Build Survival Model: Cox Proportional Hazards Model

Mingcheng Hu

Table of contents

```r
library(tidyverse)
library(survival)
library(forestplot)
library(glmnet)
library(ggfortify)
library(kableExtra) # include knitr automatically

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

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

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

```
# 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: full"
```

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

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

Load Data

```
if (include_statin == "yes") {
    data_train <-
 ↪  read.csv(paste0("/work/users/y/u/yuukias/BIOS-Material/BIOS992/data/train_data_",
 ↪  impute_type_str, "_statin.csv"),
        header = TRUE
    )
} else {
    data_train <-
 ↪  read.csv(paste0("/work/users/y/u/yuukias/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    89
```

```
colnames(data)
```

```
 [1] "event"                         "time"
 [3] "age"                           "sex"
 [5] "ethnicity"                     "BMI"
 [7] "smoking"                       "diabetes"
 [9] "systolic_bp"                   "hypertension_treatment"
[11] "total_chol"                    "hdl_chol"
[13] "education"                     "activity"
[15] "max_workload"                  "max_heart_rate"
[17] "HRV_MeanNN"                    "HRV_SDNN"
[19] "HRV_RMSSD"                     "HRV_SDSD"
[21] "HRV_CVNN"                      "HRV_CVSD"
[23] "HRV_MedianNN"                  "HRV_MadNN"
[25] "HRV_MCVNN"                     "HRV_IQRNN"
[27] "HRV_SDRMSSD"                   "HRV_Prc20NN"
[29] "HRV_Prc80NN"                   "HRV_pNN50"
[31] "HRV_pNN20"                     "HRV_MinNN"
[33] "HRV_MaxNN"                     "HRV_HTI"
[35] "HRV_TINN"                      "HRV_LF"
[37] "HRV_HF"                        "HRV_VHF"
[39] "HRV_TP"                        "HRV_LFHF"
[41] "HRV_LFn"                       "HRV_HFn"
[43] "HRV_LnHF"                      "HRV_SD1"
[45] "HRV_SD2"                       "HRV_SD1SD2"
[47] "HRV_S"                         "HRV_CSI"
```

```
[49] "HRV_CVI"                      "HRV_CSI_Modified"
[51] "HRV_PIP"                      "HRV_IALS"
[53] "HRV_PSS"                      "HRV_PAS"
[55] "HRV_GI"                       "HRV_SI"
[57] "HRV_AI"                       "HRV_PI"
[59] "HRV_C1d"                      "HRV_C1a"
[61] "HRV_SD1d"                     "HRV_SD1a"
[63] "HRV_C2d"                      "HRV_C2a"
[65] "HRV_SD2d"                     "HRV_SD2a"
[67] "HRV_Cd"                       "HRV_Ca"
[69] "HRV_SDNNd"                    "HRV_SDNNa"
[71] "HRV_ApEn"                     "HRV_ShanEn"
[73] "HRV_FuzzyEn"                  "HRV_MSEn"
[75] "HRV_CMSEn"                    "HRV_RCMSEn"
[77] "HRV_CD"                       "HRV_HFD"
[79] "HRV_KFD"                      "HRV_LZC"
[81] "HRV_DFA_alpha1"               "HRV_MFDFA_alpha1_Width"
[83] "HRV_MFDFA_alpha1_Peak"        "HRV_MFDFA_alpha1_Mean"
[85] "HRV_MFDFA_alpha1_Max"         "HRV_MFDFA_alpha1_Delta"
[87] "HRV_MFDFA_alpha1_Asymmetry"   "HRV_MFDFA_alpha1_Fluctuation"
[89] "HRV_MFDFA_alpha1_Increment"
```

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

```r
# * There are some imputed ethnicity set to "e". We will exclude them at this
 ↪  time.
data <- data %>%
    filter(ethnicity != "e")

# * We also need to manually relevel the categorical variables
data <- data %>%
    mutate(
        # Set "Never" (0) as baseline for smoking
        smoking = factor(smoking,
            levels = c("0", "1", "2", "-3"),
            labels = c("Never", "Previous", "Current", "Prefer not to
             ↪  answer")
        ),

        # Set "No" (0) as baseline for diabetes
        diabetes = factor(diabetes,
            levels = c("0", "1", "-1", "-3"),
            labels = c("No", "Yes", "Do not know", "Prefer not to answer")
        ),

        # Ensure other categorical variables are properly factored
        ethnicity = factor(ethnicity,
            levels = c("1", "2", "3", "4", "5", "6"),
            labels = c("White", "Mixed", "Asian/Asian British", "Black/Black
             ↪  British", "Chinese", "Other")
        ),
        education = factor(education,
            levels = c("1", "2", "3", "4", "5", "6", "-7", "-3"),
            labels = c(
                "College/University degree", "A levels/AS levels",
                "O levels/GCSEs", "CSEs", "NVQ/HND/HNC",
                "Other professional", "None of the above",
                "Prefer not to answer"
            )
        ),
        activity = factor(activity,
            levels = c("0", "1", "2"),
            labels = c("Low", "Moderate", "High")
        ),
        sex = factor(sex,
            levels = c("0", "1"),
            labels = c("Female", "Male")
        ),
        hypertension_treatment = factor(hypertension_treatment,
            levels = c("0", "1"),
            labels = c("No", "Yes")
        )
    )
```

```
# * 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.

```
data_complete <- na.omit(data)
```

