Draft

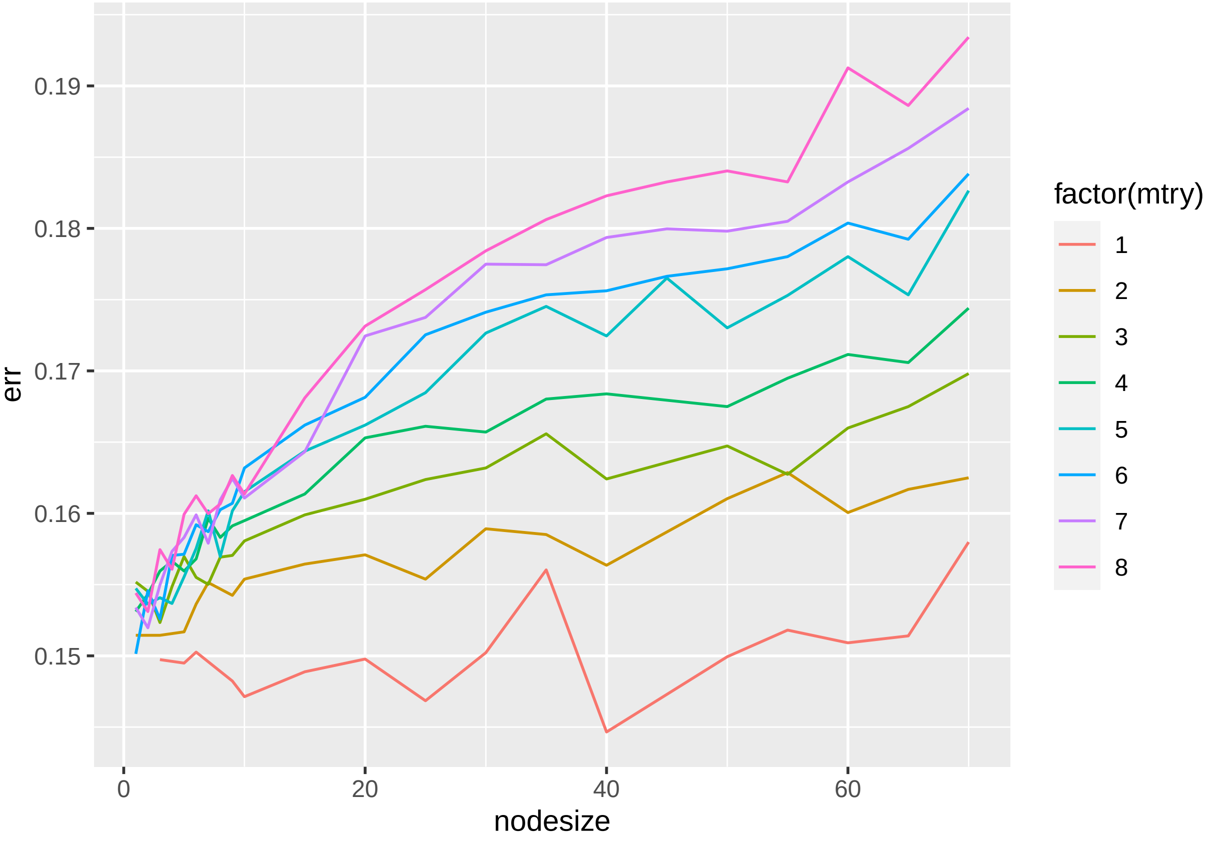
Nancy Liu

