gformula: Estimating causal effects in the presence of time-dependent confounding or mediation

Rhian Daniel, Bianca De Stavola, Simon Cousens

Centre for Statistical Methodology
London School of Hygiene and Tropical Medicine

Italian Stata Users Group Meeting · Bologna
September 20, 2012

Outline

1 Time-dependent confounding

2 Mediation

3 Notation, assumptions and causal questions

4 G-computation formula

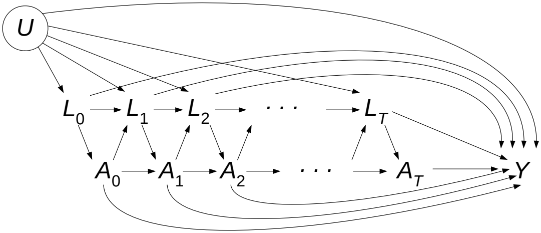
5 `gformula` in Stata

Outline

1. Time-dependent confounding
2. Mediation
3. Notation, assumptions and causal questions
4. G-computation formula
5. gformula in Stata

The setting

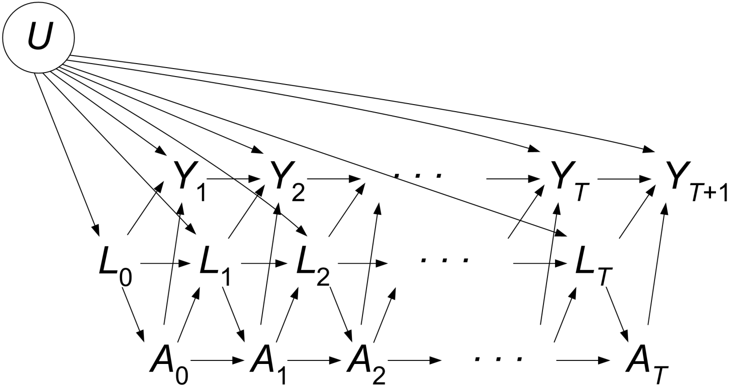
Single outcome at end of follow-up



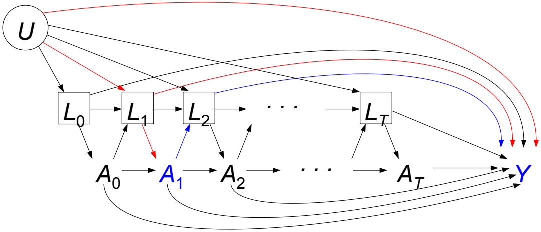
- We are interested in the causal effect of a time-varying exposure A on an outcome Y.
- This relationship is confounded by time-varying confounder L.
- L is affected by A.
- eg ART, CD4, AIDS-related death at 5 years.

The setting

Time-to-event outcome

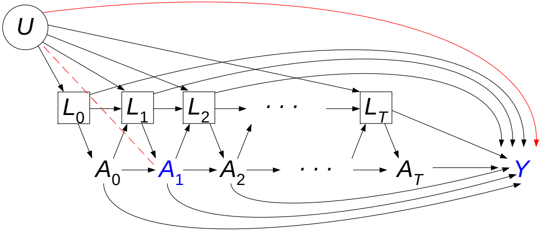


Problem with regression (1)



- What happens if we control for L in a regression model?
- Focus on the effect of A_1.
- Controlling for L_1 has blocked the red non-causal paths.
- But controlling for L_2 has blocked the blue causal pathway from A_1 to Y.

Problem with regression (2)



- In addition, since L_2 is the common effect of U and A_1, conditioning on it induces an association between them.
- This opens up an additional non-causal path.
- Thus the coefficients of $\{A_0, \ldots, A_{T-1}\}$ in a regression of Y on $\{A_0, \ldots, A_T\}$ and $\{L_0, \ldots, L_T\}$ cannot be given a causal interpretation. (NB the coefficient of A_T is OK).

