Ex.No.8 **Cox Regression**

Date: 26-09-23

Aim

To implement Cox Regression to examine how specified factors influence the rate of a particular event happening at a particular point in time, through R programming.

Procedure

1. To do programming in R, first install “RStudio” and “R” in the system. RStudio is an integrated development environment [IDE] for R and python.
2. Select the File in taskbar →open New file →R script or use shortcut “ctrl+shift+N”
3. Install the ‘survival, survminer’ package and load it in R.
4. Import the built-in dataset ‘lung’
5. Apply the Cox Regression on ‘lung’ dataset.
6. Write the program in the script and save it using the extension R.
7. Run the program by clicking Run option or use the shortcut “ctrl+enter”.
8. See the output in the console tab.

Concepts Involved

- Applying the Cox Regression on a Dataset.

COX REGRESSION

The Cox Proportional-hazard model is regression model commonly used statistical in medical research for investigating the association between the survival time of patients and one or more predictor variables.

The purpose of the model is to evaluate simultaneously the effect of several factors on survival. In other words, it allows us to examine how specified factors influence the rate of a particular event happening (e.g., infection, death) at a particular point of time. This rate is commonly referred as hazard rate. Predictor Variables (or factors) are usually termed as ‘covariates’ in the survival-analysis literature.

The Cox model is expressed by the *hazard function* denoted by h(t). Briefly, the hazard function can be
interpreted as the risk of dying at time t. It can be estimated as follow:
h(t)=h0(t)×exp(b1x1+b2x2+...+bpxp)h(t)=h0(t)×exp(b1x1+b2x2+...+bpxp)
where,
 t represents the survival time
 h(t)h(t) is the hazard function determined by a set of p covariates (x1,x2,...,xpx1,x2,...,xp)
 the coefficients (b1,b2,...,bpb1,b2,...,bp) measure the impact (i.e., the effect size) of covariates.
 the term h0h0 is called the baseline hazard. It corresponds to the value of the hazard if all the xixi are equal to zero (the quantity exp(0) equals 1). The ‘t’ in h(t) reminds us that the hazard may vary over time.

Script

```
install.packages(c("survival", "survminer"))

library("survival")

library("survminer")

coxph(formula, data, method)


data("lung")

head(lung)
```

Output

```
> head(lung)
  inst time status age sex ph.ecog ph.karno pat.karno meal.cal wt.loss
1    3  306      2  74   1       1       90       100     1175      NA
2    3  455      2  68   1       0       90        90     1225      15
3    3 1010      1  56   1       0       90        90       NA      15
4    5  210      2  57   1       1       90        60     1150      11
5    1  883      2  60   1       0      100        90       NA       0
6   12 1022      1  74   1       1       50        80      513       0
```

Script

```
res.cox <- coxph(Surv(time, status) ~ sex, data = lung)

res.cox
```

Output

```
Call:
coxph(formula = Surv(time, status) ~ sex, data = lung)

       coef exp(coef) se(coef)      z       p
sex -0.5310    0.5880   0.1672 -3.176 0.00149

Likelihood ratio test=10.63  on 1 df, p=0.001111
n= 228, number of events= 165
```

Script

```
summary(res.cox)
```

Output

```
Call:
coxph(formula = Surv(time, status) ~ sex, data = lung)

  n= 228, number of events= 165

       coef exp(coef) se(coef)      z Pr(>|z|)
sex -0.5310    0.5880   0.1672 -3.176  0.00149 **
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

    exp(coef) exp(-coef) lower .95 upper .95
sex     0.588      1.701    0.4237     0.816

Concordance= 0.579  (se = 0.021 )
Likelihood ratio test= 10.63  on 1 df,   p=0.001
Wald test            = 10.09  on 1 df,   p=0.001
Score (logrank) test = 10.33  on 1 df,   p=0.001
```

Script

```
covariates <- c("age", "sex", "ph.karno", "ph.ecog", "wt.loss")

univ_formulas <- sapply(covariates,

                 function(x) as.formula(paste('Surv(time, status)~', x)))

univ_models <- lapply( univ_formulas, function(x){coxph(x, data = lung)})

# Extract data

univ_results <- lapply(univ_models,

                function(x){

                  x <- summary(x)

                  p.value<-signif(x$wald["pvalue"], digits=2)

                  wald.test<-signif(x$wald["test"], digits=2)

                  beta<-signif(x$coef[1], digits=2);#coeficient beta

                  HR <-signif(x$coef[2], digits=2);#exp(beta)

                  HR.confint.lower <- signif(x$conf.int[,"lower .95"], 2)

                  HR.confint.upper <- signif(x$conf.int[,"upper .95"],2)

                  HR <- paste0(HR, " (",

                         HR.confint.lower, "-", HR.confint.upper, ")")

                  res<-c(beta, HR, wald.test, p.value)

                  names(res)<-c("beta", "HR (95% CI for HR)", "wald.test",

                          "p.value")

                  return(res)

                  #return(exp(cbind(coef(x),confint(x))))

                })
```

```
res <- t(as.data.frame(univ_results, check.names = FALSE))

as.data.frame(res)
```

