(count[1,1]#iCPW12+count[2,1]#iCPW22+count[3,1]#iCPW32+count[4,1]#iCPW42+count[5,1]#iCPW52
        +count[6,1]#iCPW62+count[7,1]#iCPW72+count[8,1]#iCPW82)/ (count[+ ,]);
pCPWX1X3=(count[1,1]#iCPW13+count[2,1]#iCPW23+count[3,1]#iCPW33+count[4,1]#iCPW43+count[5,1]#iCPW53
        +count[6,1]#iCPW63+count[7,1]#iCPW73+count[8,1]#iCPW83)/ (count[+ ,]);

PAF1=pCPWX2+pCPWX1X3;
PAF2=pCPWX2+pCPWX1X3;
PAF3=pCPWX1X3;
PAF4=pCPWX2+pCPWX1X3;
PAF5=pCPWX1X3;
PAF6=pCPWX2;
PAF7=pCPWX1X3;
PAF8=0;

iCPW=iCPW11 || iCPW12 || iCPW13 || iCPW21 || iCPW22 || iCPW23 || iCPW31 || iCPW32 || iCPW33 || iCPW41 || iCPW42 || iCPW43 || iCPW51 ||
iCPW52 || iCPW53 || iCPW61 || iCPW62 || iCPW63 || iCPW71 || iCPW72 || iCPW73 || iCPW81 || iCPW82 || iCPW83;
pCPW=pCPWU || pCPWX2 || pCPWX1X3;
PAF=PAF1 || PAF2 || PAF3 || PAF4 || PAF5 || PAF6 || PAF7 || PAF8;

nameEB={'EB'};
create &EB from EB [colname=nameEB];
```

```
append from EB;
nameCP={'CP'};
create &CP from CP [colname=nameCP];
append from CP;
nameicpw={'iCPW11' 'iCPW12' 'iCPW13' 'iCPW21' 'iCPW22' 'iCPW23' 'iCPW31' 'iCPW32' 'iCPW33' 'iCPW41' 'iCPW42' 'iCPW43' 'iCPW51'
'iCPW52' 'iCPW53' 'iCPW61' 'iCPW62' 'iCPW63' 'iCPW71' 'iCPW72' 'iCPW73' 'iCPW81' 'iCPW82' 'iCPW83'};
create &estimate1 from iCPW [colname=nameicpw];
append from iCPW;
namepcpw={'pCPWU' 'pCPWX2' 'pCPWX1X3'};
create &estimate2 from pCPW [colname=namepcpw];
append from pCPW;
namePAF={'PAF1' 'PAF2' 'PAF3' 'PAF4' 'PAF5' 'PAF6' 'PAF7' 'PAF8'};
create &estimate3 from PAF [colname=namePAF];
append from PAF;
quit;
%mend;
```

%*cpindex* (obsdata=survival, setdata=fornlp, csdata=csdata,EB=EB, CP=CP, estimate1=iCPW, estimate2=pCPW, estimate3=PAF);

****C. model fitting: bootstrapped 95% CI**

```
/*create dataset with category of risk-factor profiles*/
```

```
        data category;
        input a1 $ @@;
        cards;
        1 2 3 4 5 6 7 8
        ;
        run;

   /*bootstrapped 95%CI*/

%macro ci (setdata=,csdata=, N=, NumTrials=, bciEB=, bciCP=, bestimate1=, bestimate2=, bestimate3=, bciicpw=, bcipcpw=,bciPAF=);

   /*present variance-covariance matrix*/
       ods trace on;
        ods output    CovMat=VarB;
           proc nlp data=&setdata cov=2 pcov;
             parms beta2=0, beta13=0;
             profile beta2 beta13 / alpha=0.05;
             array z[3,5] z1-z15;
       * Breslow likelihood;
             caseprod = 1; _ncases=_ncases;
                do j = 1 to _ncases;
                   caseprod = caseprod*(1+z{2,j}*beta2+z{1,j}*z{3,j}*beta13);
                end;
```

```
            sum=0; _ntot=_ntot;
              do j = 1 to _ntot;
                 sum = sum + (1+z{2,j}*beta2+z{1,j}*z{3,j}*beta13);
            end;
        L = log(caseprod / sum**_ncases) ;
    max L;
    run;
  ods trace off;

********the following three steps are all with the same SAS codes presented in part 1C**********
 /*delta method*/
 /*random sampling of regression coefficients from log-normal distribution*/
 /*random sampling of case subjects*/
**********************************************************************************


  proc iml;
    reset print;
    use bootbeta;
    read all into bootbeta [colname=varNames];
      bEB=bootbeta`;
      bCP=bootbeta`;
    use bFreqCount1;
```

