

Build Survival Model: XGBoost

Mingcheng Hu

Table of contents

Load Data	2
XGBoost	4
Data Preparation	4
Hyperparameter Tuning	5
Variable Selection	8
Cross Validation to Select the Best Number of Features	11
Model Fitting	13

```
library(tidyverse)
library(survival)
library(xgboost)
library(caret)
library(survcomp)
library(parallel)
library(mcprogress) # wrap mclapply with progress bar.
library(kableExtra) # include knitr automatically
library(mlr3) # hyperparameter tuning
library(mlr3tuning)
library(paradox)

source("/work/users/y/u/youukias/BIOS-Material/BIOS992/utils/csv_utils.r")
# * Don't use setwd() for Quarto documents!
# setwd("/work/users/y/u/youukias/BIOS-Material/BIOS992/data")

adjust_type <- ifelse(exists("params"), params$adjust_type, "minimal") #
  ↪ options: "minimal", "partial", "full"
impute_type <- ifelse(exists("params"), params$impute_type, "unimputed") #
  ↪ options: "unimputed", "imputed"
```

```
include_statin <- ifelse(exists("params"), params$include_statin, "no") #
  ↪ options: "yes", "no"

# hyperparameter tuning trials
n_trials <- 50 # * It is recommended to set it to n_params * (8~10)
n_folds <- 10
set.seed(1234)
```

```
# string of parameters
adjust_type_str <- switch(adjust_type,
  minimal = "minimal",
  partial = "partial",
  full = "full"
)
print(paste0("Model Adjustment Type: ", adjust_type_str))
```

```
[1] "Model Adjustment Type: minimal"
```

```
impute_type_str <- switch(impute_type,
  unimputed = "unimputed",
  imputed = "imputed"
)
print(paste0("Data Imputation Type: ", impute_type_str))
```

```
[1] "Data Imputation Type: unimputed"
```

Load Data

```
if (include_statin == "yes") {
  data_train <-
  ↪ read.csv(paste0("/work/users/y/u/youkias/BIOS-Material/BIOS992/data/train_data_",
  ↪ impute_type_str, "_statin.csv"),
    header = TRUE
  )
} else {
  data_train <-
  ↪ read.csv(paste0("/work/users/y/u/youkias/BIOS-Material/BIOS992/data/train_data_",
  ↪ impute_type_str, ".csv"),
```

```

    header = TRUE
  )
}

data_train <- data_train[, -1] # the first column is the index generated by
↪ sklearn
(dim(data_train))

```

```
[1] 28127    100
```

```

data <- select_subset(data_train, type = adjust_type)
(dim(data))

```

```
[1] 28127    48
```

```
colnames(data)
```

```

[1] "event"           "time"
[3] "HRV_SD1"         "HRV_SD2"
[5] "HRV_SD1SD2"      "HRV_S"
[7] "HRV_CSI"         "HRV_CVI"
[9] "HRV_CSI_Modified" "HRV_PIP"
[11] "HRV_IALS"        "HRV_PSS"
[13] "HRV_PAS"         "HRV_GI"
[15] "HRV_SI"          "HRV_AI"
[17] "HRV_PI"          "HRV_C1d"
[19] "HRV_C1a"         "HRV_SD1d"
[21] "HRV_SD1a"        "HRV_C2d"
[23] "HRV_C2a"         "HRV_SD2d"
[25] "HRV_SD2a"        "HRV_Cd"
[27] "HRV_Ca"          "HRV_SDNNd"
[29] "HRV_SDNNa"       "HRV_ApEn"
[31] "HRV_ShanEn"      "HRV_FuzzyEn"
[33] "HRV_MSEn"        "HRV_CMSEn"
[35] "HRV_RCMSEn"      "HRV_CD"
[37] "HRV_HFD"         "HRV_KFD"
[39] "HRV_LZC"         "HRV_DFA_alpha1"
[41] "HRV_MFDFA_alpha1_Width" "HRV_MFDFA_alpha1_Peak"
[43] "HRV_MFDFA_alpha1_Mean"  "HRV_MFDFA_alpha1_Max"

```

```
[45] "HRV_MFDFA_alpha1_Delta"      "HRV_MFDFA_alpha1_Asymmetry"
[47] "HRV_MFDFA_alpha1_Fluctuation" "HRV_MFDFA_alpha1_Increment"
```

```
data <- tibble::as_tibble(data)
```

```
# * It is very hard to compare the HR as different predictors are on
  ↪ different magnitudes, so we need to normalize them.
time_col <- data$time
event_col <- data$event
data <- data %>%
  select(-c(time, event)) %>%
  mutate(across(where(is.numeric), scale)) %>%
  mutate(
    time = time_col,
    event = event_col
  )
```

Note now the interpretation of HR is different! For example, if HR=1.16 for the predictor in the univariate model fitted using scaled data, it means that each standard deviation increase is associated with 16% higher risk of event.

```
# For XGBoost model, we create a validation set for early stopping.
set.seed(1234)
train_index <- createDataPartition(
  data$event, # stratify by event
  p = 0.8,
  list = FALSE
)

train_data <- data[train_index, ]
val_data <- data[-train_index, ]
```

