G-formula via multiple imputation

Jonathan W. Bartlett

Department of Medical Statistics
LSHTM
www.thestatsgeek.com

Centre for Statistical Methodology Seminar
LSHTM
8th February 2023

Acknowledgements

This is joint work with Camila Olarte Parra (LSHTM) and Rhian Daniel (Cardiff).

This work was supported by a UK Medical Research Council grant (MR/T023953/1).

This is work in progress...

Outline

Brief overview

In many settings we are interested in effects of time-varying treatments or exposures.

Our exploration of this area stems from analysis of RCTs where patients may take rescue medication or discontinue randomised treatment.

Rescue treatment over time then constitutes the time-varying treatment. Here, we were interested in estimating the effects of randomised treatment, removing effects of subsequent rescue.

But of course there are also many observational settings where the treatment or exposure received may vary over time.

Brief overview

In this setting, we generally need to adjust for confounders.

Not just baseline confounders, but also time-varying confounders.

To do this 'correctly' requires use of so called G-methods, developed by Jamie Robins and co-workers [2, 1].

This talk will be about one of these methods, called **G-formula** (or sometime G-computation).

I will show how this can be implemented using multiple imputation methods.

Moreover, I will describe how we can accommodate missing data in exposures and confounders as part of the procedure.

Outline

Time-varying treatments and confounders

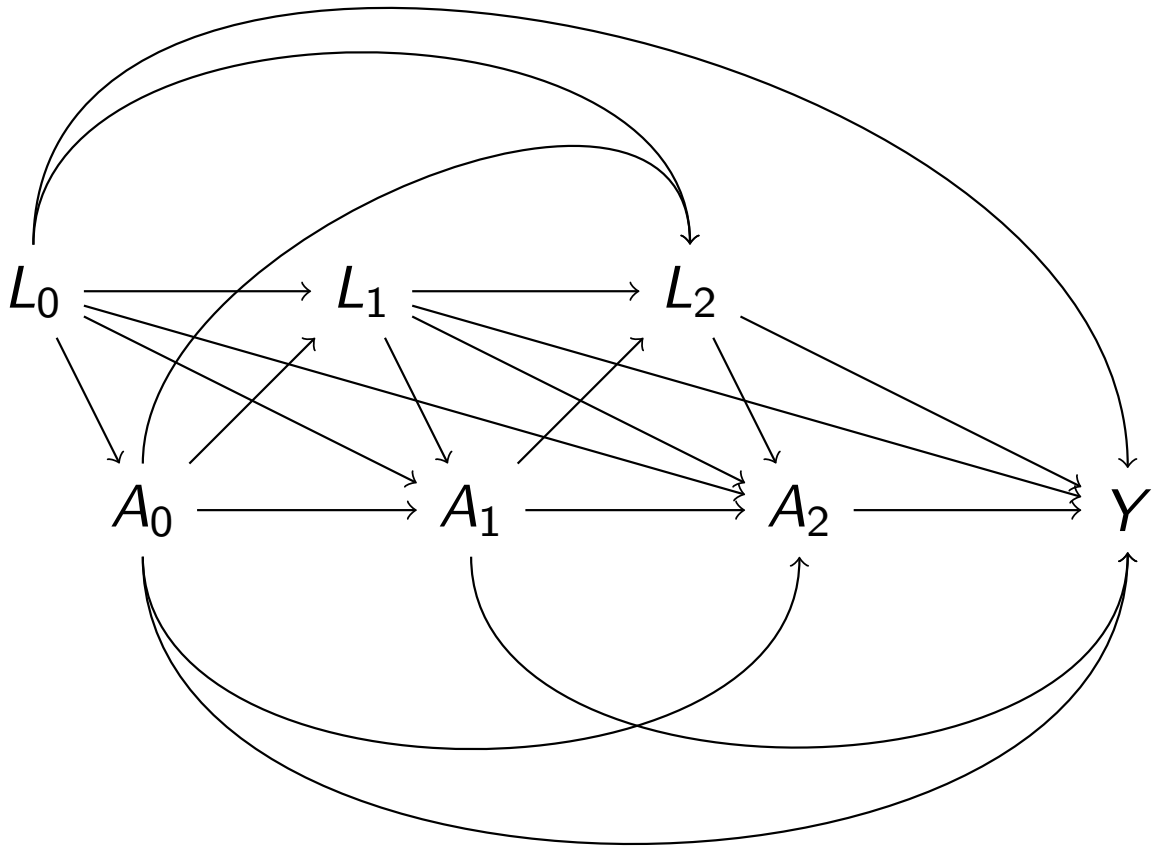
The setting under consideration is the 'standard' time-varying treatment and confounding setup.