Outline

1 Time-dependent confounding

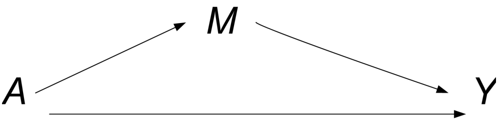
2 Mediation

3 Notation, assumptions and causal questions

4 G-computation formula

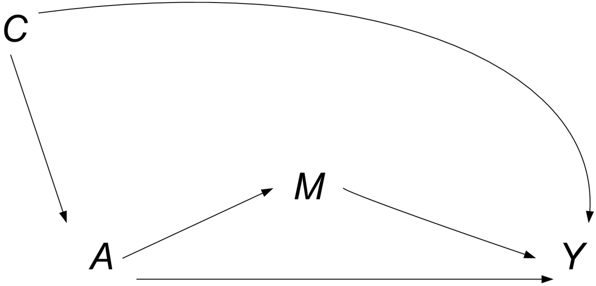
5 gformula in Stata

The mediation setting



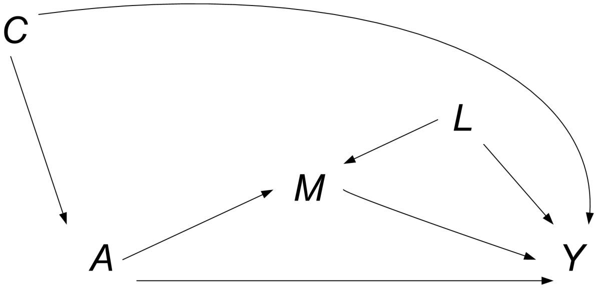
In the mediation setting, we are interested in separating the causal effect of A on Y into an effect through M (indirect) and an effect not through M (direct).

The mediation setting



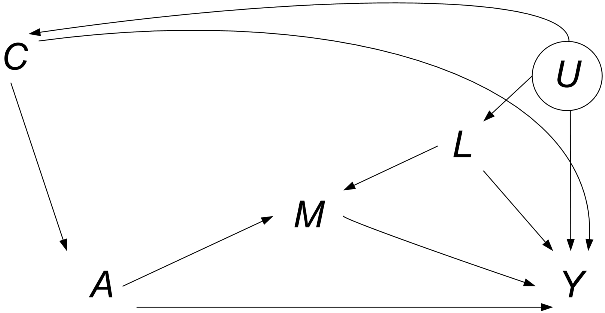
Typically there will be exposure–outcome confounding.

The mediation setting



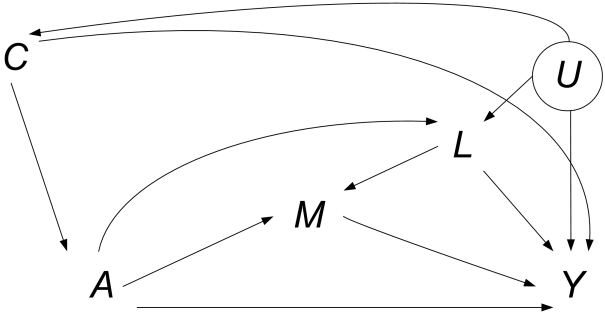
As well as mediator–outcome confounding.

The mediation setting



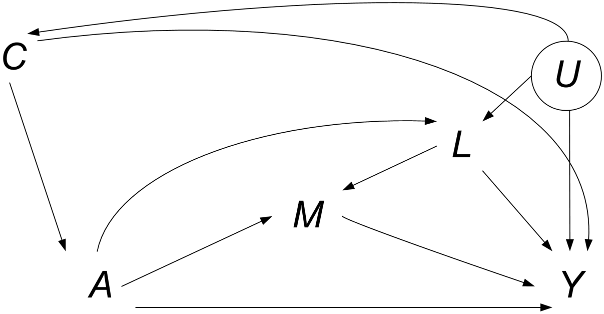
These confounders need not be purely causal for the outcome.

The mediation setting



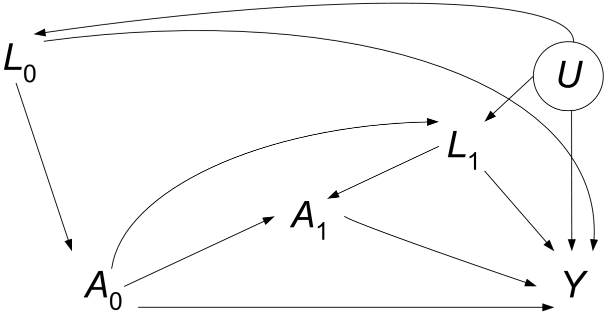
Standard methods fail when the mediator–outcome confounders are affected by the exposure.

