CHL 5209 Assignment 4

Belina Jang

April 2, 2025

Question 1

(a)

Model 1: `model1 <- coxph(Surv(time,status)~ transplant, data=dat)`

$$\lambda_i(t) = \lambda_0(t)\exp(\beta_1 \cdot \text{transplant}_i)$$

transplant_i is a binary variable indicating whether individual i received a transplant or not.

Model 2: `model2 <- coxph(Surv(time,status) ~ tt(wait), tt=function(x,t, ...) I(x<t), data=dat)`

$$\lambda_i(t) = \lambda_0(t)\exp(\beta_1 \cdot I(t > \text{wait}_i))$$

I prefer the model 2, even though the treatment effect was not found to be significant, it is more appropriate due to immortal time bias.

In model 1, individuals were classified based on whether they have received a transplant or not. Those who received transplant had survived long enough to receive the transplant in the first place. This results in immortal time bias, which gives the illusion of better survival for those who received transplant. In model 2, this issue was corrected by using a time-dependent covariate. Here, the effect of transplant is only considered after the transplant has been received.

(b)

Create a long format dataset that could be used to fit model 2. Show your dataset. Note that two patients (IDs 3 and 6) received the transplant on the same day as they entered the study; don't modify the times, instead you can remove the rows with zero follow-up duration from your dataset. Verify the above results by fitting model 2 to your long format dataset using coxph.

```r
library(survival)
library(dplyr)

q1data <- data.frame(
  id=1:10,
  sex=c(1,1,1,1,1,0,0,0,0,0),
  time=c(3,5,5,6,8,4,7,8,9,10),
  status=c(1,0,1,1,0,0,1,0,1,0),
  transplant=c(0,0,1,0,1,1,1,0,1,1),
  wait=c(NA,NA,0,NA,6,0,5,NA,5,3)
)

q1data %>% kableExtra::kable("html", caption = "Given data")
```

Table 1: Given data

id	sex	time	status	transplant	wait
1	1	3	1	0	NA
2	1	5	0	0	NA
3	1	5	1	1	0
4	1	6	1	0	NA
5	1	8	0	1	6
6	0	4	0	1	0
7	0	7	1	1	5
8	0	8	0	0	NA
9	0	9	1	1	5
10	0	10	0	1	3

```r
# long format dataset: done manually
q1data_long <- data.frame(
  id=c(1,2,3,4,5,5,6,7,7,8,9,9,10,10),
  start=c(0,0,0,0,0,6,0,0,5,0,0,5,0,3),
  stop=c(3,5,5,6,6,8,4,5,7,8,5,9,3,10),
  event=c(1,0,1,1,0,0,0,0,1,0,0,1,0,0),
  transplant_status=c(0,0,1,0,0,1,1,0,1,0,0,1,0,1)
)

q1data_long %>% kableExtra::kable("html", caption = "Long format dataset")
```

Table 2: Long format dataset

id	start	stop	event	transplant_status
1	0	3	1	0
2	0	5	0	0
3	0	5	1	1
4	0	6	1	0
5	0	6	0	0
5	6	8	0	1
6	0	4	0	1
7	0	5	0	0
7	5	7	1	1
8	0	8	0	0
9	0	5	0	0
9	5	9	1	1
10	0	3	0	0
10	3	10	0	1

```
# Fit the model again
cox_td <- coxph(Surv(start, stop, event) ~ transplant_status, data =
 ↪  q1data_long)
summary(cox_td)
```

```
Call:
coxph(formula = Surv(start, stop, event) ~ transplant_status,
    data = q1data_long)

  n= 14, number of events= 5

                    coef exp(coef) se(coef)     z Pr(>|z|)
transplant_status 0.3246    1.3835   1.1322 0.287    0.774

                  exp(coef) exp(-coef) lower .95 upper .95
transplant_status     1.383     0.7228    0.1504     12.73
```

```
Concordance= 0.538  (se = 0.126 )
Likelihood ratio test= 0.08  on 1 df,   p=0.8
Wald test            = 0.08  on 1 df,   p=0.8
Score (logrank) test = 0.08  on 1 df,   p=0.8
```

(c)

Write the Cox partial likelihood function for the parameters in model 2 and an R function returning the value of the partial log-likelihood at given parameter value(s). Use the R optim function to verify the above results (maximum likelihood estimates and their standard errors).