5/13/2022

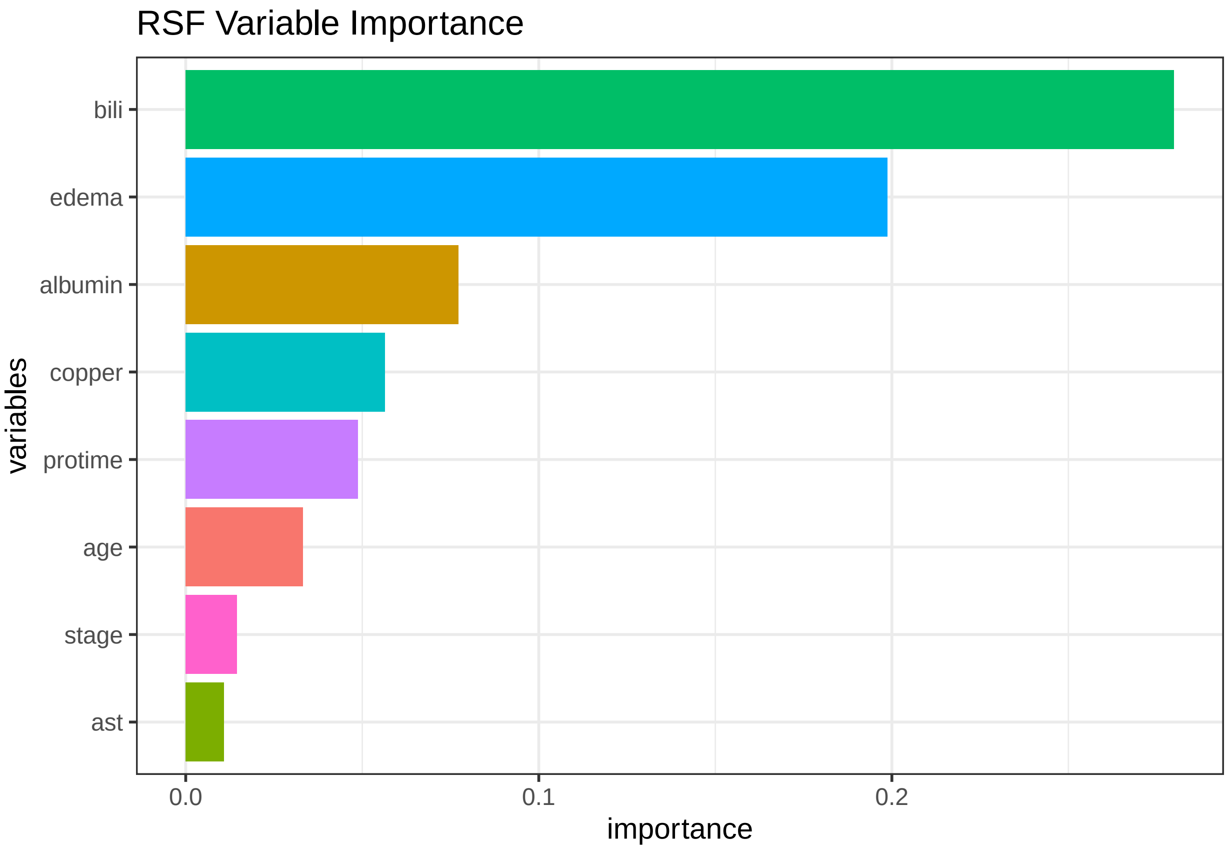
```
####### code!