The link between the two settings



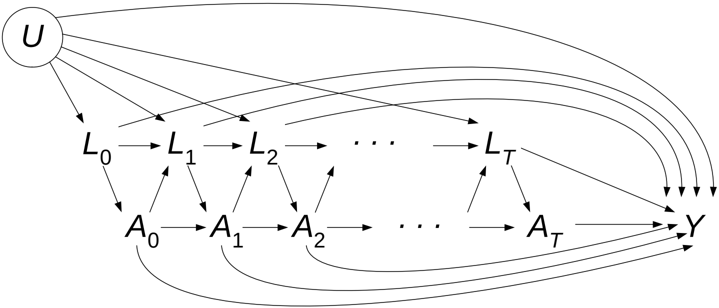
Changing the labels. . .

The link between the two settings



. . . we see that this setting is a special case of. . .

The link between the two settings



. . . the time-dependent confounding setting.

Outline

1 Time-dependent confounding

2 Mediation

3 Notation, assumptions and causal questions

4 G-computation formula

5 gformula in Stata

The actual data

- For each subject we observe:
  - The exposure at each of $T+1$ occasions: $A_0, A_1, \ldots, A_t, \ldots, A_T$.
  - The confounder at each of $T+1$ occasions: $L_0, L_1, \ldots, L_t, \ldots, L_T$ where L_t is measured just before A_t for each t.
  - The outcome, Y, measured on the $(T+1)$st occasion.
- We write $\bar{A}_t = (A_0, A_1, \ldots, A_t)$ for the *history* of A up to time t.
- Similarly, we write $\bar{L}_t = (L_0, L_1, \ldots, L_t)$ for the history of L up to time t.
- We also use the shorthand $\bar{A}$ for $\bar{A}_T$ and $\bar{L}$ for $\bar{L}_T$.

The counterfactual data

- For every possible value $\bar{a}$ of $\bar{A}$, we write $Y^{\bar{a}}$ for the *potential outcome* associated with $\bar{a}$, *i.e.* the value that Y would have taken, had exposure been manipulated to $\bar{a}$.
- We only observe $Y = Y^{\bar{A}}$. All the other potential outcomes are *counterfactual*.

Key Assumption

To make progress in estimating the causal effect of $\bar{A}$ on Y, we will need to assume:

- *No unmeasured confounders*

$$A_t \perp\!\!\!\perp Y^{\bar{a}} \big| \bar{A}_{t-1}, \bar{L}_t \ \forall t, \bar{a}$$

What does this mean?

We are really saying that the observational study needs to be 'close' to a *conditionally sequentially randomised trial*, where, at each time t, we look at a patient's history up to that point, and use this history to determine how to weight a biased coin, which then determines A_t.

Causal questions

- Causal inference in this setting involves the comparison of some aspect(s) of the distribution of $Y^{\bar{a}}$, eg $E\left(Y^{\bar{a}}\right)$, for different values of $\bar{a}$.
- We may ask which of the following regimes:
  - $\bar{a} = (1, 1, 1, \ldots, 1)$
  - $\bar{a} = (0, 0, 0, \ldots, 0)$
  - $\bar{a} = (1, 0, 1, 0, \ldots)$
  - $\bar{a} = (0, 1, 0, 1, \ldots)$
  - $\ldots$

  is optimal to minimise (maximise), say, $E\left(Y^{\bar{a}}\right)$.
- We may also be interested in dynamic regimes:
  - At what level of CD4 count should we start treating with ART?
- For the mediation setting, specific comparisons of potential outcomes correspond to direct and indirect effects. (See Bianca's talk).

A marginal structural model

- For time-varying exposures, comparing each pair of expected potential outcomes is infeasible (because there are so many POs).
- We can instead summarise these comparisons by using a *marginal structural model*:

$$E\left(Y^{\bar{a}}\right) = g\left(\bar{a};\gamma\right)$$

MSMs: examples

- Examples of MSMs:

$$E\left(Y^{\bar{a}}\right)=\gamma_0+\gamma_1\sum_{t=0}^{T}a_t \tag{1}$$

$$E\left(Y^{\bar{a}}\right)=\gamma_0+\gamma_1 a_T \tag{2}$$

$$E\left(Y^{\bar{a}}\right)=\gamma_0+\gamma_1 a_T+\gamma_2 a_{T-1}+\gamma_3 a_T a_{T-1}+\gamma_4\sum_{t=0}^{T-2}a_t \tag{3}$$

- $\gamma_1=0$ in (1) & (2) and $\gamma_1=\gamma_2=\gamma_3=\gamma_4=0$ in (3) correspond to the causal null hypothesis.