Cox partial likelihood function for the parameters in model 2:

$$L(\beta) = \prod_{i=1}^{n} \left(\frac{\exp(\beta x_i(t_i))}{\sum_{I=1} Y_I(t_i)\exp(\beta x_I(t_i))} \right)^{e_i}$$

Which follows:

$$l(\beta) = \sum_{i:e_i=1} (\beta x_i(t_i) - \log(\sum_{I=1} e^{\beta x_I(t_i)}))$$

where $x_i(t_i)$ is the value of the time-dependent covariate at event time t_i, $\mathrm{R}(t_i)$ is the risk set at time t_i and δ_i is the event indicator at time t_i.

```r
# partial log-likelihood function
loglike <- function(beta, data) {
  result <- 0
  for (i in 1:nrow(data)) {
    if (data$event[i] == 1) {
      ti <- data$stop[i]
      xi <- data$transplant_status[i]
      # currently at risk
      risk <- data$start < ti & data$stop >= ti
      xj <- data$transplant_status[risk]
      result <- result + (beta * xi) - log(sum(exp(beta * xj)))
    }
  }
  return(-result)
}

# use optim to estimate beta
fit <- optim(par=0, fn=loglike, data=q1data_long, hessian=TRUE)
```

```r
beta_hat <- fit$par
se_beta <- sqrt(1 / fit$hessian)

cat("MLE (beta):", round(beta_hat, 4), "\n Standard error:", round(se_beta,
 ↪  4), "\n")
```

```
MLE (beta): 0.3242
 Standard error: 1.1322
```

Question 2

Fit a Cox model for mortality in the brain dataset including the treatment arm indicator and all the prognostic factors. Report the results in such a form that someone could use the model for prognostic purposes, to calculate the six month absolute risk of death for a new patient.

```r
brain <- read.csv("brain.csv")
brain <- na.omit(brain) # complete cases analysis

# fit the Cox model
cox_model <- coxph(Surv(weeks, event) ~ treat + resect75 + age + interval +
 ↪  karn + race + local + male + nitro + as.factor(path) + grade, data =
 ↪  brain)

summary(cox_model)
```

```
Call:
coxph(formula = Surv(weeks, event) ~ treat + resect75 + age +
    interval + karn + race + local + male + nitro + as.factor(path) +
    grade, data = brain)

  n= 221, number of events= 206

                      coef exp(coef)  se(coef)      z Pr(>|z|)
treat            -0.396796  0.672471  0.144512 -2.746 0.006037 **
resect75         -0.443613  0.641714  0.164990 -2.689 0.007172 **
age               0.017294  1.017445  0.006029  2.868 0.004127 **
interval         -0.139448  0.869838  0.047318 -2.947 0.003208 **
karn             -0.376704  0.686119  0.160759 -2.343 0.019115 *
race              0.592330  1.808197  0.270284  2.192 0.028415 *
local            -0.466002  0.627506  0.176343 -2.643 0.008228 **
male             -0.239337  0.787150  0.153227 -1.562 0.118294
```

```
nitro             0.480875  1.617490  0.154996  3.102 0.001919 **
as.factor(path)2 -0.640822  0.526859  0.217740 -2.943 0.003250 **
as.factor(path)3 -0.804242  0.447427  0.223189 -3.603 0.000314 ***
as.factor(path)4 -0.623663  0.535977  0.429548 -1.452 0.146528
grade            -0.857132  0.424377  0.293056 -2.925 0.003447 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                 exp(coef) exp(-coef) lower .95 upper .95
treat               0.6725     1.4871    0.5066    0.8927
resect75            0.6417     1.5583    0.4644    0.8867
age                 1.0174     0.9829    1.0055    1.0295
interval            0.8698     1.1496    0.7928    0.9544
karn                0.6861     1.4575    0.5007    0.9402
race                1.8082     0.5530    1.0646    3.0712
local               0.6275     1.5936    0.4441    0.8866
male                0.7871     1.2704    0.5829    1.0629
nitro               1.6175     0.6182    1.1937    2.1917
as.factor(path)2    0.5269     1.8980    0.3438    0.8073
as.factor(path)3    0.4474     2.2350    0.2889    0.6929
as.factor(path)4    0.5360     1.8658    0.2309    1.2439
grade               0.4244     2.3564    0.2389    0.7537

Concordance= 0.705  (se = 0.019 )
Likelihood ratio test= 104.3  on 13 df,   p=2e-16
Wald test            = 94.59  on 13 df,   p=2e-14
Score (logrank) test = 99.52  on 13 df,   p=2e-15
```