A_k denotes treatment at time/visit k, $k = 0, \ldots, K$.

L_k denotes time-varying confounders at visit k.

Y denotes the final outcome of interest.

Directed acyclic graph (DAG)



Potential outcomes and estimands

Let Y^{a_0,a_1,a_2} denote potential outcome if baseline treatment A_0 is set to value a_0, treatment at time 1 A_1 is set to a_1, and treatment at time 2 A_2 is set to a_2.

Causal estimands are then constrasts of the distributions of Y^{a_0,a_1,a_2} for different values of a_0, a_1, a_2.

For example, the effect of treatment at all times versus no treatment is

$$E(Y^{1,1,1}) - E(Y^{0,0,0})$$

Identification assumptions

Consistency: Interventions on treatment/exposure well defined so that we can assume $Y = Y^{a_0,a_1,a_2}$ if $A_0 = a_0$, $A_1 = a_1$, $A_2 = a_2$

Conditional exchangeability (no unmeasured confounding)

$$Y^{a_0,a_1,a_2} \perp\!\!\!\perp A_k | \bar{A}_{k-1} = \bar{a}_{k-1}, \bar{L}_k$$

This holds under the earlier DAG. The key is that we measure all common causes of time-varying treatment and final outcome.

Outline

Estimation of causal estimands

So called G-methods have been developed by Jamie Robins and colleagues for estimating causal estimands in this setting [2, 1]:

- G-formula (sometimes known as G-computation)
- Inverse probability weighting
- G-estimation of structural nested models

In this talk I will focus on G-formula...

G-formula

Estimation of $E(Y^{\bar{a}}) = E(Y^{a_0,a_1,a_2})$ is based on

$$E(Y^{\bar{a}}) = \int_{l_0} \int_{l_1} \int_{l_2} E(Y|a_0, a_1, a_2, l_0, l_1, l_2) f(l_2|a_0, a_1, l_0, l_1) f(l_1|a_0, l_0) f(l_0) dl_2 dl_1 dl_0$$

This requires we specify and fit models for

- $f(L_0)$ (in fact, we typically empirically average across this, avoiding need for a model)
- $f(L_1|A_0, L_0)$
- $f(L_2|A_0, A_1, L_0, L_1)$
- $f(Y|A_0, A_1, A_2, L_0, L_1, L_2)$ (in fact, all we need is a model for $E(Y|A_0, A_1, A_2, L_0, L_1, L_2)$)

In general the integrals above are intractable.

Thus in practice implementations (e.g. gformula in `Stata`) use Monte-Carlo integration.

G-formula by Monte-Carlo integration/simulation

To estimate $E(Y^{a_0,a_1,a_2})$, based on fitted models, for every individual we:

- simulate L_0^* from $f(L_0)$ (or just use original, i.e. $L_0^* = L_0$)
- simulate L_1^* from $f(L_1|A_0 = a_0, L_0^*)$
- simulate L_2^* from $f(L_2|A_0 = a_0, A_1 = a_1, L_0^*, L_1^*)$
- simulate Y^* from $f(Y|A_0 = a_0, A_1 = a_1, A_2 = a_2, L_0^*, L_1^*, L_2^*)$

  (or just calculate $E(Y|A_0 = a_0, A_1 = a_1, A_2 = a_2, L_0^*, L_1^*, L_2^*)$
- calculate mean of Y^* across individuals

  (or average $E(Y|A_0 = a_0, A_1 = a_1, A_2 = a_2, L_0^*, L_1^*, L_2^*)$ across individuals)

G-formula and imputation

For those (like me) more familar with missing data methods, G-formula can be seen as a form of single stochastic imputation of the longitudinal history under the treatment regime of interest.

In fact, to reduce Monte-Carlo error, implementations of G-formula create multiple imputations of these, and then average the imputed Y^* across individuals and across imputations.

For inference, implementations in Stata and R use non-parametric bootstrapping.

The close links between MI and G-formula by simulation begs the question - could we use Rubin's combination rules, rather than bootstrapping, for inference?

Outline

G-formula via multiple imputation - earlier work

Westreich *et al* previously highlighted close connections between G-formula and MI in single time point setting [5]:

L	A	Y^0	Y^1
2.5	0	3.4	NA
7.3	1	NA	4.5
4.6	1	NA	5.7
4.2	0	4.2	NA

Impute Y^0 for those with $A = 1$, using L. Impute Y^1 for those with $A = 0$, using L.

