

Function Hierarchy for MultiCure

Lauren J Beesley

Department of Biostatistics, University of Michigan

Contact: lbeesley@umich.edu

September 13, 2017

This vignette briefly outlines how the functions in package *MultiCure* fit together. In particular, the purpose of this vignette is to document how the user-facing functions are related to other functions included in *MultiCure*. This may be useful for users looking to modify the provided code for their own purposes.

1 Function Hierarchies

In this section, we provide diagrams describing the relationships between the functions within the package **MultiCure**. **Figure 1** shows the functions called within function MultiCure. Function COVIMPUTEINITIALIZE is user-specified and initializes the missing covariate values if there are any. If there are no missing covariates, this function can be left unspecified. The M Step of the EM and MCEM algorithms are performed by functions MStep_WEIB and MStep_COX for Weibull and Cox baseline hazards respectively. The E Step is performed using the functions listed.

Figure 1: Functions used within MultiCure

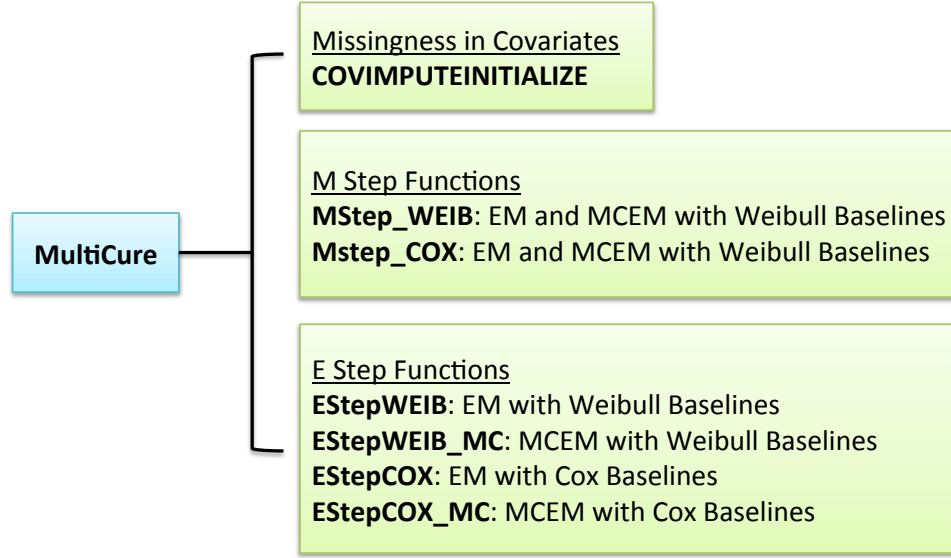


Figure 2 shows the functions used within the E Step functions of the EM and MCEM algorithms. Estimation of the baseline hazards is only necessary when we have Cox baseline hazards, and the function used depends on whether we are using an EM or MCEM algorithm. When we have covariate missingness, function COVIMPUTEFUNCTION obtains a single imputation of the covariate set. When we have unequal censoring of the two outcomes, we perform imputation using one of the listed functions. For Weibull baseline hazards, two different default options for performing this imputation are provided, and four options are provided for Cox baseline hazards. If desired, the user can also specified his/her own function for dealing with the unequal censoring (through imputation).

