AUC_WUT_INN

S_16_373

3/29/2022

Load the libraries

```
library(tidyverse)
library(survival)
library(ggfortify)
library(survminer)
library(rms)
library(dynpred)
library(MASS)
library(CPE)
library(scatterplot3d)
library(plotly)
```

Read the data sets

```
Home <- read.csv("~/Research/Research---ST426/Data/NewData/Merge/Home/Home.csv")
Overseas <- read.csv("~/Research/Research---ST426/Data/NewData/Merge/Overseas/Overseas.csv")
```

Merge two data sets

```
data_merge <- rbind(Home , Overseas)
```

Filter the data of WU Tharanga

```
bt15 <- data_merge %>% filter(grepl('WU Tharanga' , Striker))
```

1)Analysis for two different innings

First innings

Filter the first innings

```
WUT_1 <- bt15 %>% filter(innings == 1)
```

Cox model for first innings

```
cox_WUT_1 <- coxph(Surv(cum_balls , wicket) ~
                     cum_runs+cum_singles+cum_doubles+cum_boundaries+ridge(cum_dotBalls), data = WUT_1)
summary(cox_WUT_1)
```

```
## Call:
## coxph(formula = Surv(cum_balls, wicket) ~ cum_runs + cum_singles +
##     cum_doubles + cum_boundaries + ridge(cum_dotBalls), data = WUT_1)
##
##   n= 874, number of events= 25
##
##                       coef     se(coef) se2     Chisq DF p
## cum_runs             -0.01295 0.07377  0.07242  0.03 1  8.6e-01
## cum_singles          -0.59086 0.13788  0.13645 18.36 1  1.8e-05
## cum_doubles           0.14300 0.30462  0.30319  0.22 1  6.4e-01
## cum_boundaries       -0.55921 0.37231  0.37131  2.26 1  1.3e-01
## ridge(cum_dotBalls) -0.39500 0.07586  0.05340 27.11 1  1.9e-07
##
##                      exp(coef) exp(-coef) lower .95 upper .95
## cum_runs                0.9871     1.0130    0.8542    1.1407
## cum_singles             0.5539     1.8055    0.4227    0.7257
## cum_doubles             1.1537     0.8668    0.6351    2.0960
## cum_boundaries          0.5717     1.7493    0.2756    1.1859
## ridge(cum_dotBalls)     0.6737     1.4844    0.5806    0.7817
##
## Iterations: 3 outer, 14 Newton-Raphson
## Degrees of freedom for terms= 1.0 1.0 1.0 1.0 0.5
## Concordance= 0.977  (se = 0.005 )
## Likelihood ratio test= 116.5  on 4.42 df,   p=<2e-16
```

Finding the best model using AIC

```
model_WUT_1 <- stepAIC(cox_WUT_1)
```

```
## Start:  AIC=170.42
## Surv(cum_balls, wicket) ~ cum_runs + cum_singles + cum_doubles +
##     cum_boundaries + ridge(cum_dotBalls)
##
##                            Df    AIC
## - cum_doubles         1.03709 168.22
## <none>                        170.42
## - cum_runs            1.08063 182.68
## - cum_boundaries      1.05776 185.07
## - cum_singles         0.94797 199.93
## - ridge(cum_dotBalls) 0.42380 218.24
```

```
##
## Step:  AIC=168.22
## Surv(cum_balls, wicket) ~ cum_runs + cum_singles + cum_boundaries +
##     ridge(cum_dotBalls)
##
##                            Df    AIC
## <none>                        168.22
## - cum_runs            1.02440 181.61
## - cum_boundaries      1.05283 184.75
## - cum_singles         0.92638 204.91
## - ridge(cum_dotBalls) 0.38671 217.18
```

Best Model

```
cox_WUT_1_new <- coxph(Surv(cum_balls , wicket) ~
                         cum_runs + cum_singles + cum_boundaries+ ridge(cum_dotBalls), data = WUT_1)
summary(cox_WUT_1_new)
```

```
## Call:
## coxph(formula = Surv(cum_balls, wicket) ~ cum_runs + cum_singles +
##     cum_boundaries + ridge(cum_dotBalls), data = WUT_1)
##
##   n= 874, number of events= 25
##
##                       coef      se(coef) se2     Chisq DF p
## cum_runs             0.004074 0.06384  0.06123  0.00 1  9.5e-01
## cum_singles         -0.605314 0.13421  0.13310 20.34 1  6.5e-06
## cum_boundaries      -0.629851 0.34469  0.34191  3.34 1  6.8e-02
## ridge(cum_dotBalls) -0.399918 0.07558  0.05340 28.00 1  1.2e-07
##
##                       exp(coef) exp(-coef) lower .95 upper .95
## cum_runs                 1.0041     0.9959    0.8860    1.1379
## cum_singles              0.5459     1.8318    0.4196    0.7102
## cum_boundaries           0.5327     1.8773    0.2711    1.0468
## ridge(cum_dotBalls)      0.6704     1.4917    0.5781    0.7774
##
## Iterations: 3 outer, 14 Newton-Raphson
## Degrees of freedom for terms= 0.9 1.0 1.0 0.5
## Concordance= 0.978  (se = 0.004 )
## Likelihood ratio test= 116.6  on 3.39 df,   p=<2e-16
```