```
read all into bcscount [colname=varNames];
    bcount=bcscount[, 'count'];
        biCPW11=100;   biCPW12=0;   biCPW13=0;
        biCPW21=100;   biCPW22=0;   biCPW23=0;
        biCPW31=100/(bCP[1,1]+1);   biCPW32=100-biCPW31;   biCPW33=0;
       biCPW41=100;   biCPW42=0;   biCPW43=0;
       biCPW51=100/(bCP[1,1]+1);   biCPW52=100-biCPW51;   biCPW53=0;
       biCPW61=100/(bCP[2,1]+1);   biCPW62=0;   biCPW63=100-biCPW61;
       biCPW71=100/(bCP[1,1]+1);   biCPW72=100-biCPW71;   biCPW73=0;
       biCPW81=100/(bCP[1,1]+bCP[2,1]+1);   biCPW82=bCP[1,1]*biCPW81;   biCPW83=bCP[2,1]*biCPW81;

bpCPWU=(bcount[1,1]*biCPW11+bcount[2,1]*biCPW21+bcount[3,1]*biCPW31+bcount[4,1]*biCPW41+bcount[5,1]*biCPW51
          +bcount[6,1]*biCPW61+bcount[7,1]*biCPW71+bcount[8,1]*biCPW81)/ (bcount[+ ,]);
bpCPWX2=(bcount[1,1]*biCPW12+bcount[2,1]*biCPW22+bcount[3,1]*biCPW32+bcount[4,1]*biCPW42+bcount[5,1]*biCPW52
          +bcount[6,1]*biCPW62+bcount[7,1]*biCPW72+bcount[8,1]*biCPW82)/ (bcount[+ ,]);
bpCPWX1X3=(bcount[1,1]*biCPW13+bcount[2,1]*biCPW23+bcount[3,1]*biCPW33+bcount[4,1]*biCPW43+bcount[5,1]*biCPW53
          +bcount[6,1]*biCPW63+bcount[7,1]*biCPW73+bcount[8,1]*biCPW83)/ (bcount[+ ,]);

    bPAF1=bpCPWX2+bpCPWX1X3;
    bPAF2=bpCPWX2+bpCPWX1X3;
    bPAF3=bpCPWX1X3;
    bPAF4=bpCPWX2+bpCPWX1X3;
    bPAF5=bpCPWX1X3;
```

bPAF6=bpCPWX2;
bPAF7=bpCPWX1X3;
bPAF8=0;

biCPW=biCPW11 || biCPW12 || biCPW13 || biCPW21 || biCPW22 || biCPW23 || biCPW31 || biCPW32 || biCPW33 || biCPW41 || biCPW42 || biCPW43 || biCPW51 || biCPW52 || biCPW53 || biCPW61 || biCPW62 || biCPW63 || biCPW71 || biCPW72 || biCPW73 || biCPW81 || biCPW82 || biCPW83;
bpCPW=bpCPWU || bpCPWX2 || bpCPWX1X3;
bPAF=bPAF1 || bPAF2 || bPAF3 || bPAF4 || bPAF5 || bPAF6 || bPAF7 || bPAF8;

namebicpw={'biCPW11' 'biCPW12' 'biCPW13' 'biCPW21' 'biCPW22' 'biCPW23' 'biCPW31' 'biCPW32' 'biCPW33' 'biCPW41' 'biCPW42' 'biCPW43' 'biCPW51' 'biCPW52' 'biCPW53' 'biCPW61' 'biCPW62' 'biCPW63' 'biCPW71' 'biCPW72' 'biCPW73' 'biCPW81' 'biCPW82' 'biCPW83'};
create estimate1 from biCPW [colname=namebicpw];
append from biCPW;
namebpcpw={'bpCPWU' 'bpCPWX2' 'bpCPWX1X3'};
create estimate2 from bpCPW [colname=namebpcpw];
append from bpCPW;
namebPAF={'bPAF1' 'bPAF2' 'bPAF3' 'bPAF4' 'bPAF5' 'bPAF6' 'bPAF7' 'bPAF8'};
create estimate3 from bPAF [colname=namebPAF];
append from bPAF;
namebEB={'bEB1' 'bEB2'};
create booteb from bootbeta [colname=namebEB];
append from bootbeta;

```
namebCP={'bCP1' 'bCP2'};
create bootcp from bootbeta [colname=namebCP];
append from bootbeta;
quit;
proc append base=bEB       data=booteb;
run;
proc append base=bCP       data=bootcp;
run;
proc append base=&bestimate1       data=estimate1;
run;
proc append base=&bestimate2       data=estimate2;
run;
proc append base=&bestimate3       data=estimate3;
run;

%end;

  ************************************************************************************************************
   Using the dataset of bEB, bCP, &bestimate1, &bestimate2, &bestimate3 to calculate value of 2.5th and 97.5th percentile (same codes as 1C)
  ************************************************************************************************************