Univariate Cox Proportional Hazards Model

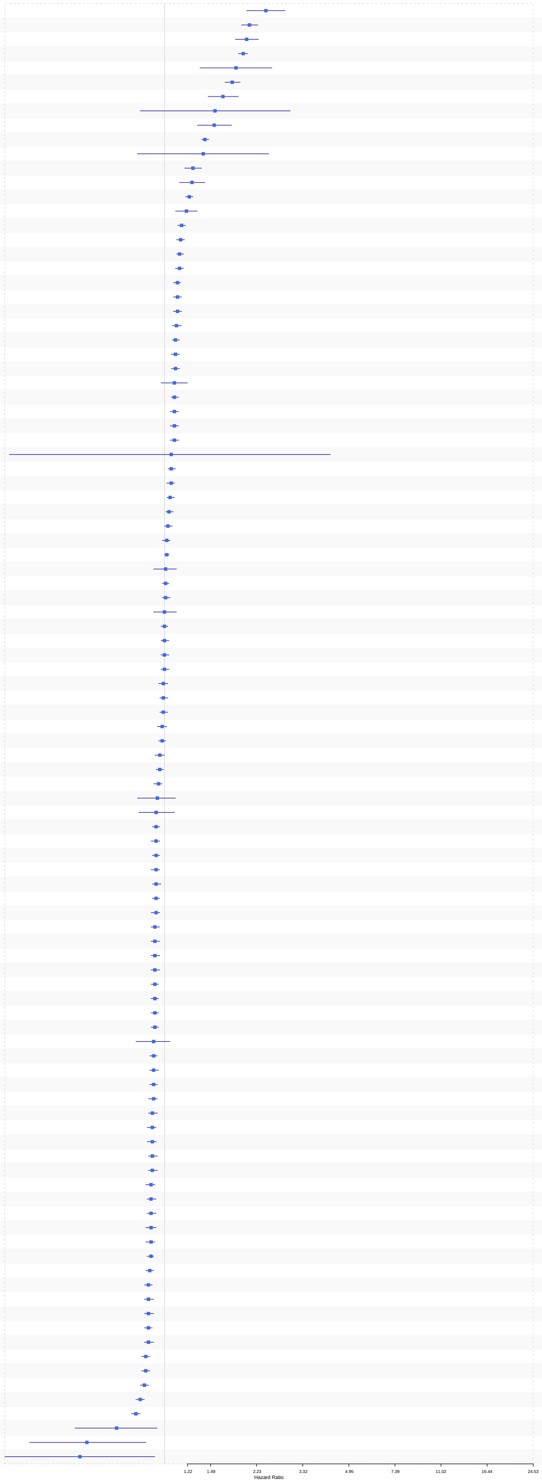
```
if (!("time" %in% colnames(data) && "event" %in% colnames(data))) {
    stop("time and event columns are required")
}
predictors <- colnames(data)[!colnames(data) %in% c("time", "event")]

results_univariate <- map_dfr(predictors, function(predictor) {
    formula <- as.formula(paste("Surv(time, event) ~", predictor))
    # cox_model_single <- coxph(Surv(time, event) ~ get(predictor), data =
     ↪  data)  # equivalent way
    cox_model_single <- coxph(formula, data = data)

    coef <- coef(cox_model_single) # log hazard ratio
    se <- sqrt(diag(vcov(cox_model_single)))

    hr <- exp(coef)
    lower_ci <- exp(coef - 1.96 * se)
    upper_ci <- exp(coef + 1.96 * se)
    p_value <-
 ↪  summary(cox_model_single)$coefficients[1:dim(summary(cox_model_single)$coefficients)[1],
 ↪  5]

    if (determine_type(predictor) == "categorical") {
        # exclude -1, -3, -7 in names
        return(
            data.frame(
                predictor = names(coef),
                hr = hr,
                lower_ci = lower_ci,
                upper_ci = upper_ci,
                p_value = p_value
            )
        )
    } else {
        return(
            data.frame(
                predictor = predictor,
                hr = hr,
                lower_ci = lower_ci,
                upper_ci = upper_ci,
                p_value = p_value
            )
        )
    }
})
results_univariate$hr <- round(results_univariate$hr, 2)
results_univariate$lower_ci <- round(results_univariate$lower_ci, 2)
results_univariate$upper_ci <- round(results_univariate$upper_ci, 2)
results_univariate$ci <- paste0("(", results_univariate$lower_ci, ",",
 ↪  results_univariate$upper_ci, ")")
results_univariate$p_value <- round(results_univariate$p_value, 3)
results_univariate <- results_univariate %>% arrange(desc(hr)) # sort
 ↪  descendingly by HR
```

```
# Create forest plot
results_univariate %>%
    forestplot(
        labeltext = c(predictor, hr, ci, p_value),
        mean = hr,
        lower = lower_ci,
        upper = upper_ci,
        xlab = "Hazard Ratio",
        title = "Univariate Cox Models",
        xlog = TRUE, # * Make sure the CI are not symmetric and need to be
         ↪  transformed
        boxsize = 0.2,
        xticks = c(0.2, 0.4, 0.8, 1.2, 1.6, 2.0, 2.4, 2.8, 3.2),
        clip = c(0.2, 3.2),
        zero = 1
```

```r
    ) %>%
    fp_set_style(
        box = "royalblue",
        line = "darkblue",
        summary = "royalblue"
    ) %>%
    fp_add_header(
        predictor = c("Predictor", ""),
        hr = c("Hazard Ratio", "per SD increase"),
        ci = c("95% CI", ""),
        p_value = c("p-value", "")
    ) %>%
    fp_decorate_graph(
        box = gpar(lty = 2, col = "lightgray"),
        graph.pos = 4
    ) %>% # change the position of forest plot
    fp_set_zebra_style("#f9f9f9")
```