Test pH assumptions

```
test_WUT_1 <- cox.zph(cox_WUT_1_new)
test_WUT_1
```

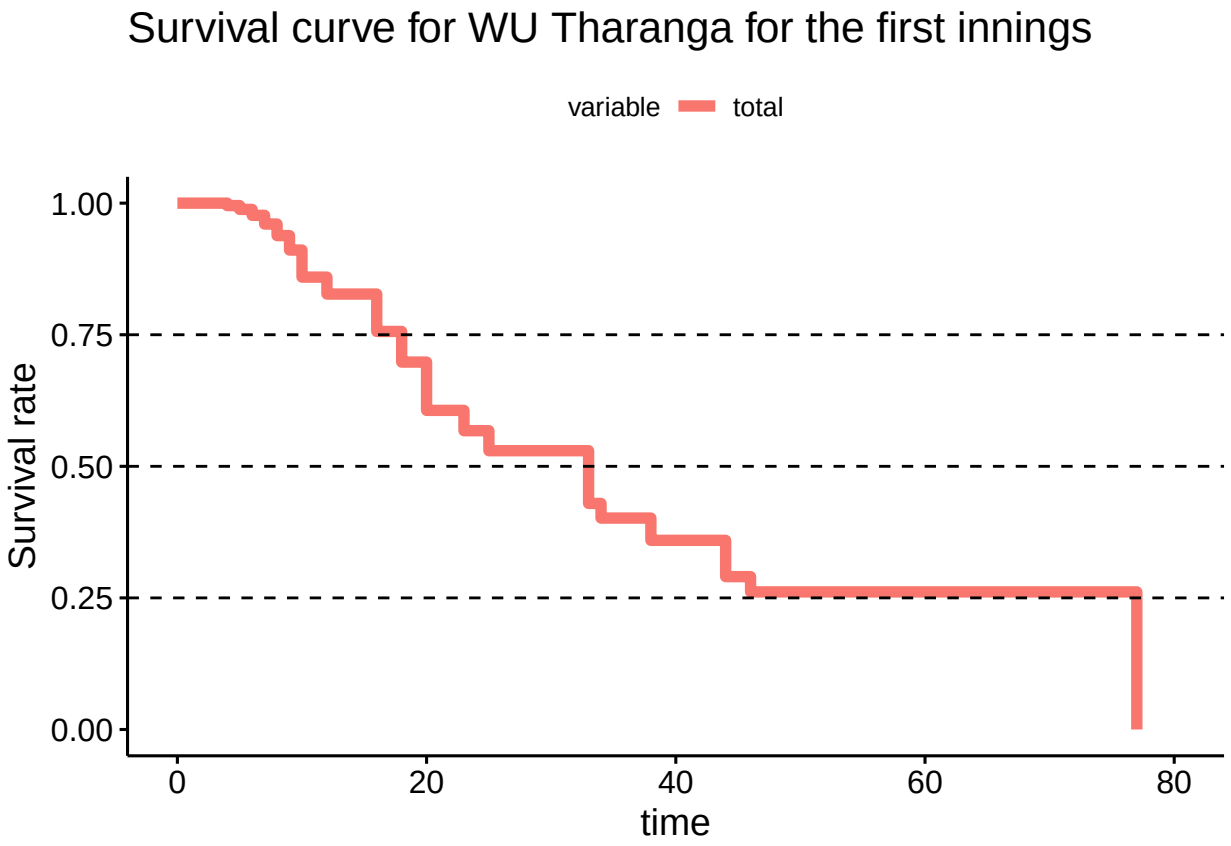
```
##                         chisq   df    p
## cum_runs             1.16e+00 0.92 0.26
```

```
## cum_singles          3.50e-03 0.98 0.95
## cum_boundaries       5.98e-01 0.98 0.43
## ridge(cum_dotBalls) 4.93e-08 0.50 0.99
## GLOBAL               2.28e+00 3.39 0.59
```

