eAppendix for "Causal inference with a mediated proportional hazards regression model"

1. Code for computing and plotting the natural direct and indirect effects using a mediated Cox proportional hazards regression model with Breslow estimator of cumulative baseline hazard

1.1. SAS code for estimating parameters of a mediated Cox proportional hazards regression model

```
proc surveyselect data=mediation_Hui NOPRINT seed=1
     out=BootCases_cox(rename=(Replicate=SampleID))
     method=urs              /* resample with replacement */
     samprate=1              /* each bootstrap sample has N observations */
     /* OUTHITS                 use OUTHITS option to suppress the frequency
var */
     reps=500      /* generate NumSamples bootstrap resamples */
       outhits;     /*Includes a distinct copy of each selected unit*/
run;

proc sort data=BootCases_cox;
by SampleID time_Recurrent_AKI_LOWHIGH descending Recurrent_AKI_LOWHIGH;
run;

data BootCases_cox;
set BootCases_cox;
by SampleID time_Recurrent_AKI_LOWHIGH descending Recurrent_AKI_LOWHIGH;
if (time_Recurrent_AKI_LOWHIGH=lag(time_Recurrent_AKI_LOWHIGH)) &
(Recurrent_AKI_LOWHIGH=1) then d+1;
if (time_Recurrent_AKI_LOWHIGH^=lag(time_Recurrent_AKI_LOWHIGH)) &
(Recurrent_AKI_LOWHIGH=1) then d=1;
if (Recurrent_AKI_LOWHIGH=0) then d=0;
if (first.time_Recurrent_AKI_LOWHIGH) & (Recurrent_AKI_LOWHIGH=1) then
d_first=1;
else d_first=0;
if (last.Recurrent_AKI_LOWHIGH) & (Recurrent_AKI_LOWHIGH=1) then d_last=1;
else d_last=0;
age_60=age-60;
run;

proc sort data=BootCases_cox;
by SampleID descending time_Recurrent_AKI_LOWHIGH  Recurrent_AKI_LOWHIGH;
run;

ods output ParameterEstimates=parameter;
proc nlmixed data=BootCases_cox;
by SampleID;
parms beta_intercept=0
        beta_V0_AKI=0
        beta_V0_CKD=0
      beta_female=0
        beta_diabetes=0
      lg_uNGAL_std=1
      theta_V0_AKI=0.32
        theta_V0_CKD=0.87
      theta_female=-0.06
        theta_diabetes=0.11
      theta_lg_uNGAL=0.12
      theta_V0_AKI_lg_uNGAL=0.13;
```

```
lg_uNGAL_mean=beta_intercept+
              (V0_AKI*beta_V0_AKI)+
              (V0_CKD*beta_V0_CKD)+
              (female*beta_female)+(diabetes*beta_diabetes);
ll_lg_uNGAL=-log(lg_uNGAL_std)-0.5*(((lg_uNGAL-
lg_uNGAL_mean)/lg_uNGAL_std)**2);
linear_predictor=(V0_AKI*theta_V0_AKI)+
                 (V0_CKD*theta_V0_CKD)+
                 (female*theta_female)+
                 (diabetes*theta_diabetes)+
                 (lg_uNGAL*theta_lg_uNGAL)+
                 (V0_AKI*lg_uNGAL*theta_V0_AKI_lg_uNGAL);

zlag_numerator=zlag(numerator);
zlag_denominator=zlag(denominator);
if (d=0) then numerator=0;
if (d=1) then numerator=linear_predictor;
if (d>1) then numerator=zlag_numerator+linear_predictor;
denominator=zlag_denominator+exp(linear_predictor);
if (Recurrent_AKI_LOWHIGH=0) then l_time=1;
if (Recurrent_AKI_LOWHIGH=1) & (d_last=0) then l_time=1;
if (Recurrent_AKI_LOWHIGH=1) & (d_last=1) then
l_time=(exp(numerator)/(denominator**d));
ll_time=log(l_time);
ll=ll_time+ll_lg_uNGAL;
model time_Recurrent_AKI_LOWHIGH ~ general(ll);
run;

proc sort data=parameter;
by SampleID;
run;

proc transpose data = parameter out=para_cox;
by SampleID;
id parameter;
var estimate;
run;
```

1.2 SAS code for Breslow estimator of cumulative baseline hazard in Cox proportional hazards model

```
data lg_uNGAL_3co_cox_cum(drop=i);
  do j = 1 to n;
   do i = 1 to 1533;
         set para_cox nobs=n point=j;
         output;
         end;
      end;
   stop;
   run;

data lg_uNGAL_3co_cox_cum;
set lg_uNGAL_3co_cox_cum;
drop _NAME_;
   run;

data lg_uNGAL_3co_cox_cum;
merge BootCases_cox lg_uNGAL_3co_cox_cum ;
```

```
run;

proc sort data=lg_uNGAL_3co_cox_cum;
by sampleid time_Recurrent_AKI_LOWHIGH;
run;

proc sort data=lg_uNGAL_3co_cox_cum;
by SampleID descending time_Recurrent_AKI_LOWHIGH  d;run;


data lg_uNGAL_3co_cox_cum;
set lg_uNGAL_3co_cox_cum;
by sampleid;
cox=exp((V0_AKI*theta_V0_AKI)+(V0_CKD*theta_V0_CKD)
    +(female*theta_female)+(diabetes*theta_diabetes)
    +(lg_uNGAL*theta_lg_uNGAL)
    +(theta_V0_AKI_lg_uNGAL*V0_AKI*lg_uNGAL));

if first.sampleid then hazard_de=0;
hazard_de+cox;
hazard=d/hazard_de;

run;

data lg_uNGAL_3co_cox_cum;
set lg_uNGAL_3co_cox_cum;
by SampleID descending time_Recurrent_AKI_LOWHIGH;
if last.time_Recurrent_AKI_LOWHIGH ;
run;

/*******************************************************/
proc sort data=lg_uNGAL_3co_cox_cum;
by  sampleid time_Recurrent_AKI_LOWHIGH ;
run;

data lg_uNGAL_3co_cox_cum;
set lg_uNGAL_3co_cox_cum;
by sampleid;
if first.sampleid then cum=0;
cum+hazard;
run;

data lg_uNGAL_3co_cox_cum;
set lg_uNGAL_3co_cox_cum;
if d=0 then delete;
run;
```

