## SNCFTMshell SAS MACRO

The SNCFTMshell macro implements in SAS the estimation of population risks under a hypothetical static intervention (on a binary exposure variable referred to as "treatment") using a Structural Nested Cumulative Failure Time Model. This method is described in detail in [Picciotto et al., 2011]. The algorithm consists of six main steps:

1. modeling the censoring variable(s) to generate inverse probability weights;
2. modeling the treatment variable;
3. g-estimation of the structural model for outcome using the treatment model to adjust for confounding (and weighted to adjust for censoring);
4. using the parameter estimate to compute the expected cumulative risk at each time under no treatment (weighted to adjust for censoring);
5. (optional) using the parameter estimate and the cumulative risks under no treatment to calculate the cumulative risk at each time under continuous treatment or a treatment regimen specified by the user;
6. comparing the cumulative risks under no treatment and (if 5 has been done) under continuous treatment to the risk under no intervention, and (if 5 has be done) comparing the cumulative risks under no treatment and under continuous treatment to one another.

Confidence intervals for estimates of cumulative incidence and relative risk are generated by repeating the procedure described above for many bootstrapped samples.

### Macro Structure

The macro is comprised of several nested macros; the structure is outlined below. Of these only SNCFTMshell requires parameters.

**Table 1. Outline of SNCFTMshell Macro**

```
SNCFTMshell (requires parameters)
    GESTDATAPREP
    do 0 to nsamples
        SNCFTMSAMPLES
        WEIGHTS
        CUMFT
        [check if parameter estimate is valid; if not, and running on original data, then terminate]
        (if psionly = 0)
            LIFEAFTERDEATH
            BLIPDOWNUP
        end
    SNCFTMRESULTS
    SURVCURV
```

The GESTDATAPREP macro does some preparatory manipulation of the data and generates macro variables for later use.

The SNCFTMSAMPLES macro does the bootstrap sampling of the data (starting with sample zero which is the actual data).

The WEIGHTS macro runs regressions predicting the censoring variables as functions of treatment and covariate history and creates a weighted population using the predicted probabilities from these regressions. These weights will be used in the BLIPDOWNUP macro.

The CUMFT macro performs g-estimation using a model for treatment and using the user-specified "blip function" form of the SNCFTM. This macro uses many other sub-macros not described here. It produces the parameter estimate psi and also (for the original dataset, optionally) a graph of the parameter psi vs. the test statistic from the estimating equation.

The LIFEAFTERDEATH macro creates a dataset that includes weights calculated in the WEIGHTS macro, and adds observations after the event of interest is observed for use in calculating the hypothetical outcomes under no treatment.

The BLIPDOWNUP macro has two steps, though the second is optional:
Blipping down (in a submacro called BLIPDOWN) computes for each person-time observation the expected hypothetical risk at that time under the intervention "never treat", using the g-estimate and the data from LIFEAFTERDEATH. From the weighted mean of these, it then outputs the expected risk in the study population at each time had no one ever been treated.
Blipping up computes the overall hypothetical risk under the intervention of interest (either "always treat" or a user-specified treatment regime), using the g-estimate and the risks under no treatment obtained in BLIPDOWN. Blipping up is optional; in some applications, the intervention of interest is "never treated" (as, for example, in a study of a harmful exposure). Also, the macro will not perform this calculation for one of the choices of structural model (*blipfunction*=6), since in that case there is no closed-form method for doing so.

The SNCFTMRESULTS macro takes the intervention results (possibly generated over multiple bootstraps) and manipulates them to output the final data. Risk ratios and risk differences are calculated to compare the risk(s) under the interventions "never treat" and (if blipping up has been done) the intervention of interest to the risk under no intervention, and also (if blipping up has been done) to compare the risk under the intervention of interest to the risk under "never treat". This macro generates 95% confidence intervals for the estimated risk, risk ratios and risk differences using the distributions for these estimates in the bootstrapped samples.

The SURVCURV macro takes the point estimates of risk at each time (obtained in BLIPDOWNUP) and creates a single graph showing survival curves for no intervention, for the intervention "never treat", and (if blipping up has been done) for the intervention of interest.

All of these main macros are defined in the sncftmshell.sas program. There are also a number of utility macros used by SNCFTMshell which are defined in the associated programs. These are included by default as part of sncftmshell.sas and so only the one line "%include '[path]/sncftmshell.sas'" is needed to access the macro. The macro runs many SAS procedures, so it is usually a good idea to specify the SAS "nonotes" option before running it.

