

Installation Instructions

Eric Novik, Generable Inc.

18 Jun 2021

Contents

Introduction	1
Installing RStan	1
Installing RStanArm Survival Branch	2
Getting Help and Other Resouces	3
References	4

Introduction

This markdown document explains how to install required software for the Bayesian Survival and Joint Models using Rstanarm for the NCB 2021 confernece class taught by Jacqueline Buros.

Please, exexecute all steps and verify that the software works before the class. We will not have time to troubleshoot installation problems during the class but if you experience any problems please send us an email to ncb2021@generable.com.

Stan (Carpenter et al. 2017) is a programming language with its own compiler and in this class, we will be accessing Stan from R directly via the RStan package and indirectly from an RStanArm package which provides a set of pre-compiled Stan models including Survival models.

Installing RStan

For this class, you should be using R version 4.0 or later and we recommend using RStudio version 1.4 or later. To install RStan, execute the following commands.

```
install.packages("rstan", dependencies = TRUE)
```

On most systems, this should work without a problem, but if you are experiencing installation issues, take a look a the RStan Getting Started document.

Execute the following code to test your installation.

```
# In general, you should write your Stan code in a separate file  
# but this will work for testing:  
stanmodelcode <- "  
data {  
  int<lower=0> N;
```

```

    real y[N];
  }

parameters {
  real mu;
}

model {
  target += normal_lpdf(mu | 0, 10);
  target += normal_lpdf(y | mu, 1);
}
"
library(rstan)
y <- rnorm(20)
dat <- list(N = 20, y = y);

# This will compile the model and start the sampling process
# It may take 10-30 seconds depending on your machine
fit <- stan(model_code = stanmodelcode,
            model_name = "example",
            data = dat)

print(fit)

```

The last command should produce something like the following.

Inference for Stan model: example.
 4 chains, each with iter=2000; warmup=1000; thin=1;
 post-warmup draws per chain=1000, total post-warmup draws=4000.

	mean	se_mean	sd	2.5%	25%	50%	75%	97.5%	n_eff	Rhat
mu	-0.08	0.01	0.23	-0.54	-0.24	-0.07	0.08	0.38	1236	1
lp__	-30.48	0.02	0.78	-32.72	-30.63	-30.18	-29.99	-29.94	1246	1

In addition, install the following R packages.

```

p <- c('survival', 'bayesplot', 'RISCA', 'tidyverse', 'remotes')
install.packages(p)

# helper functions
remotes::install_github('generable/rgeco')

```

Installing RStanArm Survival Branch

The other major package we will use is RStanArm. The advantage of RStanArm is that it comes with a large set of pre-compiled models that have been tested for computational efficiency.

As of this writing, the Survival functions are not available on CRAN, and so you have to install the package directly from GitHub as follows.

```
# This will take a while -- more than a few minutes as lots of Stan
# models are being compiled. Go have a cup of coffee if that's your thing.
remotes::install_github("stan-dev/rstanarm", ref = "feature/survival", build_vignettes = FALSE)
```

To test if the installation worked execute the following code.

```
library(rstanarm)
library(survival)
library(dplyr)

data(leukemia) # from Survival package
fit <- stan_surv(formula = Surv(time, status) ~ x,
                 data = leukemia,
                 basehaz = "exp")

print(fit)
```

After executing the print command, you should see something like this.

```
stan_surv
baseline hazard: exponential
formula:         Surv(time, status) ~ x
observations:    23
events:          18 (78.3%)
right censored:  5 (21.7%)
delayed entry:   no
-----
              Median MAD_SD exp(Median)
(Intercept)   -4.1    0.4    NA
xNonmaintained 0.9    0.5    2.6
-----
* For help interpreting the printed output see ?print.stanreg
* For info on the priors used see ?prior_summary.stanreg
```

Getting Help and Other Resources

For information on the Bayesian Survival analysis see the paper *Bayesian Survival Analysis Using the rstanarm R* (Brilleman et al. 2020). For a description of the Joint Models in Bayesian context see the *Joint longitudinal and time-to-event models for multilevel hierarchical data* paper (Brilleman et al. 2019).

For the introduction to the RStanArm package see *How to Use the rstanarm Package* vignette. For more information on Joint Models in RStanArm see this and this.

Stan is extremely well documented. Stan User Manual and other Stan documentation are available here.

If you have a question about Stan including the language and models, check out the user-friendly community forum.

We hope you enjoy the workshop. If there anything we can do to improve, please send us feedback to ncb2021@generable.com.

References

- Brilleman, Samuel L, Michael J Crowther, Margarita Moreno-Betancur, Jacqueline Buros Novik, James Dunyak, Nidal Al-Huniti, Robert Fox, Jeff Hammerbacher, and Rory Wolfe. 2019. “Joint Longitudinal and Time-to-Event Models for Multilevel Hierarchical Data.” *Statistical Methods in Medical Research* 28 (12): 3502–15. <https://doi.org/10.1177/0962280218808821>.
- Brilleman, Samuel L., Eren M. Elci, Jacqueline Buros Novik, and Rory Wolfe. 2020. “Bayesian Survival Analysis Using the Rstanarm R Package.” *arXiv:2002.09633 [Stat]*, February. <http://arxiv.org/abs/2002.09633>.
- Carpenter, Bob, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. 2017. “Stan: A Probabilistic Programming Language.” *Journal of Statistical Software* 76 (1): 1–32. <https://doi.org/10.18637/jss.v076.i01>.