pbc_use <- pbc %>%
  mutate(status = as.integer(ifelse(status == 2, 1, 0))) %>%
  select(-id)

pbc_use$trt <- as.factor(pbc_use$trt)

pbc_use$ascites <- as.factor(pbc_use$ascites)

pbc_use$hepato <- as.factor(pbc_use$hepato)

pbc_use$spiders <- as.factor(pbc_use$spiders)


set.seed(1)
train_ind_pbc <- sample(1:nrow(pbc_use), nrow(pbc_use)*0.75)
train_pbc <- pbc_use[train_ind_pbc,]
test_pbc <- pbc_use[-train_ind_pbc,]

# random forest
pbc_rf <- rfsrc(Surv(time, status) ~ age + edema + bili + albumin +
  copper + ast + protime + stage + age:edema + age:copper +
  bili:ast, data = pbc_use)


# We can find the optimal mtry and nodesize using OOB
tuning <- tune(formula = Surv(time, status) ~ age + edema + bili + albumin + copper +
       ast + protime + stage + age:edema + age:copper + bili:ast,
     data = pbc_use,
     mtryStart = ncol(pbc_use) / 2,
     nodesizeTry = c(c(1:9), seq(10, 100, by = 5)),
     ntreeTry =500,
     doBest = TRUE)

res <- as.data.frame(tuning$results)
# print optimal nodesize and mtry
tuning$optimal # wtf lmao
```

```
## nodesize     mtry
##       40        1
```

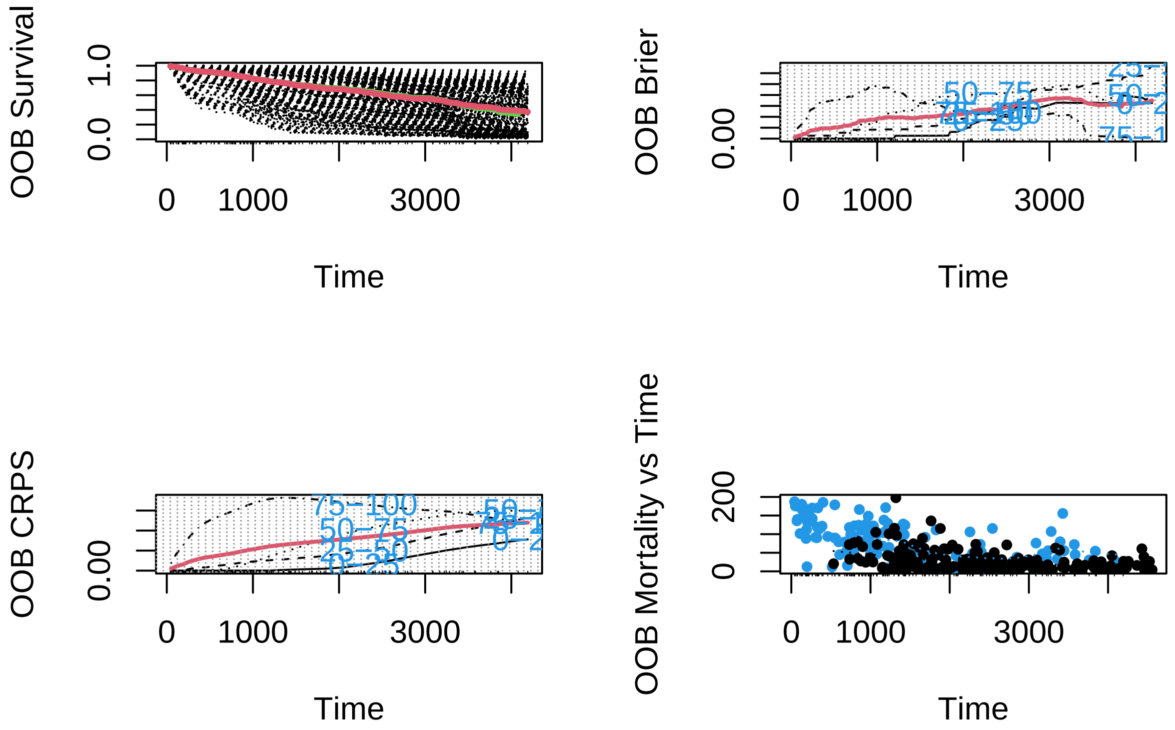
```
node_size <- c(seq(5, 100, by = 5))