Calculate difference in means of Y^1 and Y^0 in imputed datasets.

Westreich *et al* stated that Rubin's variance estimator cannot be used because each individual contributes to both treated and untreated calculations.

G-formula via multiple imputation - earlier work

Here individual's observed outcome is retained.

In the longitudinal setting, G-formula simulates confounders and outcomes for all individuals afresh.

Indeed, in this setting, it may be that no individuals followed precisely the treatment regime of interest.

We can nonetheless use MI to implement G-formula...

G-formula via MI

For the longitudinal setup earlier, we can use MI to estimate $\mu = E(Y^{\bar{a}})$ in a G-formula type approach by:

1. Augment observed data with additional n_{syn} rows, setting L_0, L_1, L_2, Y to missing in the augmented rows to missing, and A_0, A_1, A_2 to value $\bar{a} = (a_0, a_1, a_2)$.
2. Run MI on the augmented dataset, generating M imputations.
3. For imputation m ($m = 1, \ldots, M$), calculate mean of Y from the augmented part of the dataset.
4. Average estimated means across M imputations (denoted $\hat{\mu}$) as estimator of $\mu = E(Y^{\bar{a}})$.

G-formula via MI - data structure

E.g. data structure for $\bar{a} = (1, 1, 1)$ is

R	L_0	A_0	L_1	A_1	L_2	A_2	Y
1	-0.3	0	0.5	0	2.2	1	1.3
1	2.3	1	4.2	1	4.6	1	5.5
1	-0.5	1	0.4	0	0.8	1	1.9
0	NA	1	NA	1	NA	1	NA
0	NA	1	NA	1	NA	1	NA
0	NA	1	NA	1	NA	1	NA

$R = 1$ indicates originally observed data

$R = 0$ indicates augmented data

G-formula via MI - implementation details

We have a block monotone missingness pattern in the augmented dataset.

Due to our earlier model assumptions, we can impute sequentially moving forwards in time:

1. Impute L^0
2. Impute $L_1|A_0, L_0$
3. Impute $L_2|A_0, A_1, L_0, L_1$
4. Impute $Y|A_0, A_1, A_2, L_0, L_1, L_2$

This means if we use for example chained equations MI software, there is no need to iterate around models.

We specify imputation equations as per above, and set iterations to 1.

Contrasts of treatment regimes

In practice we are interested in contrasts of the form $E(^{\bar{a}_1}) - E(^{\bar{a}_2})$ for regimes $\bar{a}_1$ and $\bar{a}_2$.

To estimate this, add augmented rows with $\bar{A} = \bar{a}_1$ and another set with $\bar{A} = \bar{a}_2$.

In the imputed datasets, calculate difference in sample means.

Inference for G-formula via MI estimator

How to estimate $\text{Var}(\hat{\mu})$ and conduct inference?

Ordinarily with MI we use Rubin's rules.

Estimate variance in each imputation and average these, yielding within-imputation variance $\hat{V}$.

Estimate variance of estimated means across M imputations, yielding between-imputation variance $\hat{B}$.

Then $\widehat{\text{Var}}(\hat{\mu}) = (1 + M^{-1})\hat{B} + \hat{V}$.

Unfortunately this does not work here - Rubin's variance estimator is much larger than the true $\text{Var}(\hat{\mu})$.

This is due to a form of uncongeniality - the imputation and analysis models are being fitted to different portions of the dataset.

Multiple imputation for synthetic samples/populations

Within the survey sampling field, there is an established literature on using MI to impute partially or fully synthetic datasets.

The motivation here is concern over confidentiality if survey data were released to analysts.

MI is used to impute/simulate variables for new/synthetic individuals, ensuring confidentiality of original participants.

I will describe MI for generating synthetic samples/populations, based on work by Raghunathan, Reiter and Rubin [3].

Inference for finite population setup

We have data from a sample of size n_{obs} from a finite population of size N.

The population data consist of $\mathcal{P} = (X, Y)$ where $X = (X_i; i = 1, \ldots, N)$ and $Y = (Y_i; i = 1, \ldots, N)$.

X corresponds to background / administrative information, assumed known for all N members of the population.

Y is only observed in those sampled from the population.

$Y_{inc} = (Y_i; i = 1, \ldots, n_{\text{obs}})$ denotes the observed values of Y from the sample.

$Y_{exc} = (Y_i; i = n_{\text{obs}} + 1, \ldots, N)$ denotes the unobserved values of Y for those individuals not in the sample.

Synthetic MI algorithm

Impute/simulate data for M synthetic populations of size N by drawing their data from the posterior predictive distribution given the observed sample (size n_{obs}).