XGBoost

Data Preparation

```

total_x <- as.matrix(data %>% select(-c(time, event)))
total_y <- data %>% select(time, event)
# * Note format of label should be different when using Cox model and AFT
  ↪ model.
# define For uncensored labels, use [a,a]
# define For right-censored labels, use [a,Inf]
total_y_lower_bound <- data$time
total_y_upper_bound <- ifelse(data$event == 1, data$time, Inf)

train_x <- as.matrix(train_data %>% select(-c(time, event)))
train_y_lower_bound <- train_data$time
train_y_upper_bound <- ifelse(train_data$event == 1, train_data$time, Inf)
dtrain <- xgb.DMatrix(
  data = train_x,
  label_lower_bound = train_y_lower_bound,
  label_upper_bound = train_y_upper_bound
)

val_x <- as.matrix(val_data %>% select(-c(time, event)))
val_y_lower_bound <- val_data$time
val_y_upper_bound <- ifelse(val_data$event == 1, val_data$time, Inf)
dval <- xgb.DMatrix(
  data = val_x,
  label_lower_bound = val_y_lower_bound,
  label_upper_bound = val_y_upper_bound
)

```

Hyperparameter Tuning

Since there is no

```

# Ref Barnwal, A., Cho ,Hyunsu, & and Hocking, T. (2022). Survival Regression
  ↪ with Accelerated Failure Time Model in XGBoost. Journal of Computational
  ↪ and Graphical Statistics, 31(4), 1292-1302.
  ↪ https://doi.org/10.1080/10618600.2022.2067548
param_set <- ParamSet$new(params = list(
  learning_rate = p_dbl(
    lower = log10(0.001),
    upper = log10(1.0),
    trafo = function(x) 10^x
  ),

```

```

max_depth = p_int(
  lower = 2,
  upper = 10
),
min_child_weight = p_dbl(
  lower = log10(0.001),
  upper = log10(100.0),
  trafo = function(x) 10^x
),
reg_alpha = p_dbl(
  lower = log10(0.001),
  upper = log10(100.0),
  trafo = function(x) 10^x
),
reg_lambda = p_dbl(
  lower = log10(0.001),
  upper = log10(100.0),
  trafo = function(x) 10^x
),
aft_loss_distribution_scale = p_dbl(
  lower = 0.5,
  upper = 2.0
)
))

```

```

tune_xgb <- function(params_trial) {
  model <- xgb.train(
    params = c(
      list(
        objective = "survival:aft",
        eval_metric = "aft-nloglik",
        aft_loss_distribution = "normal"
      ),
      params_trial
    ),
    data = dtrain,
    nrounds = 500,
    early_stopping_rounds = 10,
    watchlist = list(train = dtrain, val = dval),
    verbose = 0
  )
}

```

```

    pred <- predict(model, dval)
    pred <- -pred
    # return(list(score = min(model$evaluation_log$val_aft_nloglik)))
    return(list(score = concordance.index(pred, val_data$time,
      ↪ val_data$event)$c.index))
  }

```

```

tuning_results <- pmclapply(1:n_trials, function(i) {
  params_trial <- generate_design_random(param_set, n = 1)$data
  params_trial <- param_set$trafo(params_trial)
  score <- tune_xgb(params_trial)
  return(data.frame(trial = i, score = score$score, params = params_trial))
}, title = "Tuning XGBoost hyperparameters")

```

```

tuning_results <- bind_rows(tuning_results) # convert list of lists to a
  ↪ data frame
tuning_results_best <- tuning_results[which.max(tuning_results$score), ]

```

```

model_params <- list(
  learning_rate = tuning_results_best$params.learning_rate,
  max_depth = tuning_results_best$params.max_depth,
  min_child_weight = tuning_results_best$params.min_child_weight,
  reg_alpha = tuning_results_best$params.reg_alpha,
  reg_lambda = tuning_results_best$params.reg_lambda,
  aft_loss_distribution_scale =
    ↪ tuning_results_best$params.aft_loss_distribution_scale
)
print("Best hyperparameters:")

```

```
[1] "Best hyperparameters:"
```

```
print(model_params)
```

```

$learning_rate
[1] 0.0318649

```

```

$max_depth
[1] 4

```

```
$min_child_weight  
[1] 60.69201
```

```
$reg_alpha  
[1] 0.1984304
```

```
$reg_lambda  
[1] 1.034276
```

```
$aft_loss_distribution_scale  
[1] 1.599628
```

Variable Selection

```
# * As mentioned in the paper, we use AFT model instead of Cox model.  
xgb_var_select <- xgb.train(  
  params = c(  
    list(  
      objective = "survival:aft",  
      eval_metric = "aft-nloglik",  
      aft_loss_distribution = "normal"  
    ),  
    model_params  
  ),  
  data = dtrain,  
  nrounds = 100,  
  early_stopping_rounds = 10,  
  watchlist = list(train = dtrain, val = dval)  
)
```

```
[1] train-aft-nloglik:18.376400 val-aft-nloglik:18.387193  
Multiple eval metrics are present. Will use val_aft_nloglik for early stopping.  
Will train until val_aft_nloglik hasn't improved in 10 rounds.
```

```
[2] train-aft-nloglik:17.366598 val-aft-nloglik:17.377856  
[3] train-aft-nloglik:16.418685 val-aft-nloglik:16.430397  
[4] train-aft-nloglik:15.528796 val-aft-nloglik:15.540951  
[5] train-aft-nloglik:14.693309 val-aft-nloglik:14.705896  
[6] train-aft-nloglik:13.908830 val-aft-nloglik:13.921839
```