**Data Structure**

The data must be arranged with one record per subject per time point until failure or censoring occurs. Additional information about the dataset structure (including required variables and coding) is summarized in Table 2 below. Each record must include the values of each time-varying covariate as measured at that timepoint (in whatever format the covariate is to be used as a predictor), as well as measurements at previous timepoints if those are to be used as predictors, and the fixed/baseline covariate values should be carried forward to each observation. Each record must also include indicators for whether the dichotomous outcome, death from another cause, or loss to follow-up (if these kinds of censoring are present in the data) occurred in the period between that time point and the next. No records should be included for a subject following any of those events. For each individual, there must be an observation for the minimum value of the time variable.

**Table 2. Summary of Required Dataset in the case of survival analysis for a dichotomous outcome***

| ***Parameter*** | **Description** | **Required to Run** | **Required Variables** |
|---|---|---|---|
| id | Subject index | YES | *id* |
| time | Time index | YES | *time* (minimum must be 0 or 1) |
| treat1 | Indicator for treatment/exposure | YES | *treat1* (1 if treated, 0 otherwise) |
| outc | Indicator for outcome | YES | *outc* (1 if outcome, 0 otherwise) |
| censdead | Indicator for censored due to death | NO | *censdead* (1 if censored, 0 otherwise) |
| censlost | Indicator for censored due to lost | NO | *censlost* (1 if censored, 0 otherwise) |
| censpred | Predictors of censoring | YES* | *var1,var2,…* |
| treat1pred | Predictors of treatment | YES** | *cov1,cov2,…* |

* This is a list of variable names (separated by spaces) that are to be used as predictors of censoring in order to calculate inverse probability weights. If there is censoring in the dataset, then this parameter is required.
** This is a list of variable names (separated by spaces) that are to be used as predictors of treatment to adjust for confounding via g-estimation. All potential confounders (whether fixed or time-varying) should be included here.

Note that the lagged value of the treatment variable, if included in the dataset, must be set to 0 at the first time point, because the intervention cannot start before the study period begins. If some individuals were actually treated before time 0, this can be adjusted for by including a baseline variable with the name *treat1* followed by the suffix "*preb*" in the lists of predictors for treatment and/or censoring.

## Models

*Structural model*
An SNCFTM models the effect of a final "blip" of treatment at time m on the outcome at a subsequent time k. Thus we model the ratio of two counterfactual risks at time k (conditional on the measured past) under treatment regimes that differ only at time m: the observed treatment history before time m; treatment or not at time m (in the numerator and denominator respectively); and no treatment after time m. The model for the outcome is fit using weighted g-estimation.

There are six choices of structural model, and the user may specify which model is to be fit using the *blipfunction* parameter in the call, as listed in Table 3. In the formulas in the table, $R_m$ represents the binary treatment variable at time m, Rpreb is a baseline covariate representing "prebaseline treatment", $X_0$ represents an arbitrary binary baseline covariate, and $X_m$ represents an arbitrary time-varying covariate.

**Table 3. Structural models for the effect of a final treatment at time m on the risk at time k.**

| *blipfunction** | Model Formula | Required Additional Parameters |
|---|---|---|
| 1 | $1+[\exp(psi*R_m)-1]/(k-m)$ | - |
| 2 | $\exp(psi*R_m)$ | - |
| 3 | $1+[\exp\{(psi_1+psi_2*R_m)*R_{m-1}\}-1]/(k-m)$ | - |
| 4 | $1+[\exp\{(psi_1+psi_2*X_0)*R_m\}-1]/(k-m)$ | *blem*** |
| 5 | $1+[\exp\{(psi_1+psi_2*Rpreb)*R_m\}-1]/(k-m)$ | *** |
| 6 | $1+[\exp\{(psi_1+psi_2*X_m)*R_m\}-1]/(k-m)$ | *tvem***** |

* Note: the *blipfunction* numbers in the macro do not correspond to the equation numbers in [Picciotto et al. 2011].
**The parameter *blem* is the variable name of the baseline effect modifier, the binary variable $X_0$.
***Results for *blipfunction*=5 will be identical to those obtained for *blipfunction*=4 using *blem*=Rpreb.
****The parameter *tvem* is the variable name of the time-varying effect modifier, the binary variable $X_m$.