The population size N is typically too large to release to analysts.

Thus instead Raghunathan *et al* propose drawing a random sample (of size n_{syn} we can choose) from each synthetic population, and releasing these.

Analysing samples from synthetic populations

We estimate the parameter of interest and a corresponding variance from each imputation, as usual in MI.

The variance estimate is

$$(1 + M^{-1})\hat{B} - \hat{V}$$

where $\hat{B}$ and $\hat{V}$ are between and within-imputation variance.

Note this is not the same as Rubin's MI variance estimator, which is

$$(1 + M^{-1})\hat{B} + \hat{V}$$

Estimating normal mean with known variance

Raghunathan *et al* demonstrated that Rubin's variance estimator is biased upwards (considerably) when used with the synthetic MI approach.

As mentioned earlier, this is due to uncongeniality between the imputation and analysis procedures.

To see concretely the issue, consider the simple setting of estimating the mean μ of a normal distribution with known variance σ^2, based on a sample of size n_{obs} with observed sample mean $\bar{Y}$.

Estimating normal mean with known variance

We use synthetic imputation, creating samples each of size n_{syn}.

To do this, for $m = 1, \ldots, M$, we first take a posterior draw $\tilde{\mu}_{(m)} \sim N(\bar{Y}, \frac{\sigma^2}{n_{\text{obs}}})$.

In imputation m, we impute for $i = 1, \ldots n_{\text{syn}}$ as

$$Y_{i(m)} = \tilde{\mu}_{(m)} + \epsilon_{i(m)} \text{ where } \epsilon_{i(m)} \sim N(0, \sigma^2)$$

We then estimate μ by its sample mean in each imputation and average these across the M imputation:

$$\hat{\mu} = \frac{1}{M} \sum_{m=1}^{M} \hat{\mu}_m$$

Estimating normal mean with known variance

One can show (see pre-print) that

$$\mathrm{Var}(\hat{\mu}) = \frac{\sigma^2}{n_{\mathrm{syn}} M} + (1 + M^{-1}) \frac{\sigma^2}{n_{\mathrm{obs}}}$$

As $M \to \infty$, this converges to $\frac{\sigma^2}{n_{\mathrm{obs}}}$, the variance of the observed sample mean.

Estimating normal mean with known variance

The within-imputation variance is $\hat{V} = \frac{\sigma^2}{n_{\text{syn}}}$.

We show the between-imputation $\hat{B}$ unbiasedly estimates $\frac{\sigma^2}{n_{\text{obs}}} + \frac{\sigma^2}{n_{\text{syn}}}$.

Thus Rubin's variance estimates

$$\begin{aligned}(1 + M^{-1})\hat{B} + \hat{V} &= (1 + M^{-1})\left\{\frac{\sigma^2}{n_{\text{obs}}} + \frac{\sigma^2}{n_{\text{syn}}}\right\} + \frac{\sigma^2}{n_{\text{syn}}} \\ &= \text{Var}(\hat{\mu}) + \frac{2\sigma^2}{n_{\text{syn}}}\end{aligned}$$

While Raghunathan *et al* 's variance estimator estimates

$$\begin{aligned}(1 + M^{-1})\hat{B} - \hat{V} &= (1 + M^{-1})\left\{\frac{\sigma^2}{n_{\text{obs}}} - \frac{\sigma^2}{n_{\text{syn}}}\right\} - \frac{\sigma^2}{n_{\text{syn}}} \\ &= \text{Var}(\hat{\mu})\end{aligned}$$

Synthetic MI pooling rules for G-formula MI

In our pre-print, we use asymptotic theory for MI estimators of Robins and Wang [4] to show that Raghunathan *et al* 's variance estimator is asymptotically unbiased for the G-formula via MI estimator.

Simulation setup

To evaluate G-formula via MI approach, we performed simulations.

$n_{\text{obs}} = n_{\text{syn}} = 500$

$10,000$ simulations per scenario.

We simulated with two intermediate follow-ups and a final outcome Y.

Sequential imputation using `mice` package in R, with M synthetic imputations.

Simulation setup

$$
\begin{aligned}
L_0 &\sim N(0,1) \\
P(A_0 = 1|L_0) &= \text{expit}(L_0) \\
L_1 &\sim N(A_0 + L_0, 1) \\
P(A_1 = 1|A_0, L_0, L_1) &= \text{expit}(A_0 + L_1) \\
L_2 &\sim N(A_1 + L_1, 1) \\
P(A_2 = 1|A_0, A_1, L_0, L_1, L_2) &= \text{expit}(A_1 + L_2) \\
Y &\sim N(A_2 + L_2, 1)
\end{aligned}
$$

We target $E(Y^{1,1,1}) - E(Y^{0,0,0})$, which has true value 3.

