gesttools: General Purpose G-Estimation in R

Daniel Tompsett, Stijn Vansteelandt, Oliver Dukes, Bianca De Stavola

Observational Studies, Volume 8, Issue 1, 2022, pp. 1-28 (Article)

Published by University of Pennsylvania Press
DOI: https://doi.org/10.1353/obs.2022.0003

Observational Studies 8.1 (2022) Submitted Feb. 2022; Published May 2022

gesttools: General Purpose G-Estimation in R

Daniel Tompsett **d.tompsett@ucl.ac.uk**
GOS Institute of Child Health
30 Guilford Street London WC1N 1EH

Stijn Vansteelandt **stijn.vansteelandt@ugent.be**
Department of Applied Mathematics
Ghent University
Krijgslaan 281, S9 9000 Gent, Belgium

Oliver Dukes **oliver.dukes@ugent.be**
Department of Applied Mathematics
Ghent University
Krijgslaan 281, S9 9000 Gent, Belgium

Bianca De Stavola **b.destavola@ucl.ac.uk**
GOS Institute of Child Health
30 Guilford Street London WC1N 1EH

Abstract

In this paper we present **gesttools**, a series of general purpose, user friendly functions with which to perform g-estimation of structural nested mean models (SNMMs) for time-varying exposures and outcomes in R. The package implements the g-estimation methods found in Vansteelandt and Sjolander (2016) and Dukes and Vansteelandt (2018), and is capable of analysing both end of study and time-varying outcome data that are either binary or continuous, or exposure variables that are either binary, continuous, or categorical. It also allows for the fitting of SNMMs with time-varying causal effects, effect modification by other variables, or both, as well as support for censored data using inverse weighting. We outline the theory underpinning these methods, as well as describing the SNMMs that can be fitted by the software. The package is demonstrated using simulated, and real-world inspired datasets.

Keywords: g-estimation, time-varying confounding, effect modification, R

1. Introduction

Applying causal inference to longitudinal observational studies is challenging when one aims to quantify the joint effect of a sequence of exposures on subsequent outcomes. The likely presence of time-varying variables associated with both exposures and outcome that are also affected by earlier exposures, i.e. time-varying confounding, leads to analytical complexities that standard regression-adjustment methods cannot address (Robins (1986, 2000a); Vansteelandt and Sjölander (2016)). Causal inference in such cases is typically handled by one of the three "g-methods": inverse probability weighting (IPW) of marginal structural models (MSMs) (Robins (2000a)), g-estimation of structural nested mean models (SNMMs) (Robins et al. (1992a)), or g-computation (Robins (1986)).

Recently, Vansteelandt and Sjölander (2016) showed how to yield g-estimators of causal effects for continuous outcomes via generalised estimating equations (GEE) which can be implemented using standard software. This'trick' was extended to include binary and count outcomes by Dukes and Vansteelandt (2018) (Dukes and Vansteelandt (2018)). This paper presents **gesttools**, which implements the algorithms described in Vansteelandt and Sjölander (2016) and Dukes and Vansteelandt (2018) in the statistical software R (R Core Team (2019)), to provide a flexible framework for performing g-estimation in R. The functions of the package can be downloaded from CRAN, (`https://CRAN.R-project.org/package=gesttools`) (Tompsett et al. (2020)), or from the GitHub repository `https://github.com/danieltompsett/gesttools` (Tompsett et al. (2020)).

Software for g-computation is relatively common, for example the `gformula` set of software packages provided for R,SAS and STATA (Lin et al. (2020); Logan (2019); Daniel et al. (2011)). However there is a lack of standard software implementation for g-estimation due to its relative complexity (Vansteelandt and Joffe (2014); Vansteelandt and Sjölander (2016)). Our search found two notable R packages implementing g-estimation in R, **DTRreg** (Wallace et al. (2017a,b)), and **ivtools** (Sjölander and Martinussen (2019)). The former mostly focuses on estimating dynamic treatment regimes for data with an end of study outcome. The latter focuses on two stage least squares, and g-estimation for settings containing an instrumental variable, focusing on data with time to event, or end of study outcomes. G-estimation for survival outcomes has also recently been considered in Seaman et al. (2019), which fits structural nested cumulative survival time models (SNCSTMs), based on work in Dukes et al. (2019), but is not currently released as a formal software package. Other examples of g-estimation software for survival time outcomes can be found with `stgest` command in STATA (Sterne and Tilling (2002)) and the `SNCFTM` macro in SAS (Picciotto et al. (2012)).

The design principle of **gesttools** is to provide a suite of user-friendly and versatile functions for general purpose g-estimation for a wide variety of exposure types, outcome types and SNMMs. Notable features of the package include g-estimation of both end-of-study and time-varying outcome data, the ability to define effect modification by specific covariates, specification of both overall or time specific causal effects, and the choice to fit a pre-selected list of SNMM types. The package supports exposure variables that are binary, continuous, or categorical and outcome variables that are either binary or continuous. Confidence intervals are obtained via a bootstrap function.

A brief theoretical overview is given in section 2, including a description of the SNMMs that can be fit by **gesttools**, and the interpretation of its fitted parameters. The methodology behind the g-estimation methods used by **gesttools** is then described in section 3. Section 4 describes the main functions of the package, with examples shown in section 5. The paper then concludes in section 6.

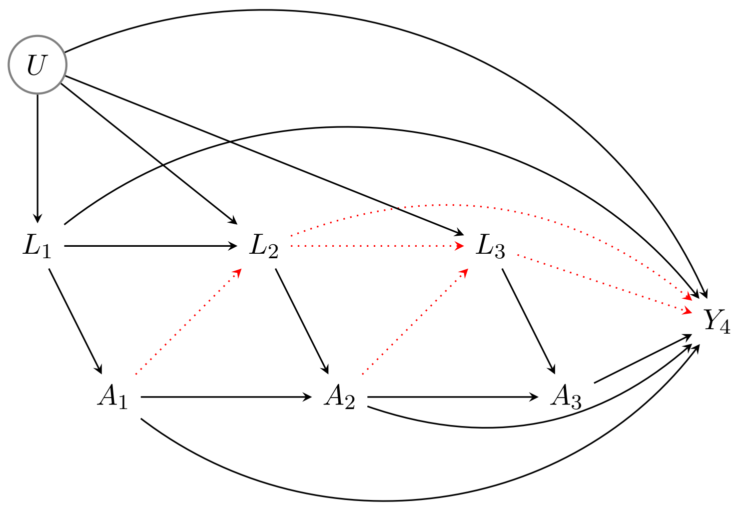
2. Structural Nested Mean Models

2.1 Overview

Suppose for now we have data with an end of study outcome variable. Let A_t denote the exposure variable, measured at times $t = 1, \ldots, T$, and Y_{T+1} the outcome of interest measured at the end of the study, at time $T+1$. Suppose also that there is a set of time-varying confounders of the exposure-outcome relationships, L_t, also measured at $t = 1, \ldots, T$, causally preceding the exposures at each time t. Furthermore, let U represent unmeasured variables associated with L_t, and Y_{T+1} but not with $A_t, \forall t$ (Figure 1).

Longitudinal settings such as these pose analytical challenges because of the time-varying confounding induced by L_t. If we wished to estimate the joint causal effect of A_1, A_2, and A_3 onto

Figure 1: Directed Acyclic Graph (DAG) of a typical data setup for a time-varying causal effect and time-varying confounding.



Y_4, we would need to control for L_1, L_2, and L_3 in order to remove the spurious associations they create between A_1, A_2, A_3 and Y_4, but at the same time we would *not* want to control for L_2 and L_3, because they lie on the causal path from A_1 to Y, and A_2 to Y, respectively. These correspond to the red paths in Figure 1. Controlling for L_2 and L_3 would also introduce collider bias, as new non-causal paths would be opened by this conditioning between A_1 and Y_4, and A_2 and Y_4, via U. As a result standard regression-adjustment methods would give biased estimates of the joint causal effect of A_1, A_2, and A_3 when time-varying confounding is present (Daniel et al. (2013)). A stated in the introduction, g-methods, such as g-estimation of SNMMs can deal with this issue.

2.2 SNMMs for End-of-Study Outcomes

Let $\overline{a}_t$ be the exposure history up to time t and $Y_{T+1}(\overline{a}_t, 0)$ the outcome that would have occurred had the exposure been set to its observed values up to time t, and set to 0 afterwards (if A is binary, 0 denotes no exposure). A general linear SNMM is defined as

$$E(Y_{T+1}(\overline{a}_t, 0) - Y_{T+1}(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \overline{l}_t) = \psi z_t a_t,\ \forall t = 1, \dots, T \tag{1}$$

where $\overline{l}_t$ is the covariate history up to t, z_t is a vector that could include a function of t and/or l_t (in addition to a column of "1"), and ψ is a vector containing the causal effect of A_t on Y_{T+1}, having the same dimensions as z_t. This model captures the effect of setting the exposure to its observed values up to time t and then to no exposure after time t versus setting it to its observed value up to time $t-1$ and then to no exposure thereafter.