Plot the survival curve

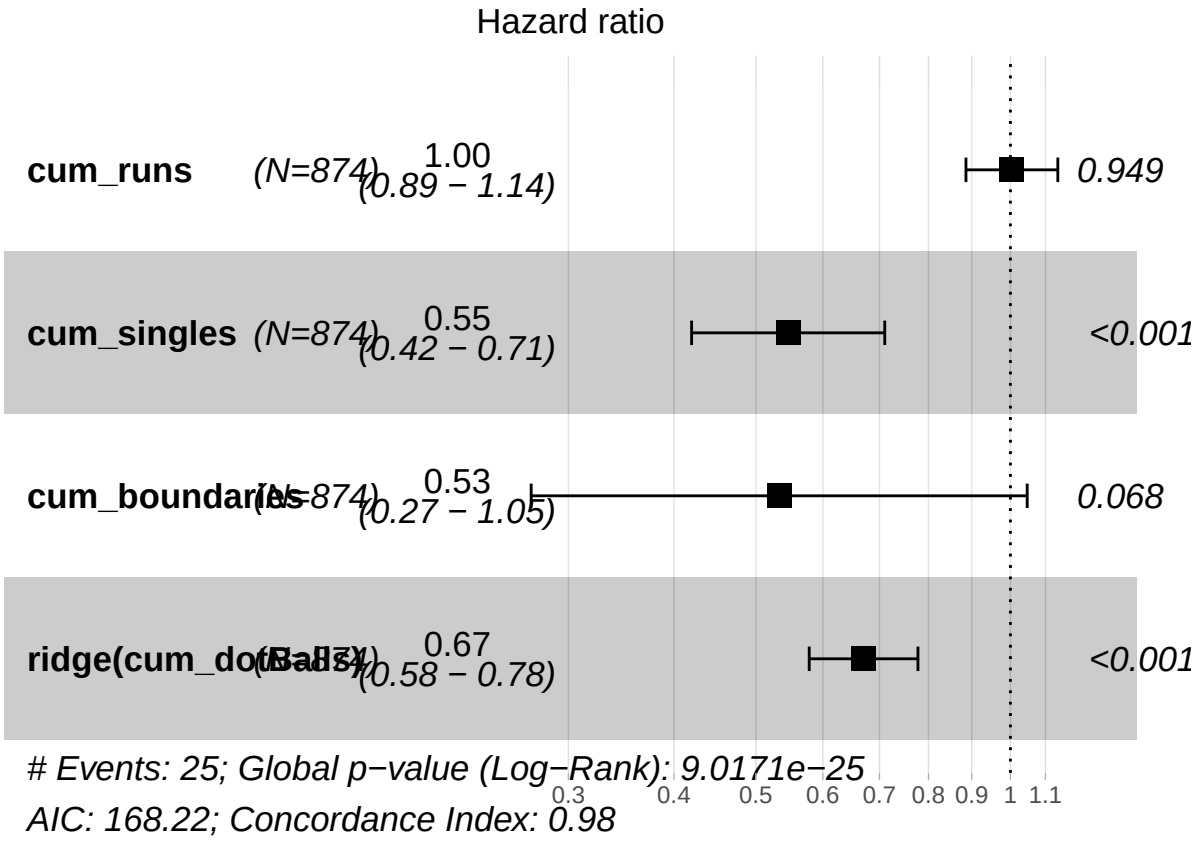
```
cox_plot_WUT_1 <- ggadjustedcurves(cox_WUT_1_new , data = WUT_1 , size = 2,ggtheme = theme_survminer())
                  geom_hline(yintercept = c(0.25,0.5,0.75) , linetype = "dashed") +xlim(0,80) +
                    ggtitle("Survival curve for WU Tharanga for the first innings") +
  theme(axis.title = element_text(size = 15),axis.text = element_text(size = 14))

cox_plot_WUT_1
```



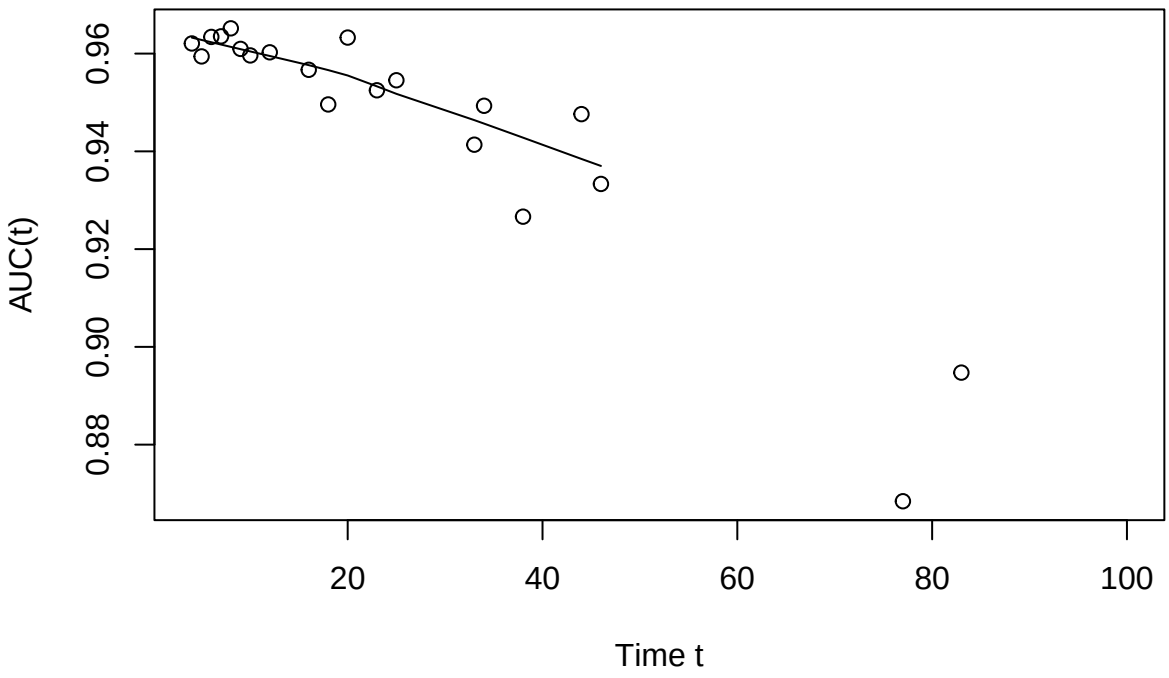
coeffient plot

```
coef_plot_WUT_1 <- ggforest(cox_WUT_1_new , data = WUT_1,fontsize = 1.2) +
  theme(axis.title = element_text(size = 15),axis.text = element_text(size = 14))
coef_plot_WUT_1
```