The 6 months absolute risk of death for a new patient can be calculated using the equations below:

$$Risk_i(t) = 1 - S_i(t)$$

where $S_i(t)$ is the survival probability at time t for patient i.

$$S_i(t) = \exp\ (-L_0(t)\exp(\beta' x_i))$$

where $L_0(t)$ is the baseline cumulative hazard at time t.

The linear predictor for patient i is given by:

$$\begin{aligned}\hat{\beta}' x_i = & -0.374 \cdot \text{treat} - 0.448 \cdot \text{resect75} + 0.0185 \cdot \text{age} - 0.125 \cdot \text{interval} \\ & - 0.376 \cdot \text{karn} + 0.553 \cdot \text{race} - 0.456 \cdot \text{local} - 0.221 \cdot \text{male} \\ & + 0.457 \cdot \text{nitro} - 0.376 \cdot \text{path} - 0.811 \cdot \text{grade}\end{aligned}$$

Based on the results, calculate the model-based six month survival probability and risk of death for a 70-year old white male receiving standard of care with local radiation, with less than 75% resection in the new operation one year after the previous operation, having Karnofsky score of more than 70, no previous exposure to nitrosoureas, glioblastoma pathology and inactive grade.

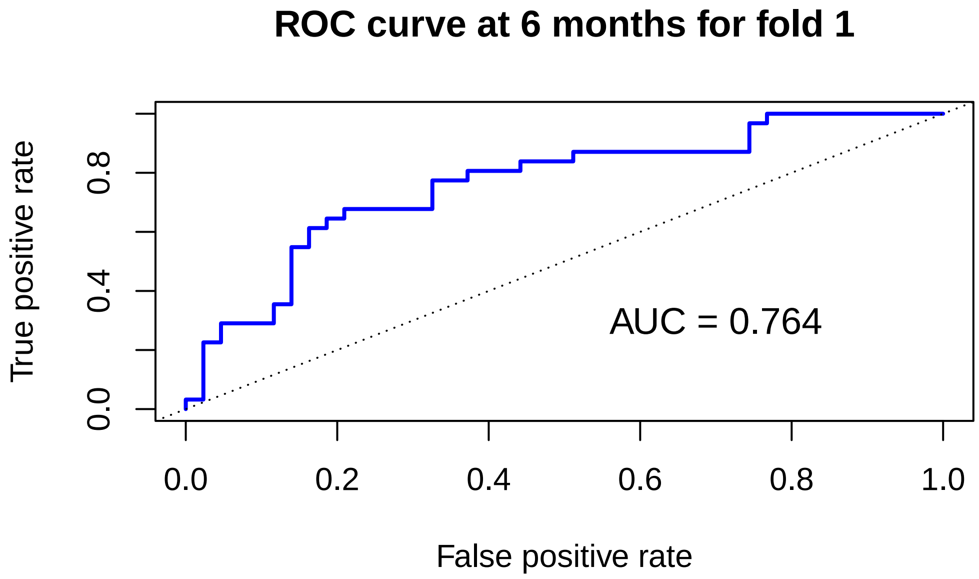
```r
# new patient info
new_patient <- data.frame(
  treat = 0, # standard of care
  resect75 = 0, # <75% resection
  age = 70,
  interval = 1.0, # one year after the previous operation
  karn = 1, # Karnofsky performance score > 70
  race = 1, # white
  local = 1, # local radiation
  male = 1,
  nitro = 0, # no previous exposure
  path = 1, # Gliobastoma
  grade = 0 # inactive
)

# Calculate 6 month absolute risks:
s <- 6 * 52/12 # 6 months in weeks
L0 <- basehaz(cox_model, centered = TRUE)
L0s <- L0$hazard[findInterval(s, L0$time)]

# lp
lp <- predict(cox_model, newdata = new_patient, type = "lp")
survival_prob <- exp(-L0s * exp(lp))
risk <- 1.0 - exp(-L0s * exp(lp))

cat("Model-based six month survival probability:", round(survival_prob, 4),
 ↪  "\nsix month risk of death:", round(risk, 4), "\n")
```

```
Model-based six month survival probability: 0.3134
six month risk of death: 0.6866
```

Question 3.