G-estimation as implemented by Vansteelandt and Sjölander (2016) exploits the fact that the causal effect of A_T on Y_{T+1}, ψ,can be identified using regression models via adjustment for previous exposure and confounders times, that is $\overline{A}_{T-1}$ and $\overline{L}_T$, and a propensity score variable for the likelihood of being exposed. After which ψz_T is used to predict the potential outcome under no exposure at the previous time period, $Y(\overline{a}_{T-1}, 0)$.

$$H_{T-1} = Y_{T+1} - \psi z_T A_T,$$

where H_{T-1} denotes the potential outcome under no exposure at time T. The same process is then repeated to find the causal effect of A_{T-1} on H_{T-1}, controlling for $\overline{A}_{T-2}$ and $\overline{L}_{T-1}$, and so forth up to the identification of the causal effect of A_1 on H_1.

As with all g-methods the following three assumptions are sufficient for identification of causal effects: counterfactual consistency, where an individual's counterfactual outcome under a specific set of exposures is equal to their outcome had it been their observed exposure history; positivity, that there is a non-zero possibility of receiving every feasible set of exposures; and conditional exchangeability, specifically that $Y_{T+1}(\overline{a}_t, 0) \perp\!\!\!\perp A_t | \overline{L}_t, \overline{A}_{t-1} = \overline{a}_{t-1}$, for all feasible sets of exposures and for all t (Vansteelandt and Sjölander (2016); Robins (2000a); Hernán and Robins (2020)), which amounts to a sequential no unmeasured confounding assumption. We make an additional parametric assumption that the SNMM correctly specifies the causal relationship between exposure and outcome.

This g-estimation method can be applied to settings with binary, continuous and, as shown in section 3.4, categorical exposure variables, an advantage over IPW of MSMs. Furthermore, the causal effect ψ may vary with time, or be modified by time-varying covariates via the specification of z_t. Furthermore, g-estimation is more robust to model misspecification than g-computation, as it does not require postulating a model for the distribution of L_t given the exposure and covariate history $\overline{A}_{t-1}$ and $\overline{L}_{t-1}$ for each time t (Hernán and Robins (2020)).

With binary or count outcomes, SNMMs can be specified on the risk ratio scale, for example, as

$$\frac{E(Y_{T+1}(\overline{a}_t, 0)|\overline{a}_{t-1}, \bar{l}_t)}{E(Y_{T+1}(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \bar{l}_t)} = exp(\psi z_t a_t), \qquad \forall t = 1, \dots, T. \tag{2}$$

For purposes of simplicity, we will present the following sections 2 and 3 assuming a continuous outcome and SNMM of the form in equation 1. These sections remain relevant to binary outcome SNMMs of equation 2 with minimal changes to the methods. This is discussed in section 3.4.

Model Specification

The package **gesttools** allows users to choose from four specific types of SNMMs, based on the form of ψz_t. For an end-of-study outcome Y_{T+1}, the linear SNMMs we consider are as follows

Type 1: Overall Effect

The simplest SNMM sets $z_t = 1$,

$$E\{Y_{T+1}(\overline{a}_t, 0) - Y_{T+1}(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \bar{l}_t\} = \psi a_t, \quad \forall t = 1, \dots, T.$$

This model encodes a causal effect ψ of A_t on Y_T, the same for all exposure times. It may also be interpreted as an overall, or average effect of exposure at any time on the end of study outcome. If exposure is a treatment of some sort, ψ may be interpreted as the effect of the last portion of treatment at any time on the outcome.

Type 2: Modified Overall Effect

The other notable form for a SNMM is to allow effect modification of the causal effect by some covariate L^*. In this case $z_t = (1, L_t^*)'$, leading to

$$E\{Y_{T+1}(\overline{a}_t, 0) - Y_{T+1}(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \overline{l}_t\} = (\psi_0 + \psi_1 l_t^*)a_t, \quad \forall t = 1, \ldots, T.$$

Here, ψ_0 represents the overall effect of exposure at any time on outcome when $L_t^* = 0$, which is modified by an amount ψ_1 for each unit increase in the value of L_t^*.

Type 3: Time-Varying Effect

The package will also give users the option to allow for a separate causal effect for each exposure time $t = 1, \ldots, T$, in the form

$$E\{Y_{T+1}(\overline{a}_t, 0) - Y_{T+1}(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \overline{l}_t\} = \psi_t a_t, \quad \forall t = 1, \ldots, T.$$

Now $\psi = (\psi_1, \ldots, \psi_T)$ where ψ_t is specifically the effect of A_t on Y_T, and z_t is a vector of zeros with a 1 in the t'th position, so that $\psi z_t = \psi_t$. For example when $t = 1$ $\psi z_t = (\psi_1, \ldots, \psi_T)(1, \ldots, 0)'$.

Type 4: Modified Time-Varying Effect

A time-varying equivalent of SNMM type 2, denoted type 4 is specified as

$$E\{Y_{T+1}(\overline{a}_t, 0) - Y_{T+1}(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \overline{l}_t\} = (\psi_{0t} + \psi_{1t} l_t^*)a_t, \quad \forall t = 1, \ldots, T$$

Now $\psi = (\underline{\psi_1}, \ldots, \underline{\psi_T})$ where $\underline{\psi_t} = (\psi_{0t}, \psi_{1t})$, with ψ_{0t} denoting the effect of A_t on Y_{T+1} when $L_t^* = 0$, modified by an amount ψ_{1t} for each unit increase in L_t^*. Because $\psi z_t = (\psi_{0t} + \psi_{1t} l_t^*)$, z_t is a matrix of zeros with $(1, l_t^*)$ in the t'th row.

Equivalent specifications are available for binary outcomes modeled using equation 2.

Average Causal Effects and Structural Nested Mean Models

A key estimand in causal analysis is the average causal effect (ACE), sometimes known as the average treatment effect. With a single exposure and outcome, the ACE is defined as the difference in potential outcomes under exposure and no exposure. In the case of a time-varying binary exposure, several definitions are possible, one of which is as the difference in potential outcomes of always being exposed, versus never being exposed. For example, if $T = 2$

$$ACE = E(Y_3(1, 1) - Y_3(0, 0)).$$

SNMMs do not provide direct insight into the ACE as they specify the effect of exposure versus no exposure at a given time, given that exposure after that time is set to 0 (or a suitable reference value).

However in Vansteelandt and Sjölander (2016) it is shown that, at least for a continuous exposure, some SNMMs imply specific Marginal Structural Models (MSMs), for which the identification of the ACE is trivial. In particular SNMM type 1 implies an MSM of the form

$$E(Y_{T+1}(\overline{a_t})) = \alpha_0 + \psi \sum_{i=1}^{T} a_t$$

where α_0 is some constant value. It is then trivial to see that for a continuous Y_{T+1}, the ACE is simply $T\psi$. Equivalently, SNMM type 3 implies the MSM

$$E(Y_{T+1}(\overline{a_t})) = \alpha_0 + \sum_{i=1}^{T} \psi_t a_t$$

and thus the ACE is simply $\sum_{i=1}^{T} \psi_t$.

For binary outcomes or SNMMs with effect modification, such as in types 2 and 4, there is no obvious MSM to serve as an analogue and thus no obvious way to define an ACE, without invoking additional models.

2.3 SNMMs for Time-varying Outcomes

Let Y_s be a continuous time-varying outcome measured over times $s = 2, \ldots, T+1$. A linear SNMM is defined for each s as follows

$$E(Y_s(\overline{a}_t, 0) - Y_s(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \bar{l}_t) = \psi z_{st} a_t, \tag{3}$$

for all $s = 2, \ldots, T+1$ and $t < s$. We can define similar SNMM types as those for end-of-study outcomes.

Types 1 and 2

These are the same as in section 2.2, encoding an overall, or effect modified causal effect for all t and s as

$$E\{Y_s(\overline{a}_t, 0) - Y_s(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \bar{l}_t\} = \psi a_t, \quad \forall s = 2, \ldots, T+1 \text{ and } t < s$$

and

$$E\{Y_s(\overline{a}_t, 0) - Y_s(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \bar{l}_t\} = (\psi_0 + \psi_1 l_t^*) a_t, \quad \forall s = 2, \ldots, T+1 \text{ and } t < s.$$

respectively.

Type 3

To allow for a time-varying causal effect with multiple outcomes, we define $c = s - t$ as the ***step length***, that is number of time periods between exposure and outcome. A causal effect can then be encoded for each $c = 1, \ldots, T$, that is for each step length between exposure and outcome separately by specifying the SNMM.

$$E\{Y_s(\overline{a}_{s-c}, 0) - Y_s(\overline{a}_{s-c-1}, 0)|\overline{a}_{s-c-1}, \bar{l}_{s-c}\} = \psi_{s-c} a_{s-c}, \quad \forall c = 1, \ldots, T \text{ and } s > c.$$

Now $\psi = (\psi_{s-1}, \ldots, \psi_{s-T})$ and z_{st} is a vector of zeros of length T with a 1 in the c'th position.

By replacing the earlier ψ_t with ψ_{s-c} in this way, we encode the causal parameter as the effect of exposure on outcome c time periods later, that is the effect of A_{s-c} on Y_s, $\forall s > c$. For example ψ_{s-1} represents the overall effect of exposure on the subsequent outcome.