Area under the curve

```
AUC(Surv(cum_balls , wicket) ~
      cum_runs + cum_singles + cum_boundaries+ ridge(cum_dotBalls), data = WUT_1)
```




```
## $AUCt
##    time       AUC
## 1     4 0.9620733
## 2     5 0.9594241
## 3     6 0.9634146
## 4     7 0.9635344
## 5     8 0.9651669
## 6     9 0.9609610
## 7    10 0.9596273
## 8    12 0.9602649
## 9    16 0.9566855
## 10   18 0.9495968
## 11   20 0.9632829
## 12   23 0.9524941
## 13   25 0.9545455
## 14   33 0.9413681
## 15   34 0.9493243
## 16   38 0.9266409
## 17   44 0.9476190
## 18   46 0.9333333
## 19   77 0.8684211
## 20   83 0.8947368
## 21  100       NaN
##
## $AUC
## [1] 0.95671
```

percentage of matches won when the player is in the specific match

```
match_status_WUT_1 <- unique(WUT_1[c("Match_Number","winning_status")])
tot_matches_WUT_1 <- nrow(match_status_WUT_1)
win_matches_WUT_1 <- sum(match_status_WUT_1$winning_status==1)

win_percent_WUT_1 <- (win_matches_WUT_1/tot_matches_WUT_1)*100

win_percent_WUT_1
```

```
## [1] 21.42857
```

Runs contributed

```
match_played_WUT_1 <- data_merge %>% filter(Match_Number %in% unique(WUT_1$Match_Number))
tot_runs_WUT_1 <- match_played_WUT_1 %>% summarise(sum_runs = sum(tot_runs))
WUT_runs_1 <- WUT_1 %>%  summarise(sum_runs = sum(tot_runs))

runs_contrib_percent_WUT_1 <- (WUT_runs_1/tot_runs_WUT_1)*100
runs_contrib_percent_WUT_1
```

```
##   sum_runs
## 1 11.92701
```

Second Innings

Filter the second innings

```
WUT_2 <- bt15 %>% filter(innings == 2)
```

Cox model for second innings

```
cox_WUT_2 <- coxph(Surv(cum_balls , wicket) ~
                     cum_runs+cum_singles+cum_doubles+cum_boundaries+ridge(cum_dotBalls), data = WUT_2)
summary(cox_WUT_2)
```

```
## Call:
## coxph(formula = Surv(cum_balls, wicket) ~ cum_runs + cum_singles +
##     cum_doubles + cum_boundaries + ridge(cum_dotBalls), data = WUT_2)
##
##   n= 980, number of events= 18
##
##                      coef     se(coef) se2     Chisq DF p
```

```
## cum_runs            -0.04774 0.04255  0.04153  1.26 1  2.6e-01
## cum_singles         -0.40068 0.12537  0.12485 10.21 1  1.4e-03
## cum_doubles          0.31124 0.26047  0.25941  1.43 1  2.3e-01
## cum_boundaries       0.04417 0.25157  0.25110  0.03 1  8.6e-01
## ridge(cum_dotBalls) -0.23982 0.05027  0.03396 22.76 1  1.8e-06
##
##                     exp(coef) exp(-coef) lower .95 upper .95
## cum_runs               0.9534     1.0489    0.8771    1.0363
## cum_singles            0.6699     1.4928    0.5239    0.8564
## cum_doubles            1.3651     0.7325    0.8193    2.2745
## cum_boundaries         1.0452     0.9568    0.6383    1.7113
## ridge(cum_dotBalls)    0.7868     1.2710    0.7130    0.8682
##
## Iterations: 3 outer, 14 Newton-Raphson
## Degrees of freedom for terms= 1.0 1.0 1.0 1.0 0.5
## Concordance= 0.981  (se = 0.005 )
## Likelihood ratio test= 80.83  on 4.39 df,   p=<2e-16
```