1.3. MATLAB code and SAS code for computing and plotting the natural direct and indirect effects based on the Breslow estimator of cumulative baseline hazard

1.3.1. MATLAB code for computing $\lambda_{T_{aM_{a^*}}}(t|c)$, $\lambda_{T_{a^*M_{a^*}}}(t|c)$ and $\lambda_{T_{aM_a}}(t|c)$ by integration

```
kcox=[]
kcoxa0a0=[]
```

```
kcoxa1a1=[]

for j=1:122497
    rcox=[]
    rcoxa0a0=[]
    rcoxa1a1=[]

%hazard for T(a,Ma*)
%a=1 a*=0
%female=1  diabetes=1 V0_CKD=1

fun=@(m)akilguNGAL3cocox500.hazard(j).*exp(akilguNGAL3cocox500.theta_V0_AKI(j
)*1+akilguNGAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0_AKI
_lg_uNGAL(j)*1.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox500.t
heta_diabetes(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1-
akilguNGAL3cocox500.cum(j).*exp(akilguNGAL3cocox500.theta_V0_AKI(j)*1+akilguN
GAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0_AKI_lg_uNGAL(j
)*1.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox500.theta_diabet
es(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1)-(m-
akilguNGAL3cocox500.beta_intercept(j)- akilguNGAL3cocox500.beta_V0_AKI(j)*0-
akilguNGAL3cocox500.beta_female(j)*1-akilguNGAL3cocox500.beta_diabetes(j)*1-
akilguNGAL3cocox500.beta_V0_CKD(j)*1).^2./(2*
akilguNGAL3cocox500.lg_uNGAL_std(j)^2))

q=integral(fun,-Inf,Inf)

fun1=@(m)exp(-
akilguNGAL3cocox500.cum(j).*exp(akilguNGAL3cocox500.theta_V0_AKI(j)*1+akilguN
GAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0_AKI_lg_uNGAL(j
)*1.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox500.theta_diabet
es(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1)-(m-
akilguNGAL3cocox500.beta_intercept(j)- akilguNGAL3cocox500.beta_V0_AKI(j)*0-
akilguNGAL3cocox500.beta_female(j)*1-akilguNGAL3cocox500.beta_diabetes(j)*1-
akilguNGAL3cocox500.beta_V0_CKD(j)*1).^2./(2*
akilguNGAL3cocox500.lg_uNGAL_std(j)^2))

        q1=integral(fun1,-Inf,Inf)

        s=q/q1
        rcox=[rcox s]

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%hazard for T(a*,Ma*)
%a=1 a*=0
%female=1  diabetes=1 V0_CKD=1
funa0a0=@(m)akilguNGAL3cocox500.hazard(j).*exp(akilguNGAL3cocox500.theta_V0_A
KI(j)*0+akilguNGAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0
_AKI_lg_uNGAL(j)*0.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox5
00.theta_diabetes(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1-
akilguNGAL3cocox500.cum(j).*exp(akilguNGAL3cocox500.theta_V0_AKI(j)*0+akilguN
GAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0_AKI_lg_uNGAL(j
)*0.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox500.theta_diabet
es(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1)-(m-
akilguNGAL3cocox500.beta_intercept(j)- akilguNGAL3cocox500.beta_V0_AKI(j)*0-
akilguNGAL3cocox500.beta_female(j)*1-akilguNGAL3cocox500.beta_diabetes(j)*1-
```

```
akilguNGAL3cocox500.beta_V0_CKD(j)*1).^2./(2*
akilguNGAL3cocox500.lg_uNGAL_std(j)^2))

qa0a0=integral(funa0a0,-Inf,Inf)

fun1a0a0=@(m)exp(-
akilguNGAL3cocox500.cum(j).*exp(akilguNGAL3cocox500.theta_V0_AKI(j)*0+akilguN
GAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0_AKI_lg_uNGAL(j
)*0.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox500.theta_diabet
es(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1)-(m-
akilguNGAL3cocox500.beta_intercept(j)- akilguNGAL3cocox500.beta_V0_AKI(j)*0-
akilguNGAL3cocox500.beta_female(j)*1-akilguNGAL3cocox500.beta_diabetes(j)*1-
akilguNGAL3cocox500.beta_V0_CKD(j)*1).^2./(2*
akilguNGAL3cocox500.lg_uNGAL_std(j)^2))

q1a0a0=integral(fun1a0a0,-Inf,Inf)

sa0a0=qa0a0/q1a0a0
rcoxa0a0=[rcoxa0a0 sa0a0]

  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%hazard for T(a,Ma)
%a=1 a*=0
%female=1  diabetes=1 V0_CKD=1
funa1a1=@(m)akilguNGAL3cocox500.hazard(j).*exp(akilguNGAL3cocox500.theta_V0_A
KI(j)*1+akilguNGAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0
_AKI_lg_uNGAL(j)*1.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox5
00.theta_diabetes(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1-
akilguNGAL3cocox500.cum(j).*exp(akilguNGAL3cocox500.theta_V0_AKI(j)*1+akilguN
GAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0_AKI_lg_uNGAL(j
)*1.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox500.theta_diabet
es(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1)-(m-
akilguNGAL3cocox500.beta_intercept(j)- akilguNGAL3cocox500.beta_V0_AKI(j)*1-
akilguNGAL3cocox500.beta_female(j)*1-akilguNGAL3cocox500.beta_diabetes(j)*1-
akilguNGAL3cocox500.beta_V0_CKD(j)*1).^2./(2*
akilguNGAL3cocox500.lg_uNGAL_std(j)^2))

qa1a1=integral(funa1a1,-Inf,Inf)

fun1a1a1=@(m)exp(-
akilguNGAL3cocox500.cum(j).*exp(akilguNGAL3cocox500.theta_V0_AKI(j)*1+akilguN
GAL3cocox500.theta_lg_uNGAL(j).*m+akilguNGAL3cocox500.theta_V0_AKI_lg_uNGAL(j
)*1.*m+akilguNGAL3cocox500.theta_female(j)*1+akilguNGAL3cocox500.theta_diabet
es(j)*1+akilguNGAL3cocox500.theta_V0_CKD(j)*1)-(m-
akilguNGAL3cocox500.beta_intercept(j)- akilguNGAL3cocox500.beta_V0_AKI(j)*1-
akilguNGAL3cocox500.beta_female(j)*1-akilguNGAL3cocox500.beta_diabetes(j)*1-
akilguNGAL3cocox500.beta_V0_CKD(j)*1).^2./(2*
akilguNGAL3cocox500.lg_uNGAL_std(j)^2))

q1a1a1=integral(fun1a1a1,-Inf,Inf)

sa1a1=qa1a1/q1a1a1
rcoxa1a1=[rcoxa1a1 sa1a1]
```

```
    kcox=[kcox rcox]
    kcoxa0a0=[kcoxa0a0 rcoxa0a0]
    kcoxa1a1=[kcoxa1a1 rcoxa1a1]

end
save ('kcox')
save ('kcoxa0a0')
save ('kcoxa1a1')
```

