

# LTMLE manual

08-01-2023

The aim of this document is to provide practical advice on how to shqp typical observational data in order to conduct a longitudinal targeted maximum likelihood estimation (LTMLE) analysis.

This practical manual is based on the `data.table` package and further the `heaven` package which is available on [github/tagteam/heaven](https://github.com/tagteam/heaven).

## Raw data formatting

First we will create the basic data necessary for the analysis as they often are available. All basic information of time dependent variables is based on dates of events. The following block creates the data for this example

```
library(data.table)
library(heaven)
set.seed(21)
n <- 10
W <- rnorm(n) # A baseline variable
ID<-1:n
startDate<-as.Date("2010-01-01")+sample(0:365,n,replace=TRUE)
L1<-startDate+sample(0:365,n,replace=TRUE)
L2<-startDate+sample(0:365,n,replace=TRUE)
ebd <- pmax(L1,L2)+sample(0:180,n,replace=TRUE)
base <- data.table(ID,startDate,dummy=1,W)
select <- sample(1:3,n,replace=TRUE) # event, censor, competing
covariatesEvents <- data.table(ID,L1,L2,select,ebd)
covariatesEvents[select==3,Y:=ebd]
covariatesEvents[select==2,C:=ebd]
covariatesEvents[select==1,compete:=ebd]
covariatesEvents[,c("select","ebd"):=NULL]
select2 <- sample(1:2,n,replace=TRUE)
select3 <- sample(1:2,n,replace=TRUE)
timeVarCov <- data.table(ID,select2,startDate)
timeVarCov[select2==1,':='(inn=startDate,out=startDate+sample(0:720,1),var='L3')]
timeVarCov[select2==2,':='(inn=startDate+75,out=startDate+75+sample(0:720,1))]
timeVarCov2 <- data.table(ID,select3,startDate)
timeVarCov2[select3==1,':='(inn=startDate,out=startDate+sample(180:720,1),var='A')]
timeVarCov<-rbind(timeVarCov,timeVarCov2,fill=TRUE)
timeVarCov<-timeVarCov[!is.na(inn) & !is.na(var),.(ID,var,inn,out)]
timeVarCov[,value:='1']
```

To use this manual, the raw data should be formatted as three datasets. The first is contains the individual identification (ID), the date of entry into the study (startDate), variables that are fixed at baseline (W) and a dummy variable that should be either zero or one

`base`

The second dataset includes dates of variables that only change once during follow-up. This includes covariates such as diseases (L) and also how follow-up ends in either outcome (Y), censoring (C) and competing risk

(compete)

`covariatesEvents`

The third dataset includes data for variables that may change multiple times including multiple times for each individual. For the example it includes a third covariate (L3) and the treatment of interest (A). This dataset includes the identification (ID), a variable with the name of the time varying variable (var), start and end of each period (inn/out) and a variable (value) with the content “1” indicating “exposure”

`timeVarCov`

## Splitting by time varying events

The first step is to fix the end time of the analysis LTMLE used a range of time intervals of equal length and the same number of time intervals for all individuals.

For the example 4 time intervals of 180 days is chosen and therefore

```
MyData[endDate:=startDate+4*180]
```

The data are next split by all time dependent variables and events. The order of splitting is not important.

To split by the covariates and events where each is characterized by a single date the `lexisTwo` function is used

```
longSplit <- lexisTwo(indat=base,
                      covariatesEvents,
                      invars=c('ID','startDate','endDate','dummy')
                      splitvars = c('ID',"L1","L2","Y","C","compete"))
```

Next the data are split by the time dependent variables with potentially multiple changes during the study

```
longSplit <- lexisFromTo(indat=base,
                        covariatesEvents,
                        invars=c('ID','startDate','endDate','dummy')
                        splitvars = c('ID','inn','out','value','var'))
```

Finally, the data are split by the selected time periods, in the current case four periods of 180 days

```
longSplit <- lexisSeq(indat=base,
                     timeVarCov,
                     invars=c('ID','startDate','endDate','dummy')
                     splitvars = c('ID','inn','out','value','var'))
```