Number of imputations M

The variance estimator $(1 + M^{-1})\hat{B} - \hat{V}$ can be negative, due to noise in $\hat{B}$ as estimate of true between-imputation variance.

To examine how large M needs to be, we evaluated $M = \{5, 10, 25, 50, 100\}$.

If on a given dataset the estimated variance was negative, we added new sets of M imputations until it became non-negative.

We report the mean and max. value of M required across the $10,000$ simulations.

Simulation results

M	Bias	Emp. SE	Est. SE	95% CI	Mean M	Max M
5	-0.002	0.242	0.236	99.4	6.2	25
10	0.001	0.229	0.223	98.4	10.4	30
25	0.000	0.223	0.220	95.6	25.0	50
50	0.000	0.217	0.219	95.2	50.0	50
100	0.004	0.218	0.219	95.0	100.0	100

- Estimates are unbiased for true effect ($= 3$).
- For $M \geq 25$ 95% coverage is reasonable (but more simulations needed).
- Negative variance issue is rare with M as low as 25, and never occurred with $M \geq 50$.

Interim conclusions

MI provides a potentially convenient route to performing G-formula, exploiting existing software for MI.

Inference seems reliable for M as low as 25, which is computationally far fewer than number of bootstraps typically used (e.g. $1,000$).

Outline

G-formula and missing data

In practice there will typically (always?!) be some missing data on baseline and time-varying confounders and treatment variables.

Given we've seen that MI can be used to perform G-formula when data are complete, can we use it to impute any missing data as well?

Yes. We impute the combination of the missing actual data and the missing potential outcome data (in the augmented part).

Imputing missing data with MI G-formula

There are (at least) two approaches:

1. Impute missing actual data and missing potential outcome data in one go.
2. First impute missing actual data, then impute missing potential outcome data conditional on these imputations.

We believe option 2 is more attractive.

We only have to impute the (usually) small amount of missing actual data to create a monotone pattern. The remaining missing (potential outcome) values are then imputed as before, with no iterations required at the second stage.

This approach is well established in the context of using MI with longitudinal data where the missingness pattern is almost monotone.

Simulation setup

To the previous setup, we made some values in L_1, A_1, L_2, A_2, Y independently missing completely at random.

We varied the probability $p = \{0.05, 0.1, 0.25, 0.5\}$ that values in each were missing. Note p refers to the marginal probability that values in **each** variable are missing.

We imputed missing values $M = 50$ using `mice` with default settings.

For each imputed dataset, we then augmented as described previously, and imputed missing potential outcomes once as described earlier.

Inferences were then again based on Raghunathan's variance estimator.

Simulation results

π	Bias	Emp. SE	Mean est. SE	95% CI
0.05	-0.001	0.225	0.224	95.4
0.10	-0.003	0.231	0.231	95.3
0.25	-0.008	0.259	0.258	95.4
0.50	-0.011	0.360	0.361	95.0

- As expected, estimator becomes more variable with increasing missingness.
- Raghunathan variance estimator and 95% CI performing well.

Outline

Conclusions

- MI seems like an attractive route to implementing G-formula, particularly when some data are missing.
- More details in our pre-print `https://arxiv.org/abs/2301.12026`
- R package implementation which utilises `mice` in gFormulaMI
- Here we considered static (fixed) treatment regimes.
- For dynamic treatment regimes, specify treatment rules via custom imputation method functionality in `mice` package.

References I

[1] M. Hernán and J. Robins.
Causal Inference: What If.
Boca Raton: Chapman & Hall/CRC, 2020.

[2] A. I. Naimi, S. R. Cole, and E. H. Kennedy.
An introduction to g methods.
International Journal of Epidemiology, 46(2):756–762, 2017.

[3] T. E. Raghunathan, J. P. Reiter, and D. B. Rubin.
Multiple imputation for statistical disclosure limitation.
Journal of Official Statistics, 19(1):1, 2003.

[4] J. M. Robins and N. Wang.
Inference for imputation estimators.
Biometrika, 85:113–124, 2000.

References II

[5] D. Westreich, J. K. Edwards, S. R. Cole, R. W. Platt, S. L. Mumford, and E. F. Schisterman.

Imputation approaches for potential outcomes in causal inference.

International Journal of Epidemiology, 44(5):1731–1737, 2015.