1.3.2. SAS code for computing the natural direct and indirect effects based on $\lambda_{T_{aM_{a^*}}}(t|c)$*,* $\lambda_{T_{a^*M_{a^*}}}(t|c)$ *and* $\lambda_{T_{aM_a}}(t|c)$

```
/***calculate indirect and direct effect**********/
proc sort data=Akilgungal3cocox500_result;
by time_Recurrent_AKI_LOWHIGH sampleid;
run;

data Akilgungal3cocox500_plot;
set Akilgungal3cocox500_result;
indirect=kcoxa1a1/kcox;
direct=kcox/kcoxa0a0;
run;

proc univariate  data=Akilgungal3cocox500_plot noprint;
var indirect direct;
by time_Recurrent_AKI_LOWHIGH;
output out=effect_interval500 pctlpts=2.5 97.5 pctlpre=indirect direct
mean=indirect_mean direct_mean;
run;

/***calculate VanderWeele****/
data VanderWeele;
set Para_cox;
indirect_vander=exp((theta_lg_uNGAL*beta_V0_AKI+theta_V0_AKI_lg_uNGAL*
beta_V0_AKI)*(1-0));
direct_vander=exp((theta_V0_AKI +
theta_V0_AKI_lg_uNGAL*(beta_intercept+ beta_V0_AKI*0
                    +beta_female*1+beta_diabetes*1+beta_V0_CKD*1+
theta_lg_uNGAL*(lg_uNGAL_std**2)))*(1-0)

+0.5*(theta_V0_AKI_lg_uNGAL)*(theta_V0_AKI_lg_uNGAL)*(lg_uNGAL_std)*(l
g_uNGAL_std)*(1-0));
run;

proc univariate  data=VanderWeele noprint;
var indirect_vander direct_vander;

output out=vanderweele_interval pctlpts=2.5 97.5
pctlpre=indirectvander directvander mean=indirectvander_mean
directvander_mean;
run;
```

```
data VanderWeele1(drop=i);
  do j = 1 to n;
   do i = 1 to 334;
         set vanderweele_interval nobs=n point=j;
         output;
         end;
      end;
   stop;
   run;

/****************/

data effect_interval500;
merge effect_interval500 VanderWeele1;
run;
```

1.3.3. MATLAB code for plotting the natural direct and indirect effects

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%import data "akilguNGAL3cocoxplot200"

x=akilguNGAL3cocoxplot500.time_Recurrent_AKI_LOWHIGH
indirect_mean=akilguNGAL3cocoxplot500.indirect_mean
indirect_lower=akilguNGAL3cocoxplot500.indirect2_5
indirect_upper=akilguNGAL3cocoxplot500.indirect97_5
direct_mean=akilguNGAL3cocoxplot500.direct_mean
direct_lower=akilguNGAL3cocoxplot500.direct2_5
direct_upper=akilguNGAL3cocoxplot500.direct97_5
indirect_vander_mean=akilguNGAL3cocoxplot500.indirectvander_mean
indirect_vander_lower=akilguNGAL3cocoxplot500.indirectvander2_5
indirect_vander_upper=akilguNGAL3cocoxplot500.indirectvander97_5
direct_vander_mean=akilguNGAL3cocoxplot500.directvander_mean
direct_vander_lower=akilguNGAL3cocoxplot500.directvander2_5
direct_vander_upper=akilguNGAL3cocoxplot500.directvander97_5