Check the discrimination and calibration of the model you fitted in Q2 (choose one statistic and one graphical presentation to describe each).

```r
# Discrimination
library(survivalROC)
brain <- read.csv("brain.csv")
brain <- na.omit(brain) # complete cases analysis

set.seed(0)

# 3 fold cross validation
k <- 3
folds <- sample(rep(1:k, length.out = nrow(brain)))
aucs_cv <- numeric(k)

# save predictions for calibration
brain$cv_pred <- NA

for (i in 1:k){
  test <- brain[folds == i,]
  train <- brain[folds != i,]

  # fit the Cox model
  cox_model <- coxph(Surv(weeks, event) ~ treat + resect75 + age + interval +
↪   karn + race + local + male + nitro + as.factor(path) + grade, data =
↪   train)
  s <- 6 * 52/12 # 6 months in weeks

  L0 <- basehaz(cox_model, centered = TRUE)
  L0s <- L0$hazard[findInterval(s, L0$time)]
  # new lp
  lp <- predict(cox_model, newdata = test, type = "lp")
  risk <- 1 - exp(-L0s * exp(lp)) # update risk

  roc <- survivalROC(Stime = test$weeks,
                     status = test$event,
                     marker = risk,
                     predict.time = s,
                     method = "KM")
  aucs_cv[i] <- roc$AUC
  brain$cv_pred[folds == i] <- risk

  if (i==1) {
    plot(roc$FP, roc$TP, type = "s", col = "blue", lwd = 2,
         xlab = "False positive rate", ylab = "True positive rate",
         main = paste0("ROC curve at 6 months for fold ", i))
    abline(a = 0, b = 1, lty = "dotted")
    text(0.7, 0.3, labels = paste("AUC =", round(roc$AUC, 3)), cex = 1.2)
  }
}
```



```
# mean AUC
cat("Mean 3 fold cross-validated AUC:", mean(aucs_cv), "\n")
```

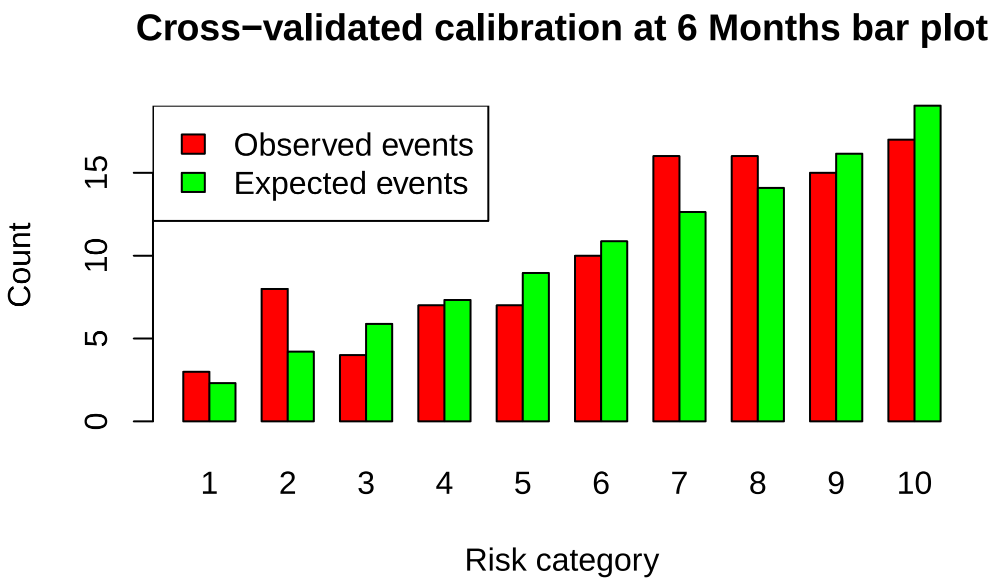
```
Mean 3 fold cross-validated AUC: 0.7505099
```

Discrimination measures how well the model discriminate between those who will experience the event and those who will not.

