**Appendix. Main SAS code of the simulation for scenario Ref**

**1. Generation of target population**

\*1.1 Simulating confounders L1-L6;

%include 'D:\Simulation\Programs\RandMVBinary.sas'; /\*Wicklin R. Simulating Data with SAS. Cary, NC: SAS Institute Inc; 2013.\*/

%let id=Ref;

ods output L=population;

**proc** **iml**;

load module=\_all\_;

p = {**0.2** **0.2** **0.2** **0.2** **0.2** **0.2**}; /\*positive probability\*/

R = {**1** **0** **0** **0** **0** **0**, **0** **1** **0** **0** **0** **0**, **0** **0** **1** **0** **0** **0**, **0** **0** **0** **1** **0** **0**, **0** **0** **0** **0** **1** **0**, **0** **0** **0** **0** **0** **1**}; /\*correlation matrix\*/

L = RandMVBinary(**10000**, p, R);

DiffMean = p - mean(L);

DiffCorr = R - corr(L);

call svd(U,Q,V,R);

print L;

**quit**;

**data** population;

set population;

rename col1=L1 col2=L2 col3=L3 col4=L4 col5=L5 col6=L6;

**run**;

\*1.2 Simulating exposure A, risk factor R, mediator M, outcome Y, and collider C;

**data** pop;

set population;

lnA=log(**0.2**/**0.8**)+log(**2**)\*L1+log(**2**)\*L2+log(**2**)\*L3+log(**2**)\*L4+log(**2**)\*L5+log(**2**)\*L6;

pA=exp(lnA)/(**1**+exp(lnA));

A=rand('bernoulli',pA);

lnR=log(**0.2**/**0.8**)+log(**2**)\*L1+log(**2**)\*L2+log(**2**)\*L3+log(**2**)\*L4;

pR=exp(lnR)/(**1**+exp(lnR));

R=rand('bernoulli',pR);

lnM=log(**0.2**/**0.8**)+log(**2**)\*A;

pM=exp(lnM)/(**1**+exp(lnM));

M=rand('bernoulli',pM);

**run**;

ods output ParameterEstimates=pe;

**proc** **reg** data=pop;

model M=A;

**run**;

**data** \_null\_;

set pe;

if variable='A';

call symput ('pe',estimate);

**run**;

**data** pop;

set pop;

lnY=log(**0.2**/**0.8**)+(log(**2**)-&pe\*log(**2**))\*A+log(**2**)\*M+log(**2**)\*R+log(**2**)\*L1+log(**2**)\*L2+log(**2**)\*L3+log(**2**)\*L4+log(**2**)\*L5+log(**2**)\*L6;

pY=exp(lnY)/(**1**+exp(lnY));

Y=rand('bernoulli',pY);

lnC=log(**0.2**/**0.8**)+log(**2**)\*A+log(**2**)\*Y;

pC=exp(lnC)/(**1**+exp(lnC));

C=rand('bernoulli',pC);

**run**;

**proc** **sort** data=pop;

by Y;

**run**;

**proc export** data=pop outfile="D:\Simulation\Results\population &id..csv" replace;

**run**;

**2. Generation of case-control studies**

\*2.1 Sampling cases and controls;

**%macro** sample(repeat=, data=, nstudy=);

%do s=**1** %to &nstudy;

proc surveyselect data=&data method=srs n=**100** out=sample\_&repeat.\_&s;

strata Y;

run;

%end;

**%mend** sample;

\*2.2 Estimating ORs;

**%macro** sampleOR(repeat=, nstudy=);

%do s=**1** %to &nstudy;

/\*Crude OR\*/

ods output RelativeRisks=sampleOR\_&repeat.\_&s.\_Crude;

proc freq data=sample\_&repeat.\_&s;

tables A\*Y /relrisk;

run;

data sampleOR\_&repeat.\_&s.\_Crude;

length type $20.;

set sampleOR\_&repeat.\_&s.\_Crude;

if statistic='Odds Ratio';

type='Crude';

rename Value=sampleOR;

keep Value LowerCL UpperCL type;

run;

/\*ORs insufficiently adjusted for confounders\*/

%do nvar=**1** %to **6**;

ods output OddsRatios=sampleOR\_&repeat.\_&s.\_Adjust&nvar;

proc logistic data=sample\_&repeat.\_&s;

model Y(event='1')=A L1-L&nvar;

run;

data sampleOR\_&repeat.\_&s.\_Adjust&nvar;

length type $20.;

set sampleOR\_&repeat.\_&s.\_Adjust&nvar;

if effect='A';

type="Adjust&nvar";

rename OddsRatioEst=sampleOR;

keep OddsRatioEst LowerCL UpperCL type;

run;

%end;