figure

plot(x,indirect_mean,'b',x,indirect_lower,'b--',x,indirect_upper,'b--
',x,indirect_vander_mean,'c',x,indirect_vander_lower,'c--
',x,indirect_vander_upper,'c--')
title('plot for indirect effect on hazard ratio')
xlabel('time')
ylabel('hazard ratio')
legend({'indirect effect from integration','indirect effect from integration
lower bound' ,'indirect effect from integration upper bound','indirect effect
by VanderWeele','indirect effect by VanderWeele lower bound', 'indirect
effect by VanderWeele upper bound'},'Position',[0.5 0.1 0.4 0.1])


ax1=gca
ax1_pos=get(ax1,'position')  %get the location of the current plot
                             % which is [0.1300    0.1100    0.7750    0.8150]
                             % ax1_pos(1) and ax1_pos(2) are the coordinate
of the legend windows
```

```
                             % ax1_pos(3) and ax1_pos(4) are width and height
of this windows

pos1 = ax1_pos(2)            % save the original bottom position
ax1_pos(2)=ax1_pos(2)+2.5*pos1
ax1_pos(4)=ax1_pos(4)-2.5*pos1

set(ax1,'position',ax1_pos)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
figure

plot(x,direct_mean,'r',x,direct_lower,'r--',x,direct_upper,'r--
',x,direct_vander_mean,'m',x,direct_vander_lower,'m--
',x,direct_vander_upper,'m--')
title('plot for direct effect on hazard ratio')
xlabel('time')
ylabel('hazard ratio')
legend({'direct effect from integration','direct effect from integration
lower bound' ,'direct effect from integration upper bound','direct effect by
VanderWeele','direct effect by VanderWeele lower bound', 'direct effect by
VanderWeele upper bound'},'Position',[0.5 0.1 0.4 0.1])


ax1=gca
ax1_pos=get(ax1,'position')  %get the location of the current plot
                             % which is [0.1300    0.1100    0.7750    0.8150]
                             % ax1_pos(1) and ax1_pos(2) are the coordinate
of the legend windows
                             % ax1_pos(3) and ax1_pos(4) are width and height
of this windows

pos1 = ax1_pos(2)            % save the original bottom position
ax1_pos(2)=ax1_pos(2)+2.5*pos1
ax1_pos(4)=ax1_pos(4)-2.5*pos1

set(ax1,'position',ax1_pos)
```

2. Code for Computing and plotting the natural direct and indirect effects using a mediated proportional hazards regression model with piecewise constant hazard.

2.1. SAS code for estimating parameters of a mediated proportional hazards regression model with piecewise constant hazard.

```
proc surveyselect data=mediation_Hui NOPRINT seed=1
     out=BootCases(rename=(Replicate=SampleID))
     method=urs                /* resample with replacement */
     samprate=1                /* each bootstrap sample has N
observations */
     /* OUTHITS                   use OUTHITS option to suppress the
frequency var */
```

```
     reps=500      /* generate NumSamples bootstrap resamples */
      outhits;     /*Includes a distinct copy of each selected unit*/
run;

proc sort data=BootCases;
by sampleid time_Recurrent_AKI_LOWHIGH;
run;

data AKI_lowhigh_1_8;
set BootCases;
if Recurrent_AKI_lowhigh=1;
run;

data NOAKI_lowhigh_1_8;
set BootCases;
if Recurrent_AKI_lowhigh=0;
run;

proc univariate  data=AKI_LOWHIGH_1_8 noprint;
by sampleid;
var time_Recurrent_AKI_LOWHIGH;
output out=percentiles_18 pctlpts=12.5 25 37.5 50 62.5 75 87.5 100
pctlpre=time_Recurrent_AKI_LOWHIGH;
run;

data percentiles_1_8(drop=i);
  do j = 1 to n;
   do i = 1 to 1533;
         set percentiles_18 nobs=n point=j;
         output;
         end;
      end;
   stop;
   run;
data BootCases_group;
merge BootCases percentiles_1_8;
group1=0;
group2=0;
group3=0;
group4=0;
group5=0;
group6=0;
group7=0;
group8=0;

if 0<=time_Recurrent_AKI_LOWHIGH<=time_Recurrent_AKI_LOWHIGH12_5 then
group1=1;
if
time_Recurrent_AKI_LOWHIGH12_5<time_Recurrent_AKI_LOWHIGH<=time_Recurr
ent_AKI_LOWHIGH25 then group2=1;
```

```
if
time_Recurrent_AKI_LOWHIGH25<time_Recurrent_AKI_LOWHIGH<=time_Recurren
t_AKI_LOWHIGH37_5 then group3=1;
if
time_Recurrent_AKI_LOWHIGH37_5<time_Recurrent_AKI_LOWHIGH<=time_Recurr
ent_AKI_LOWHIGH50 then group4=1;
if
time_Recurrent_AKI_LOWHIGH50<time_Recurrent_AKI_LOWHIGH<=time_Recurren
t_AKI_LOWHIGH62_5 then group5=1;
if
time_Recurrent_AKI_LOWHIGH62_5<time_Recurrent_AKI_LOWHIGH<=time_Recurr
ent_AKI_LOWHIGH75 then group6=1;
if
time_Recurrent_AKI_LOWHIGH75<time_Recurrent_AKI_LOWHIGH<=time_Recurren
t_AKI_LOWHIGH87_5 then group7=1;
if time_Recurrent_AKI_LOWHIGH87_5<time_Recurrent_AKI_LOWHIGH then
group8=1;