Univariate Cox Models

Predictor	Hazard Ratio per SD increase	95% CI	p−value
diabetesYes	2.41	(2.04,2.85)	0
hypertension_treatmentYes	2.09	(1.95,2.25)	0
educationNone of the above	2.04	(1.85,2.26)	0
age	1.98	(1.9,2.06)	0
educationPrefer not to answer	1.86	(1.36,2.54)	0
sexMale	1.8	(1.69,1.93)	0
educationNVQ/HND/HNC	1.66	(1.46,1.9)	0
diabetesDo not know	1.55	(0.81,2.98)	0.188
educationOther professional	1.54	(1.33,1.79)	0
systolic_bp	1.42	(1.38,1.47)	0
smokingPrefer not to answer	1.4	(0.79,2.47)	0.248
smokingPrevious	1.28	(1.19,1.38)	0
smokingCurrent	1.27	(1.14,1.42)	0
BMI	1.24	(1.2,1.28)	0
educationO levels/GCSEs	1.21	(1.1,1.33)	0
HRV_PIP	1.16	(1.12,1.2)	0
HRV_IALS	1.15	(1.11,1.19)	0
HRV_SD1SD2	1.14	(1.11,1.18)	0
HRV_HFD	1.14	(1.1,1.18)	0
HRV_PAS	1.12	(1.08,1.15)	0
HRV_GI	1.12	(1.08,1.16)	0
HRV_SI	1.12	(1.08,1.16)	0
HRV_AI	1.11	(1.07,1.16)	0
HRV_PSS	1.1	(1.07,1.14)	0
HRV_ApEn	1.1	(1.06,1.14)	0
HRV_CMSEn	1.1	(1.06,1.14)	0
educationA levels/AS levels	1.09	(0.97,1.22)	0.144
HRV_VHF	1.09	(1.06,1.13)	0
HRV_C2d	1.09	(1.05,1.13)	0
HRV_Cd	1.09	(1.05,1.13)	0
HRV_RCMSEn	1.09	(1.05,1.13)	0
diabetesPrefer not to answer	1.06	(0.26,4.22)	0.939
total_chol	1.06	(1.03,1.1)	0
HRV_C1d	1.06	(1.02,1.09)	0.002
HRV_pNN20	1.05	(1.02,1.09)	0.003
HRV_MFDFA_alpha1_Asymmetry	1.04	(1.01,1.08)	0.011
HRV_MSEn	1.03	(1,1.07)	0.065
HRV_pNN50	1.02	(0.98,1.05)	0.315
HRV_KFD	1.02	(1,1.04)	0.051
activityHigh	1.01	(0.91,1.11)	0.923
HRV_ShanEn	1.01	(0.98,1.04)	0.593
HRV_MFDFA_alpha1_Width	1.01	(0.98,1.05)	0.411
activityModerate	1	(0.91,1.11)	0.928
HRV_HF	1	(0.97,1.03)	0.913
HRV_TP	1	(0.97,1.04)	0.931
HRV_MFDFA_alpha1_Fluctuation	1	(0.97,1.04)	0.901
HRV_MFDFA_alpha1_Increment	1	(0.97,1.04)	0.771
HRV_LFHF	0.99	(0.95,1.03)	0.622
HRV_HFn	0.99	(0.96,1.03)	0.732
HRV_LnHF	0.99	(0.96,1.03)	0.674
HRV_S	0.98	(0.94,1.02)	0.277
HRV_MFDFA_alpha1_Mean	0.98	(0.95,1.01)	0.188
HRV_TINN	0.96	(0.92,1)	0.029
HRV_FuzzyEn	0.96	(0.93,0.99)	0.009
HRV_C1a	0.95	(0.91,0.98)	0.002
educationCSEs	0.94	(0.79,1.1)	0.424
ethnicityMixed	0.93	(0.8,1.09)	0.401
HRV_CVSD	0.93	(0.9,0.96)	0
HRV_MadNN	0.93	(0.89,0.96)	0
HRV_MCVNN	0.93	(0.9,0.96)	0
HRV_IQRNN	0.93	(0.89,0.96)	0
HRV_HTI	0.93	(0.9,0.97)	0
HRV_LF	0.93	(0.9,0.96)	0
HRV_SD1d	0.93	(0.89,0.96)	0
HRV_RMSSD	0.92	(0.89,0.96)	0
HRV_SDSD	0.92	(0.89,0.96)	0
HRV_SD1	0.92	(0.89,0.96)	0
HRV_SD1a	0.92	(0.89,0.96)	0
HRV_C2a	0.92	(0.89,0.95)	0
HRV_Ca	0.92	(0.89,0.95)	0
HRV_MFDFA_alpha1_Max	0.92	(0.89,0.95)	0
HRV_MFDFA_alpha1_Delta	0.92	(0.89,0.95)	0
ethnicityAsian/Asian British	0.91	(0.78,1.05)	0.2
HRV_PI	0.91	(0.88,0.94)	0
HRV_SDNNd	0.91	(0.88,0.95)	0
HRV_CD	0.91	(0.88,0.94)	0
HRV_MFDFA_alpha1_Peak	0.91	(0.87,0.94)	0
HRV_SDNN	0.9	(0.87,0.94)	0
HRV_MedianNN	0.9	(0.86,0.93)	0
HRV_Prc80NN	0.9	(0.86,0.93)	0
HRV_SD2d	0.9	(0.87,0.94)	0
HRV_SDNNa	0.9	(0.87,0.94)	0
HRV_MeanNN	0.89	(0.85,0.92)	0
HRV_LFn	0.89	(0.86,0.93)	0
HRV_SD2	0.89	(0.86,0.93)	0
HRV_CSI_Modified	0.89	(0.85,0.93)	0
HRV_SD2a	0.89	(0.85,0.92)	0
HRV_LZC	0.89	(0.86,0.91)	0
HRV_DFA_alpha1	0.88	(0.85,0.91)	0
max_workload	0.87	(0.84,0.9)	0
HRV_CVNN	0.87	(0.84,0.91)	0
HRV_Prc20NN	0.87	(0.84,0.91)	0
HRV_MinNN	0.87	(0.84,0.9)	0
HRV_MaxNN	0.87	(0.84,0.91)	0
HRV_CSI	0.85	(0.82,0.88)	0
HRV_CVI	0.85	(0.82,0.88)	0
HRV_SDRMSSD	0.84	(0.81,0.87)	0
hdl_chol	0.81	(0.78,0.84)	0
max_heart_rate	0.78	(0.75,0.81)	0
ethnicityOther	0.66	(0.46,0.94)	0.023
ethnicityBlack/Black British	0.51	(0.31,0.85)	0.01
ethnicityChinese	0.48	(0.25,0.92)	0.028



Multivariate Cox Proportional Hazards Model

```
cox_model_full <- coxph(Surv(time, event) ~ ., data = data)
summary(cox_model_full)
```

```
cox_model_full_complete <- coxph(Surv(time, event) ~ ., data = data_complete)
summary(cox_model_full_complete)
```