[7] train-aft-nloglik:13.172177 val-aft-nloglik:13.185598
 [8] train-aft-nloglik:12.480371 val-aft-nloglik:12.494194
 [9] train-aft-nloglik:11.830616 val-aft-nloglik:11.844833
 [10] train-aft-nloglik:11.220296 val-aft-nloglik:11.234897
 [11] train-aft-nloglik:10.646957 val-aft-nloglik:10.661934
 [12] train-aft-nloglik:10.108300 val-aft-nloglik:10.123645
 [13] train-aft-nloglik:9.602172 val-aft-nloglik:9.617876
 [14] train-aft-nloglik:9.126554 val-aft-nloglik:9.142609
 [15] train-aft-nloglik:8.679553 val-aft-nloglik:8.695951
 [16] train-aft-nloglik:8.259396 val-aft-nloglik:8.276132
 [17] train-aft-nloglik:7.864425 val-aft-nloglik:7.881489
 [18] train-aft-nloglik:7.493080 val-aft-nloglik:7.510467
 [19] train-aft-nloglik:7.143904 val-aft-nloglik:7.161606
 [20] train-aft-nloglik:6.815530 val-aft-nloglik:6.833541
 [21] train-aft-nloglik:6.506676 val-aft-nloglik:6.524991
 [22] train-aft-nloglik:6.216142 val-aft-nloglik:6.234754
 [23] train-aft-nloglik:5.942804 val-aft-nloglik:5.961707
 [24] train-aft-nloglik:5.685607 val-aft-nloglik:5.704795
 [25] train-aft-nloglik:5.443563 val-aft-nloglik:5.463031
 [26] train-aft-nloglik:5.215747 val-aft-nloglik:5.235489
 [27] train-aft-nloglik:5.001290 val-aft-nloglik:5.021301
 [28] train-aft-nloglik:4.799379 val-aft-nloglik:4.819654
 [29] train-aft-nloglik:4.609251 val-aft-nloglik:4.629786
 [30] train-aft-nloglik:4.430193 val-aft-nloglik:4.450982
 [31] train-aft-nloglik:4.261533 val-aft-nloglik:4.282572
 [32] train-aft-nloglik:4.102645 val-aft-nloglik:4.123929
 [33] train-aft-nloglik:3.952939 val-aft-nloglik:3.974483
 [34] train-aft-nloglik:3.811860 val-aft-nloglik:3.833659
 [35] train-aft-nloglik:3.678893 val-aft-nloglik:3.700957
 [36] train-aft-nloglik:3.553552 val-aft-nloglik:3.575858
 [37] train-aft-nloglik:3.435381 val-aft-nloglik:3.457927
 [38] train-aft-nloglik:3.323955 val-aft-nloglik:3.346748
 [39] train-aft-nloglik:3.218872 val-aft-nloglik:3.241890
 [40] train-aft-nloglik:3.119756 val-aft-nloglik:3.142991
 [41] train-aft-nloglik:3.026256 val-aft-nloglik:3.049711
 [42] train-aft-nloglik:2.938040 val-aft-nloglik:2.961694
 [43] train-aft-nloglik:2.854800 val-aft-nloglik:2.878662
 [44] train-aft-nloglik:2.776240 val-aft-nloglik:2.800306
 [45] train-aft-nloglik:2.702094 val-aft-nloglik:2.726356
 [46] train-aft-nloglik:2.632096 val-aft-nloglik:2.656549
 [47] train-aft-nloglik:2.566012 val-aft-nloglik:2.590666
 [48] train-aft-nloglik:2.503611 val-aft-nloglik:2.528439
 [49] train-aft-nloglik:2.444681 val-aft-nloglik:2.469694