run;

ods output ParameterEstimates=parameter;
proc nlmixed data= BootCases_group;
by sampleid;
parms
      beta_intercept=0

      beta_V0_AKI=0

      beta_V0_CKD=0
      beta_female=0
      beta_diabetes=0

      lg_uNGAL_std=1

      lambda1=1
      lambda2=1
      lambda3=1
      lambda4=1
      lambda5=1
      lambda6=1
      lambda7=1
      lambda8=1

     theta_V0_AKI=0
     theta_V0_CKD=0

     theta_female=0
     theta_diabetes=0
     theta_lg_uNGAL=0
     theta_V0_AKI_lg_uNGAL=0;

lg_uNGAL_mean=beta_intercept+(V0_AKI*beta_V0_AKI)+(V0_CKD*beta_V0_CKD)
```

```
            +(female*beta_female)
            +(diabetes*beta_diabetes);

ll_lg_uNGAL=-log(lg_uNGAL_std)-0.5*(((lg_uNGAL-
lg_uNGAL_mean)/lg_uNGAL_std)**2);

baseline_group1=lambda1*group1;
baseline_group2=lambda2*group2;
baseline_group3=lambda3*group3;
baseline_group4=lambda4*group4;
baseline_group5=lambda5*group5;
baseline_group6=lambda6*group6;
baseline_group7=lambda7*group7;
baseline_group8=lambda8*group8;

baseline_group=baseline_group1+baseline_group2+baseline_group3+baselin
e_group4

+baseline_group5+baseline_group6+baseline_group7+baseline_group8;


cox=exp((V0_AKI*theta_V0_AKI)+(V0_CKD*theta_V0_CKD)
          +(female*theta_female)+(diabetes*theta_diabetes)
          +(lg_uNGAL*theta_lg_uNGAL)
          +(theta_V0_AKI_lg_uNGAL*V0_AKI*lg_uNGAL));

hazard=baseline_group*cox;

cum1=( time_Recurrent_AKI_LOWHIGH12_5 - 0.0)*lambda1;
cum2=( time_Recurrent_AKI_LOWHIGH25-
time_Recurrent_AKI_LOWHIGH12_5)*lambda2;
cum3=( time_Recurrent_AKI_LOWHIGH37_5-
time_Recurrent_AKI_LOWHIGH25)*lambda3;
cum4=( time_Recurrent_AKI_LOWHIGH50-
time_Recurrent_AKI_LOWHIGH37_5)*lambda4;
cum5=( time_Recurrent_AKI_LOWHIGH62_5-
time_Recurrent_AKI_LOWHIGH50)*lambda5;
cum6=( time_Recurrent_AKI_LOWHIGH75-
time_Recurrent_AKI_LOWHIGH62_5)*lambda6;
cum7=( time_Recurrent_AKI_LOWHIGH87_5-
time_Recurrent_AKI_LOWHIGH75)*lambda7;

if ( 0.0<=time_Recurrent_AKI_LOWHIGH<=time_Recurrent_AKI_LOWHIGH12_5)
then cumulative_hazard=(time_Recurrent_AKI_LOWHIGH- 0.0)*lambda1;
if ( time_Recurrent_AKI_LOWHIGH12_5<time_Recurrent_AKI_LOWHIGH<=
time_Recurrent_AKI_LOWHIGH25) then
cumulative_hazard=cum1+((time_Recurrent_AKI_LOWHIGH-
time_Recurrent_AKI_LOWHIGH12_5)*lambda2);
if ( time_Recurrent_AKI_LOWHIGH25<time_Recurrent_AKI_LOWHIGH<=
time_Recurrent_AKI_LOWHIGH37_5) then
cumulative_hazard=cum1+cum2+((time_Recurrent_AKI_LOWHIGH-
time_Recurrent_AKI_LOWHIGH25)*lambda3);
```

```
if ( time_Recurrent_AKI_LOWHIGH37_5<time_Recurrent_AKI_LOWHIGH<=
time_Recurrent_AKI_LOWHIGH50) then
cumulative_hazard=cum1+cum2+cum3+((time_Recurrent_AKI_LOWHIGH-
time_Recurrent_AKI_LOWHIGH37_5)*lambda4);
if ( time_Recurrent_AKI_LOWHIGH50<time_Recurrent_AKI_LOWHIGH<=
time_Recurrent_AKI_LOWHIGH62_5) then
cumulative_hazard=cum1+cum2+cum3+cum4+((time_Recurrent_AKI_LOWHIGH-
time_Recurrent_AKI_LOWHIGH50)*lambda5);
if ( time_Recurrent_AKI_LOWHIGH62_5<time_Recurrent_AKI_LOWHIGH<=
time_Recurrent_AKI_LOWHIGH75) then
cumulative_hazard=cum1+cum2+cum3+cum4+cum5+((time_Recurrent_AKI_LOWHIG
H- time_Recurrent_AKI_LOWHIGH62_5)*lambda6);
if ( time_Recurrent_AKI_LOWHIGH75<time_Recurrent_AKI_LOWHIGH<=
time_Recurrent_AKI_LOWHIGH87_5) then
cumulative_hazard=cum1+cum2+cum3+cum4+cum5+cum6+((time_Recurrent_AKI_L
OWHIGH-time_Recurrent_AKI_LOWHIGH75)*lambda7);
if ( time_Recurrent_AKI_LOWHIGH87_5<time_Recurrent_AKI_LOWHIGH     )
then
cumulative_hazard=cum1+cum2+cum3+cum4+cum5+cum6+cum7+((time_Recurrent_
AKI_LOWHIGH-time_Recurrent_AKI_LOWHIGH87_5)*lambda8);