When *blipfunction*=1 or 2, the macro will run a one-parameter model. Note that the model of *blipfunction*=1 is compatible with data generated under a particular structural nested accelerated failure time model.[Young et al., 2008]
Otherwise, the macro will run a two parameter model:
When *blipfunction*=3, the model includes an interaction term between current and prior treatment. The prior treatment variable name should be the name of the treatment variable followed by the suffix "*_1I*".
When *blipfunction*=4 or 5, the model evaluates effect modification by a baseline variable $X_0$ (if *blipfunction*=4, $X_0$ is specified by the parameter *blem*). The option *blipfunction*=5 represents a particular choice of $X_0$ for *blipfunction*=4: the baseline effect modifier is the pre-baseline indicator for "treatment", whose name in the dataset should be the name of the treatment variable with the suffix "*preb*" appended to it.
When *blipfunction*=6, the model evaluates effect modification by a time-varying variable $X_m$, specified by the parameter *tvem*. When this structural model is chosen, the program will not blip up since there is no closed-form way to do so under this model.
Note that two-parameter models take much longer to run than one-parameter models.

In addition to the structural model for the effect of treatment on the outcome, we use models to calculate the weights to adjust for censoring and we use a model for treatment to adjust for confounding.

*Weighting models*
The models for the weights predict the probability of remaining uncensored. It is possible to model two different types of censoring separately, specified in the call of the macro as the parameters *censdead* representing censoring by death from competing causes, and *censlost* representing loss to follow-up. If there is only one type of censoring, its variable name should be entered for *censdead*. If censoring is absent, both *censlost* and *censdead* should be listed as blank.

Unstabilized weights are used, and are the inverse of the cumulative probability of remaining uncensored. Included in the models are all variables included in the list given by the parameter *censpred*.

*Treatment model*

The model for probability of treatment will include all variables included in the list given by the parameter *treat1pred*. If *treat1skip* is specified (using a list of times when treatment was not measured, separated by spaces), treatment will not be predicted at those times.

The modeled probability is then used to create a residual that is one of the terms in the estimating equation employed in g-estimation. This is how bias due to confounding is removed.

Note: at present the macro does not support joint treatments involving interventions on more than one covariate, or treatments that are not binary.

**Parameters of the SNCFTMshell Macro**

data (required)

The *data* parameter specifies the dataset to be used for the analysis.

id (required)

The *id* parameter specifies the variable name for the unique identifier for subjects in the dataset *data*.

time (required)
timepoints (required)
mintime (optional, default=0)

The *time* parameter specifies the name of the time point variable in the dataset *data*. The *timepoints* parameter specifies the number of time points in the dataset *data*. The *mintime* parameter specifies the value of *time* at the beginning of the study and can be either 0 or 1 (option *mintime*=1 has not been tested extensively). All subjects must have an observation for *time = mintime*.

outc (required)

The *outc* parameter specifies the outcome variable in the dataset *data*. This must be coded with value 1 if there is an event at a given time point and 0 if not, with censored values coded as missing.

censdead (optional)
censlost (optional)

The *censdead* parameter specifies censoring due to death in the dataset *data*. The variable must be coded with values 1 if censored at a given time point and 0 otherwise. The *censlost* parameter specifies censoring due to loss to follow-up in the dataset *data*. This variable must be coded with values 1 if censored at a given time point and 0 otherwise. If the data has only one type of censoring, leave *censlost*= and use the parameter *censdead* to specify the name of the censoring variable.

censpred (required if at least one censoring variable is specified)

This parameter is a list of variables, separated by spaces, to include in the models for probability of remaining uncensored for the calculation of weights.

treat1 (required)

The *treat1* parameter specifies the name of the dichotomous variable representing time-varying treatment to be modeled; 1 should correspond to treated/exposed and 0 to untreated/unexposed.

treat1skip (required if treatment is not measured at every time)

The *treat1skip* parameter specifies time points at which *treat1* was not measured. Data from these time points will not be included in the pooled models with *treat1* as the dependent variable.

treat1class (optional)

The *treat1class* parameter specifies (by name) a variable on which to stratify the treatment models. This variable must be coded 0/1, and results in separate models for treatment within each level. This is particularly useful for modeling things like number of cigarettes, which can be modeled separately among those who did and did not smoke at the previous time point.