[50]	train-aft-nloglik:2.389022	val-aft-nloglik:2.414215
[51]	train-aft-nloglik:2.336443	val-aft-nloglik:2.361838
[52]	train-aft-nloglik:2.286777	val-aft-nloglik:2.312346
[53]	train-aft-nloglik:2.239851	val-aft-nloglik:2.265617
[54]	train-aft-nloglik:2.195513	val-aft-nloglik:2.221435
[55]	train-aft-nloglik:2.153612	val-aft-nloglik:2.179687
[56]	train-aft-nloglik:2.114006	val-aft-nloglik:2.140275
[57]	train-aft-nloglik:2.076577	val-aft-nloglik:2.102995
[58]	train-aft-nloglik:2.041192	val-aft-nloglik:2.067799
[59]	train-aft-nloglik:2.007749	val-aft-nloglik:2.034494
[60]	train-aft-nloglik:1.976125	val-aft-nloglik:2.003030
[61]	train-aft-nloglik:1.946232	val-aft-nloglik:1.973286
[62]	train-aft-nloglik:1.917969	val-aft-nloglik:1.945183
[63]	train-aft-nloglik:1.891243	val-aft-nloglik:1.918623
[64]	train-aft-nloglik:1.865968	val-aft-nloglik:1.893499
[65]	train-aft-nloglik:1.842071	val-aft-nloglik:1.869757
[66]	train-aft-nloglik:1.819436	val-aft-nloglik:1.847321
[67]	train-aft-nloglik:1.798043	val-aft-nloglik:1.826108
[68]	train-aft-nloglik:1.777821	val-aft-nloglik:1.806045
[69]	train-aft-nloglik:1.758681	val-aft-nloglik:1.787085
[70]	train-aft-nloglik:1.740574	val-aft-nloglik:1.769162
[71]	train-aft-nloglik:1.723453	val-aft-nloglik:1.752222
[72]	train-aft-nloglik:1.707241	val-aft-nloglik:1.736165
[73]	train-aft-nloglik:1.691894	val-aft-nloglik:1.720982
[74]	train-aft-nloglik:1.677392	val-aft-nloglik:1.706668
[75]	train-aft-nloglik:1.663681	val-aft-nloglik:1.693105
[76]	train-aft-nloglik:1.650688	val-aft-nloglik:1.680282
[77]	train-aft-nloglik:1.638422	val-aft-nloglik:1.668196
[78]	train-aft-nloglik:1.626769	val-aft-nloglik:1.656729
[79]	train-aft-nloglik:1.615779	val-aft-nloglik:1.645904
[80]	train-aft-nloglik:1.605359	val-aft-nloglik:1.635650
[81]	train-aft-nloglik:1.595522	val-aft-nloglik:1.625960
[82]	train-aft-nloglik:1.586231	val-aft-nloglik:1.616801
[83]	train-aft-nloglik:1.577422	val-aft-nloglik:1.608160
[84]	train-aft-nloglik:1.569112	val-aft-nloglik:1.599996
[85]	train-aft-nloglik:1.561240	val-aft-nloglik:1.592290
[86]	train-aft-nloglik:1.553766	val-aft-nloglik:1.584994
[87]	train-aft-nloglik:1.546720	val-aft-nloglik:1.578089
[88]	train-aft-nloglik:1.540035	val-aft-nloglik:1.571606
[89]	train-aft-nloglik:1.533708	val-aft-nloglik:1.565423
[90]	train-aft-nloglik:1.527734	val-aft-nloglik:1.559626
[91]	train-aft-nloglik:1.522110	val-aft-nloglik:1.554118
[92]	train-aft-nloglik:1.516782	val-aft-nloglik:1.548920

```

[93]   train-aft-nloglik:1.511705   val-aft-nloglik:1.544007
[94]   train-aft-nloglik:1.506930   val-aft-nloglik:1.539366
[95]   train-aft-nloglik:1.502441   val-aft-nloglik:1.535013
[96]   train-aft-nloglik:1.498136   val-aft-nloglik:1.530874
[97]   train-aft-nloglik:1.494090   val-aft-nloglik:1.526993
[98]   train-aft-nloglik:1.490241   val-aft-nloglik:1.523320
[99]   train-aft-nloglik:1.486621   val-aft-nloglik:1.519871
[100]  train-aft-nloglik:1.483178   val-aft-nloglik:1.516574

```

```

# Sort descendingly using gain
xgb_importance <- xgb.importance(model = xgb_var_select)
# Other attributes: Gain, Cover, Frequency
vars_ranked <- xgb_importance$Feature

```

Cross Validation to Select the Best Number of Features

```

# * xgb.cv is not available for AFT model.
set.seed(1234)
folds <- createFolds(data$event, k = n_folds)

cv_errors <- pmclapply(seq(1, length(vars_ranked), by = 1),
  ↪ function(num_vars) {
    selected_vars <- vars_ranked[1:num_vars]
    print(paste0("Selecting ", num_vars, " variables"))
    fold_errors <- sapply(folds, function(fold_idx) {
      # * We take all training data and validation data and then split them
      ↪ into folds.
      train_x_fold <- as.matrix(total_x[-fold_idx, selected_vars, drop =
↪ FALSE])
      train_y_lower_fold <- total_y_lower_bound[-fold_idx]
      train_y_upper_fold <- total_y_upper_bound[-fold_idx]

      val_x_fold <- as.matrix(total_x[fold_idx, selected_vars, drop =
↪ FALSE])
      val_y_lower_fold <- total_y_lower_bound[fold_idx]
      val_y_upper_fold <- total_y_upper_bound[fold_idx]
      val_y_fold <- total_y[fold_idx, ] # for C-index calculation

      dtrain_fold <- xgb.DMatrix(
        data = train_x_fold,