From the ROC curve at 6 months, the mean 3 fold cross validated AUC (area under the ROC curve) was 0.751, which indicates that the model has good discrimination. This means that the model can distinguish between patients who will die within 6 months and those who will not.

```
risk <- brain$cv_pred

# Check calibration:
deciles <- quantile(risk, probs=seq(0.1,0.9,by=0.1))
```

```
dcat <- cut(risk, c(0,deciles,1), labels=1:10)
n <- table(dcat)
p <- r <- rep(NA, 10)

for (k in 1:10) {
    data <- subset(brain, dcat==k)
    fit <- survfit(Surv(weeks, event) ~ 1, data=data)
    p[k] <- mean(risk[dcat==k])
    r[k] <- (1 - fit$surv[findInterval(s, fit$time)])
}

o <- n * r
e <- n * p

barplot(rbind(o, e), beside=TRUE, col=c('red','green'), names.arg=1:10,
 ↪  xlab='Risk category', ylab='Count', main='Cross-validated calibration at
 ↪  6 Months bar plot')
legend('topleft', legend=c('Observed events','Expected events'),
 ↪  fill=c('red','green'))
```



```
# H-L test p-value:
```

```r
chisq <- sum((o - e)^2/(n * p * (1 - p)))
pval <- pchisq(chisq, df=10-2, lower.tail=FALSE)
cat("Cross-validated H-L test p-value:", round(pval, 4))
```

```
Cross-validated H-L test p-value: 0.205
```

Calibration measures how well the predicted risks match with the observed absolute risks in different subgroups.

The cross-validated calibration at 6 month bar plot shows that the predicted risks are close to the observed risks in overall with a small variation, which indicates that the model is likely well-calibrated. Additionally, the cross-validated Hosmer-Lemeshow test p-value was 0.205, which indicates that there's no significant difference between the observed and expected number of deaths. These observations indicate that the model is well-calibrated.

Question 4

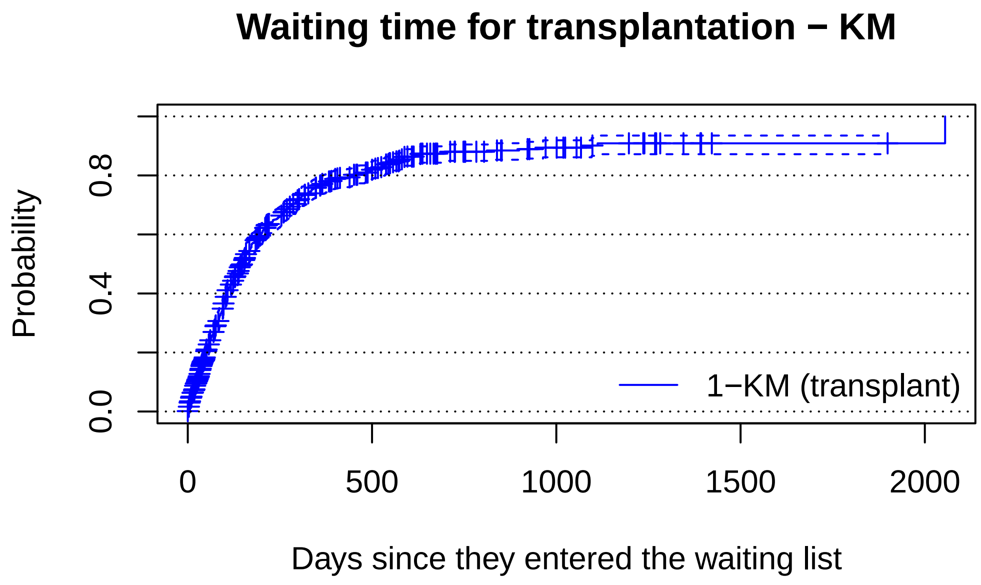
From the transplant data, estimate (i) the probability of receiving transplant using the Kaplan-Meier method and (ii) the cumulative incidence of receiving transplant using the non-parametric cumulative incidence estimator. Present the results as curves over time. Which method would you prefer and why?

```r
library(survival)
library(cmprsk)
data(transplant)
head(transplant)
```

```
  age sex abo year futime    event
1  47   m   B 1994   1197    death
2  55   m   A 1991     28      ltx
3  52   m   B 1996     85      ltx
4  40   f   O 1995    231      ltx
5  70   m   O 1996   1271 censored
6  66   f   O 1996     58      ltx
```