Type 4

We can also allow for effect modification with a type 3 SNMM as

$$E\{Y_s(\overline{a}_{s-c},0)-Y_s(\overline{a}_{s-c-1},0)|\overline{a}_{s-c-1},\overline{l}_{s-c}\} = (\psi^0_{s-c}+\psi^1_{s-c}l^*_t)a_{s-c}, \quad \forall c=1,\ldots,T \text{ and } s>c.$$

Here, the elements of $\psi zs,t$ are $\psi = (\underline{\psi_{s-1}},\ldots,\underline{\psi_{s-T}})$ with z_{st} a vector of zeros of length T with a $(1,l_t)$ in the c'th position. Here $\underline{\psi_{s-c}} = (\psi^0_{s-c},\psi^1_{s-c})$, where ψ^0_{s-c} denotes the effect of A_{s-c} on Y_s when $L^*_{s-c}=0$, modified by an amount ψ^1_{s-c} for each unit increase in L^*_{s-c}.

2.4 SNMMs for a Categorical Exposure

Suppose that A_t is a categorical exposure with 2 or more categories. These categories may take any arbitrary list of names, but we assume for simplicity they are labeled as $j=0,1\ldots,k$, where $j=0$ denotes no exposure, or some other reference category. Define binary variables A^j_t $(j=0,1,\ldots,k)$ where $A^j_t=1$ if $A_t=j$ and 0 otherwise. A SNMM can be specified to modeling the causal effect of exposure to categories $1,\ldots,k$, versus exposure to category 0 as follows

$$E(Y_{T+1}(\overline{a}_t,a^0)-Y_{T+1}(\overline{a}_{t-1},a^0)|\overline{a}_{t-1},\overline{l}_t) = \sum_{j=1}^{k}\psi^j z_t a^j_t,\ \forall t=1,\ldots,T \tag{4}$$

for an end of study outcome, or

$$E(Y_s(\overline{a}_t,a^0)-Y_s(\overline{a}_{t-1},a^0)|\overline{a}_{t-1},\overline{l}_t) = \sum_{j=1}^{k}\psi^j z_{st} a^j_t \tag{5}$$

for all $s=2,\ldots,T+1$ and $t<s$, where $Y_s(\overline{a}_t,a^0)$ is the counterfactual outcome of Y_s that would have occurred if exposure was set to its observed history up to time t and set to the reference category afterwards, and ψ^j is a vector representing the causal effect of exposure to category j versus exposure to the reference category 0. For simplicity, we set z_t to be identical for each j.

Note there now exists a separate causal vector for exposure to each category $j\in[1,k]$, versus exposure to the reference category. SNMM types 1-4 can be defined in the same way as SNMMs in sections 2.2 and 2.3 through specification of z_t, allowing for effect modification, time-varying effects, or both for each ψ^j.

3. Estimation

3.1 G-estimation for End-of-Study Continuous Outcomes

Suppose the data structure is as in Figure 1, and that we wish to fit a SNMM specified as in type 1. Vansteelandt and Sjölander (2016) states that the g-estimator of the causal effect of A_T on Y_{T+1} can be obtained as follows.

1. *PS Model*
   Fit a propensity score model for A, regressing the exposure at each time point (A_t) on the previous exposure history $\overline{A}_{t-1}$, and covariate history $\overline{L}_t$ using a logistic regression model if A is binary. For example

$$logit(P(A_t=1|\overline{a}_{t-1},\overline{l}_t)) = \eta_{0t}+\eta_{1t}a_{t-1}+\eta_{2t}l_t \quad \forall t=1,\ldots,T$$

or a normal linear regression model if A is continuous

$$E(A_t|\overline{a}_{t-1}, \overline{l}_t)) = \eta_{0t} + \eta_{1t}a_{t-1} + \eta_{2t}l_t \quad \forall t = 1, \ldots, T.$$

These models can also be consolidated into a single model for each exposure time t, by including t as a factor variable in the model (Seaman et al. (2019)). From this model estimate fitted values p_t, representing the propensity score (or predicted score if A is continuous) for exposure at time t for each individual.

2. *Adjusted Outcome Model*
Obtain an estimate of ψ, $\hat{\psi}^{(1)}$, by regressing the outcome Y_{T+1} on a_{T-1}, l_T, and the terms $z_T a_T$ and $z_T p_T$. For continuous Y this is a normal linear regression model

$$E(Y_{T+1}|a_T, \overline{l}_T, \overline{a}_{T-1}) = \beta_0 + \beta_1 a_{T-1} + \beta_2 l_T + \beta_3 z_T p_T + \psi z_T a_T.$$

It can be shown that the coefficient of $z_T a_T$ in the fitted outcome model is an estimate of the causal effect of A_T on Y_{T+1}, which we denote $\hat{\psi}^{(1)}$ (Vansteelandt and Sjölander (2016)). This can be used to predict the counterfactual outcome under no exposure after time $T-1$, which we label H_{T-1}. By definition $H_T = Y_{T+1}$ and in general $H_t = Y_{T+1} - \sum_{i=t+1}^{T} \psi Z_i A_i$. Due to its recursive nature this can be simplified to

$$H_t = H_{t+1} - \psi Z_{t+1} A_{t+1}.$$

3. Using $\hat{\psi}^{(1)}$ from step 2, estimate the counterfactual outcomes under no exposure at time T and $T-1$ respectively as

$$\hat{H}_T = Y_{T+1} \text{ and } \hat{H}_{T-1} = Y_{T+1} - \hat{\psi}^{(1)} z_T A_T$$

4. Now fit the adjusted outcome model of step 2 to $\hat{H}_T$ and $\hat{H}_{T-1}$ as follows

$$E(\hat{H}_t|\overline{a}_t, \overline{l}_t) = \beta_0 + \beta_1 a_{t-1} + \beta_2 l_{t-1} + \beta_3 z_t p_t + \psi z_t a_t.$$

for $t = T, T-1$. These models are fit simultaneously by Generalising Estimating Equations (GEE) with an independent working correlation. This results in an updated estimate $\hat{\psi}^{(2)}$ for the causal effects of A_{T-1} and A_T on Y_{T+1}.

5. Step 3 is then repeated using $\hat{\psi}^{(2)}$ to re-estimate H_T and H_{T-1} and additionally derive $\hat{H}_{T-2}$. Step 4 is then applied to all estimated H_t to obtain an updated estimate of ψ. This is repeated until H_1 is predicted and step 4 applied once more to obtain an estimate of ψ for the causal effect of A_t on Y_{T+1} for all $t = 1, \ldots, T$.

G-estimation by this method is doubly robust, in that ψ will be unbiased provided that either the propensity score model, or the outcome model is correctly specified (and the SNMM is correct). The association between L and Y is not necessarily assumed correct, in fact it is not strictly necessary to include L in the outcome model if there is no effect modification. If it is modeled correctly however, then unbiased causal effects can be obtained even if the propensity score model is mis-specified. Including the covariates L in the outcome model also leads to an gain in efficiency of the estimator. Note that by default the adjusted outcome models only condition on the previous exposure a_{T-1}. A user may wish to condition on ALL previous exposure and confounder history ($\overline{a}_{t-1}$ and $\overline{l}_{t-1}$), by including them as additional confounding variables. These exposure histories can be generated in **gesttools** using the `FormatData` function (see Implementation).

Censoring Weights and Missing Data

Suppose the data also contains censoring, that is drop-out, described by a time varying censoring indicator C_t that is set to 1 if the individual is censored by time t, and 0 otherwise. In this case, censoring weights are applied to the adjusted outcome model to account for bias caused by loss to follow up and are calculated as follows

$$w_t = \frac{I(C_{T+1} = 0)}{\prod_{i=t+1}^{T+1} P(C_i = 0|C_{i-1} = 0, \overline{a}_{i-1}, \overline{l}_{i-1})}$$

where $I(C_{T+1} = 0)$ is 1 when $C_{T+1} = 0$ and 0 otherwise. The probabilities $P(C_t = 0|C_{t-1} = 0, \overline{a}_{t-1}, \overline{l}_{t-1})$ are estimated in the same way as the propensity scores, that is from a user-specified logistic regression model such as

$$logit(P(C_t = 1|\overline{a}_{t-1}, \overline{l}_t)) = \eta_{0t} + \eta_{1t}a_{t-1} + \eta_{2t}l_t, \quad \forall t = 2, \ldots, T+1.$$

Note that such censoring weights are valid, provided that any variable used in the censoring model above, is also used in the model for the propensity score. In the package, the weights w_T are calculated prior to the propensity score estimation in step 1 and the remaining w_t are estimated at the same time as H_t in steps 3 or 5.

By default data rows with missing outcome or exposure data not due to censoring are omitted from the propensity and adjusted outcome models. Note that if A_t is missing at some time t for an individual, then all counterfactuals H_t at times $1, \ldots, t$ will also be missing, even if data on A exists prior to time t.

3.2 G-estimation For Time-Varying Outcomes

When the outcome variable Y_t varies over time and is measured at multiple time points, the SNMMs described in section 2.3 can be estimated by g-estimation as follows.