```

```

        label_lower_bound = train_y_lower_fold,
        # label_upper_bound = train_y_upper_fold
        label_upper_bound = train_y_lower_fold
    )

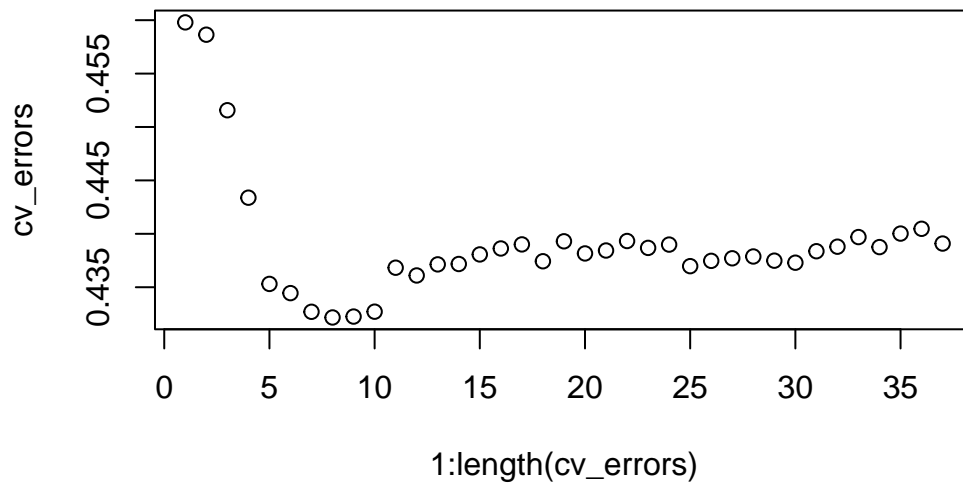
    dval_fold <- xgb.DMatrix(
        data = val_x_fold,
        label_lower_bound = val_y_lower_fold,
        # label_upper_bound = val_y_upper_fold
        label_upper_bound = val_y_lower_fold
    )

    model <- xgb.train(
        params = c(
            list(
                objective = "survival:aft",
                eval_metric = "aft-nloglik",
                aft_loss_distribution = "normal"
            ),
            model_params
        ),
        data = dtrain_fold,
        nrounds = 500,
        early_stopping_rounds = 10,
        watchlist = list(train = dtrain_fold, val = dval_fold),
        verbose = 0
    )
    # * It outputs the estimated survival time. We need to convert it to
    ↪ risk.
    pred <- predict(model, dval_fold)
    pred <- -pred

    # Use C-index to measure the performance of the model
    1 - concordance.index(pred, val_y_fold$time,
        ↪ val_y_fold$event)$c.index
    })
    print(mean(fold_errors))
    mean(fold_errors)
}, title = "Cross Validation to Select the Best Number of Features")

```

```
cv_errors <- as.numeric(cv_errors)
plot(1:length(cv_errors), cv_errors)
```



```
best_num_vars <- which.min(cv_errors)
vars_selected <- vars_ranked[1:best_num_vars]
```

```
print(paste0("The best number of features to retain is ", best_num_vars))
```

```
[1] "The best number of features to retain is 8"
```

Model Fitting

```
dtrain_selected <- xgb.DMatrix(
  data = train_x[, vars_selected],
  label_lower_bound = train_y_lower_bound,
  label_upper_bound = train_y_upper_bound
)
dval_selected <- xgb.DMatrix(
```

```

    data = val_x[, vars_selected],
    label_lower_bound = val_y_lower_bound,
    label_upper_bound = val_y_upper_bound
)
xgb_model <- xgb.train(
  params = c(
    list(
      objective = "survival:aft",
      eval_metric = "aft-nloglik",
      aft_loss_distribution = "normal"
    ),
    model_params
  ),
  data = dtrain_selected,
  nrounds = 100,
  early_stopping_rounds = 10,
  watchlist = list(train = dtrain_selected, val = dval_selected)
)

```

[1] train-aft-nloglik:18.376400 val-aft-nloglik:18.387193
Multiple eval metrics are present. Will use val_aft_nloglik for early stopping.
Will train until val_aft_nloglik hasn't improved in 10 rounds.

```

[2] train-aft-nloglik:17.366598 val-aft-nloglik:17.377856
[3] train-aft-nloglik:16.418685 val-aft-nloglik:16.430397
[4] train-aft-nloglik:15.528796 val-aft-nloglik:15.540951
[5] train-aft-nloglik:14.693309 val-aft-nloglik:14.705896
[6] train-aft-nloglik:13.908830 val-aft-nloglik:13.921839
[7] train-aft-nloglik:13.172177 val-aft-nloglik:13.185598
[8] train-aft-nloglik:12.480371 val-aft-nloglik:12.494194
[9] train-aft-nloglik:11.830616 val-aft-nloglik:11.844833
[10] train-aft-nloglik:11.220296 val-aft-nloglik:11.234897
[11] train-aft-nloglik:10.646957 val-aft-nloglik:10.661934
[12] train-aft-nloglik:10.108300 val-aft-nloglik:10.123645
[13] train-aft-nloglik:9.602172 val-aft-nloglik:9.617876
[14] train-aft-nloglik:9.126554 val-aft-nloglik:9.142609
[15] train-aft-nloglik:8.679553 val-aft-nloglik:8.695951
[16] train-aft-nloglik:8.259396 val-aft-nloglik:8.276132
[17] train-aft-nloglik:7.864425 val-aft-nloglik:7.881489
[18] train-aft-nloglik:7.493080 val-aft-nloglik:7.510467
[19] train-aft-nloglik:7.143904 val-aft-nloglik:7.161606