Finding the best model using AIC

```
model_WUT_2 <- stepAIC(cox_WUT_2)
```

```
## Start:  AIC=137.23
## Surv(cum_balls, wicket) ~ cum_runs + cum_singles + cum_doubles +
##     cum_boundaries + ridge(cum_dotBalls)
##
##                            Df    AIC
## - cum_boundaries      1.07788 135.18
## - cum_doubles         1.01352 135.78
## <none>                        137.23
## - cum_runs            1.03520 138.31
## - cum_singles         0.90426 156.78
## - ridge(cum_dotBalls) 0.38923 177.70
##
## Step:  AIC=135.18
## Surv(cum_balls, wicket) ~ cum_runs + cum_singles + cum_doubles +
##     ridge(cum_dotBalls)
##
##                            Df    AIC
## - cum_doubles         1.02496 133.54
## <none>                        135.18
## - cum_runs            0.82318 147.74
## - cum_singles         0.83179 160.17
## - ridge(cum_dotBalls) 0.31134 176.66
##
## Step:  AIC=133.55
## Surv(cum_balls, wicket) ~ cum_runs + cum_singles + ridge(cum_dotBalls)
##
##                            Df    AIC
## <none>                        133.54
## - cum_runs            0.79427 148.14
```

```
## - cum_singles         0.78382 159.69
## - ridge(cum_dotBalls) 0.28638 180.61
```

Best Model

```
cox_WUT_2_new <- coxph(Surv(cum_balls , wicket) ~
                         cum_runs + cum_singles+ ridge(cum_dotBalls), data = WUT_2)
summary(cox_WUT_2_new)
```

```
## Call:
## coxph(formula = Surv(cum_balls, wicket) ~ cum_runs + cum_singles +
##     ridge(cum_dotBalls), data = WUT_2)
##
##   n= 980, number of events= 18
##
##                       coef     se(coef) se2     Chisq DF p
## cum_runs            -0.04258 0.01981  0.01849  4.62 1  3.2e-02
## cum_singles         -0.35091 0.09133  0.08892 14.76 1  1.2e-04
## ridge(cum_dotBalls) -0.24792 0.04974  0.03402 24.85 1  6.2e-07
##
##                       exp(coef) exp(-coef) lower .95 upper .95
## cum_runs                 0.9583      1.043    0.9218    0.9963
## cum_singles              0.7040      1.420    0.5887    0.8421
## ridge(cum_dotBalls)      0.7804      1.281    0.7079    0.8603
##
## Iterations: 3 outer, 13 Newton-Raphson
## Degrees of freedom for terms= 0.9 0.9 0.5
## Concordance= 0.985  (se = 0.004 )
## Likelihood ratio test= 80.3  on 2.29 df,   p=<2e-16
```

Test pH assumptions

```
test_WUT_2 <- cox.zph(cox_WUT_2_new)
test_WUT_2
```

```
##                         chisq   df      p
## cum_runs              6.95831 0.87 0.0066
## cum_singles           6.73828 0.95 0.0086
## ridge(cum_dotBalls)   0.00177 0.47 0.7878
## GLOBAL               11.32294 2.29 0.0049
```

Extended cox model

```
WUT_2_cut <- survSplit(WUT_1 , cut = 45 , end = "cum_balls" , event = "wicket" , start = "start" , id =

WUT_2_cut$hv1 <- WUT_2_cut$cum_runs  * (WUT_2_cut$start < 45)
```

```
WUT_2_cut$hv2 <- WUT_2_cut$cum_runs  * (WUT_2_cut$start >= 45)

WUT_2_cut$hv3 <- WUT_2_cut$cum_singles   * (WUT_2_cut$start < 45)
WUT_2_cut$hv4 <- WUT_2_cut$cum_singles   * (WUT_2_cut$start >= 45)

cox_WUT_2_heav <- coxph(Surv(start,cum_balls , wicket) ~
                     hv1 + hv2 +hv3 +hv4 + ridge(cum_dotBalls) + cluster (id) , data = WUT_2_cut)
summary(cox_WUT_2_heav)
```

```
## Call:
## coxph(formula = Surv(start, cum_balls, wicket) ~ hv1 + hv2 +
##     hv3 + hv4 + ridge(cum_dotBalls), data = WUT_2_cut, cluster = id)
##
##   n= 1070, number of events= 25
##
##                       coef    se(coef) se2     Chisq DF p
## hv1                  -0.1222 0.03222  0.03623 14.40 1  1.5e-04
## hv2                  -0.1080 0.02299  0.03811 22.06 1  2.6e-06
## hv3                  -0.3445 0.08980  0.09121 14.71 1  1.3e-04
## hv4                  -0.7852 0.24843  0.30260  9.99 1  1.6e-03
## ridge(cum_dotBalls) -0.3736 0.04618  0.04772 65.46 1  5.9e-16
##
##                       exp(coef) exp(-coef) lower .95 upper .95
## hv1                      0.8849      1.130    0.8308    0.9426
## hv2                      0.8976      1.114    0.8581    0.9390
## hv3                      0.7086      1.411    0.5943    0.8450
## hv4                      0.4560      2.193    0.2802    0.7421
## ridge(cum_dotBalls)      0.6882      1.453    0.6287    0.7534
##
## Iterations: 3 outer, 14 Newton-Raphson
## Degrees of freedom for terms= 0.9 0.9 0.9 1.0 0.5
## Concordance= 0.982  (se = 0.003 )
## Likelihood ratio test= 114.2  on 4.23 df,   p=<2e-16
```