1. Obtain propensity scores p_t in the same way as in step 1 for the end of study outcome implementation.

2. Obtain an initial estimate for ψ by fitting an adjusted outcome model for Y_s on a_{s-2}, l_{s-1}, $z_{s(s-1)}a_{s-1}$ and $z_{s(s-1)}p_{s-1}$, for example as in the model

$$E(Y_s|a_{s-1}, l_{s-1}, a_{s-2}, p_{s-1}) = \beta_0 + \beta_1 a_{s-2} + \beta_2 l_{s-1} + \beta_3 z_{s(s-1)} p_t + \psi z_{s(s-1)} a_{s-1},$$

   $\forall s = 2, \ldots, T+1$. The estimated of the causal parameter ψ, denoted $\hat{\psi}^{(1)}$, that captures the effect of exposure on the subsequent outcome time (that is the effect of A_{s-1} on Y_s).

3. Define H_{st} as the predicted counterfactual of the outcome Y_s given exposure is set to its history up to time t and to 0 from time $t+1$ to time s. By again setting $t = s - c$, this may be written as $H_{s(s-c)}$, which we define as the c step predicted counterfactual of Y_s. The one step predicted counterfactual is given as $H_{s(s-1)} = Y_s$, and in general

$$H_{s(s-c)} = H_{s(s-c+1)} - \psi Z_{s(s-c+1)} A_{s-c+1},$$

   for $c > 1$. We can then predict the two step counterfactual $\hat{H}_{s(s-2)} = \hat{H}_{s(s-1)} - \hat{\psi}^{(1)} Z_{s(s-1)} A_{s-1}$.

4. Now update $\hat{\psi}^{(1)}$ by fitting the adjusted outcome model

$$E(\hat{H}_{st}|A_t, L_t, A_{t-1}, P_t) = \beta_0 + \beta_1 a_{t-1} + \beta_2 l_t + \beta_3 z_{st} p_t + \psi z_{st} a_t,$$

   for $s = 2, \ldots, T+1$ and $t = s-1, s-2$, to obtain $\hat{\psi}^{(2)}$.

5. Now apply steps 3 and 4 recursively, in each case using the updated $\hat{\psi}$ to re-predict the existing counterfactuals and further predict the counterfactuals with an additional time period between exposure and outcome. Then update $\hat{\psi}$ by fitting all predicted counterfactuals using the adjusted outcome model. Repeat until $H_{s(s-T)}$ is estimated and a final estimate of ψ is reached.

The package will allow users to specify an optional argument `cutoff`, a value for c between 1 and T which will stop step 5 once $H_{s(s-c)}$ has been calculated. Note that choosing a ”wrong” value of `cutoff` does not cause bias, but may change the interpretation of ψ, which will be relevant only for causal effects up to c time periods after exposure. Missing data are handled in the same way as in section 3.1. In the case of censored data, censoring weights are also applied similarly. The censoring weights for $\hat{H}_{st}$ in the adjusted outcome model are

$$w_{st} = \frac{I(C_s = 0)}{\prod_{i=t+1}^{s} P(C_i = 0|C_{i-1} = 0, \overline{a}_{i-1}, \overline{l}_{i-1})}.$$

When implemented, these weights are calculated during step 3 of the algorithm.

3.3 G-estimation for Binary or Count Outcomes

G-estimation of SNMMs when the outcome is a binary or count variable can be more challenging. For binary data the most obvious SNMM to fit, that uses a logistic link function for the outcome data cannot be fit by standard g-estimation methods, and suffer from non-collapsibility (Tan (2019); Tchetgen Tchetgen et al. (2009); Matsouaka and Tchetgen Tchetgen (2017)). Furthermore, additive SNMMs that measure the mean difference are not recommended for binary outcomes as it is difficult to obtain a parameterisation in which the exposure effect is variation dependent of the nuisance parameters Wang et al. (2017); Robins (2000b). Hence (Wang et al. (2017) and Dukes and Vansteelandt (2018)) recommend modeling the causal risk ratio, by fitting a multiplicative SNMM using a log link function as follows.

$$\frac{E(Y_{T+1}(\overline{a}_t, 0)|\overline{a}_{t-1}, \overline{l}_t)}{E(Y_{T+1}(\overline{a}_{t-1}, 0)|\overline{a}_{t-1}, \overline{l}_t)} = exp(\psi z_t a_t) \ \forall t = 1, \ldots, T. \tag{6}$$

SNMM types 1-4 are defined in the exact same way as continuous outcomes, with the interpretation of ψ now being a causal risk ratio, rather than a causal risk difference. Work in Dukes and Vansteelandt (2018) demonstrated that such SNMMs can be fit in the same way as for continuous outcomes with only minor modifications of the above algorithms. Firstly, the adjusted outcome models in steps 2 and 4 are gamma regression models with a log link, rather than normal linear regression models, that is step 2 and step 4 now fit the model

$$E(H_t|a_{t-1}, l_t, a_t, p_t) = exp(\beta_0 + \beta_1 a_{t-1} + \beta_2 l_{t-1} + \beta_3 z_t p_t + \psi z_t a_t).$$

Secondly, the potential outcomes under no exposure at time t are estimated as

$$\hat{H}_T = Y_{T+1} \text{ and } \hat{H}_t = \hat{H}_{t+1} exp(-\hat{\psi} Z_t A_t).$$

Equivalent changes are made for datasets with time-varying binary outcomes.

If the binary outcome Y_s is an indicator of survival up to time s, then the SNMM of equation 6 is equivalent to fitting a special case of SNMMs for survival data over discrete time periods known as Structural Nested Cumulative Failure Time Models (SNCFTMs), explained in Picciotto et al. (2012) and Dukes and Vansteelandt (2018).

3.4 G-Estimation for Categorical Exposures

Work by Vansteelandt and Sjölander (2016), specifically the case study of section 4, demonstrated how g-estimation of SNMMs in the case of a categorical exposure can be performed with only minor changes to the methods described in the rest of the section.

Propensity Score

The propensity score model is now a multinomial logistic model, fitting A_t against a_{t-1} and l_t

$$log\left\{\frac{P(A_t^j|\overline{a}_{t-1},\bar{l}_t)}{P(A_t^0|\overline{a}_{t-1},\bar{l}_t)}\right\} = \eta_{0jt} + \sum_{m=1}^{k} \eta_{1mjt} a_{t-1}^m + \eta_{2jt} l_t \ \forall t = 1,\dots,T \text{ and } j = 1,\dots,k.$$

From this model we obtain fitted values $p_{jt},\ j = 1,\dots,k$, representing the propensity score of exposure to category j at time t.

Adjusted Outcome Model

For a continuous, end-of-study outcome, the first adjusted outcome model is now

$$E(Y_{T+1}|\overline{a}_t,\bar{l}_t) = \beta_0 + \sum_{j=1}^{k} \beta_{1j} a_{(T-1)}^j + \beta_2 l_{T-1} + \sum_{j=1}^{k} \beta_{3j} z_T p_{jT} + \sum_{j=1}^{k} \psi^j z_T a_T^j.$$

Counterfactuals

The counterfactual outcomes are now calculated as

$$H_T = Y_{T+1} \text{ and } H_t = H_{t+1} - f(A_{t+1})$$

where

$$f(A_{t+1}) = \begin{cases} 0 : \text{if } A_{t+1} = 0 \\ \psi_j Z_t : \text{ if} A_{t+1} = j \end{cases}$$

We note that due to the way in which the coding is performed (see section 4), and how the counterfactuals are calculated as above, there is no need for the user to derive the binary variables A^j, and only the original categorical exposure is needed. These steps extend naturally to binary outcomes and time-varying outcome SNMMs. Censoring weights are applied without change from the previous sections.

4. Implementation

4.1 Installation and Package Dependencies

The package can be downloaded from CRAN (Tompsett et al. (2020)) via the URL `https://CRAN.R-project.org/package=gesttools`, or found at the Github repository `https://github.com/danieltompsett/gesttools`. The package has the following dependencies: **DataCombine** and the `slide` function (Gandrud (2016)), **geeM** for fitting GEE models (McDaniel et al. (2013)), **nnet** for fitting multinomial models with `multinom` (Venables and Ripley (2002)), **rsample** and the `bootstraps` function (Kuhn et al. (2019)), **tibble** and the `as_tibble` function (Müller and Wickham (2019)), **tidyr** and the `nest_legacy` function (Wickham and Henry (2019)), **tidyselect** and the `all_of` function (Henry and Wickham (2020)),**magrittr** for the `%>%` operator (Bache and Wickham (2014)), and the **testthat** package for testing the functions (Wickham (2011)). The remaining required functions are found in the **stats** package. All dependencies are automatically loaded when installed via CRAN.

4.2 Data Setup

The data to analyse must be in long format, that is each row holds the data for an individual at some specific time t, in ascending order by time and id variable. Time periods must be labeled as numeric integers starting from 1, going up to up to time T. We assume the convention that each row contains A_t L_t and Y_{t+1}, and C_{t+1}. This implies that the censoring indicator for each row should indicate that a user is censored AFTER time t, specifically after A_t and L_t are measured, and the outcome indicates the first outcome that occurs AFTER A_t and L_t are measured. For example, data at time 1, should contain A_1, L_1, Y_2, and C_2. If Y is an end of study variable, simply repeat its value on each row. We expect the convention that censoring C occurs before the outcome Y is measured at each time.