PH Assumption Assessment

```
cox.zph(cox_model_full)
```

```
                            chisq df      p
age                      3.29e-01  1 0.5664
sex                      7.63e+00  1 0.0057
ethnicity                2.08e+00  5 0.8376
BMI                      4.91e-02  1 0.8245
smoking                  3.05e+00  3 0.3847
diabetes                 3.79e+00  3 0.2848
systolic_bp              1.48e+00  1 0.2244
hypertension_treatment   8.34e+00  1 0.0039
total_chol               7.83e-01  1 0.3762
hdl_chol                 7.09e+00  1 0.0077
education                3.98e+00  7 0.7816
activity                 8.02e-02  2 0.9607
max_workload             2.00e+00  1 0.1572
max_heart_rate           7.20e+00  1 0.0073
HRV_MeanNN               1.16e+00  1 0.2812
HRV_SDNN                 9.03e-01  1 0.3421
HRV_RMSSD                8.04e-01  1 0.3699
HRV_SDSD                 8.14e-01  1 0.3671
HRV_CVNN                 1.00e-01  1 0.7517
HRV_CVSD                 1.26e-01  1 0.7230
HRV_MedianNN             1.86e+00  1 0.1729
HRV_MadNN                5.12e-01  1 0.4745
HRV_MCVNN                4.34e-01  1 0.5099
HRV_IQRNN                8.30e-01  1 0.3623
```

```
HRV_SDRMSSD                  7.90e-02  1 0.7786
HRV_Prc20NN                  2.90e-01  1 0.5901
HRV_Prc80NN                  1.62e+00  1 0.2036
HRV_pNN50                    8.51e-01  1 0.3564
HRV_pNN20                    1.05e+00  1 0.3054
HRV_MinNN                    2.47e-03  1 0.9604
HRV_MaxNN                    6.19e-01  1 0.4313
HRV_HTI                      8.28e-02  1 0.7735
HRV_TINN                     5.50e-01  1 0.4583
HRV_LF                       5.40e-01  1 0.4625
HRV_HF                       2.76e-01  1 0.5994
HRV_VHF                      1.47e+00  1 0.2252
HRV_LFHF                     7.22e-01  1 0.3955
HRV_LFn                      5.19e-02  1 0.8198
HRV_HFn                      1.28e-01  1 0.7202
HRV_LnHF                     1.08e-01  1 0.7425
HRV_SD2                      8.08e-01  1 0.3688
HRV_SD1SD2                   4.53e-01  1 0.5007
HRV_S                        1.36e+00  1 0.2432
HRV_CSI                      2.84e-01  1 0.5942
HRV_CVI                      8.53e-02  1 0.7702
HRV_CSI_Modified             7.36e-01  1 0.3909
HRV_PIP                      1.28e-02  1 0.9099
HRV_IALS                     4.99e-02  1 0.8233
HRV_PSS                      2.95e-02  1 0.8635
HRV_PAS                      1.64e+00  1 0.2002
HRV_GI                       7.28e-01  1 0.3934
HRV_SI                       1.39e-01  1 0.7095
HRV_AI                       7.84e-01  1 0.3759
HRV_PI                       5.72e-01  1 0.4495
HRV_C1d                      3.46e-01  1 0.5566
HRV_SD1d                     7.11e-01  1 0.3992
HRV_SD1a                     8.98e-01  1 0.3434
HRV_C2d                      2.96e-01  1 0.5861
HRV_SD2d                     5.63e-01  1 0.4530
HRV_SD2a                     8.94e-01  1 0.3445
HRV_Cd                       6.93e-01  1 0.4052
HRV_SDNNd                    7.23e-01  1 0.3953
HRV_SDNNa                    9.64e-01  1 0.3261
HRV_ApEn                     9.43e-01  1 0.3316
HRV_ShanEn                   3.81e-01  1 0.5373
HRV_FuzzyEn                  1.87e-01  1 0.6657
HRV_MSEn                     8.62e-02  1 0.7691
```

```
HRV_CMSEn                    1.13e+00  1 0.2887
HRV_RCMSEn                   4.32e-01  1 0.5112
HRV_CD                       3.22e-02  1 0.8575
HRV_HFD                      7.22e-02  1 0.7881
HRV_KFD                      9.56e-01  1 0.3283
HRV_LZC                      1.11e-02  1 0.9160
HRV_DFA_alpha1               4.11e-01  1 0.5215
HRV_MFDFA_alpha1_Width       1.67e-02  1 0.8973
HRV_MFDFA_alpha1_Peak        5.40e-02  1 0.8162
HRV_MFDFA_alpha1_Mean        6.44e-02  1 0.7997
HRV_MFDFA_alpha1_Max         2.58e-01  1 0.6113
HRV_MFDFA_alpha1_Delta       2.84e-02  1 0.8662
HRV_MFDFA_alpha1_Asymmetry   1.86e-02  1 0.8915
HRV_MFDFA_alpha1_Fluctuation 5.17e-01  1 0.4721
HRV_MFDFA_alpha1_Increment   2.13e-01  1 0.6443
GLOBAL                       1.15e+02 97 0.0974
```

```
cox.zph(cox_model_full_complete)
```