treat1pred (required)

This parameter is the list of variables to include in the model for probability treatment. All potential confounders should be included here, since this is how g-estimation adjusts for confounding.

regimen (optional, default=1 1 … 1)

This parameter may be used to specify the intervention of interest: the treatment regimen for blipping up. It should be a space-separated list (of length *timepoints*) of 0s and 1s indicating whether the intervention of interest requires treatment at each respective time point. If *regimen* is not specified (and *blipup*=yes), then the macro assumes the intervention of interest is "always treat".

intlabel1 (optional, but probably a good idea)
intlabel2 (optional, but probably a good idea)

The parameters *intlabel1* and *intlabel2* are where the user can specify descriptive labels for the intervention of interest ("always treat" or a user-specified *regimen*) and "never treat", respectively.

blipfunction (optional, default=1)

This parameter allows the user to select different structural models; see table 3.

blem

This parameter gives the name of the baseline effect modifier to use in the structural model if *blipfunction*=4.

tvem

This parameter gives the name of the time-varying effect modifier to use in the structural model if *blipfunction*=6.

psionly (optional, default=0)

When *psionly*=1, macro will perform g-estimation and then stop without blipping down and up to compute the risks under interventions.

blipup (optional, default=yes)

When *blipfunction* is not 6 and *psionly*=0, the parameter *blipup* determines whether the macro computes the risk under an intervention other than never treat. When *blipup*=yes, *blipfunction*<6, and *psionly*=0, the macro will compare the risks under the intervention of interest and "never treat" to each other and to the risk under "no intervention". When *blipup*=no (or *blipfunction*=6), the macro will only compare the risk under "never treat" to the risk under no intervention.

dir (optional)
intervs_all (optional, required if *parallel* is used, see below)
weightdata (optional, default=svwts)
resultsdata (optional)

The *dir* parameter specifies a path for the directory in which to save output datasets; if not specified, datasets will be saved in the SAS work library. The *intervs_all* parameter specifies a dataset in which to store the information from parallel runs of bootstraps in order to combine them afterwards; see below. The *weightdata* parameter specifies a dataset in which to store the weights used in BLIPDOWNUP. The *resultsdata* parameter specifies a dataset in which to store the results (risks, risk differences, and risk ratios, etc.).

nointervdata (optional, default=refdata)

The parameter *nointervdata* specifies the base name for the dataset (with index 0) containing the estimated risk under no intervention in each bootstrap sample and also (when *blipup*=1, with index 2) the dataset containing the estimated risk under "never treat". Unless parameter *dir* is specified, the data will be saved in the work directory.

seed (optional, default = 7834)

The *seed* parameter specifies the random numbers seed for generating bootstraps.

nsamples (optional, default = 0)

The *nsamples* parameter specifies the number of bootstrap samples to be conducted. The default is 0, meaning that no bootstraps will be computed and the results will give point estimates for the full dataset with no confidence intervals.

parallel (optional)

The *parallel* parameter can be used for running bootstraps in parallel. If the program is to be run multiple times (say, *x*) in parallel with bootstraps using different seeds, then the same number (say, *y*) of bootstraps should be done in each of the *x* parallel runs, and certain changes must be made to the call:

*parallel* should take values 1, 2, …, *x* in the respective runs;
in each one, *seed* should take value 3*y* higher than it did in the previous one;
*dir* and *intervs_all* must be specified so that the macro can save the results data separately for each parallel run, in a format to be able to combine them afterwards, and *dir* should be the same in each parallel run of the same analysis;

if more than one analysis is being run (e.g. different treatment, set of covariates, or outcome), *dir* should be different for each one in order to avoid overwriting datasets created by the macro; and

another program, COMBINEPARALLELS, must be run once all bootstraps have finished (see below).