The outcome and exposure variables must be set up in a specific manner. They must be either a continuous variable, or if binary, written as an `as.numeric` variable taking values 0 or 1, where 1 indicates the event or exposure. This also applies to any covariate that is an effect modifier. Effect modification by categorical variables is not supported. Categorical exposures must be given as an `as.factor` variable. The censoring indicator must also be written as an `as.numeric` variable taking values 1 if censored, and 0 otherwise.

Crucially the data must be rectangular, that is there must exists a row entry for every time period for all individuals. Data rows that are missing due to censoring or missing data must be included with missing values for all variables besides the id and time variables. A function `FormatData` is provided that can add these rows for a given long format dataset.

```
FormatData(data, idvar, timevar, An, varying, Cn=NA,
GenerateHistory = FALSE ,GenerateHistoryMax = NA)
```

The required inputs are `data,idvar,timevar,An`, and `varying`, which hold (in order) the name of the data, and the variable names of the time, unique identifier (id), and exposure in quotations. Users then specify for `varying` a vector of names in quotations of the time-varying variables in the data, including the exposure, covariates, and if applicable the outcome, with the name of the censoring indicator given in `Cn`. The result is a long format dataset that is given in ascending order of time and id with missing rows added as necessary.

Users can optionally generate variables corresponding to the lagged exposure history up to d time periods prior by setting `GenerateHistory = FALSE` and `GenerateHistoryMax = d`, which may be included as covariates in the g-estimation functions.

4.3 G-Estimation Functions

The main functions of the package are `gestSingle` and `gestMultiple`, which perform g-estimation for data with a single end-of-study outcome, and a time-varying outcome respectively.

```
gestSingle(data, idvar, timevar, Yn, An, Cn, outcomemodels,
propensitymodel, censoringmodel, type, EfmVar,...)
gestMultiple(data, idvar, timevar, Yn, An, Cn, outcomemodels,
propensitymodel, censoringmodel, type, EfmVar, cutoff,...)
```

For data with a time-varying outcome, an optional input `cutoff`, sets a maximum value for the step length c in the algorithm, for which counterfactuals are calculated and the causal parameter ψ is defined. For example if `cutoff = 2` then only the effect of exposure on the two subsequent outcome time is calculated. This gives the user control over whether to model only short term effects of exposure (`cutoff = 1`), or to model longer term effects of exposure up to c outcome periods after the user was exposed. The full list of input arguments are as follows.

- `data`: A data frame in long format containing the data to be analysed.
- `idvar`: Character string specifying the name of the ID variable in data.
- `timevar`: Character string specifying the name of the time variable in the data. Note that `timevar` must specify time periods as integer values starting from 1 (must not begin at 0).
- `Yn`: Character string specifying the name of the outcome variable.
- `An`: Character string specifying the name of the exposure variable.
- `Cn`: Optional character string specifying the name of the censoring indicator variable. Cn should be a numeric vector taking values 0 or 1, with 1 indicating censored.
- `outcomemodels`: A list of formulas or formula objects specifying the outcome models for `Yn` that includes all the confounders. See notes below on how best to specify these models.
- `propensitymodel`: A formula or formula object specifying the propensity score model for An.
- `censoringmodel`: A formula or formula object specifying the censoring model for Cn.
- `type`: Value from 1-4, which will fit the corresponding SNMM type as described in section 2.
- `EfmVar`: Character string specifying the name of the effect modifying variable for types 2 or 4.
- `cutoff`: Available only for time-varying outcome g-estimation. An integer taking value from 1 up to T, where T is the maximum value of `timevar`. Instructs the function to estimate causal effects only up to `cutoff` time periods prior to outcome.

Output

If `type = 2` or `type = 4`, that is the SNMM has effect modification, `EfmVar` is taken as the effect modifier. Each function outputs a vector corresponding to the fitted causal parameter ψ, a summary of the propensity score weights and censoring weights labeled `PropensitySummary` and `CensoringSummary`, as well as the dataset, returned as a tibble dataset with `Data`, which includes the full list of propensity score and censoring weights. If `An` is the name of the exposure variable, `i` is the current time period, and `EfmVar` is the name of the effect modifier, then each element of ψ is labeled as follows for `gestSingle`

- `type = 1`: `An`: The effect of exposure at any time t on outcome.
- `type = 2`: `An`: The effect of exposure at any time t on outcome, when `EfmVar` is set to zero.

  `An:EfmVar`: The effect modification by `EfmVar`, the additional effect of A on Y for each unit increase in `EfmVar`.
- `type = 3`: `t=i.An`: The effect of exposure at time t=i on outcome.
- `type = 4`: `t=i.An`: The effect of exposure at time t=i on outcome, when `EfmVar` is set to zero.

  `t=i.An:EfmVar`: The effect modification by `EfmVar`, the additional effect of A on Y at time t=i for each unit increase in `EfmVar`.

For `gestMultiple`, the output for SNMM types 3 and 4 is instead

- `type = 3`: `c=i.An`: The effect of exposure at any time t on outcome `c=i` time periods later.
- `type = 4`: `c=i.An`: The effect of exposure at any time t on outcome `c=i` time periods later, when EfmVar is set to zero.

  `c=i.An:EfmVar`: The effect modification by EfmVar, the additional effect of exposure on outcome c=i time periods later for each unit increase in EfmVar.

When A is categorical, `An` is replaced with `Anj`, where `j` is the category level, indicating this is the effect of exposure to category `j`, versus the reference category.

Notes

The input `outcomemodels` is specified as a list of T elements (the number of exposure times) with each element being a formula, specifying an outcome model for the counterfactual outcome against exposure `An`, any confounding variables L, and any history of exposure or confounding variables. Note that these model should NOT include the propensity score. The relevant terms for the propensity score are added automatically based on propensity scores predicted from the `propensitymodel` formula. We recommend including `timevar` in `propensitymodel` to allow for propensity scores to vary with time.

For `gestSingle`, element i of `outcomemodels`, that is `outcomemodels[[i]]` contains the formula for the outcome models at time i, that is for (or up to) the counterfactuals H_i.

For `gestMultiple`, `outcomemodels[[i]]` contains the formula for the outcome models for the i-step counterfactuals, that is for (or up to) the counterfactuals $H_{s(s-i)}$. If `cutoff=i`, then `outcomemodels` only needs to be a list of i formula objects, up to the i step counterfactuals.

Every outcome model must include `An` and, if `type` is 2 or 4, every outcome model must include `An`, `EfmVar`, and an `An:EfmVar` interaction, written in that order. To ensure this, every outcome model must write `An` on the RHS before the `EfmVar` term. If not then the formula object writes the interaction term as `EfmVar:An` which will not be recognised by the code. An alternative is to write `An*EfmVar` in each model instead.

The models defined in `outcomemodels` and `propensitymodel`, should include the same confounders of exposure and outcome as well as the value of the exposure at the previous time point. This improves the chances of either the propensity or outcome model being correctly specified. If fitting problems occur, consider removing confounders from the outcome models first. Fitting problems can also occur due to the history of exposure. For example if A_{t-1} is included in the outcome model for $T = 1$, this history does not exist and A_{t-1} is simply set to the reference category for every individual. As a final note all terms included in `censoringmodel` should also be in `propensitymodel`.

If the outcome is time-varying, g-estimation become increasingly slow as T becomes large. For example, when $T = 3$, there are $3+2+1 = 6$ counterfactuals H_{st} to estimate for each individual, but when $T = 10$ there are $10+9+...+1 = 55$ to estimate. Consider using `cutoff` to avoid this issue.

4.4 Bootstrap Function

Standard errors for the causal effect estimates are obtained with the bootstrap function `gestboot`.

```
gestboot(gestfunc, data, idvar, timevar, Yn, An, Cn=NA,
outcomemodels, propensitymodel, censoringmodel=NULL,
type, EfmVar=NA, cutoff = NA, bn, alpha= 0.05 ,
onesided = "twosided", seed = NULL,...)
```

The user is required to specify which of the g-estimation functions to be fit with `gestfunc`, one of `gestSingle` or `gestMultiple` along with the functions required inputs. Users must also specify `bn`, `alpha` and `onesided`, which define the number of bootstraps to generate, the confidence level α and whether to fit a one or two sided interval (one of `"upper", " lower"` or `"twosided"`.) Confidence intervals for each element of ψ are taken as the α, $1-\alpha$ or $(\frac{\alpha}{2}, 1-\frac{\alpha}{2})$ quantiles of the ordered bootstrap estimates of each element of ψ for lower, upper, and two sided confidence intervals respectively. These intervals are labeled in the same way as in the g-estimation functions. Bonferroni corrected intervals for multiple comparisons are also generated with given confidence level $\frac{\alpha}{r}$ where r is the number of elements of ψ. A full list of the bootstrapped estimates of ψ are also output as a tibble dataset labeled `boot.results`.

5. Examples

5.1 Simulated Data

We begin by demonstrating **gesttools** using simulated datasets generated by the `dataexamples` function included with the package.

```
dataexamples(n = 1000 , seed = NULL, Censoring   = FALSE).
```

The function outputs four datasets.

- `datagest`: A dataset with a continuous end of study outcome, and binary time varying exposure. Designed to test `gestSingle`.
- `datagestcat`: A dataset with a continuous end of study outcome, and categorical time varying exposure with three categories. Designed to test `gestSingle`.
- `datagestmult`: A dataset with a continuous time varying outcome, and binary time varying exposure. Designed to test `gestMultiple`.
- `datagestmultcat`: A dataset with a continuous time varying outcome, and categorical time varying exposure with three categories. Designed to test `gestMultiple`.