```
                                chisq df      p
age                          3.29e-01  1 0.5664
sex                          7.63e+00  1 0.0057
ethnicity                    2.08e+00  5 0.8376
BMI                          4.91e-02  1 0.8245
smoking                      3.05e+00  3 0.3847
diabetes                     3.79e+00  3 0.2848
systolic_bp                  1.48e+00  1 0.2244
hypertension_treatment       8.34e+00  1 0.0039
total_chol                   7.83e-01  1 0.3762
hdl_chol                     7.09e+00  1 0.0077
education                    3.98e+00  7 0.7816
activity                     8.02e-02  2 0.9607
max_workload                 2.00e+00  1 0.1572
max_heart_rate               7.20e+00  1 0.0073
HRV_MeanNN                   1.16e+00  1 0.2812
HRV_SDNN                     9.03e-01  1 0.3421
HRV_RMSSD                    8.04e-01  1 0.3699
HRV_SDSD                     8.14e-01  1 0.3671
HRV_CVNN                     1.00e-01  1 0.7517
HRV_CVSD                     1.26e-01  1 0.7230
HRV_MedianNN                 1.86e+00  1 0.1729
HRV_MadNN                    5.12e-01  1 0.4745
```

```
HRV_MCVNN                    4.34e-01  1 0.5099
HRV_IQRNN                    8.30e-01  1 0.3623
HRV_SDRMSSD                  7.90e-02  1 0.7786
HRV_Prc20NN                  2.90e-01  1 0.5901
HRV_Prc80NN                  1.62e+00  1 0.2036
HRV_pNN50                    8.51e-01  1 0.3564
HRV_pNN20                    1.05e+00  1 0.3054
HRV_MinNN                    2.47e-03  1 0.9604
HRV_MaxNN                    6.19e-01  1 0.4313
HRV_HTI                      8.28e-02  1 0.7735
HRV_TINN                     5.50e-01  1 0.4583
HRV_LF                       5.40e-01  1 0.4625
HRV_HF                       2.76e-01  1 0.5994
HRV_VHF                      1.47e+00  1 0.2252
HRV_LFHF                     7.22e-01  1 0.3955
HRV_LFn                      5.19e-02  1 0.8198
HRV_HFn                      1.28e-01  1 0.7202
HRV_LnHF                     1.08e-01  1 0.7425
HRV_SD2                      8.08e-01  1 0.3688
HRV_SD1SD2                   4.53e-01  1 0.5007
HRV_S                        1.36e+00  1 0.2432
HRV_CSI                      2.84e-01  1 0.5942
HRV_CVI                      8.53e-02  1 0.7702
HRV_CSI_Modified             7.36e-01  1 0.3909
HRV_PIP                      1.28e-02  1 0.9099
HRV_IALS                     4.99e-02  1 0.8233
HRV_PSS                      2.95e-02  1 0.8635
HRV_PAS                      1.64e+00  1 0.2002
HRV_GI                       7.28e-01  1 0.3934
HRV_SI                       1.39e-01  1 0.7095
HRV_AI                       7.84e-01  1 0.3759
HRV_PI                       5.72e-01  1 0.4495
HRV_C1d                      3.46e-01  1 0.5566
HRV_SD1d                     7.11e-01  1 0.3992
HRV_SD1a                     8.98e-01  1 0.3434
HRV_C2d                      2.96e-01  1 0.5861
HRV_SD2d                     5.63e-01  1 0.4530
HRV_SD2a                     8.94e-01  1 0.3445
HRV_Cd                       6.93e-01  1 0.4052
HRV_SDNNd                    7.23e-01  1 0.3953
HRV_SDNNa                    9.64e-01  1 0.3261
HRV_ApEn                     9.43e-01  1 0.3316
HRV_ShanEn                   3.81e-01  1 0.5373
```

```
HRV_FuzzyEn                  1.87e-01  1 0.6657
HRV_MSEn                     8.62e-02  1 0.7691
HRV_CMSEn                    1.13e+00  1 0.2887
HRV_RCMSEn                   4.32e-01  1 0.5112
HRV_CD                       3.22e-02  1 0.8575
HRV_HFD                      7.22e-02  1 0.7881
HRV_KFD                      9.56e-01  1 0.3283
HRV_LZC                      1.11e-02  1 0.9160
HRV_DFA_alpha1               4.11e-01  1 0.5215
HRV_MFDFA_alpha1_Width       1.67e-02  1 0.8973
HRV_MFDFA_alpha1_Peak        5.40e-02  1 0.8162
HRV_MFDFA_alpha1_Mean        6.44e-02  1 0.7997
HRV_MFDFA_alpha1_Max         2.58e-01  1 0.6113
HRV_MFDFA_alpha1_Delta       2.84e-02  1 0.8662
HRV_MFDFA_alpha1_Asymmetry   1.86e-02  1 0.8915
HRV_MFDFA_alpha1_Fluctuation 5.17e-01  1 0.4721
HRV_MFDFA_alpha1_Increment   2.13e-01  1 0.6443
GLOBAL                       1.15e+02 97 0.0974
```