cumulative_hazard=cumulative_hazard*cox;
log_survival=-cumulative_hazard;
log_pdf=log(hazard)+log_survival;
ll_t=((Recurrent_AKI_LOWHIGH=1)*log_pdf)+((Recurrent_AKI_LOWHIGH=0)*lo
g_survival);
ll=ll_t+ll_lg_uNGAL;
/* or it is equal to
ll=(censored=0)*log(hazard)+log(survival);
*/
model time_Recurrent_AKI_LOWHIGH ~ general(ll);

run;

proc sort data=parameter;
by SampleID;
run;

proc transpose data = parameter out=para18;
by SampleID;
id parameter;
var estimate;
run;
```

2. 2. MATLAB code for computing and plotting the natural direct and indirect effects with the estimated parameters.

```
k18=[]
k18a0a0=[]
k18a1a1=[]

for j=1:92
```

```
    r18=[]
    r18a0a0=[]
    r18a1a1=[]

   for i=1:500

cum1=( percentiles18Hui500.time_Recurrent_AKI_LOWHIGH12_5(i) -
0.0)*para18Hui500.lambda1(i);
cum2=( percentiles18Hui500.time_Recurrent_AKI_LOWHIGH25(i)-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH12_5(i))*
para18Hui500.lambda2(i)
cum3=( percentiles18Hui500.time_Recurrent_AKI_LOWHIGH37_5(i)-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH25(i))* para18Hui500.lambda3(i)
cum4=( percentiles18Hui500.time_Recurrent_AKI_LOWHIGH50(i)-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH37_5(i))*
para18Hui500.lambda4(i)
cum5=( percentiles18Hui500.time_Recurrent_AKI_LOWHIGH62_5(i)-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH50(i))* para18Hui500.lambda5(i)
cum6=( percentiles18Hui500.time_Recurrent_AKI_LOWHIGH75(i)-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH62_5(i))*
para18Hui500.lambda6(i)
cum7=( percentiles18Hui500.time_Recurrent_AKI_LOWHIGH87_5(i)-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH75(i))* para18Hui500.lambda7(i)

if ( 0.0<=j)&&(j<= percentiles18Hui500.time_Recurrent_AKI_LOWHIGH12_5(i))

    cumulative_hazard=(j-0.0)* para18Hui500.lambda1(i)
    lambda= para18Hui500.lambda1(i)

elseif
(percentiles18Hui500.time_Recurrent_AKI_LOWHIGH12_5(i)<j)&&(j<=percentiles18H
ui500.time_Recurrent_AKI_LOWHIGH25(i))

    cumulative_hazard=cum1+((j-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH12_5(i))*
para18Hui500.lambda2(i))
    lambda= para18Hui500.lambda2(i)

elseif
(percentiles18Hui500.time_Recurrent_AKI_LOWHIGH25(i)<j)&&(j<=percentiles18Hui
500.time_Recurrent_AKI_LOWHIGH37_5(i))

    cumulative_hazard=cum1+cum2+((j-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH25(i))*  para18Hui500.lambda3(i))
    lambda= para18Hui500.lambda3(i)

elseif
(percentiles18Hui500.time_Recurrent_AKI_LOWHIGH37_5(i)<j)&&(j<=percentiles18H
ui500.time_Recurrent_AKI_LOWHIGH50(i))

    cumulative_hazard=cum1+cum2+cum3+((j-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH37_5 (i))*
para18Hui500.lambda4(i))
    lambda= para18Hui500.lambda4(i)
```

```
elseif
(percentiles18Hui500.time_Recurrent_AKI_LOWHIGH50(i)<j)&&(j<=percentiles18Hui
500.time_Recurrent_AKI_LOWHIGH62_5(i))

    cumulative_hazard=cum1+cum2+cum3+cum4+((j-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH50(i))*  para18Hui500.lambda5(i))
    lambda= para18Hui500.lambda5(i)

elseif
(percentiles18Hui500.time_Recurrent_AKI_LOWHIGH62_5(i)<j)&&(j<=percentiles18H
ui500.time_Recurrent_AKI_LOWHIGH75(i))
    cumulative_hazard=cum1+cum2+cum3+cum4+cum5+((j-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH62_5(i))*
para18Hui500.lambda6(i))
    lambda= para18Hui500.lambda6(i)

elseif
(percentiles18Hui500.time_Recurrent_AKI_LOWHIGH75(i)<j)&&(j<=percentiles18Hui
500.time_Recurrent_AKI_LOWHIGH87_5(i))

    cumulative_hazard=cum1+cum2+cum3+cum4+cum5+cum6+((j-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH75(i))*  para18Hui500.lambda7(i))
    lambda= para18Hui500.lambda7(i)

else
    cumulative_hazard=cum1+cum2+cum3+cum4+cum5+cum6+cum7+((j-
percentiles18Hui500.time_Recurrent_AKI_LOWHIGH75(i))*  para18Hui500.lambda8(i))
    lambda= para18Hui500.lambda8(i)

end

%hazard for T(a,Ma*)
%a=1 a*=0
%female=1 center2=1 age=66 smoker2=1 diabetes=1 V0_CKD=1