MSMs: more examples

- Logistic MSM:

$$E\left(Y^{\bar{a}}\right)=\frac{\exp\left(\gamma_0+\gamma_1\sum_{t=0}^{T}a_t\right)}{1+\exp\left(\gamma_0+\gamma_1\sum_{t=0}^{T}a_t\right)}$$

- Marginal structural Cox model:

$$\lambda_{T_{\bar{a}}}(t)=\lambda_0(t)\exp\left(\gamma a_t\right)$$

where $T_{\bar{a}}$ is the counterfactual time-to-event under exposure $\bar{a}$ and $\lambda_0(t)$ is an unspecified baseline hazard function.

Outline

1 Time-dependent confounding

2 Mediation

3 Notation, assumptions and causal questions

4 G-computation formula

5 gformula in Stata

G-methods

- Jamie Robins and colleagues have introduced three different methods for estimating causal effects in the presence of time-dependent confounding.
- The g-computation formula (Robins 1986, Mathematical Modelling).
- Inverse probability weighting of marginal structural models (Robins et al 2000, Epidemiology).
- G-estimation of structural nested models (Robins et al 1992, Epidemiology).

The g-computation formula

$$E\left(Y^{\bar{a}}\right) = \sum_{(l_0,\dots,l_T)} \left\{ E\left(Y \left| \bar{A} = \bar{a}, \bar{L} = \bar{l}\right.\right) \cdot \prod_{t=0}^{T} Pr\left(L_t = l_t \left| \bar{A}_{t-1} = \bar{a}_{t-1}, \bar{L}_{t-1} = \bar{l}_{t-1}\right.\right) \right\}$$

- Conditional expectations and distributions estimated using conditional univariate regression models.
- Marginalising over the conditional distribution of $L_t \left| \bar{A}_{t-1}, \bar{L}_{t-1}\right.$ deals appropriately with the time-dependent confounding.
- Summation replaced by integration when L_t continuous.
- Monte Carlo simulation when integral analytically intractable.
- This is what `gformula` does.

Outline

1 Time-dependent confounding

2 Mediation

3 Notation, assumptions and causal questions

4 G-computation formula

5 gformula in Stata

The data structure (1)

```
------------------------------------------------
id  t   y  l     a  cuma  a_lag  cuma_lag  l_lag
------------------------------------------------
1   0   .  5.20  1  1     0      0         0
1   1   0  5.52  1  2     1      1         5.20
1   2   0  5.95  0  2     1      2         5.52
1   3   0  5.23  1  3     0      2         5.95
1   4   0  5.62  0  3     1      3         5.23
1   5   0  4.96  1  4     0      3         5.62
1   6   1  5.47  1  5     1      4         4.96
------------------------------------------------
2   0   .  4.69  0  0     0      0         0
2   1   0  4.06  0  0     0      0         4.69
2   2   1  3.42  1  1     0      0         4.06
------------------------------------------------
```

The data structure (2)

```
------------------------------------------------
id  t   y  l     a  cuma  a_lag  cuma_lag  l_lag
------------------------------------------------
...
3   0   .  6.05  0  0     0      0         0
3   1   0  5.41  0  0     0      0         6.05
3   2   0  4.75  1  1     0      0         5.41
3   3   0  5.16  1  2     1      1         4.75
3   4   0  5.67  0  2     1      2         5.16
3   5   0  5.17  1  3     0      2         5.67
3   6   0  5.55  1  4     1      3         5.17
3   7   0  6.21  0  4     1      4         5.55
3   8   0  5.48  0  4     0      4         6.21
3   9   0  4.90  0  4     0      4         5.48
3   10  0  .     .  .     0      4         4.90
------------------------------------------------
```

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

All the variables involved in the analysis are listed here.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The outcome variable.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The RHS of the equations to be used for simulation.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The commands associated with these equations.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The id variable.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The time variable.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The time-changing covariates.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The intervention variables.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The interventions to be compared.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

All associational models are to be fitted after pooling across visits.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

Lagged variables.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The rules for generating them.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The MSM.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

Derived variables.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The rules for generating them.