Output

```
           beta HR (95% CI for HR) wald.test p.value
age       0.019            1 (1-1)       4.1   0.042
sex       -0.53   0.59 (0.42-0.82)        10  0.0015
ph.karno -0.016      0.98 (0.97-1)       7.9   0.005
ph.ecog    0.48        1.6 (1.3-2)        18 2.7e-05
wt.loss  0.0013         1 (0.99-1)      0.05    0.83
```

Script

```
res.cox <- coxph(Surv(time, status) ~ age + sex + ph.ecog, data = lung)

summary(res.cox)
```

Output

```
Call:
coxph(formula = Surv(time, status) ~ age + sex + ph.ecog, data = lung)

  n= 227, number of events= 164
   (1 observation deleted due to missingness)

             coef exp(coef)  se(coef)      z Pr(>|z|)
age      0.011067  1.011128  0.009267  1.194 0.232416
sex     -0.552612  0.575445  0.167739 -3.294 0.000986 ***
ph.ecog  0.463728  1.589991  0.113577  4.083 4.45e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

        exp(coef) exp(-coef) lower .95 upper .95
age        1.0111     0.9890    0.9929    1.0297
sex        0.5754     1.7378    0.4142    0.7994
ph.ecog    1.5900     0.6289    1.2727    1.9864

Concordance= 0.637  (se = 0.025 )
Likelihood ratio test= 30.5  on 3 df,   p=1e-06
Wald test            = 29.93  on 3 df,   p=1e-06
Score (logrank) test = 30.5  on 3 df,   p=1e-06
```

Script

```
# Plot the baseline survival function

ggsurvplot(survfit(res.cox, data = lung), palette = "#2E9FDF",ggtheme = theme_minimal())

# Create the new data

sex_df <- with(lung,

          data.frame(sex = c(1, 2),

                    age = rep(mean(age, na.rm = TRUE), 2),

                    ph.ecog = c(1, 1)))
```

sex_df

Output

```
  sex      age ph.ecog
1   1 62.44737       1
2   2 62.44737       1
```