The proportional hazards assumption was tested using Schoenfeld residuals. None of the variables violated the PH assumption (all $p > 0.05$), indicating that the Cox proportional hazards model was appropriate for our analysis.

Variable Selection

LASSO

```
# * LASSO doesn't allow missing values
set.seed(1234)
# x <- as.matrix(data_complete %>% select(-c(time, event)))
# * We need to explicitly use model.matrix for categorical variables
x <- model.matrix(~ . - 1 - time - event, data = data_complete)
y <- Surv(data_complete$time, data_complete$event)
cox_model_lasso.cv <- cv.glmnet(
    x,
    y,
    family = "cox",
    alpha = 1, # 1 for LASSO, 0 for Ridge
    nfolds = 10
)
```

```
# plot(cox_model_lasso.cv)  # Plot partial likelihood deviance vs log(lambda)
print(cox_model_lasso.cv$lambda.min)
```

```
[1] 0.002211303
```

```
print(cox_model_lasso.cv$lambda.1se)
```

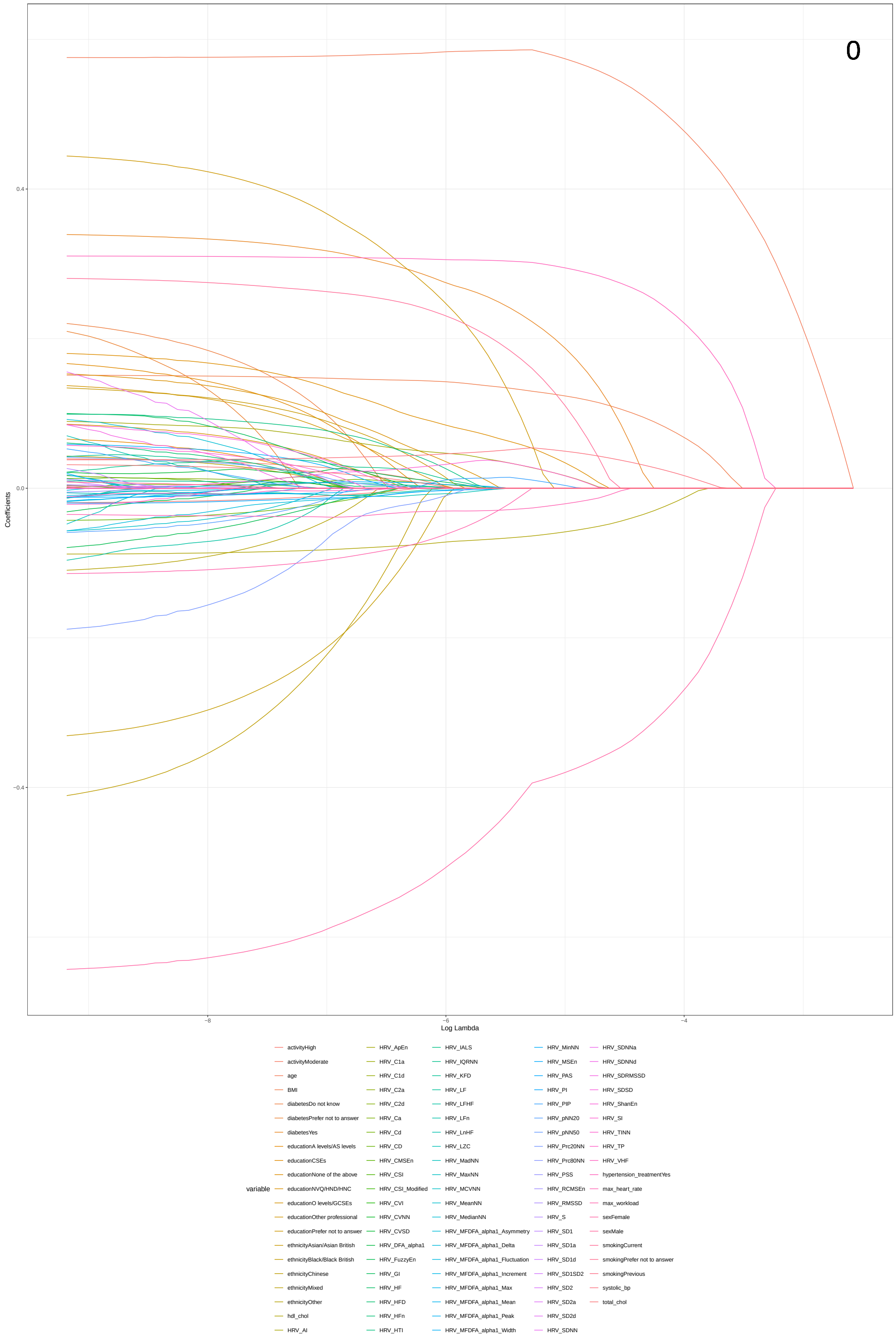
```
[1] 0.01075268
```

As mentioned in the paper, we will use the value of hyperparameter `lambda.1se` that gave the most shrunk model but still was within one standard error from the value that gave the lowest error. This is shown to produce consistently better performance than `lambda.min`.

```
cox_model_lasso <- glmnet(
    x,
    y,
    family = "cox",
    alpha = 1,
    lambda = cox_model_lasso.cv$lambda.1se
)
cox_model_lasso.coef <- coef(cox_model_lasso)
print(cox_model_lasso.coef)
selected_vars <- rownames(cox_model_lasso.coef)[which(cox_model_lasso.coef !=
 ↪  0)]
print(selected_vars)
```

```
# * To visualize the LASSO path, we should not supply lambda
cox_model_lasso_fullpath <- glmnet(
    x,
    y,
    family = "cox",
    alpha = 1
)
```

```
# plot(cox_model_lasso_fullpath, xvar = "lambda", label = TRUE)
autoplot(cox_model_lasso_fullpath, xvar = "lambda", label = TRUE, label.size
 ↪  = 15) +
    theme_bw() +
    theme(legend.position = "bottom") # better way of visualizing the LASSO
     ↪  path
```


0
0.4
0.0
−0.4
Coefficients
−8
−6
−4
Log Lambda
variable
activityHigh
activityModerate
age
BMI
diabetesDo not know
diabetesPrefer not to answer
diabetesYes
educationA levels/AS levels
educationCSEs
educationNone of the above
educationNVQ/HND/HNC
educationO levels/GCSEs
educationOther professional
educationPrefer not to answer
ethnicityAsian/Asian British
ethnicityBlack/Black British
ethnicityChinese
ethnicityMixed
ethnicityOther
hdl_chol
HRV_AI
HRV_ApEn
HRV_C1a
HRV_C1d
HRV_C2a
HRV_C2d
HRV_Ca
HRV_Cd
HRV_CD
HRV_CMSEn
HRV_CSI
HRV_CSI_Modified
HRV_CVI
HRV_CVNN
HRV_CVSD
HRV_DFA_alpha1
HRV_FuzzyEn
HRV_GI
HRV_HF
HRV_HFD
HRV_HFn
HRV_HTI
HRV_IALS
HRV_IQRNN
HRV_KFD
HRV_LF
HRV_LFHF
HRV_LFn
HRV_LnHF
HRV_LZC
HRV_MadNN
HRV_MaxNN
HRV_MCVNN
HRV_MeanNN
HRV_MedianNN
HRV_MFDFA_alpha1_Asymmetry
HRV_MFDFA_alpha1_Delta
HRV_MFDFA_alpha1_Fluctuation
HRV_MFDFA_alpha1_Increment
HRV_MFDFA_alpha1_Max
HRV_MFDFA_alpha1_Mean
HRV_MFDFA_alpha1_Peak
HRV_MFDFA_alpha1_Width
HRV_MinNN
HRV_MSEn
HRV_PAS
HRV_PI
HRV_PIP
HRV_pNN20
HRV_pNN50
HRV_Prc20NN
HRV_Prc80NN
HRV_PSS
HRV_RCMSEn
HRV_RMSSD
HRV_S
HRV_SD1
HRV_SD1a
HRV_SD1d
HRV_SD1SD2
HRV_SD2
HRV_SD2a
HRV_SD2d
HRV_SDNN
HRV_SDNNa
HRV_SDNNd
HRV_SDRMSSD
HRV_SDSD
HRV_ShanEn
HRV_SI
HRV_TINN
HRV_TP
HRV_VHF
hypertension_treatmentYes
max_heart_rate
max_workload
sexFemale
sexMale
smokingCurrent
smokingPrefer not to answer
smokingPrevious
systolic_bp
total_chol