The gformula syntax

Example I

The gformula command

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
interventions(a=1 if t<10, a=0 if t<=1 \ a=1 if t>1 & t<10, a=0
if t<=3 \ a=1 if t>3 & t<10, a=0 if t<=5 \ a=1 if t>5 & t<10,
a=0 if t<=7 \ a=1 if t>7 & t<10, a=0 if t<=9) pooled
laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l 1, a_lag:a 1,
cuma_lag:cuma 1) msm(stcox cuma_lag) derived(cuma)
derrules(cuma:cuma_lag+a) seed(79)
```

Explanation

The seed.

Results (1)

Example I

The output of the gformula command: MSM

```
G-computation formula estimates for the parameters of the specified marginal structural model

Specified MSM: stcox cuma_lag

---------------------------------------------------------------------------
           G-computation
           estimate of    Bootstrap                 Normal-based
y          Coef.          Std. Err.   z        P>|z|   [95% Conf. Interval]
---------------------------------------------------------------------------
cuma_lag   -.4620501      .0426871    -10.82   0.000   -.5457153 -.3783849
---------------------------------------------------------------------------
```

Results (2)

Example I

The output of the gformula command: log IR

```
G-computation formula estimates of the average log incidence rates under each of the specified
  interventions and under no intervention (i.e. as simulated under the observational regime).
  For comparison, the average log incidence rate in the observed data is also shown.

    Specified interventions:
      Intervention 1: a=1 if t<10
      Intervention 2: a=0 if t<=1 \ a=1 if t>1 & t<10
      Intervention 3: a=0 if t<=3 \ a=1 if t>3 & t<10
      Intervention 4: a=0 if t<=5 \ a=1 if t>5 & t<10
      Intervention 5: a=0 if t<=7 \ a=1 if t>7 & t<10
      Intervention 6: a=0 if t<=9

-------------------------------------------------------------------------------
            G-computation
            estimate of     Bootstrap                          Normal-based
y           av. log IR      Std. Err.    z         P>|z|     [95% Conf. Interval]
-------------------------------------------------------------------------------
Int. 1      -3.710399       .1178156     -31.49    0.000     -3.941313 -3.479485
Int. 2      -2.849232       .0737148     -38.65    0.000     -2.99371  -2.704754
Int. 3      -2.409732       .0742438     -32.46    0.000     -2.555247 -2.264216
Int. 4      -2.155157       .0708308     -30.43    0.000     -2.293983 -2.016331
Int. 5      -1.992489       .0690772     -28.84    0.000     -2.127878 -1.8571
Int. 6      -2.010118       .0656089     -30.64    0.000     -2.138709 -1.881526
-------------------------------------------------------------------------------
Obs. regime
simulated   -2.693125       .0648117     -41.55    0.000     -2.820153 -2.566096
observed    -2.585342
-------------------------------------------------------------------------------
```

Results (3)

Example I

The output of the gformula command: cumulative incidence

```
G-computation formula estimates of the cumulative incidence under each of the specified
  interventions and under no intervention (i.e. as simulated under the observational
  regime). For comparison, the cumulative incidence in the observed data is also shown.

    Specified interventions:
      Intervention 1: a=1 if t<10
      Intervention 2: a=0 if t<=1 \ a=1 if t>1 & t<10
      Intervention 3: a=0 if t<=3 \ a=1 if t>3 & t<10
      Intervention 4: a=0 if t<=5 \ a=1 if t>5 & t<10
      Intervention 5: a=0 if t<=7 \ a=1 if t>7 & t<10
      Intervention 6: a=0 if t<=9

--------------------------------------------------------------------------------
            G-computation
            estimate of     Bootstrap                           Normal-based
y           cum. incidence  Std. Err.    z        P>|z|     [95% Conf. Interval]
--------------------------------------------------------------------------------
Int. 1      .208            .0217588     9.56     0.000     .1653535   .2506465
Int. 2      .408            .0211903     19.25    0.000     .3664678   .4495322
Int. 3      .565            .0242743     23.28    0.000     .5174232   .6125768
Int. 4      .677            .0251431     26.93    0.000     .6277205   .7262795
Int. 5      .77             .0256334     30.04    0.000     .7197594   .8202406
Int. 6      .782            .0248577     31.46    0.000     .7332798   .8307202
--------------------------------------------------------------------------------
Obs. regime
simulated   .486            .0222683     21.82    0.000     .4423549   .5296451
observed    .519
--------------------------------------------------------------------------------
```

The gformula syntax

Example II

The gformula command: dynamic regimes

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
dynamic interventions(a=0 if t<10 & l>6.9 \ a=1 if t<10 &
l<=6.9, a=0 if t<10 & l>6.55 \ a=1 if t<10 & l<=6.55, a=0 if
t<10 & l>6.2 \ a=1 if t<10 & l<=6.2, a=0 if t<10 & l>5.3 \
a=1 if t<10 & l<=5.3, a=0 if t<10 & l>4.6 a=1 if t<10 &
l<=4.6) pooled laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l
1, a_lag:a 1, cuma_lag:cuma 1) derived(cuma)
derrules(cuma:cuma_lag+a) seed(801)
```