Test pH assumptions

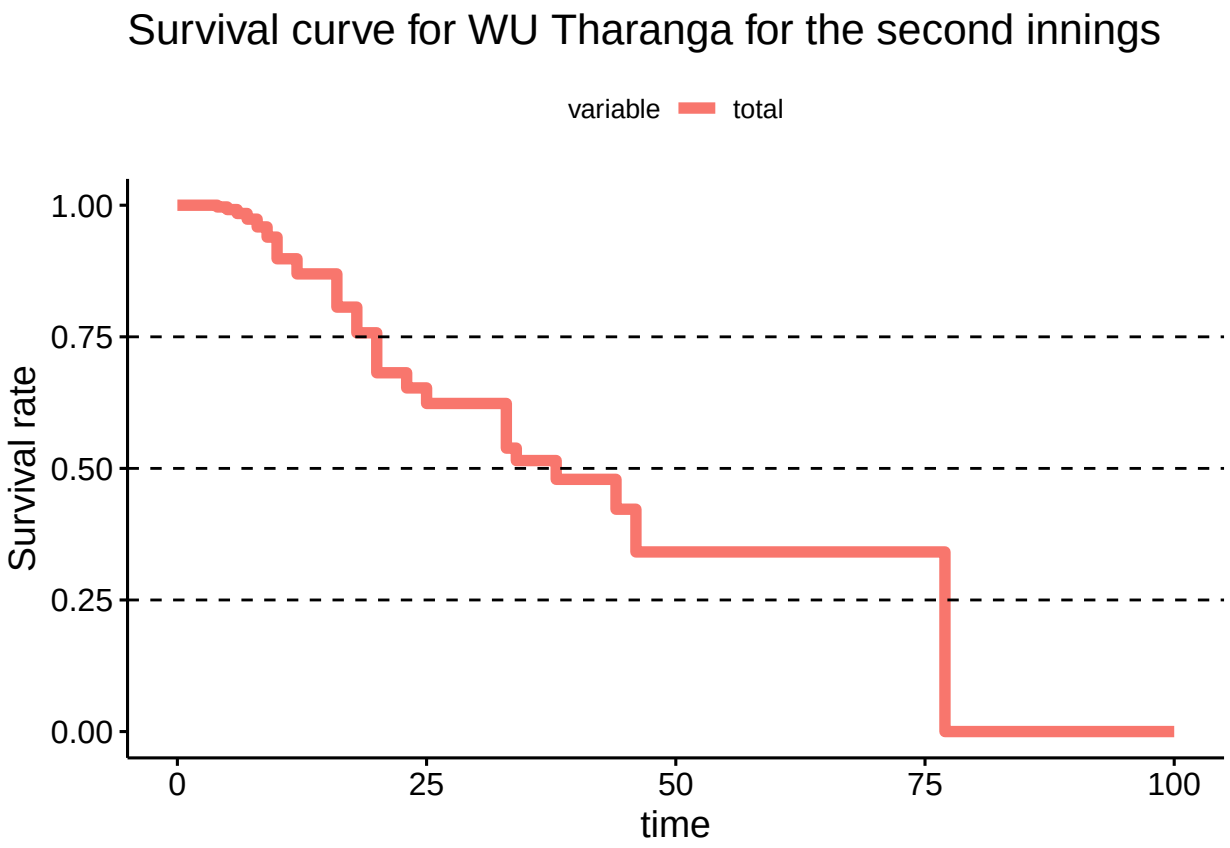
```
test_WUT_2_heav <- cox.zph(cox_WUT_2_heav)
test_WUT_2_heav
```

```
##                         chisq   df    p
## hv1                  1.61e+00 0.94 0.19
## hv2                  1.06e+00 0.91 0.27
## hv3                  2.17e+00 0.91 0.13
## hv4                  4.80e-01 0.99 0.49
## ridge(cum_dotBalls) 4.52e-05 0.48 0.91
## GLOBAL               2.95e+00 4.23 0.60
```