/\*ORs improperly adjusted for risk factor, mediator or collider\*/

ods output OddsRatios=sampleOR\_&repeat.\_&s.\_Risk;

proc logistic data=sample\_&repeat.\_&s;

model Y(event='1')=A L1-L6 R;

run;

data sampleOR\_&repeat.\_&s.\_Risk;

length type $20.;

set sampleOR\_&repeat.\_&s.\_Risk;

if effect='A';

type="Risk";

rename OddsRatioEst=sampleOR;

keep OddsRatioEst LowerCL UpperCL type;

run;

ods output OddsRatios=sampleOR\_&repeat.\_&s.\_Mediator;

proc logistic data=sample\_&repeat.\_&s;

model Y(event='1')=A L1-L6 M;

run;

data sampleOR\_&repeat.\_&s.\_Mediator;

length type $20.;

set sampleOR\_&repeat.\_&s.\_Mediator;

if effect='A';

type="Mediator";

rename OddsRatioEst=sampleOR;

keep OddsRatioEst LowerCL UpperCL type;

run;

ods output OddsRatios=sampleOR\_&repeat.\_&s.\_Collider;

proc logistic data=sample\_&repeat.\_&s;

model Y(event='1')=A L1-L6 C;

run;

data sampleOR\_&repeat.\_&s.\_Collider;

length type $20.;

set sampleOR\_&repeat.\_&s.\_Collider;

if effect='A';

type="Collider";

rename OddsRatioEst=sampleOR;

keep OddsRatioEst LowerCL UpperCL type;

run;

ods output OddsRatios=sampleOR\_&repeat.\_&s.\_All;

proc logistic data=sample\_&repeat.\_&s;

model Y(event='1')=A L1-L6 R M C;

run;

data sampleOR\_&repeat.\_&s.\_All;

length type $20.;

set sampleOR\_&repeat.\_&s.\_All;

if effect='A';

type="All";

rename OddsRatioEst=sampleOR;

keep OddsRatioEst LowerCL UpperCL type;

run;

/\*All ORs\*/

data sampleOR\_&repeat.\_&s;

length study $20.;

set sampleOR\_&repeat.\_&s.\_:;

es=log(sampleOR); se=(log(sampleOR)-log(LowerCL))/**1.96**;

study="s\_&repeat.\_&s"; repeat=&repeat; nstudy=&s;

run;

%end;

data sampleOR\_&repeat;

set sampleOR\_&repeat.\_1-sampleOR\_&repeat.\_&nstudy;

run;

proc datasets lib=work;

delete sample\_&repeat.\_: sampleOR\_&repeat.\_:;

run;

**%mend** sampleOR;

**3. Generation of Meta-analyses**

options mautosource sasautos=(' D:\Simulation\Programs');

**%macro** metaOR(repeat=, nstudy=, type=);

data sampleOR;

set sampleOR\_&repeat;

if type="&type";

w=**1**/(se\*\***2**);

run;

%***mainverse***(dataset=sampleOR,sampsize=no); /\*Senn S, Weir J, Hua TA, et al. Creating a suite of macros for meta-analysis in SAS: A case study in collaboration. Stat Probabil Lett. 2011; 81 (7): 842-857.\*/

data metaOR\_&repeat.\_&type;

length type $20.;

set inv\_var;

if type='Combined';

metaOR=exp(es); lowerCL=exp(lower); upperCL=exp(upper);

repeat=&repeat; type="&type";

run;

proc sql; create table q as select es, se, w, sum(w) as denominator, sum(w\*es) as numerator from sampleOR; run;

data q; set q; e=w\*((es-numerator/denominator)\*\***2**); run;

proc sql; create table i2 as select sum(e) as q from q; run;

data i2; set i2; i2=max(**0**,(q-&nstudy+**1**)/q); p=**1**-cdf('CHISQUARE',q,&nstudy-**1**); run;

data metaOR\_&repeat.\_&type;

merge metaOR\_&repeat.\_&type i2;

run;

**%mend** metaOR;

**4. Repetition**

**%macro** repeat(repeat=, data=, nstudy=, id=);

%do r=**1** %to &repeat;

%***sample***(repeat=&r,data=&data,nstudy=&nstudy);

%***sampleOR***(repeat=&r,nstudy=&nstudy);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Crude);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Adjust1);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Adjust2);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Adjust3);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Adjust4);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Adjust5);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Adjust6);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Risk);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Mediator);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=Collider);

%***metaOR***(repeat=&r,nstudy=&nstudy,type=All);

data metaOR\_&r;

set metaOR\_&r.\_:;

run;

dm log 'clear' continue;

dm odsresults 'clear' continue;

proc datasets lib=work;

delete metaOR\_&r.\_:;

run;

%end;

data metaOR;

set metaOR\_1-metaOR\_&repeat;

run;

proc sort data=metaOR;

by study;

run;

proc export data=metaOR outfile="D:\Simulation\Results\metaor &id..csv" replace;

run;

data sampleOR\_full;

length study $20.;

set sampleOR\_1-sampleOR\_&repeat;

run;

proc export data=sampleOR\_full outfile="D:\Simulation\Results\sampleor &id..csv" replace;

run;

**%mend** repeat;

ods html close;

%***repeat***(repeat=**1000**,data=pop,nstudy=**20**,id=Ref);

ods html;