fun=@(m)lambda*exp(para18Hui500.theta_V0_AKI(i)*1+para18Hui500.theta_lg_uNGAL
(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*1.*m+para18Hui500.theta_female(i
)*1+para18Hui500.theta_diabetes(i)*1+ para18Hui500.theta_V0_CKD(i)*1-
cumulative_hazard.*exp(para18Hui500.theta_V0_AKI(i)*1+para18Hui500.theta_lg_u
NGAL(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*1.*m+para18Hui500.theta_fema
le(i)*1+para18Hui500.theta_diabetes(i)*1+ para18Hui500.theta_V0_CKD(i)*1)-(m-
para18Hui500.beta_intercept(i)-para18Hui500.beta_V0_AKI(i)*0-
para18Hui500.beta_female(i)*1-para18Hui500.beta_diabetes(i)*1-
para18Hui500.beta_V0_CKD(i)*1).^2./(2*para18Hui500.lg_uNGAL_std(i)^2))
        q=integral(fun,-Inf,Inf)

      fun1=@(m)exp(-
cumulative_hazard.*exp(para18Hui500.theta_V0_AKI(i)*1+para18Hui500.theta_lg_u
NGAL(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*1.*m+
para18Hui500.theta_female(i)*1+para18Hui500.theta_diabetes(i)*1+
para18Hui500.theta_V0_CKD(i)*1)-(m- para18Hui500.beta_intercept(i)-
para18Hui500.beta_V0_AKI(i)*0-para18Hui500.beta_female(i)*1-
para18Hui500.beta_diabetes(i)*1-
para18Hui500.beta_V0_CKD(i)*1).^2./(2*para18Hui500.lg_uNGAL_std(i)^2))
        q1=integral(fun1,-Inf,Inf)
```

```
        s=q/q1
        r18=[r18 s]

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%hazard for T(a*,Ma*)
%a=1 a*=0
%female=1 center2=1 age=66 smoker2=1 diabetes=1 V0_CKD=1

funa0a0=@(m)lambda*exp(para18Hui500.theta_V0_AKI(i)*0+para18Hui500.theta_lg_u
NGAL(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*0.*m+
para18Hui500.theta_female(i)*1+para18Hui500.theta_diabetes(i)*1+
para18Hui500.theta_V0_CKD(i)*1-
cumulative_hazard.*exp(para18Hui500.theta_V0_AKI(i)*0+para18Hui500.theta_lg_u
NGAL(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*0.*m+
para18Hui500.theta_female(i)*1+para18Hui500.theta_diabetes(i)*1+
para18Hui500.theta_V0_CKD(i)*1)-(m-para18Hui500.beta_intercept(i)-
para18Hui500.beta_V0_AKI(i)*0- para18Hui500.beta_female(i)*1-
para18Hui500.beta_diabetes(i)*1-
para18Hui500.beta_V0_CKD(i)*1).^2./(2*para18Hui500.lg_uNGAL_std(i)^2))
        qa0a0=integral(funa0a0,-Inf,Inf)

      fun1a0a0=@(m)exp(-
cumulative_hazard.*exp(para18Hui500.theta_V0_AKI(i)*0+para18Hui500.theta_lg_u
NGAL(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*0.*m+
para18Hui500.theta_female(i)*1+para18Hui500.theta_diabetes(i)*1+
para18Hui500.theta_V0_CKD(i)*1)-(m-para18Hui500.beta_intercept(i)-
para18Hui500.beta_V0_AKI(i)*0- para18Hui500.beta_female(i)*1-
para18Hui500.beta_diabetes(i)*1-
para18Hui500.beta_V0_CKD(i)*1).^2./(2*para18Hui500.lg_uNGAL_std(i)^2))
        q1a0a0=integral(fun1a0a0,-Inf,Inf)

        sa0a0=qa0a0/q1a0a0
        r18a0a0=[r18a0a0 sa0a0]

  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%hazard for T(a,Ma)
%a=1 a*=0
%female=1 black=1 age=61

funa1a1=@(m)lambda*exp(para18Hui500.theta_V0_AKI(i)*1+para18Hui500.theta_lg_u
NGAL(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*1.*m+
para18Hui500.theta_female(i)*1+para18Hui500.theta_diabetes(i)*1+
para18Hui500.theta_V0_CKD(i)*1-
cumulative_hazard.*exp(para18Hui500.theta_V0_AKI(i)*1+para18Hui500.theta_lg_u
NGAL(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*1.*m+
para18Hui500.theta_female(i)*1+para18Hui500.theta_diabetes(i)*1+
para18Hui500.theta_V0_CKD(i)*1)-(m-para18Hui500.beta_intercept(i)-
para18Hui500.beta_V0_AKI(i)*1- para18Hui500.beta_female(i)*1-
para18Hui500.beta_diabetes(i)*1-
para18Hui500.beta_V0_CKD(i)*1).^2./(2*para18Hui500.lg_uNGAL_std(i)^2))
        qa1a1=integral(funa1a1,-Inf,Inf)
```

```
      fun1a1a1=@(m)exp(-
cumulative_hazard.*exp(para18Hui500.theta_V0_AKI(i)*1+para18Hui500.theta_lg_u
NGAL(i).*m+para18Hui500.theta_V0_AKI_lg_uNGAL(i)*1.*m+
para18Hui500.theta_female(i)*1+para18Hui500.theta_diabetes(i)*1+
para18Hui500.theta_V0_CKD(i)*1)-(m-para18Hui500.beta_intercept(i)-
para18Hui500.beta_V0_AKI(i)*1- para18Hui500.beta_female(i)*1-
para18Hui500.beta_diabetes(i)*1-
para18Hui500.beta_V0_CKD(i)*1).^2./(2*para18Hui500.lg_uNGAL_std(i)^2))
        q1a1a1=integral(fun1a1a1,-Inf,Inf)

        sa1a1=qa1a1/q1a1a1
        r18a1a1=[r18a1a1 sa1a1]

    end
    k18=[k18 r18]
    k18a0a0=[k18a0a0 r18a0a0]
    k18a1a1=[k18a1a1 r18a1a1]
end

kk18=reshape(k18,500,92)
kk18a0a0=reshape(k18a0a0,500,92)
kk18a1a1=reshape(k18a1a1,500,92)


save ('kk18')
save ('kk18a0a0')
save ('kk18a1a1')