Plot the survival curve

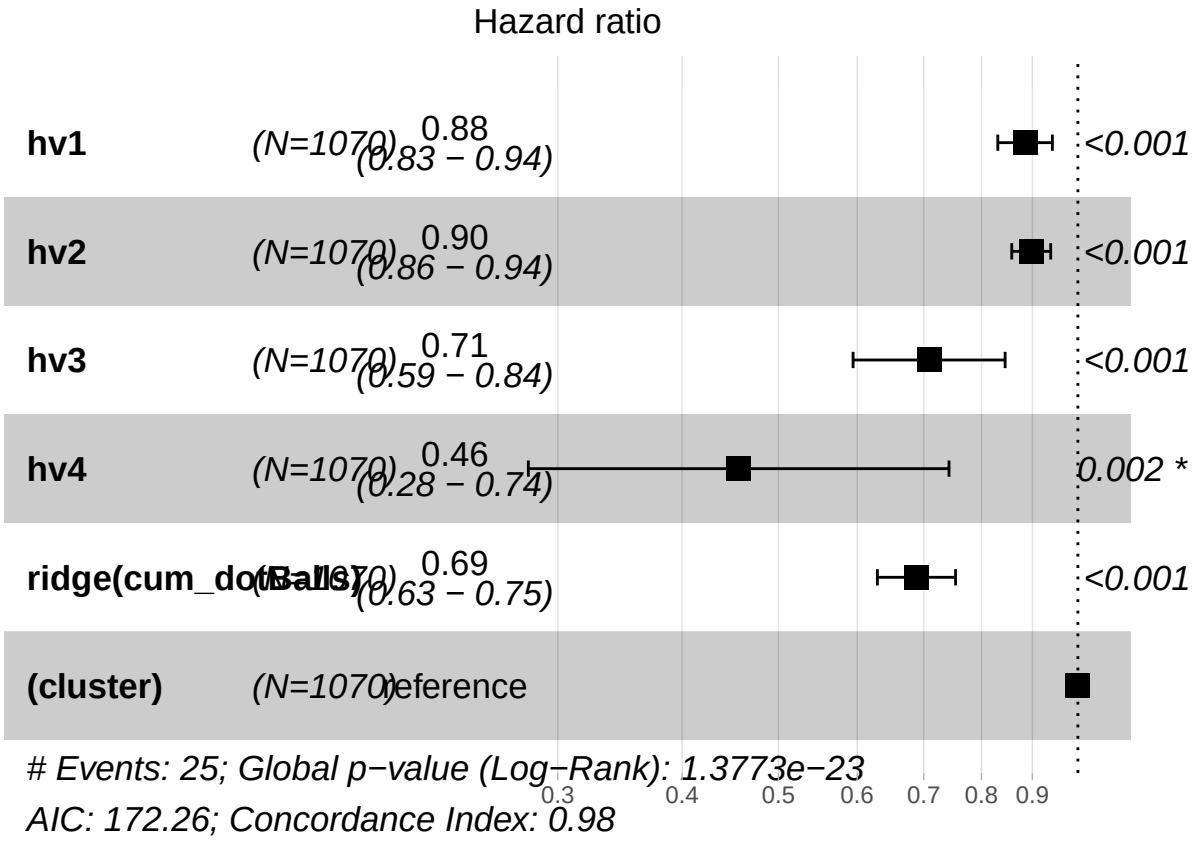
```
cox_plot_WUT_2 <- ggadjustedcurves(cox_WUT_2_heav , data = WUT_2_cut , size = 2,ggtheme = theme_survmin
                  geom_hline(yintercept = c(0.25,0.5,0.75) , linetype = "dashed")  +
                    ggtitle("Survival curve for WU Tharanga for the second innings") +
                    theme(text = element_text(size = 20))

cox_plot_WUT_2
```