Each parallel run will produce several datasets, differentiated only by the suffix *parallel*, that will be combined later when running COMBINEPARALLELS.

outputs(optional, default = yes)

The *outputs* parameter specifies whether any output should be printed in the log file. In general, this should be left as yes.

survcurv (optional)

This parameter provides the full path and file name in which to save a pdf file containing the survival plot (the .pdf extension should not be included in the file name as the macro will add it).

survaxis (optional)

If not specified, SAS will use default settings for the survival plot, and will include only the portion of the vertical axis containing data. This parameter allows the user to define the vertical scale of the survival plot in the format:

*min* to 1.000 by *delta*

srchbnd (optional, default=3.0)

starting_point (optional, default=0.0 or 0.0 0.0)

The parameter *srchbnd* provides the width of interval in which to search for each parameter of the structural model, and *starting_point* specifies the center of the interval for grid search and/or optimization procedure in the g-estimation procedure. These can be changed if program does not find a minimum within the original search grid. They should be specified as real number(s) using at least one digit after the decimal point so that the macro does not run into errors trying to assume they are integers. It is probably better to change *srchbnd* than *starting_point*, and the macro may not work if the interval is not a strict subset of the interval (-10,10).

est_method (optional, default=2)

This parameter specifies the optimization procedure to be used:

0=no estimation done

1=Nelder-Mead Simplex method

2=Newton-Raphson method

3=Simulated annealing

4=bisection method (only for 1-parameter structural models: *blipfunction*=1 or 2)

In general, method 2 is the fastest, but occasionally SAS errors will occur where it fails to optimize in a fatal way and exits the program. In those cases, method 1 is recommended. Methods 3 and 4 have not been tested extensively within the context of this shell.

save_graph (optional, default=0)
graph_path (required if *save_graph*=1)
graph_path2 (required if *save_graph*=1 and *blipfunction*=3, 4, or 5)

When *save_graph*=1, macro saves the graph of test statistic vs. psi in a pdf file with the path and filename given by *graph_path*, without the .pdf extension. In the case of a 2-parameter model, this will be a three-dimensional graph shown from several angles, and in that case *graph_path2* should be specified as well to save the contour graph (which will otherwise overwrite the 3D graph). The contour graph shows contours at 1, 2, 3, 4, 5, and 6. The chi level for a two-parameter model is 5.99, so the contour at 6 gives an idea of the 95% confidence region for the parameter.

**COMBINEPARALLELS Macro and parameters**

This macro, to be run separately after multiple parallel runs of SNCFTMshell, combines the results from the parallel runs and gives the output that would have been obtained if the bootstraps had all resulted from a single run of SNCFTMshell. This macro is in sncftmshell.sas by default, and it uses the submacro SNCFTMRESULTS, so the line %include "[path]/sncftmshell.sas" should be at the top of the program in which this macro is called.

*If different analyses will be run, e.g. for different datasets, treatments, outcomes, or covariates, these must be done in separate directories using the parameter* ***dir****. When doing parallel bootstrap runs, the SNCFTMshell macro creates some datasets with names not specified by the user, which will be overwritten if more than one such analysis is run in the same directory.*

The macro COMBINEPARALLELS takes the following parameters, several of which should be given the same values as those from the original SNCFTMshell runs.

nsets (required)

This is a number counting the datasets of each type previously saved for use in %sncftmresults. The suffixes on the names of those datasets should go from 1 to *nsets*. This also corresponds to the number of parallel runs of SNCFTMshell.

nsamples (required)

This is the total number of bootstrap samples in the combined dataset to be created.

ssize (required)

This is the number of individuals in the original dataset.

npsi (required)

The number of parameters estimated in the original runs of SNCFTMshell: *npsi*=1 for blip functions 1 and 2, while *npsi*=2 for blip functions 3, 4, and 5.

data (required)
outc (required)
time (required)
timepoints (required)
censdead (optional)
censlost (optional)

These parameters should take the same values they did in the original runs of SNCFTMshell. The parameter *data* is the original dataset used, *outc* is the name of the outcome variable, *time* is the name of the time variable, and *timepoints* is the maximum number of observations per individual. The parameters *censdead* and *censlost* give the names of the censoring variables. If censoring of one type (or both) is absent from the dataset, these can be specified as blank.

nointervdata (optional, default=refdata)
blipup (optional, default=yes)
These parameters should take the same values they did in the original runs of SNCFTMshell. The parameter *nointervdata* is the base name of the files containing the reference data, and will be followed by two indices. If *blipup*=yes then the macro compares the intervention of interest to both "never treat" and "no intervention", so there will be two previously saved datasets of reference data for each parallel run, indexed with 0 (natural course) and 2 (never treat) followed by the index of the parallel run. These datasets will be combined into two reference datasets (indexed 0 and 2), each containing all bootstrap reference results corresponding to the respective reference intervention. If *blipup*=no then the macro compares never treat to natural course only. There will only be one previously saved set of reference data saved for each parallel run, indexed with 0 followed by the index of the parallel run; these will be combined into one reference dataset called *nointervdata* followed by 0.