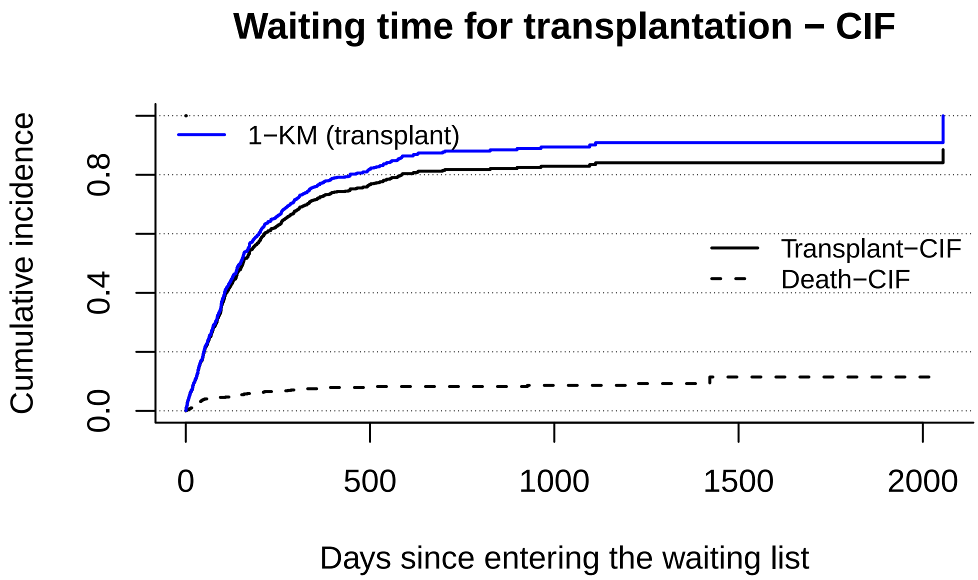
```r
transplant <- na.omit(transplant) # complete cases analysis

# create event_code
# 0 = censored/withdraw
# 1 = transplant (ltx)
# 2 = death (competing risk)
```

```
transplant$event_code <- ifelse(transplant$event == "ltx", 1,
                           ifelse(transplant$event == "death", 2, 0))

# (i) Kaplan-Meier estimate (treat event_code0&2 as censored)
fit_km <- survfit(Surv(futime, event_code==1) ~ 1, data = transplant)

plot(fit_km, col = "blue", xlab = "Days since they entered the waiting list",
 ↪  ylim=c(0,1), ylab = "Probability", lwd=1, conf.int=T, mark.time=TRUE,
     main = "Waiting time for transplantation - KM", fun="event")
legend("bottomright", "1-KM (transplant)", col = "blue", lwd = 1, bty = "n")
abline(h=seq(0,1,0.2), lty='dotted')
```



```
# (ii) Cumulative incidence function

# using cmprsk package
cifit <- cuminc(transplant$futime, transplant$event_code, cencode=0)

plot(cifit, curvlab=c('Transplant-CIF','Death-CIF'), xlab="Days since
 ↪  entering the waiting list", ylab="Cumulative incidence", lwd=1.5,
 ↪  ylim=c(0,1), cex = 0.00001)
legend("right", legend=c("Transplant-CIF","Death-CIF"), lwd=1.5, lty=c(1,2),
 ↪  bty='n', cex=0.8)
```

```
abline(h=seq(0,1,0.2), lty='dotted',lwd=0.5)
lines(fit_km, lwd=1.5, col="blue", conf.int=F, fun="event")
legend("topleft", col="blue", legend=c("1-KM (transplant)"), lwd=1.5,
 ↪  bty='n', cex=0.8)
title("Waiting time for transplantation - CIF")
```