Data are generated on `n` individuals, comprising of an id variable "id", time variable "time", continuous outcome Y (time-varying or end of study), time-varying binary exposure A, time-varying confounder L, and baseline confounder U, over $T = 3$ time periods. If `Censoring = TRUE` the data are appropriately censored with a censoring indicator C_t. For `datagest`, `datagestmult`, the data are simulated as follows.

- Baseline covariate: $U \sim N(0,1)$
- Covariates $L_t \sim N(1 + L_{t-1} + 0.5A_{t-1} + U)$, $t = 1,2,3$, $A_0 = 0$
- Exposure: $A_t \sim Bin(1, expit(1 + 0.1L_t + 0.1A_{t-1}))$ $t = 1,2,3$.
- Censoring indicator: $C_t \sim Bin(1, expit(-1 + 0.001 * L_{t-1} + 0.001 * A_{t-1}))$ $t = 2,3,4$.
- Time-varying outcome: $Y_t \sim N(1 + A_t + \gamma_t A_{t-1} + \sum_{i=1}^{t} L_t + U, 1)$ $t = 2,3,4$
- Or an End-of-study outcome: $Y_4 \sim N(1 + 0.5A_2 + A_3 + L_1 + L_2 + L_3 + U, 1)$.

where we set $(\gamma_1, \gamma_2, \gamma_3) = (0, 1/2, 1/2)$. For `datagestcat`, and `datagestmultcat`, A_t is categorical variable taking values "a" (the reference category), "b" or "c". We define the coefficient for each category of A_t in the models via the $\zeta(A_t)$ function

$$\zeta(A_t) = \begin{cases} 1: \text{ if } A_t = "a" \\ 2: \text{ if } A_t = "b" \\ 3: \text{ if } A_t = "c" \end{cases}$$

We then generate A_t from the following multinomial distribution

- $P(A_t = "a") = 1 - \frac{3}{5} * expit(1 + 0.1 * L_t + \zeta(A_{t-1}))$
- $P(A_t = "b") = \frac{1}{5} * expit(1 + 0.1 * L_t + \zeta(A_{t-1}))$
- $P(A_t = "c") = \frac{2}{5} * expit(1 + 0.1 * L_t + \zeta(A_{t-1}))$.

Now L, Y,U and C are generated as before, with A_t replaced by ζA_t.

We first demonstrate `gestSingle`, by generating an appropriate dataset with n=1000 individuals, and no censoring with the following code.

We make two notes. Firstly we use `FormatData` to create the lagged exposure `Lag1A`. Secondly, we allow the propensity score to vary with the time period by adding time as a variable in `propensitymodel`.

```
R> datas<- dataexamples(n = 1000, seed = 123, Censoring = FALSE)
R> data<-datas$datagest
R> data<-FormatData(data=data,idvar="id",timevar="time",An="A",
+   varying=c("A","L"),GenerateHistory=TRUE,GenerateHistoryMax=1)
R> idvar<- "id"
R> timevar<- "time"
R> Yn<- "Y"
R> An<- "A"
R> Cn<-NA
R> outcomemodels=list("Y~A+U+L+Lag1A","Y~A+U+L+Lag1A",
+   "Y~A+U+L+Lag1A")
R> propensitymodel=c("A~L+U+as.factor(time)+Lag1A")
R> censoringmodel=NULL
R> EfmVar=NA
R> type<- 1
R> gestSingle(data, idvar, timevar, Yn, An, Cn, outcomemodels,
+   propensitymodel, censoringmodel, type, EfmVar)

$psi
       A
1.146566

$Data
# A tibble: 3,000 x 10
       A     Y      L        U    id  time   L1A   int   prs     w
   <dbl> <dbl>  <dbl>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1     1  4.62 -0.556 -0.560       1     1     0     1 0.753     1
 2     0  4.62  1.86  -0.560       1     2     1     1 0.783     1
 3     1  4.62  1.60  -0.560       1     3     0     1 0.774     1
 4     0  1.49 -0.270 -0.230       2     1     0     1 0.758     1
 5     0  1.49 -0.907 -0.230       2     2     0     1 0.719     1
 6     1  1.49  0.859 -0.230       2     3     0     1 0.767     1
 7     1 16.4   2.54   1.56        3     1     0     1 0.801     1
 8     1 16.4   3.72   1.56        3     2     1     1 0.814     1
 9     1 16.4   6.08   1.56        3     3     1     1 0.854     1
10     0  7.46  0.938  0.0705      4     1     0     1 0.775     1
# ... with 2,990 more rows

$PropensitySummary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.6275  0.7680  0.7892  0.7923  0.8173  0.9169

$CensoringSummary
   Mode    NA's
logical       1
```

```
attr(,"class")
[1] "Results"
```

The output gives the causal effect of exposure on outcome, presumed to be the same at all 3 exposure times, note that the true value is 1. We also see a preview of the data with the fitted propensity scores `prs` and a summary of the propensity scores. We can obtain 95% confidence intervals via the `gestboot` function using 1000 bootstrapped datasets.

```
R> gestfunc<- gestSingle
R> start<- Sys.time()
R> gestboot(gestfunc, data, idvar, timevar, Yn, An, Cn,
+   outcomemodels, propensitymodel, censoringmodel=NULL,
+   type = 1, EfmVar, bn = 1000, alpha = 0.05,
+   onesided = "twosided", seed = 123)
R> end<- Sys.time()
R> end - start

$original
       A
1.146566

$mean.boot
       A
1.148633

$conf
       2.5%   97.5%
A 0.9006799 1.38613

$conf.Bonferroni
       2.5%   97.5%
A 0.9006799 1.38613

$boot.results
# A tibble: 1,000 x 1
       A
   <dbl>
 1 1.02
 2 1.09
 3 1.16
 4 0.966
 5 0.877
 6 1.09
 7 1.13
```

```
 8 1.27
 9 1.24
10 1.23
# ... with 990 more rows

attr(,"class")
[1] "Results"
> end<- Sys.time()
> end - start
Time difference of 6.607419 mins
```

The output is straightforward; `original` gives the estimated value of ψ for the original dataset, and `mean.boot` displays the average fitted value of ψ over the 1000 bootstrapped datas. The $1-\alpha\%$ and Bonferroni corrected confidence intervals for each element of ψ are given in `conf` and `conf.Bonferroni`. A full list of estimates of ψ for each bootstrapped dataset is also given as a tibble in `boot.results`. We note that the average bootstrapped value of ψ closely matches that estimated form the original data, and that the confidence intervals include the true effect 1. Note that in versions of R prior to version 3.6.0, results may differ slightly due to a change in the method of random number generation.

We can also demonstrate g-estimation for a dataset with a time-varying outcomes and censoring using `gestMultiple`. In particular, we will run g-estimation of SNMM type 3, a time-varying causal effect, and specify long term effects of exposure up to 2 subsequent time periods by setting `cutoff = 2`

```
R> datas<- dataexamples(n = 1000, seed = 123, Censoring = TRUE)
R> data<- datas$datagestmult
R> data<-FormatData(data=data,idvar="id",timevar="time",An="A",
+   Cn="C",varying=c("Y","A","L"),GenerateHistory=TRUE,
+   GenerateHistoryMax=1)
R> Cn<-"C"
R> outcomemodels=list("Y~A+U+L+Lag1A","Y~A+U+L+Lag1A",
+   "Y~A+U+L+Lag1A")
> propensitymodel=c("A~L+U+as.factor(time)+Lag1A")
> censoringmodel=c("C~L+U+as.factor(time)")
> EfmVar=NA
> type<- 3
> cutoff<-2
> gestMultiple(data, idvar, timevar, Yn, An, Cn, outcomemodels,
+ propensitymodel, censoringmodel, type, EfmVar, cutoff)

$psi
   c=1.A    c=2.A
1.073489 1.122045
```

```
$Data
# A tibble: 3,000 x 16
      id      U  time      Y     A      L     C  Lag1Y Lag1A
   <dbl>  <dbl> <dbl>  <dbl> <dbl>  <dbl> <dbl>  <dbl> <dbl>
 1     1 -0.560     1  0.204     1 -0.556     0  0         0
 2     1 -0.560     2 NA         0  1.86      1  0.204     1
 3     1 -0.560     3 NA        NA NA         1 NA         0
 4    10 -0.446     1  2.26      0  0.107     0  0         0
 5    10 -0.446     2  3.50      1  2.18      0  2.26      0
 6    10 -0.446     3 NA         0  2.08      1  3.50      1
 7   100 -1.03      1 NA         1 -0.271     1  0         0
 8   100 -1.03      2 NA        NA NA         1 NA         1
 9   100 -1.03      3 NA        NA NA         1 NA        NA
10  1000 -0.249     1  3.88      1  0.290     0  0         0
# ... with 2,990 more rows, and 7 more variables: Lag1L <dbl>,
#   int <dbl>, prs <dbl>, cps <dbl>, C0 <int>, cprod <dbl>,
#   w <dbl>

$PropensitySummary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
 0.6089  0.7627  0.7866  0.7903  0.8157  0.9442     701

$CensoringSummary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
 0.6783  0.7275  0.7411  0.7399  0.7527  0.7926     701

attr(,"class")
[1] "Results"
Warning message:
In gestMultiple(data, idvar, timevar, Yn, An, Cn, outcomemodels,  :
Variables included in censoringmodel should ideally be included
in propensitymodel else propensity scores may be invalid.
>
```

Here the parameter `c=1.A` is the short term effect of A, that is the overall effect of exposure on the subsequent outcome period, i.e A_{s-1} on Y_s, with `c=2.A` the longer term effect of exposure on outcome 2 time periods later. The effect `c=3.A` was not estimated due to the cutoff option. As before, the true effects are both 1. Note the warning message given which occurs whenever `Cn` is supplied reminding the user than any variables used to model the censoring score must also be used in the propensity score model.