```

[20]	train-aft-nloglik:6.815530	val-aft-nloglik:6.833541
[21]	train-aft-nloglik:6.506676	val-aft-nloglik:6.524991
[22]	train-aft-nloglik:6.216142	val-aft-nloglik:6.234754
[23]	train-aft-nloglik:5.942804	val-aft-nloglik:5.961707
[24]	train-aft-nloglik:5.685607	val-aft-nloglik:5.704795
[25]	train-aft-nloglik:5.443563	val-aft-nloglik:5.463031
[26]	train-aft-nloglik:5.215747	val-aft-nloglik:5.235489
[27]	train-aft-nloglik:5.001290	val-aft-nloglik:5.021301
[28]	train-aft-nloglik:4.799379	val-aft-nloglik:4.819654
[29]	train-aft-nloglik:4.609251	val-aft-nloglik:4.629786
[30]	train-aft-nloglik:4.430193	val-aft-nloglik:4.450982
[31]	train-aft-nloglik:4.261533	val-aft-nloglik:4.282572
[32]	train-aft-nloglik:4.102645	val-aft-nloglik:4.123929
[33]	train-aft-nloglik:3.952939	val-aft-nloglik:3.974483
[34]	train-aft-nloglik:3.811860	val-aft-nloglik:3.833659
[35]	train-aft-nloglik:3.678893	val-aft-nloglik:3.700957
[36]	train-aft-nloglik:3.553552	val-aft-nloglik:3.575858
[37]	train-aft-nloglik:3.435381	val-aft-nloglik:3.457927
[38]	train-aft-nloglik:3.323955	val-aft-nloglik:3.346748
[39]	train-aft-nloglik:3.218872	val-aft-nloglik:3.241890
[40]	train-aft-nloglik:3.119756	val-aft-nloglik:3.142991
[41]	train-aft-nloglik:3.026256	val-aft-nloglik:3.049711
[42]	train-aft-nloglik:2.938040	val-aft-nloglik:2.961694
[43]	train-aft-nloglik:2.854800	val-aft-nloglik:2.878662
[44]	train-aft-nloglik:2.776240	val-aft-nloglik:2.800306
[45]	train-aft-nloglik:2.702093	val-aft-nloglik:2.726366
[46]	train-aft-nloglik:2.632094	val-aft-nloglik:2.656558
[47]	train-aft-nloglik:2.566009	val-aft-nloglik:2.590656
[48]	train-aft-nloglik:2.503608	val-aft-nloglik:2.528429
[49]	train-aft-nloglik:2.444678	val-aft-nloglik:2.469685
[50]	train-aft-nloglik:2.389019	val-aft-nloglik:2.414205
[51]	train-aft-nloglik:2.336445	val-aft-nloglik:2.361809
[52]	train-aft-nloglik:2.286775	val-aft-nloglik:2.312320
[53]	train-aft-nloglik:2.239851	val-aft-nloglik:2.265556
[54]	train-aft-nloglik:2.195509	val-aft-nloglik:2.221389
[55]	train-aft-nloglik:2.153611	val-aft-nloglik:2.179660
[56]	train-aft-nloglik:2.114010	val-aft-nloglik:2.140234
[57]	train-aft-nloglik:2.076585	val-aft-nloglik:2.102959
[58]	train-aft-nloglik:2.041208	val-aft-nloglik:2.067759
[59]	train-aft-nloglik:2.007768	val-aft-nloglik:2.034461
[60]	train-aft-nloglik:1.976149	val-aft-nloglik:2.003001
[61]	train-aft-nloglik:1.946258	val-aft-nloglik:1.973258
[62]	train-aft-nloglik:1.917998	val-aft-nloglik:1.945146

[63]	train-aft-nloglik:1.891268	val-aft-nloglik:1.918583
[64]	train-aft-nloglik:1.865997	val-aft-nloglik:1.893456
[65]	train-aft-nloglik:1.842092	val-aft-nloglik:1.869723
[66]	train-aft-nloglik:1.819488	val-aft-nloglik:1.847289
[67]	train-aft-nloglik:1.798115	val-aft-nloglik:1.826088
[68]	train-aft-nloglik:1.777880	val-aft-nloglik:1.806055
[69]	train-aft-nloglik:1.758755	val-aft-nloglik:1.787095
[70]	train-aft-nloglik:1.740658	val-aft-nloglik:1.769156
[71]	train-aft-nloglik:1.723549	val-aft-nloglik:1.752197
[72]	train-aft-nloglik:1.707336	val-aft-nloglik:1.736145
[73]	train-aft-nloglik:1.692009	val-aft-nloglik:1.720998
[74]	train-aft-nloglik:1.677491	val-aft-nloglik:1.706658
[75]	train-aft-nloglik:1.663789	val-aft-nloglik:1.693116
[76]	train-aft-nloglik:1.650834	val-aft-nloglik:1.680295
[77]	train-aft-nloglik:1.638569	val-aft-nloglik:1.668184
[78]	train-aft-nloglik:1.626980	val-aft-nloglik:1.656726
[79]	train-aft-nloglik:1.616013	val-aft-nloglik:1.645902
[80]	train-aft-nloglik:1.605619	val-aft-nloglik:1.635675
[81]	train-aft-nloglik:1.595811	val-aft-nloglik:1.626007
[82]	train-aft-nloglik:1.586490	val-aft-nloglik:1.616846
[83]	train-aft-nloglik:1.577670	val-aft-nloglik:1.608201
[84]	train-aft-nloglik:1.569361	val-aft-nloglik:1.600035
[85]	train-aft-nloglik:1.561494	val-aft-nloglik:1.592321
[86]	train-aft-nloglik:1.554054	val-aft-nloglik:1.585026
[87]	train-aft-nloglik:1.547007	val-aft-nloglik:1.578126
[88]	train-aft-nloglik:1.540372	val-aft-nloglik:1.571616
[89]	train-aft-nloglik:1.534049	val-aft-nloglik:1.565456
[90]	train-aft-nloglik:1.528098	val-aft-nloglik:1.559662
[91]	train-aft-nloglik:1.522466	val-aft-nloglik:1.554190
[92]	train-aft-nloglik:1.517136	val-aft-nloglik:1.549026
[93]	train-aft-nloglik:1.512087	val-aft-nloglik:1.544120
[94]	train-aft-nloglik:1.507345	val-aft-nloglik:1.539506
[95]	train-aft-nloglik:1.502849	val-aft-nloglik:1.535142
[96]	train-aft-nloglik:1.498597	val-aft-nloglik:1.531030
[97]	train-aft-nloglik:1.494588	val-aft-nloglik:1.527128
[98]	train-aft-nloglik:1.490773	val-aft-nloglik:1.523450
[99]	train-aft-nloglik:1.487165	val-aft-nloglik:1.519987
[100]	train-aft-nloglik:1.483756	val-aft-nloglik:1.516732

```
dtrain_full <- xgb.DMatrix(
  data = train_x,
  label_lower_bound = train_y_lower_bound,
```



```

    label_upper_bound = train_y_upper_bound
)
# We also fit the full model
xgb_model_full <- xgb.train(
  params = c(
    list(
      objective = "survival:aft",
      eval_metric = "aft-nloglik",
      aft_loss_distribution = "normal"
    ),
    model_params
  ),
  data = dtrain_full,
  nrounds = 100,
  early_stopping_rounds = 10,
  watchlist = list(train = dtrain_full, val = dval)
)