Stepwise Selection based on BIC

```
# * Stepwise selection doesn't allow missing values
cox_model_step <- MASS::stepAIC(cox_model_full_complete,
    direction = "both",
    k = log(nrow(data)), # Use BIC instead of AIC
    trace = FALSE
)
```

```
summary(cox_model_step)
```

```
Call:
coxph(formula = Surv(time, event) ~ age + sex + BMI + hypertension_treatment +
    hdl_chol + max_workload + HRV_Prc20NN + HRV_HTI, data = data_complete)

  n= 26729, number of events= 3372

                              coef exp(coef) se(coef)      z Pr(>|z|)
age                        0.59860   1.81957  0.02337 25.611  < 2e-16 ***
sexMale                    0.68880   1.99133  0.05268 13.075  < 2e-16 ***
BMI                        0.17108   1.18659  0.01871  9.143  < 2e-16 ***
hypertension_treatmentYes  0.34591   1.41327  0.03939  8.782  < 2e-16 ***
hdl_chol                  -0.08969   0.91421  0.02204 -4.069 4.72e-05 ***
max_workload              -0.15139   0.85952  0.02642 -5.730 1.00e-08 ***
HRV_Prc20NN               -0.13808   0.87103  0.03054 -4.521 6.15e-06 ***
HRV_HTI                    0.09787   1.10282  0.02348  4.169 3.06e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                          exp(coef) exp(-coef) lower .95 upper .95
age                          1.8196     0.5496    1.7381    1.9049
sexMale                      1.9913     0.5022    1.7960    2.2079
BMI                          1.1866     0.8428    1.1439    1.2309
hypertension_treatmentYes    1.4133     0.7076    1.3083    1.5267
hdl_chol                     0.9142     1.0938    0.8756    0.9546
max_workload                 0.8595     1.1634    0.8161    0.9052
HRV_Prc20NN                  0.8710     1.1481    0.8204    0.9248
HRV_HTI                      1.1028     0.9068    1.0532    1.1547

Concordance= 0.712  (se = 0.004 )
```

```
Likelihood ratio test= 1960  on 8 df,   p=<2e-16
Wald test            = 1800  on 8 df,   p=<2e-16
Score (logrank) test = 1937  on 8 df,   p=<2e-16
```

Summary of Variable Selection

We will compare the selection of variables from all models we have built.

```
# Obtain the selected variables from all models
variable_names_all <- colnames(data) %>%
    setdiff(c("time", "event"))

variable_names_univariate <- results_univariate %>%
    filter(p_value < 0.05) %>%
    pull(predictor)

variable_names_multivariate <- summary(cox_model_full_complete)$coefficients
 ↪  %>%
    as.data.frame() %>%
    rownames_to_column(var = "predictor") %>% # transpose, "predictor" will
     ↪  now be the column name
    filter(`Pr(>|z|)` < 0.05) %>%
    pull(predictor)

variable_names_lasso <-
 ↪  rownames(cox_model_lasso.coef)[which(cox_model_lasso.coef != 0)]

variable_names_step <- summary(cox_model_step)$coefficients %>%
    as.data.frame() %>%
    filter(`Pr(>|z|)` < 0.05) %>%
    rownames()
```

```
# * We need to handle the categorical variables for correct displaying
get_base_variable_name <- function(variable_name) {
    if
     ↪  (grepl("^(sex|ethnicity|education|smoking|diabetes|activity|hypertension_treatment)[A
     ↪  variable_name)) {
        return(sub("([a-z_]+)[A-Z].*", "\\1", variable_name))
    }
    return(variable_name)
}
```

```
variable_selection_matrix <- matrix(
    0,
    nrow = length(variable_names_all),
    ncol = 4 # univariate, multivariate, lasso, stepwise
)
colnames(variable_selection_matrix) <- c("univariate", "multivariate",
 ↪  "lasso", "stepwise")
rownames(variable_selection_matrix) <- variable_names_all

for (variable in variable_names_all) {
    if (determine_type(variable) == "categorical") {
        if (variable %in% sapply(variable_names_univariate,
         ↪  get_base_variable_name)) {
            variable_selection_matrix[variable, "univariate"] <- 1
        }
        if (variable %in% sapply(variable_names_multivariate,
         ↪  get_base_variable_name)) {
            variable_selection_matrix[variable, "multivariate"] <- 1
        }
        if (variable %in% sapply(variable_names_lasso,
         ↪  get_base_variable_name)) {
            variable_selection_matrix[variable, "lasso"] <- 1
        }
        if (variable %in% sapply(variable_names_step,
         ↪  get_base_variable_name)) {
            variable_selection_matrix[variable, "stepwise"] <- 1
        }
    } else {
        if (variable %in% variable_names_univariate) {
            variable_selection_matrix[variable, "univariate"] <- 1
        }
        if (variable %in% variable_names_multivariate) {
            variable_selection_matrix[variable, "multivariate"] <- 1
        }
        if (variable %in% variable_names_lasso) {
            variable_selection_matrix[variable, "lasso"] <- 1
        }
        if (variable %in% variable_names_step) {
            variable_selection_matrix[variable, "stepwise"] <- 1
        }
    }
}
```

```
symbol_selected <- "*"

selection_table <- data.frame(
    Variable = variable_names_all,
    Univariate = ifelse(variable_selection_matrix[, "univariate"] == 1,
     ↪  symbol_selected, ""),
    Multivariate = ifelse(variable_selection_matrix[, "multivariate"] == 1,
     ↪  symbol_selected, ""),
    LASSO = ifelse(variable_selection_matrix[, "lasso"] == 1,
     ↪  symbol_selected, ""),
    Stepwise = ifelse(variable_selection_matrix[, "stepwise"] == 1,
     ↪  symbol_selected, "")
) %>%
    mutate(Num_Selected = rowSums(variable_selection_matrix)) %>%
    arrange(desc(Num_Selected), Variable) %>%
    as.data.frame() %>%
    remove_rownames()

variable_categories <- sapply(variable_names_all, determine_category)
category_colors <- c(
    "covariate" = "#FFB6C1", #
    "time"      = "#1E90FF", #
    "frequency" = "#32CD32", #
    "poincare"  = "#FF4500", #
    "entropy"   = "#FF8C00", #
    "fractal"   = "#FFD700", #
    "unknown"   = "#000000" #
)
category_colors_names <- c(
    "covariate"  = "pink", #
    "time"       = "blue", #
    "frequency"  = "green", #
    "poincare"   = "red", #
    "entropy"    = "orange", #
    "fractal"    = "gold" #
)
category_legend <- sapply(names(category_colors_names), function(cat) {
    sprintf(
        "%s: %s",
        tools::toTitleCase(cat),
        tools::toTitleCase(category_colors_names[cat])
    )
}) %>%
    paste(collapse = "; ")


selection_table %>%
    kbl(
        caption = "Variable Selection by Different Models",
        align = c("|l", "c", "c", "c", "c", "c|"),
        col.names = c("Variable", "Univariate", "Multivariate", "LASSO",
         ↪  "Stepwise", "Selected Times"),
        longtable = TRUE
    ) %>%
    kable_styling(
        bootstrap_options = c("striped", "hover", "condensed", "responsive"),
        position = "center",
        font_size = 9,
        latex_options = c("repeat_header", "striped", "HOLD_position")
    ) %>%
    # Add color for different categories of variables
    column_spec(1,
        color =
         ↪  category_colors[variable_categories[selection_table$Variable]],
        bold = TRUE
    ) %>%
    # Add a header colname for four columns: Univariate, Multivariate, LASSO,
     ↪  Stepwise
    add_header_above(c(
        " " = 1,
        "Selection Methods" = 4,
        " " = 1
    )) %>%
    footnote(
        general = sprintf("%s", category_legend),
        general_title = "Note:"
    )
```

```
Warning: 'xfun::attr()' is deprecated.
```

```
Use 'xfun::attr2()' instead.
See help("Deprecated")

Warning: 'xfun::attr()' is deprecated.
Use 'xfun::attr2()' instead.
See help("Deprecated")
```