The gformula syntax

Example II

The gformula command: dynamic regimes

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
dynamic interventions(a=0 if t<10 & l>6.9 \ a=1 if t<10 &
l<=6.9, a=0 if t<10 & l>6.55 \ a=1 if t<10 & l<=6.55, a=0 if
t<10 & l>6.2 \ a=1 if t<10 & l<=6.2, a=0 if t<10 & l>5.3 \
a=1 if t<10 & l<=5.3, a=0 if t<10 & l>4.6 a=1 if t<10 &
l<=4.6) pooled laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l
1, a_lag:a 1, cuma_lag:cuma 1) derived(cuma)
derrules(cuma:cuma_lag+a) seed(801)
```

Explanation

Compare dynamic regimes.

The gformula syntax

Example II

The gformula command: dynamic regimes

```
gformula y a l a_lag l_lag cuma cuma_lag id t, out(y) eq(y:l_lag
cuma_lag, l:l_lag a_lag, a:l a_lag) com(y:logit, l:regress,
a:logit) idvar(id) tvar(t) varyingcovariates(l) intvars(a)
dynamic interventions(a=0 if t<10 & l>6.9 \ a=1 if t<10 &
l<=6.9, a=0 if t<10 & l>6.55 \ a=1 if t<10 & l<=6.55, a=0 if
t<10 & l>6.2 \ a=1 if t<10 & l<=6.2, a=0 if t<10 & l>5.3 \
a=1 if t<10 & l<=5.3, a=0 if t<10 & l>4.6 a=1 if t<10 &
l<=4.6) pooled laggedvars(l_lag a_lag cuma_lag) lagrules(l_lag:l
1, a_lag:a 1, cuma_lag:cuma 1) derived(cuma)
derrules(cuma:cuma_lag+a) seed(801)
```

Explanation

The interventions to be compared.

Summary (1)

- Controlling for confounders of later relationships affected by earlier exposures is problematic using standard methods.
- This situation arises often in practice, when investigating causal effects of time-changing exposures, and when disentangling effects into path-specific components.
- One method for addressing this issue under the assumption of no unmeasured confounding is the g-computation formula.
- When implemented by Monte Carlo simulation, it is very flexible, allowing dynamic as well as static regimes to be compared.
- Multivariate exposures and confounders of all types, and continuous, binary, time-to-event outcomes can all be dealt with, and the form of the specified models is flexible too.

Summary (2)

- The `gformula` command in Stata allows us to implement this procedure.
- It is heavy on parametric assumptions; in particular, we must specify a model for each $\left[L_t \,\middle|\, \bar{L}_{t-1}, \bar{A}_{t-1}\right]$.
- Alternative semiparametric methods (IPW of MSMs, g-estimation of SNMs) avoid this need.

References (1)

Robins JM (1986)
A new approach to causal inference in mortality studies with sustained exposure periods — Application to control of the healthy worker survivor effect.
Mathematical Modelling, 7:1393–1512.

Robins JM, Hernán MA (2009)
Estimation of the causal effects of time-varying exposures.
In *Longitudinal Data Analysis*, Fitzmaurice G, Davidian M, Verbeke G, Molenberghs G (eds). New York: Chapman and Hall/CRC Press; 553-599.

References (2)

- Taubman SL, Robins JM, Mittleman MA and Hernán MA (2009)
  Intervening on risk factors for coronary heart disease: an application of the parametric g-formula.
  International Jounral of Epidemiology, 38:1599–1611.

- Daniel RM, De Stavola BL, Cousens SN (2011)
  gformula: Estimating causal effects in the presence of time-varying confounding or mediation using the g-computation formula.
  The Stata Journal, 11(4):479–517.