Finally, we show a brief demonstration for a categorical exposure.

```
R> datas<- dataexamples(n = 1000 ,seed = 123, Censoring = FALSE)
R> data<- datas$datagestcat
R> data<-FormatData(data=data,idvar="id",timevar="time",An="A",
```

```
+   varying=c("Y","A","L"),GenerateHistory=TRUE,
+   GenerateHistoryMax=1)
R> outcomemodels=list("Y~A+U+L+A:L+Lag1A",
+   "Y~A+U+L+A:L+Lag1A","Y~A+U+L+A:L+Lag1A")
R> propensitymodel=c("A~L+U+as.factor(time)+Lag1A")
R> EfmVar<-"L"
R> gestSingle(data, idvar, timevar, Yn, An, Cn=NA, outcomemodels,
+   propensitymodel, censoringmodel=NULL, type=2, EfmVar)

# weights:  24 (14 variable)
initial  value 3295.836866
iter  10 value 3049.680743
final  value 3029.176746
converged
$psi
         Ab        Ab:L          Ac        Ac:L
0.862709487 0.006566811 1.739354556 0.039175505

$Data
# A tibble: 3,000 x 12
      id      U  time     Y A          L Lag1Y Lag1A  Lag1L
   <dbl>  <dbl> <dbl> <dbl> <fct>  <dbl> <dbl> <fct>  <dbl>
 1     1 -0.560     1  7.62 a     -0.556  0    a       0
 2     1 -0.560     2  7.62 c      1.86   7.62 a      -0.556
 3     1 -0.560     3  7.62 a      3.10   7.62 c       1.86
 4    10 -0.446     1  9.54 b      0.107  0    a       0
 5    10 -0.446     2  9.54 a      3.18   9.54 b       0.107
 6    10 -0.446     3  9.54 b      3.08   9.54 a       3.18
 7   100 -1.03      1  8.00 c     -0.271  0    a       0
 8   100 -1.03      2  8.00 a      1.66   8.00 c      -0.271
 9   100 -1.03      3  8.00 c      2.98   8.00 a       1.66
10  1000 -0.249     1  5.02 a      0.290  0    a       0
# ... with 2,990 more rows, and 3 more variables: int <dbl>,
#   prs <dbl[,2]>, w <dbl>

$PropensitySummary
       b                  c
 Min.   :0.09199   Min.   :0.2656
 1st Qu.:0.13249   1st Qu.:0.3186
 Median :0.17779   Median :0.3433
 Mean   :0.16967   Mean   :0.3530
 3rd Qu.:0.19801   3rd Qu.:0.3870
 Max.   :0.26527   Max.   :0.4870

$CensoringSummary
```

```
   Mode    NA's
logical       1

attr(,"class")
[1] "Results"
```

Now ψ displays the causal effects for each ψ^j, that contains the causal effects of exposure to category j versus the reference. For example `Ab` is the overall effect of exposure to level "b" versus the reference level "a", when the effect modifier $L = 0$. Coefficients containing L are the effect modifications, that is `Ab:L` is the additional effect of exposure to level "b" for each unit increase in L (or when L is set to 1 if L is binary).

5.2 The QMUL Clinical Effectiveness Group Database

As a simple case study, we demonstrate **gesttools** on a simulated dataset that is inspired by an observational study of diabetic patients from the Queen Mary University of London (QMUL) clinical effectiveness group (CEG) database (Malawana et al. (2018)). The CEG database includes patient data from three primary care trusts in East London. We are interested in a core cohort of approximately 45,282 type 2 diabetic (T2D) patients with data collected in 6 monthly intervals from 2011 to 2017.

The question of interest is a comparison of the impact of two "second line" treatments on blood glucose levels, measured by HbA1c level. Initial treatment for T2D involves use of Metformin. When this fails two second line treatments are considered

- Metformin plus insulin, the baseline treatment
- Metformin plus Sulphonylureas

which will represent the binary exposure of interest. The simulated data will create a simplified version of this data, consisting of $n = 4902$ eligible T2D patients who failed first line therapy. Each patient has $T = 6$ exposure times, measured every 6 months, starting from when they were first recorded (their baseline), and outcome (HbA1c levels) measured every 6 months from baseline up to 3 years.

We are interested in the comparison of effects of second line treatments on Hba1c levels three years after baseline, and, taking HbA1c as a time-varying outcome, the short term effect of treatment at each time on HbA1c 6-12 months afterwards. The data, which we label `dataQMUL` is generated by code found with the supporting material, along with the code required for analysis.

- We have baseline confounders `sex` (0=female, 1=male), centred age `age` and its square value `ageSQ`, and the centred log value of HbA1c at baseline $(t = 1)$ `HbA1cB` and its squared value, `Hba1cBSQ`.
- Second line treatment exposure `Treatment` over $t = 1, \ldots, 6$ taking value 0 if Metformin plus insulin and 1 if Metformin plus Sulphonylureas.
- Time varying outcome over $t = 2, \ldots, 7$ as the centred log HbA1c level `HbA1c`.

- An end of study outcome, that is HbA1c at 3 years after baseline (T=7), labeled `HbA1cEND`.
- Time-varing confounders equal to the previous value of `treatment` (`TreatmentL`), the previous outcome, labeled `HbA1cL`, and its squared value `HbA1cLSQ`.

We first use `gestSingle` to investigate the effect of `Treatment` on `HbA1cEND`, fitting a type 1 SNMM with bn=1000 bootstraps for a 95% confidence interval using `gestboot`.

```
R> idvar = "id"
R> timevar = "time"
R> Yn = "HbA1cEND"
R> An = "Treatment"
R>
R> outcomemodels<-list(
+   "Hba1cEND~Treatment+TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ",
+   "Hba1cEND~Treatment+TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ",
+   "Hba1cEND~Treatment+TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ",
+   "Hba1cEND~Treatment+TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ",
+   "Hba1cEND~Treatment+TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ",
+   "Hba1cEND~Treatment+TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ")
R>
R> propensitymodel<-c(
+   "Treatment~TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ+
+   as.factor(time)")
R>
R> gestboot(gestSingle,data, idvar, timevar, Yn, An,
+   Cn=NA, outcomemodels, propensitymodel, censoringmodel=NULL,
+   type=1, EfmVar=NA, bn=1000, seed=123)

$original
  Treatment
0.008680741

$mean.boot
  Treatment
0.008697338

$conf
                 2.5%      97.5%
Treatment 0.003469881 0.01425178

$conf.Bonferroni
                 2.5%      97.5%
Treatment 0.003469881 0.01425178

$boot.results
```

```
# A tibble: 1,000 x 1
   Treatment
       <dbl>
 1   0.00646
 2   0.00900
 3   0.0100
 4   0.00852
 5   0.00739
 6   0.0129
 7   0.0138
 8   0.00860
 9   0.00394
10   0.0106
# ... with 990 more rows

attr(,"class")
[1] "Results"
```

We note a small, but significant overall increase in log HbA1c levels at the end of follow up of about 0.008 when on Sulphonylureas, compared to insulin, in combination with Metformin. As this is a type 1 SNMM the ACE can be calculated as shown in section 2.2 as $0.08 * 6 \approx 0.64$, representing the effect on HbA1c levels at end of study when always on Sulphonylureas compared to always on insulin.

We now analyse the effect of `Treatment` on time varying `HbA1c`. We set `cutoff =2` to indicate that we are interested in the effect of Treatment on HbA1c level 1 or 2 time periods (6 months to a year) later. Note that as we only calculate up to the two step counterfactuals, `outcomemodels` needs only two formulas.

```
R> Yn = "HbA1c"
R> outcomemodels<-list(
+   "Hba1cEND~Treatment+TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ",
+   "Hba1cEND~Treatment+TreatmentL+HbA1cL+sex+age+ageSQ+HbA1cLSQ")
R>
R> gestboot(gestMultiple,data, idvar, timevar, Yn, An,
+   Cn=NA, outcomemodels, propensitymodel, censoringmodel=NULL,
+   type=3, EfmVar=NA, cutoff=2, bn=1000, seed=123)

$original
c=1.Treatment c=2.Treatment
  -0.01923040    0.02648829

$mean.boot
c=1.Treatment c=2.Treatment
  -0.01939645    0.02637108
```

```
$conf
                       2.5%       97.5%
c=1.Treatment -0.02407830 -0.01432729
c=2.Treatment  0.02107008  0.03144697

$conf.Bonferroni
                      1.25%      98.75%
c=1.Treatment -0.02437562 -0.01356692
c=2.Treatment  0.02016529  0.03211326

$boot.results
# A tibble: 1,000 x 2
   `c=1.Treatment` `c=2.Treatment`
             <dbl>           <dbl>
 1         -0.0243          0.0262
 2         -0.0242          0.0232
 3         -0.0191          0.0267
 4         -0.0170          0.0290
 5         -0.0188          0.0239
 6         -0.0216          0.0242
 7         -0.0220          0.0232
 8         -0.0225          0.0230
 9         -0.0183          0.0276
10         -0.0212          0.0244
# ... with 990 more rows

attr(,"class")
[1] "Results"
```

We note that there is a reduction in HbA1c levels 6 months after taking a Sulphonylureas combination, compared to an insulin combination of around -0.02, but that after a year, those on a Sulphonylureas combination had an increase in HbA1c levels of around 0.03. Both are shown to be significant.