```

```

[1] train-aft-nloglik:18.376400 val-aft-nloglik:18.387193
Multiple eval metrics are present. Will use val_aft_nloglik for early stopping.
Will train until val_aft_nloglik hasn't improved in 10 rounds.

```

```

[2] train-aft-nloglik:17.366598 val-aft-nloglik:17.377856
[3] train-aft-nloglik:16.418685 val-aft-nloglik:16.430397
[4] train-aft-nloglik:15.528796 val-aft-nloglik:15.540951
[5] train-aft-nloglik:14.693309 val-aft-nloglik:14.705896
[6] train-aft-nloglik:13.908830 val-aft-nloglik:13.921839
[7] train-aft-nloglik:13.172177 val-aft-nloglik:13.185598
[8] train-aft-nloglik:12.480371 val-aft-nloglik:12.494194
[9] train-aft-nloglik:11.830616 val-aft-nloglik:11.844833
[10] train-aft-nloglik:11.220296 val-aft-nloglik:11.234897
[11] train-aft-nloglik:10.646957 val-aft-nloglik:10.661934
[12] train-aft-nloglik:10.108300 val-aft-nloglik:10.123645
[13] train-aft-nloglik:9.602172 val-aft-nloglik:9.617876
[14] train-aft-nloglik:9.126554 val-aft-nloglik:9.142609
[15] train-aft-nloglik:8.679553 val-aft-nloglik:8.695951
[16] train-aft-nloglik:8.259396 val-aft-nloglik:8.276132
[17] train-aft-nloglik:7.864425 val-aft-nloglik:7.881489
[18] train-aft-nloglik:7.493080 val-aft-nloglik:7.510467
[19] train-aft-nloglik:7.143904 val-aft-nloglik:7.161606
[20] train-aft-nloglik:6.815530 val-aft-nloglik:6.833541