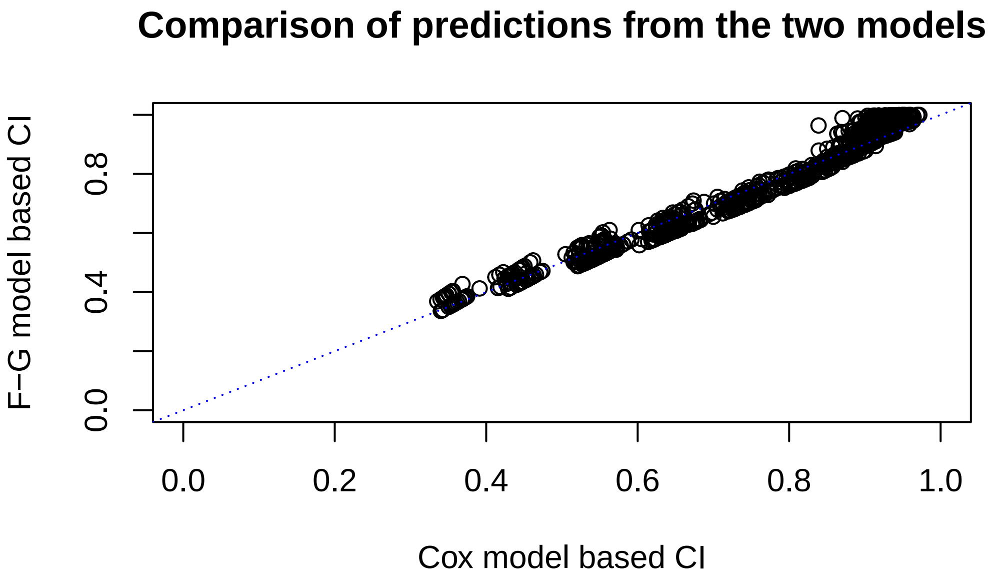


I would prefer the non-parametric cumulative incidence estimator over the Kaplan-Meier (KM) method for estimating the probability of receiving transplant. The KM method treats all other outcomes other than treatment, such as death or withdrawal, as non-informative sensoring, which may not be appropriate in the presence of competing risks. In contrast, the non-parametric cumulative incidence estimator allows us to set death as a competing event (risk), which is more realistic. Therefore, it provides a more reliable estimate for the probability of receiving transplant over time.

Question 5

Use fitted Cox-type (cause-specific hazard) and Fine & Gray (subdistribution hazard) models adjusted for age, sex, ABO blood group and year to calculate individual-level one-year cumulative incidences of receiving the transplant. Present these in a scatterplot. How do the results compare between the models? (You don't have to validate the models, just comment on whether the cause-specific hazard and subdistribution hazard model results are "similar".)

```
transplant$abo <- as.factor(transplant$abo)
transplant$event_code <- as.factor(transplant$event_code)
transplant$id <- 1:nrow(transplant)
transplant$age_c <- transplant$age - mean(transplant$age)
transplant$year_c <- transplant$year - mean(transplant$year)

# Comparison of predictions: calculate 12-month cumulative incidences of
 ↪  receiving transplant for everyone in the dataset:
cox_model <- coxph(Surv(futime, factor(event_code)) ~ age + sex + abo + year,
                   id = id, data = transplant)
ciall <- rep(NA, nrow(transplant))
fgciall <- rep(NA, nrow(transplant))
s <- 365

cifit <- survfit(cox_model, newdata=transplant)
idx <- findInterval(s, cifit$time)

ciall <- NULL
for (i in 1:nrow(transplant)) {
    ciall <- c(ciall, cifit$pstate[idx,i,2])
}

mm <- model.matrix(~ age_c + sex + abo + year_c, data = transplant)[, -1]

fgmodel <- crr(transplant$futime, transplant$event_code, cov1=mm, failcode=1,
 ↪  cencode=0)

fgci <- predict(fgmodel, cov1=mm)
fgciall <- fgci[findInterval(s, fgci[,1]),2:ncol(fgci)]

plot(ciall, fgciall, xlab='Cox model based CI', ylab='F-G model based CI',
 ↪  xlim=c(0,1),ylim=c(0,1),
     main='Comparison of predictions from the two models')
abline(a=0, b=1, lty='dotted', col='blue')
```



The scatterplot shows that the predictions from the two models are close to the line of equality (dotted line). This indicates that the results from the cause-specific hazard (Cox-type) and subdistribution hazard (Fine & Gray) models are similar.