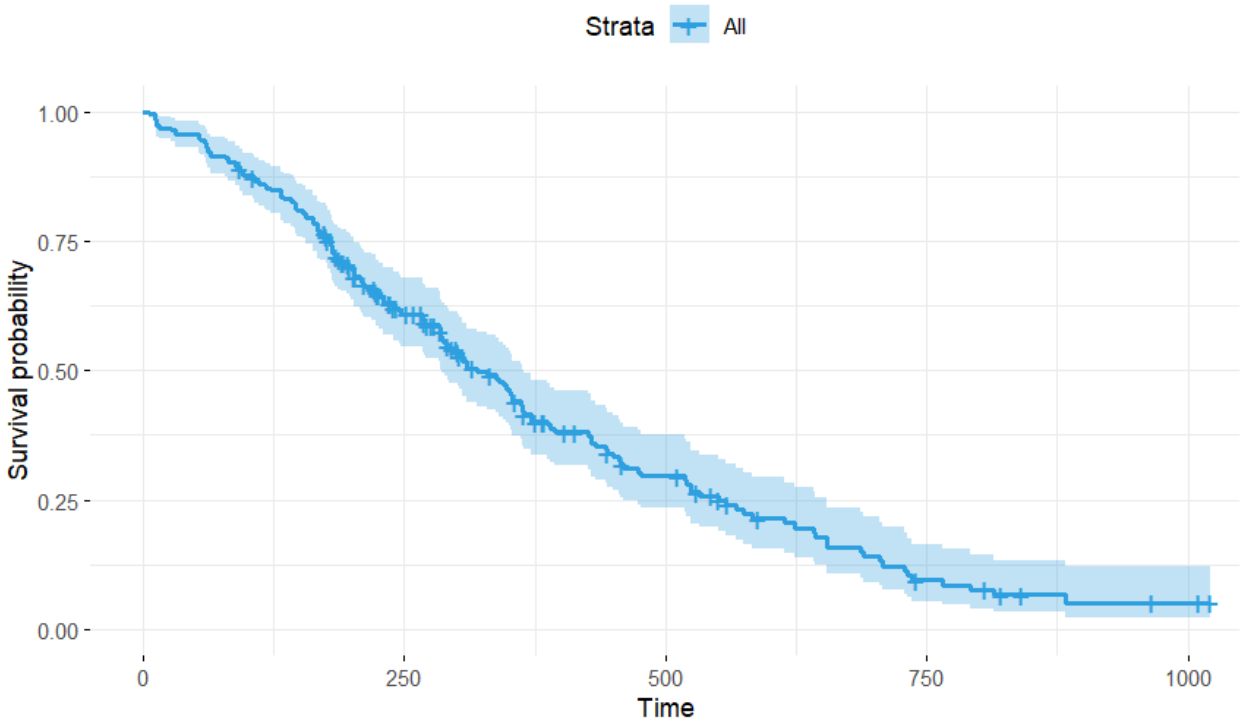
Script

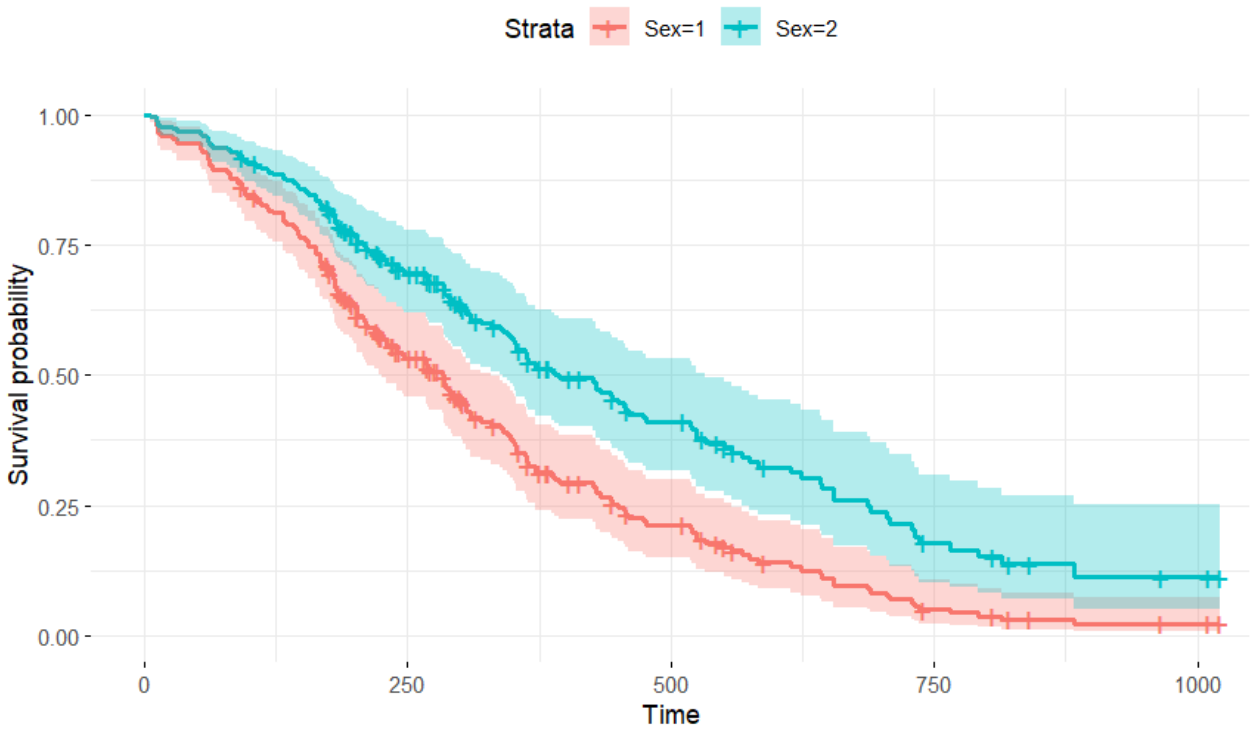
```
fit <- survfit(res.cox, newdata = sex_df)

ggsurvplot(fit,data=sex_df, conf.int = TRUE, legend.labs=c("Sex=1", "Sex=2"),

        ggtheme = theme_minimal())
```

Output





Result

Thus the Cox Regression is successfully implemented in R programming.