%mend;
```

%ci (setdata=fornlp,csdata=csdata, N=10000, NumTrials=***3***, bciEB=ciEB, bciCP=ciCP, bestimate1=biCPW, bestimate2=bpCPW, bestimate3=bPAF, bciicpw=bciicpw, bcipcpw=bcipcpw, bciPAF=bciPAF);

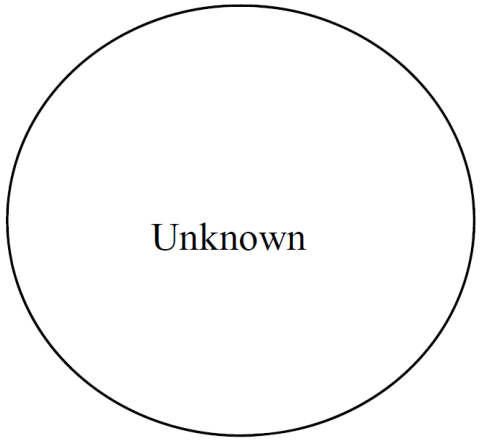
eAppendix3

Assume that we are interested in the relation between two dichotomous risk factors, X_1 and X_2, and a disease in a follow-up of a population. Four classes of sufficient causes can be identified: (1) the all-unknown class: containing neither of X_1 and X_2 as its component causes; (2) the X_1 class: containing X_1 but not X_2 as its component causes; (3) the X_2 class: containing X_2 but not X_1 as its component causes; and (4) the X_1 and X_2 class: containing both X_1 and X_2 as its component causes. We assume the completion rate (10^{-3}) for the all-unknown class (also the disease rate for people exposed to neither of the two risk factors) is 5.0. The ratios of the completion rate (the CP index) of the X_1 class, the X_2 class, and the X_1 and X_2 class and that of the all-unknown class are 1.2, 0.8, 1.5, respectively.

We simulate the survival time (time of censoring or time of failure for developing disease due to the particular class of sufficient causes) for each subject in the population from the exponential distribution. After 1,000 simulations, the means (medians) of CP values for the X_1 class, the X_2 class, and the X_1 and X_2 class are 1.154 (1.148), 0.774 (0.765), 1.492 (1.492), respectively, showing that the CP estimates are approximately unbiased.

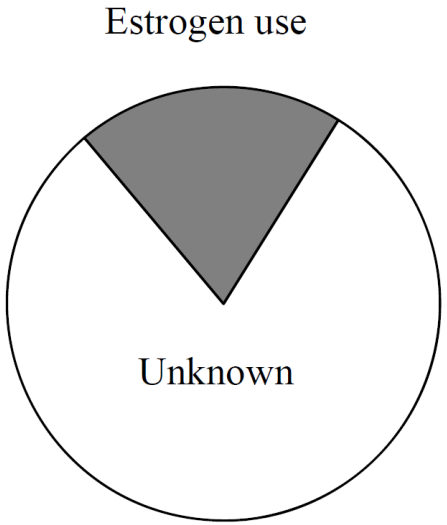
eFigure

Estimates and 95% confidence intervals of sufficient-cause related indices for the Leisure World Cohort Study of Endometrial Cancer[6] (CPW: causal-pie weight; CP: completion potential



Population-wide CPW (%)=22.3 (12.4-36.5)

CP=1.0



Population-wide CPW (%)=77.7 (63.5-87.6)

CP=7.0 (2.7-17.9)

eFigure.

eTable. Estimates of individual-based causal-pie weights (CPWs) and the population attributable fractions (PAFs) for the Bone Marrow Transplant Patients Study.[7]

Risk-Factor Profiles or Target Level			Individual-based CPW[c] (%)			PAF (%)	
CMV[a]	FAB[b]	MTX[a]	all-unknown	FAB	CMV and MTX	Estimate	95%CI
0	0	0	100.0	0.0	0.0	37.0	25.1, 48.8
1	0	0	100.0	0.0	0.0	37.0	25.1, 48.8
0	1	0	42.0	58.0	0.0	13.7	7.1, 21.4
0	0	1	100.0	0.0	0.0	37.0	25.1, 48.8
1	1	0	42.0	58.0	0.0	13.7	7.1, 21.4
1	0	1	39.0	0.0	61.0	23.4	14.2, 33.5
0	1	1	42.0	58.0	0.0	13.7	7.1, 21.4
1	1	1	25.3	35.1	39.0	0.0	

[a] CMV (cytomegalovirus infection), MTX (methotrexate use): 0 for no exposure (or with an intervention); 1 for exposure (or without an intervention)

[b] FAB (French-American-British disease classification grade): 0 for lower grade (or with an intervention); 1 for higher grade (or without an intervention)

[c] for classes of sufficient causes