

# Simulation Study

Alvin Sheng

4/6/2019

```
library(survival)
library(glmnet)
```

```
## Loading required package: Matrix
## Loading required package: foreach
## Loaded glmnet 2.0-16
```

```
library(polyspline)
library(knitr)
library(EnvStats)
```

```
##
## Attaching package: 'EnvStats'
## The following object is masked from 'package:Matrix':
##
##      print
## The following objects are masked from 'package:stats':
##
##      predict, predict.lm
## The following object is masked from 'package:base':
##
##      print.default
```

## Simulating Survival Time with a Weibull Distribution

This function is based on `simulate_data` in <https://cran.r-project.org/web/packages/rsimsum/vignettes/relhaz.html>

```
## Simulate survival times with censoring, based on a weibull baseline hazard
##
## This function simulates survival times with censoring, according to a weibull
## baseline hazard that the user parameterizes. The survival/censoring times are
## simulated for user-given covariates and coefficients.
## @param x model matrix (including intercept) of x-values for the Cox model of survival times
## @param fcts_select subset of fcts from a hare object containing the coefficients of interest.
## @param params parameters shape and scale for the baseline Weibull distribution, by default the expon
## @param FUN random generation function for the distribution of censoring times, expected to be unifor
## @param ... arguments for FUN, the random generation function
## @return
## @export
simulate_weibull <- function(x, fcts_select, params = list(shape = 1, scale = 1), FUN, ...) {

  n <- nrow(x)
```

```

# extract unique list of covariates selected
cov_nums <- sort(fcts_select[,1][fcts_select[,1] != 0])
cov_names <- colnames(x)[cov_nums]
x_select <- x[,cov_names]

# extract the coefficient values from fcts_select
betas <- fcts_select[,5][fcts_select[,1] != 0]

# simulate survival times according to Bender et al. (2005)
u <- runif(n)
time <- (-log(u) / (params$scale * exp(x_select %*% betas)))^(1 / params$shape)

# Censoring
# **
cid = 1

# return a dataframe
data.frame(time, cid, x)
}

```

```

load("actg175.RData")

x <- model.matrix( ~ trt + age + wtkg + hemo + drugs +
                   karnof + oprior + preanti + race +
                   gender + symptom + offtrt + cd40 +
                   cd80, actg175)

# x <- readRDS("actg175_mat.rds")

nphm_hare <- readRDS("nphm_hare.rds")

# extracting the coefficients for basis functions
# that do not correspond to knots and/or tensor products
fcts <- nphm_hare$fcts
fcts_select <- fcts[fcts[,2] == 0 & is.na(fcts[,3]),]

```

```
set.seed(513)
```

```
test1 <- simulate_weibull(x, fcts_select)
```

```
summary(test1$time)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
## 0.000e+00 1.990e+02 9.290e+02 9.439e+06 4.714e+03 1.988e+10
```

```
head(test1)
```

```
##      time cid X.Intercept. trtZDV.ddi trtZDV.ZAL trtddi age  wtkg
## 1502   9.396669   1         1         0         0     1  30 64.6380
## 1432  66.509367   1         1         0         0     0  36 74.3904
## 1832 2668.521917   1         1         0         1     0  38 78.0192
## 1331  33.319335   1         1         0         1     0  43 64.4112
##  886  195.606678   1         1         0         0     0  46 86.1000
##  221  543.111773   1         1         0         0     0  38 69.5000
##      hemo1 drugs1 karnof oprior1 preanti race1 gender1 symptom1 offtrt1
```

##	1502	0	0	90	0	0	0	1	1	0
##	1432	0	0	80	0	381	0	1	1	1
##	1832	0	0	90	0	1099	0	1	0	0
##	1331	0	0	70	0	29	0	1	1	1
##	886	0	1	90	0	1423	0	1	0	1
##	221	0	0	100	1	113	1	1	0	0
##		cd40	cd80							
##	1502	131	227							
##	1432	173	724							
##	1832	416	1663							
##	1331	164	944							
##	886	311	1867							
##	221	375	986							