6. Concluding Remarks

The paper introduces and demonstrates a series of functions forming the package **gesttools** for general purpose g-estimation of SNMMs. The package provides a variety of options to users in terms of choice of model specifications, and is applicable to a number of different variable types for both exposure and outcome. These functions have user friendliness in mind, allowing the choice of SNMMs via simple options and a simple specification of the relevant propensity and outcome models.

This implementation of g-estimation retains the double robustness property of the theory they are based on, allowing for unbiased estimates provided that either the propensity or outcome model are correctly specified. We hope that the accessibility of these functions will encourage use of g-estimation by practitioners.

A notable area of future improvement is in the analysis of survival, or time to event outcome data. The package is capable of fitting Structural Nested Cumulative Failure Time Models (SNCFTMs) to survival data in the case of discrete time periods, by treating failure time as a repeated measurement binary outcome (using the methods of Picciotto et al. (2012) and Dukes and Vansteelandt (2018)). However, the packages ability to analyse survival data is limited, for example when survival and exposure are treated as continuous measurements. Other work, such as in Seaman et al. (2019) are capable of fitting SNCSTMs for continuous exposure and outcome measurements, and are more specialised for the analysis of survival data in general. The possibility of implementing additional functionality to **gesttools**, such as the handling of competing risks or outcomes is consideration for future work.

Acknowledgments

We would like to thank the Medical Research Council (MRC), that funded this work, and CRAN for hosting the package of this paper. Daniel Tompsett and Bianca De Stavola were funded by the MRC Methodology Grant, grant number: MR/R025215/1. Stijn Vansteelandt was funded by the Fund for Scientific Research, Flanders, grant number: G019317N. Oliver Dukes was funded by the Ghent Special Research Fund (BOF), grant number: BOF.01P08419.

References

Stefan Milton Bache and Hadley Wickham. ***magrittr**: A Forward-Pipe Operator for R*, 2014. URL `https://CRAN.R-project.org/package=magrittr`. R package version 1.5.

Rhian Daniel, Bianca De Stavola, and Simon Cousens. gformula: Estimating causal effects in the presence of time-varying confounding or mediation using the g-computation formula. *Stata Journa*, 11(4):479–517(39), 2011.

Rhian Daniel, S Cousens, Bianca De Stavola, M Kenward, and Jonathan Sterne. Methods for dealing with time-dependent confounding. *Statistics in Medicine*, 32:1584–1618, 04 2013. doi: 10.1002/sim.5686.

O Dukes and S Vansteelandt. A Note on G-Estimation of Causal Risk Ratios. *American Journal of Epidemiology*, 187:1079–1084, 2018.

Oliver Dukes, Torben Martinussen, Eric J. Tchetgen Tchetgen, and Stijn Vansteelandt. On doubly robust estimation of the hazard difference. *Biometrics*, 75(1):100–109, 2019.

Christopher Gandrud. ***DataCombine**: Tools for Easily Combining and Cleaning Data Sets*, 2016. URL `https://CRAN.R-project.org/package=DataCombine`. R package version 0.2.21.

Lionel Henry and Hadley Wickham. ***tidyselect**: Select from a Set of Strings*, 2020. URL `https://CRAN.R-project.org/package=tidyselect`. R package version 1.1.0.

M Hernán and J Robins. *Causal Inference:What If*. Chapman and Hall/CRC, Boca Raton, 2020.

Max Kuhn, Fanny Chow, and Hadley Wickham. ***rsample**: General Resampling Infrastructure*, 2019. URL https://CRAN.R-project.org/package=rsample. R package version 0.0.5.

Victoria Lin, Sean McGrath, Zilu Zhang, Roger W. Logan, Lucia C. Petito, Jessica G. Young, and Miguel A. Hernán. ***gfoRmula**: Parametric G-Formula*, 2020. URL https://CRAN.R-project.org/package=gfoRmula. R package version 0.3.1.

Roger Logan. GFORMULA-SAS. https://github.com/CausalInference/GFORMULA-SAS, 2019. [Online; accessed 09-06-2021].

M Malawana, S Kerry, R Mathur, and J Robson. Hba1c and hypoglycaemia in intensively treated type 2 diabetes: a retrospective cohort study in primary care. *JRSM Open*, 9(7), 2018.

Roland A. Matsouaka and Eric J. Tchetgen Tchetgen. Instrumental variable estimation of causal odds ratios using structural nested mean models. *Biostatistics*, 18(3):465–476, 02 2017. doi: 10.1093/biostatistics/kxw059.

Lee S. McDaniel, Nicholas C. Henderson, and Paul J. Rathouz. Fast pure R implementation of GEE: application of the Matrix package. *The R Journal*, 5: 181–187, 2013. URL https://journal.r-project.org/archive/2013-1/mcdaniel-henderson-rathouz.pdf.

Kirill Müller and Hadley Wickham. ***tibble**: Simple Data Frames*, 2019. URL https://CRAN.R-project.org/package=tibble. R package version 2.1.3.

Sally Picciotto, Miguel A. Hernán, John H. Page, Jessica G. Young, and James M. Robins. Structural nested cumulative failure time models to estimate the effects of interventions. *Journal of the American Statistical Association*, 107(499):886–900, 2012.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2019. URL https://www.R-project.org/.

J Robins. 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, 1986.

J Robins. *Statistical Models in Epidemiology: The Environment and Clinical Trials*. Springer-Verlag, New York, 2000a.

J Robins, D Blevins, G Ritter, and M Wulfsohn. G-estimation of the effect of prophylaxis therapy for pneumo-cystis Carinii pneumonia on the survival of aids patients. *Epidemiology*, 3:319–336, 1992a.

JM Robins. *Marginal structural models versus structural nested models as tools for causal inference. In Statistical models in epidemiology, the environment, and clinical trials*. Springer-Verlag, New York, 2000b.

S Seaman, O Dukes, R Keogh, and S Vansteelandt. Adjusting for time-varying confounders in survival analysis using structural nested cumulative survival time models. *Biometrics*, pages 1–12, 2019.

A Sjölander and T Martinussen. Instrumental Variable Estimation with the R Package **ivtools**. *Epidemiol.Methods*, 8, 2019.

Jonathan Sterne and Kate Tilling. G-estimation of causal effects, allowing for time-varying confounding. *The Stata Journal*, 2(2):164–182, 2002.

Zhiqiang Tan. On doubly robust estimation for logistic partially linear models. *Statistics and Probability Letters*, 155:108577, 2019.

Eric J. Tchetgen Tchetgen, James M. Robins, and Andrea Rotnitzky. On doubly robust estimation in a semiparametric odds ratio model. *Biometrika*, 97(1):171–180, 12 2009.

Daniel Tompsett, Stijn Vansteelandt, Oliver Dukes, and Bianca De Stavola. ***gesttools**: General Purpose G-Estimation for End of Study or Time-Varying Outcomes*, 2020. URL `https://CRAN.R-project.org/package=gesttools`. R package.

S Vansteelandt and A Sjölander. Revisiting g-estimation of the Effect of a Time-varying Exposure Subject to Time-varying Confounding. *Epidemiol.Methods*, 5:37–56, 2016.

Stijn Vansteelandt and Marshall Joffe. Structural nested models and g-estimation: The partially realized promise. *Statist. Sci.*, 29(4):707–731, 11 2014.

W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. URL `http://www.stats.ox.ac.uk/pub/MASS4`. ISBN 0-387-95457-0.

M Wallace, E Moodie, and D Stephens. Dynamic treatment regimen estimation via regression-based techniques: Introducing R package **DTRreg**. *Journal of Statistical Software*, 80(2):1–20, 2017a.

Michael P Wallace, Erica E M Moodie, and David A Stephens. An R package for g-estimation of structural nested mean models. *Epidemiology*, 28:e18–e20, 2017b.

Linbo Wang, Thomas Richardson, and James Robins. Congenial causal inference with binary structural nested mean models. 09 2017.

Hadley Wickham. testthat: Get started with testing. *The R Journal*, 3:5–10, 2011. URL `https://journal.r-project.org/archive/2011-1/RJournal_2011-1_Wickham.pdf`.

Hadley Wickham and Lionel Henry. ***tidyr**: Tidy Messy Data*, 2019. URL `https://CRAN.R-project.org/package=tidyr`. R package version 1.0.0.