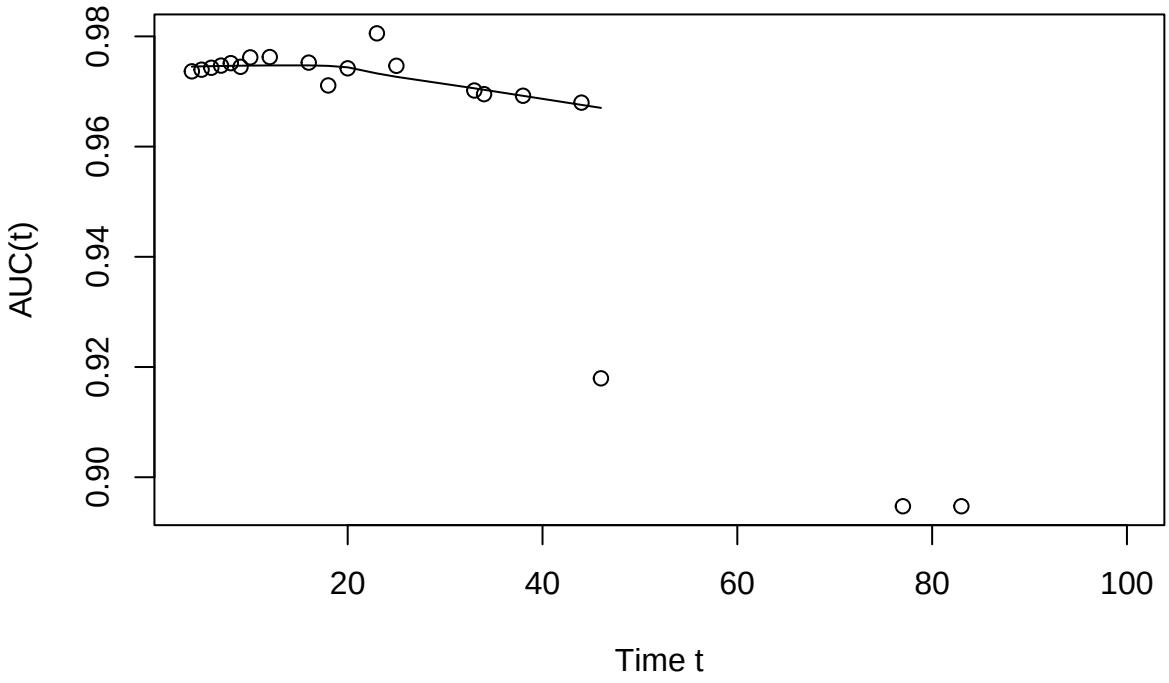
coeffient plot

```
coef_plot_WUT_2 <- ggforest(cox_WUT_2_heav , data = WUT_2_cut,fontsize = 1.2)
coef_plot_WUT_2
```



Area under the curve

```
AUC(Surv(start,cum_balls , wicket) ~
                    hv1 + hv2 +hv3 +hv4 + ridge(cum_dotBalls) + cluster (id) , data = WUT_2_cut)
```




```
## $AUCt
##    time       AUC
## 1     4 0.9736575
## 2     5 0.9739583
## 3     6 0.9743041
## 4     7 0.9746975
## 5     8 0.9751412
## 6     9 0.9744780
## 7    10 0.9761905
## 8    12 0.9762500
## 9    16 0.9752407
## 10   18 0.9710983
## 11   20 0.9742033
## 12   23 0.9805511
## 13   25 0.9746622
## 14   33 0.9701789
## 15   34 0.9695122
## 16   38 0.9692308
## 17   44 0.9679803
## 18   46 0.9179487
## 19   77 0.8947368
## 20   83 0.8947368
## 21  100       NaN
##
## $AUC
## [1] 0.9727967
```

percentage of matches won when the player is in the specific match

```
match_status_WUT_2 <- unique(WUT_2[c("Match_Number","winning_status")])
tot_matches_WUT_2 <- nrow(match_status_WUT_2)
win_matches_WUT_2 <- sum(match_status_WUT_2$winning_status==1)

win_percent_WUT_2 <- (win_matches_WUT_2/tot_matches_WUT_2)*100

win_percent_WUT_2
```

```
## [1] 27.27273
```

Runs contributed

```
match_played_WUT_2 <- data_merge %>% filter(Match_Number %in% unique(WUT_2$Match_Number))
tot_runs_WUT_2 <- match_played_WUT_2 %>% summarise(sum_runs = sum(tot_runs))
WUT_runs_2 <- WUT_2 %>%  summarise(sum_runs = sum(tot_runs))

runs_contrib_percent_WUT_2 <- (WUT_runs_2/tot_runs_WUT_2)*100
runs_contrib_percent_WUT_2
```

```
##   sum_runs
## 1 18.88664
```

Compare different innings

Create the data frame

```
PlayerName_WUT <- "WU Tharanga"
Innings <- c(1,2)
Survival_prob_WUT <- c(0.95671,0.9727967)
winning_percent_WUT <- c(21.42857,27.27273)
runs_contrib_percent_WUT <- c(11.92701 ,18.88664)

comp_innings_WUT <- data.frame(PlayerName_WUT,Innings,Survival_prob_WUT,runs_contrib_percent_WUT,winning
names(comp_innings_WUT) <- c("PlayerName","Innings","Survival_prob","runs_contrib_percent","winning_per
comp_innings_WUT
```

```
##    PlayerName Innings Survival_prob runs_contrib_percent winning_percent
## 1 WU Tharanga       1     0.9567100             11.92701        21.42857
## 2 WU Tharanga       2     0.9727967             18.88664        27.27273
```