Table 1: Variable Selection by Different Models

	Selection Methods				
Variable	Univariate	Multivariate	LASSO	Stepwise	Selected Times
BMI	*	*	*	*	4
age	*	*	*	*	4
hdl_chol	*	*	*	*	4
hypertension_treatment	*	*	*	*	4
sex	*	*	*	*	4
HRV_HTI	*	*		*	3
diabetes	*	*	*		3
max_workload	*	*		*	3
HRV_ApEn	*	*			2
HRV_FuzzyEn	*	*			2
HRV_Prc20NN	*			*	2
education	*	*			2
max_heart_rate	*		*		2
smoking	*	*			2
systolic_bp	*		*		2
HRV_AI	*				1
HRV_C1a	*				1
HRV_C1d	*				1
HRV_C2a	*				1
HRV_C2d	*				1
HRV_CD	*				1
HRV_CMSEn	*				1
HRV_CSI	*				1
HRV_CSI_Modified	*				1
HRV_CVI	*				1
HRV_CVNN	*				1
HRV_CVSD	*				1
HRV_Ca	*				1
HRV_Cd	*				1
HRV_DFA_alpha1	*				1
HRV_GI	*				1
HRV_HFD	*				1
HRV_IALS	*				1
HRV_IQRNN	*				1
HRV_LF	*				1
HRV_LFn	*				1
HRV_LZC	*				1

Table 1: Variable Selection by Different Models *(continued)*

Variable	Univariate	Multivariate	LASSO	Stepwise	Selected Times
HRV_MCVNN	*				1
HRV_MFDFA_alpha1_Asymmetry	*				1
HRV_MFDFA_alpha1_Delta	*				1
HRV_MFDFA_alpha1_Max	*				1
HRV_MFDFA_alpha1_Peak	*				1
HRV_MadNN	*				1
HRV_MaxNN	*				1
HRV_MeanNN	*				1
HRV_MedianNN	*				1
HRV_MinNN	*				1
HRV_PAS	*				1
HRV_PI	*				1
HRV_PIP	*				1
HRV_PSS	*				1
HRV_Prc80NN	*				1
HRV_RCMSEn	*				1
HRV_RMSSD	*				1
HRV_SD1	*				1
HRV_SD1SD2	*				1
HRV_SD1a	*				1
HRV_SD1d	*				1
HRV_SD2	*				1
HRV_SD2a	*				1
HRV_SD2d	*				1
HRV_SDNN	*				1
HRV_SDNNa	*				1
HRV_SDNNd	*				1
HRV_SDRMSSD	*				1
HRV_SDSD	*				1
HRV_SI	*				1
HRV_ShanEn		*			1
HRV_TINN	*				1
HRV_VHF	*				1
HRV_pNN20	*				1
ethnicity	*				1
total_chol	*				1
HRV_HF					0
HRV_HFn					0
HRV_KFD					0
HRV_LFHF					0
HRV_LnHF					0
HRV_MFDFA_alpha1_Fluctuation					0
HRV_MFDFA_alpha1_Increment					0
HRV_MFDFA_alpha1_Mean					0
HRV_MFDFA_alpha1_Width					0
HRV_MSEn					0
HRV_S					0
HRV_TP					0

Table 1: Variable Selection by Different Models *(continued)*

Variable	Univariate	Multivariate	LASSO	Stepwise	Selected Times
HRV_pNN50					0
activity					0

Note:

Covariate: Pink; Time: Blue; Frequency: Green; Poincare: Red; Entropy: Orange; Fractal: Gold