```

[21]	train-aft-nloglik:6.506676	val-aft-nloglik:6.524991
[22]	train-aft-nloglik:6.216142	val-aft-nloglik:6.234754
[23]	train-aft-nloglik:5.942804	val-aft-nloglik:5.961707
[24]	train-aft-nloglik:5.685607	val-aft-nloglik:5.704795
[25]	train-aft-nloglik:5.443563	val-aft-nloglik:5.463031
[26]	train-aft-nloglik:5.215747	val-aft-nloglik:5.235489
[27]	train-aft-nloglik:5.001290	val-aft-nloglik:5.021301
[28]	train-aft-nloglik:4.799379	val-aft-nloglik:4.819654
[29]	train-aft-nloglik:4.609251	val-aft-nloglik:4.629786
[30]	train-aft-nloglik:4.430193	val-aft-nloglik:4.450982
[31]	train-aft-nloglik:4.261533	val-aft-nloglik:4.282572
[32]	train-aft-nloglik:4.102645	val-aft-nloglik:4.123929
[33]	train-aft-nloglik:3.952939	val-aft-nloglik:3.974483
[34]	train-aft-nloglik:3.811860	val-aft-nloglik:3.833659
[35]	train-aft-nloglik:3.678893	val-aft-nloglik:3.700957
[36]	train-aft-nloglik:3.553552	val-aft-nloglik:3.575858
[37]	train-aft-nloglik:3.435381	val-aft-nloglik:3.457927
[38]	train-aft-nloglik:3.323955	val-aft-nloglik:3.346748
[39]	train-aft-nloglik:3.218872	val-aft-nloglik:3.241890
[40]	train-aft-nloglik:3.119756	val-aft-nloglik:3.142991
[41]	train-aft-nloglik:3.026256	val-aft-nloglik:3.049711
[42]	train-aft-nloglik:2.938040	val-aft-nloglik:2.961694
[43]	train-aft-nloglik:2.854800	val-aft-nloglik:2.878662
[44]	train-aft-nloglik:2.776240	val-aft-nloglik:2.800306
[45]	train-aft-nloglik:2.702094	val-aft-nloglik:2.726356
[46]	train-aft-nloglik:2.632096	val-aft-nloglik:2.656549
[47]	train-aft-nloglik:2.566012	val-aft-nloglik:2.590666
[48]	train-aft-nloglik:2.503611	val-aft-nloglik:2.528439
[49]	train-aft-nloglik:2.444681	val-aft-nloglik:2.469694
[50]	train-aft-nloglik:2.389022	val-aft-nloglik:2.414215
[51]	train-aft-nloglik:2.336443	val-aft-nloglik:2.361838
[52]	train-aft-nloglik:2.286777	val-aft-nloglik:2.312346
[53]	train-aft-nloglik:2.239851	val-aft-nloglik:2.265617
[54]	train-aft-nloglik:2.195513	val-aft-nloglik:2.221435
[55]	train-aft-nloglik:2.153612	val-aft-nloglik:2.179687
[56]	train-aft-nloglik:2.114006	val-aft-nloglik:2.140275
[57]	train-aft-nloglik:2.076577	val-aft-nloglik:2.102995
[58]	train-aft-nloglik:2.041192	val-aft-nloglik:2.067799
[59]	train-aft-nloglik:2.007749	val-aft-nloglik:2.034494
[60]	train-aft-nloglik:1.976125	val-aft-nloglik:2.003030
[61]	train-aft-nloglik:1.946232	val-aft-nloglik:1.973286
[62]	train-aft-nloglik:1.917969	val-aft-nloglik:1.945183
[63]	train-aft-nloglik:1.891243	val-aft-nloglik:1.918623

[64]	train-aft-nloglik:1.865968	val-aft-nloglik:1.893499
[65]	train-aft-nloglik:1.842071	val-aft-nloglik:1.869757
[66]	train-aft-nloglik:1.819436	val-aft-nloglik:1.847321
[67]	train-aft-nloglik:1.798043	val-aft-nloglik:1.826108
[68]	train-aft-nloglik:1.777821	val-aft-nloglik:1.806045
[69]	train-aft-nloglik:1.758681	val-aft-nloglik:1.787085
[70]	train-aft-nloglik:1.740574	val-aft-nloglik:1.769162
[71]	train-aft-nloglik:1.723453	val-aft-nloglik:1.752222
[72]	train-aft-nloglik:1.707241	val-aft-nloglik:1.736165
[73]	train-aft-nloglik:1.691894	val-aft-nloglik:1.720982
[74]	train-aft-nloglik:1.677392	val-aft-nloglik:1.706668
[75]	train-aft-nloglik:1.663681	val-aft-nloglik:1.693105
[76]	train-aft-nloglik:1.650688	val-aft-nloglik:1.680282
[77]	train-aft-nloglik:1.638422	val-aft-nloglik:1.668196
[78]	train-aft-nloglik:1.626769	val-aft-nloglik:1.656729
[79]	train-aft-nloglik:1.615779	val-aft-nloglik:1.645904
[80]	train-aft-nloglik:1.605359	val-aft-nloglik:1.635650
[81]	train-aft-nloglik:1.595522	val-aft-nloglik:1.625960
[82]	train-aft-nloglik:1.586231	val-aft-nloglik:1.616801
[83]	train-aft-nloglik:1.577422	val-aft-nloglik:1.608160
[84]	train-aft-nloglik:1.569112	val-aft-nloglik:1.599996
[85]	train-aft-nloglik:1.561240	val-aft-nloglik:1.592290
[86]	train-aft-nloglik:1.553766	val-aft-nloglik:1.584994
[87]	train-aft-nloglik:1.546720	val-aft-nloglik:1.578089
[88]	train-aft-nloglik:1.540035	val-aft-nloglik:1.571606
[89]	train-aft-nloglik:1.533708	val-aft-nloglik:1.565423
[90]	train-aft-nloglik:1.527734	val-aft-nloglik:1.559626
[91]	train-aft-nloglik:1.522110	val-aft-nloglik:1.554118
[92]	train-aft-nloglik:1.516782	val-aft-nloglik:1.548920
[93]	train-aft-nloglik:1.511705	val-aft-nloglik:1.544007
[94]	train-aft-nloglik:1.506930	val-aft-nloglik:1.539366
[95]	train-aft-nloglik:1.502441	val-aft-nloglik:1.535013
[96]	train-aft-nloglik:1.498136	val-aft-nloglik:1.530874
[97]	train-aft-nloglik:1.494090	val-aft-nloglik:1.526993
[98]	train-aft-nloglik:1.490241	val-aft-nloglik:1.523320
[99]	train-aft-nloglik:1.486621	val-aft-nloglik:1.519871
[100]	train-aft-nloglik:1.483178	val-aft-nloglik:1.516574

SHAP?