Figure 2: Functions used within E Step

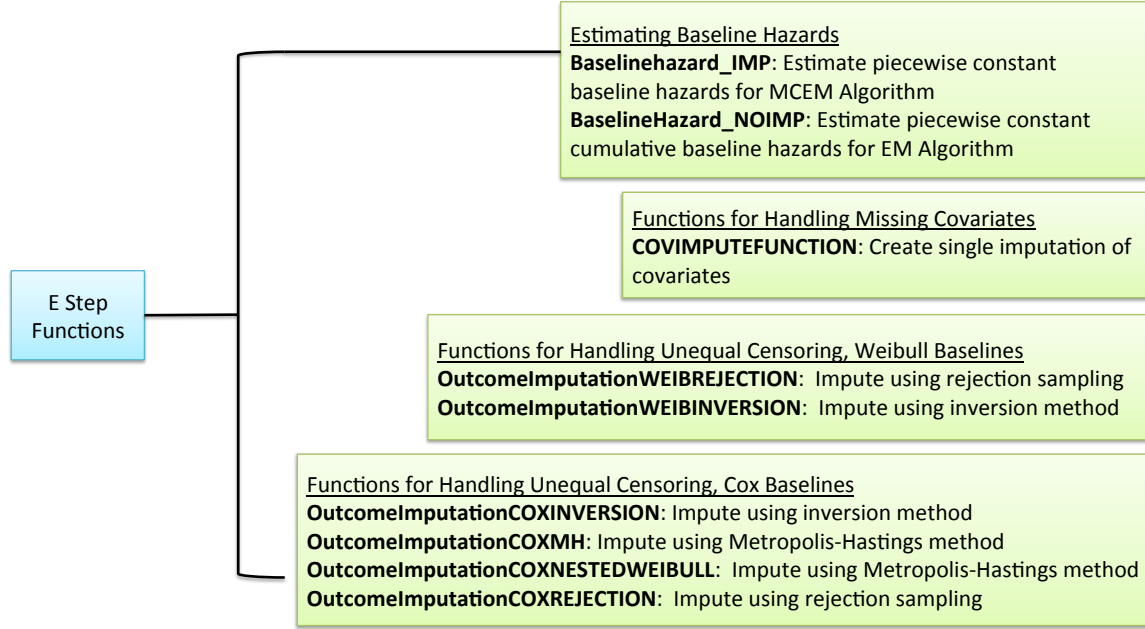


Figure 3 shows the functions used within the M Step functions of the EM and MCEM algorithms. These two functions obtain the augmented version of the data used for each of the maximizations.

Figure 3: Functions used within M Step

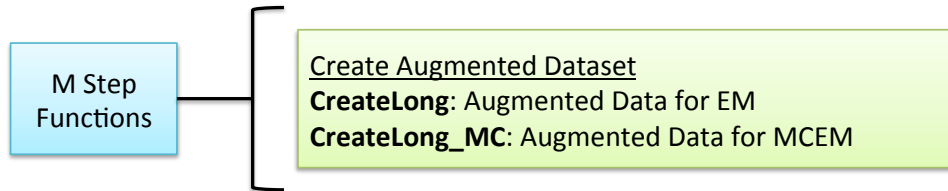


Figure 4 shows the functions used within the function ProperDraws_MC, which is used to obtain "proper" imputations for the missing data for the MCEM algorithm. This function is not used in the EM algorithm.

Figure 4: Functions used within ProperDraws_MC

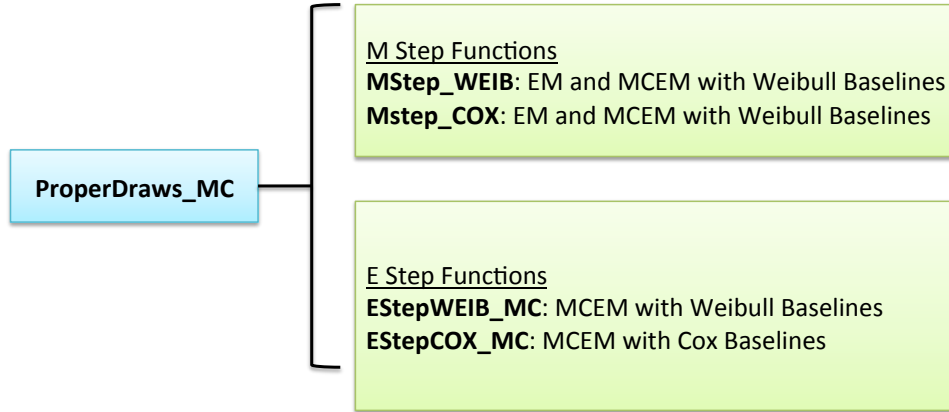


Figure 5 shows the functions used within each of the functions used to estimate standard errors. The function **VarianceEM** obtains standard errors for the EM algorithm via bootstrap. The function **VarianceMCEM_BOOT** estimates the standard errors for the MCEM algorithm via bootstrap for each imputed dataset and then combines across imputed datasets using Rubin's Rules. The function **VarianceMCEM_NOBOOT** estimates standard errors for each imputed dataset by fitting a multistate cure model and use the standard software-estimated standard errors. Then, these imputation-specific standard errors are combined across imputations using Rubin's Rules.

Figure 5: Functions used within Variance Estimation Functions