%%%%%%plot%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
x=1:92
indirect=kk18a1a1./kk18
indirect_mean=mean(indirect)
indirect_lower=prctile(indirect,[2.5],1)
indirect_upper=prctile(indirect,[97.5],1)

direct=kk18./kk18a0a0
direct_mean=mean(direct)
direct_lower=prctile(direct,[2.5],1)
direct_upper=prctile(direct,[97.5],1)

save ('indirect')
save ('direct')

save ('indirect_mean')
save ('indirect_lower')
save ('indirect_upper')

save ('direct_mean')
save ('direct_lower')
save ('direct_upper')

% vanderweele
indirect_vander= exp((para18Hui500.theta_lg_uNGAL.* para18Hui500.beta_V0_AKI
+ para18Hui500.theta_V0_AKI_lg_uNGAL.* para18Hui500.beta_V0_AKI).*(1-0))
```

```
indirect_vander_mean_1=mean(indirect_vander)
indirect_vander_mean=repelem(indirect_vander_mean_1,92)

indirect_vander_lower_1=prctile(indirect_vander,[2.5],1)
indirect_vander_lower=repelem(indirect_vander_lower_1,92)

indirect_vander_upper_1=prctile(indirect_vander,[97.5],1)
indirect_vander_upper= repelem(indirect_vander_upper_1,92)


direct_vander=exp(para18Hui500.theta_V0_AKI +
para18Hui500.theta_V0_AKI_lg_uNGAL.*( para18Hui500.beta_intercept+
para18Hui500.beta_V0_AKI.*0+
para18Hui500.beta_female*1+para18Hui500.beta_diabetes*1+para18Hui500.beta_V0_
CKD*1+ para18Hui500.theta_lg_uNGAL.*( para18Hui500.lg_uNGAL_std.^2)).*(1-
0)+0.5.*( para18Hui500.theta_V0_AKI_lg_uNGAL).*( para18Hui500.theta_V0_AKI_lg
_uNGAL).*( para18Hui500.lg_uNGAL_std).*( para18Hui500.lg_uNGAL_std).*(1-0))

direct_vander_mean_1=mean(direct_vander)
direct_vander_mean=repelem(direct_vander_mean_1,92)
direct_vander_lower_1=prctile(direct_vander,[2.5],1)
direct_vander_lower=repelem(direct_vander_lower_1,92)
direct_vander_upper_1=prctile(direct_vander,[97.5],1)
direct_vander_upper= repelem(direct_vander_upper_1,92)


save('indirect_vander')
save('indirect_vander_mean_1')
save('indirect_vander_lower_1')
save('indirect_vander_upper_1')

save('direct_vander')
save('direct_vander_mean_1')
save('direct_vander_lower_1')
save('direct_vander_upper_1')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure
plot(x,indirect_mean,'b',x,indirect_lower,'b--',x,indirect_upper,'b--
',x,indirect_vander_mean,'c',x,indirect_vander_lower,'c--
',x,indirect_vander_upper,'c--')
title('plot for indirect effect on hazard ratio')
xlabel('time')
ylabel('hazard ratio')
ylim([1.01 1.12])
legend({'indirect effect from integration','indirect effect from integration
lower bound' ,'indirect effect from integration upper bound','indirect effect
by VanderWeele','indirect effect by VanderWeele lower bound', 'indirect
effect by VanderWeele upper bound'},'Position',[0.5 0.1 0.4 0.1])

ax1=gca
ax1_pos=get(ax1,'position')  %get the location of the current plot
                             % which is [0.1300    0.1100    0.7750    0.8150]
```

```
                                % ax1_pos(1) and ax1_pos(2) are the coordinate
of the legend windows
                                % ax1_pos(3) and ax1_pos(4) are width and height
of this windows

pos1 = ax1_pos(2)               % save the original bottom position
ax1_pos(2)=ax1_pos(2)+2.5*pos1
ax1_pos(4)=ax1_pos(4)-2.5*pos1

set(ax1,'position',ax1_pos)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
figure
plot(x,direct_mean,'r',x,direct_lower,'r--',x,direct_upper,'r--
',x,direct_vander_mean,'m',x,direct_vander_lower,'m--
',x,direct_vander_upper,'m--')
title('plot for direct effect on hazard ratio')
xlabel('time')
ylabel('hazard ratio')
legend({'direct effect from integration','direct effect from integration
lower bound' ,'direct effect from integration upper bound','direct effect by
VanderWeele','direct effect by VanderWeele lower bound', 'direct effect by
VanderWeele upper bound'},'Position',[0.5 0.1 0.4 0.1])

ax1=gca
ax1_pos=get(ax1,'position')  %get the location of the current plot
                                % which is [0.1300    0.1100    0.7750    0.8150]
                                % ax1_pos(1) and ax1_pos(2) are the coordinate
of the legend windows
                                % ax1_pos(3) and ax1_pos(4) are width and height
of this windows

pos1 = ax1_pos(2)               % save the original bottom position
ax1_pos(2)=ax1_pos(2)+2.5*pos1
ax1_pos(4)=ax1_pos(4)-2.5*pos1

set(ax1,'position',ax1_pos)
```