dir (required unless work directory was used in the original parallel runs)
Gives the directory path for where to save the datasets (must be the same as the one where datasets were saved in the original SNCFTMshell runs).

outputs (optional, default=yes)
Allows the user to decide whether to print results. In general, should be left as yes.

resultsdata (optional)
Allows the user to specify a file name for saving the results.

nocens (optional, default=0)
If there is no censoring in the dataset, this should be set to 1.

deldata (optional, default=no)
Specifies whether to delete all the datasets created in the parallel runs after having combined them. Safest to leave at "no", though after the final results have been successfully obtained, the macro could be run again with *deldata*=yes to avoid wasting disk space.

**Sample Code**

The file sncftmsample.sas creates a sample dataset and then runs the SNCFTMshell macro on that data. The code for the macro call is also included below with an explanation. The sample dataset has 6 timepoints (numbered 0 to 5), an outcome of diabetes (dia), the fixed covariate baseline age (baseage), and the time-varying covariates time, hypertension (hbp) and body mass index (bmi), which has been log-transformed. Censoring can occur by death due to competing risks or loss to follow-up. The treatment A is defined as "having BMI<=25" (actually, "log(bmi)<=3.22"). The variables are generated randomly so there is no reason for the prediction of outcome to be good or for the interventions specified to have any effect other than noise.

```
options nonotes;
%include "sncftmshell.sas";
/*CALL TO MACRO*/

%sncftmshell(
   data= sampledata,
   id=id,
   time=time,
   timepoints = 6,
   mintime=0,
   outc = dia,

   censdead = censdead,
   censlost = censlost,
   censpred = baseage hbp_b bmi_b time_1 time_2 time_3 time_4 time_5 hbp bmi bmi_l1,
   weightdata=natcowts,

   nointervdata = refdata

   psionly=0,
   blipup=yes,
   blipfunction=1,
   intlabel1= BMI <= 25,
   intlabel2= BMI > 25,

   treat1=A,
   treat1pred=baseage hbp_b bmi_b time_1 time_2 time_3 time_4 time_5 hbp bmi_l1 bmi_l2,
   parallel=,
   seed= 741079,
   nsamples = 10,
  );
```

This code runs the SNCFTMshell, specifying the dataset "sampledata", the id variable "id", the time variable "time", the number of timepoints (6), the minimum time (0), the outcome "dia", censoring variables "censdead" and "censlost", the treatment variable (A), and the model to use (*blipfunction*=1). Labels have been specified for the intervention of interest, which is "always treat", and for "never treat". The weights will be saved in a dataset called "natcowts" and the reference data in datasets called "refdata0" (risk under no intervention) and "refdata2" (risk under "never treat"). The macro will first perform g-estimation, and then (because *psionly*=0) blip down to find the risk under the intervention "never treat", and finally (because *blipup*=yes) blip up to find the risk under the intervention "always treat".

Censoring by competing risks ("censdead") and by loss-to-follow-up ("censlost") is adjusted for using inverse probability weights. Treatment A (an indicator for BMI above or below 25) and the two censoring variables are modeled by logistic regressions. The models for probabilities of treatment and of censoring include time indicators, baseline age, baseline hypertension and BMI ("hbp_b" and "bmi_b"), and time-varying hypertension and BMI. Hypertension enters models as an indicator for being hypertensive in the previous period. We have not included A in the model for censoring, instead using the 2 previous measures of BMI itself.

The random number seed is set to 741079 to ensure rerunning the code yields the same results. Confidence intervals are based on 10 bootstrapped repeats.

## LICENSE



## CITATIONS

Young, J. G., Hernán, M. A., Picciotto, S., and Robins, J. M. (2008), "Simulation from Structural Survival Models under Complex Time-Varying Data Structures," in Joint Statistical Meetings, American Statistical Association.

Picciotto, S., Hernán, M. A., Page, J.H., Young, J.G., Robins, J.M. “Structural Nested Cumulative Failure Time Models to Estimate the Effect of Hypothetical Interventions”, in preparation.

## FUNDING SOURCE

This work was partially funded by NIH grant R01 HL080644.