ggplot(res) +
  geom_line(aes(x = nodesize, y = err, colour = factor(mtry)))
```



```
# variable importance
rf_imp_pbc <- data.frame(importance = vimp(pbc_rf)$importance %>% sort(decreasing = T))
rf_imp_pbc$variables <- rownames(rf_imp_pbc)

ggplot(rf_imp_pbc, aes(x = reorder(variables, importance, decreasing = TRUE),
                        y = importance,
                        fill = variables)) +
  geom_col() +
  coord_flip() +
  labs(x = "variables", title = "RSF Variable Importance") + theme_bw() +
  theme(legend.position = "none")
```

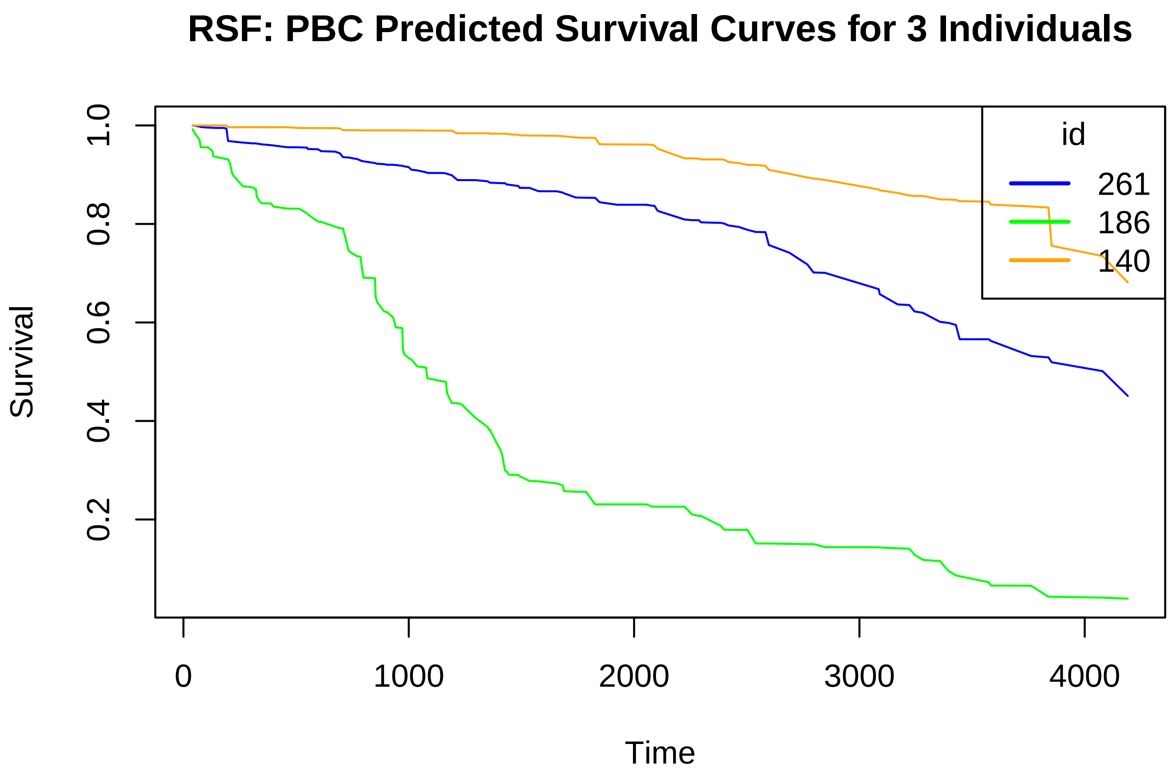


```
plot.survival.rfsrc(pbc_rf)
```



```
########################## CIF CODE START ##########################
pbc_cif <- pecCforest(Surv(time, status) ~ age + edema + bili + albumin +
  copper + ast + protime + stage + age:edema + age:copper +
  bili:ast,
  data = pbc_use)


set.seed(3)
rand_inds_pbc <- sample(1:nrow(pbc_use), 3)
cif.predict3_pbc <- treeresponse(pbc_cif$forest,
                                 newdata = pbc_use[rand_inds_pbc,] )

########################## CIF CODE END ##########################
```

```
# predict survial probability

# randomly select three individuals
set.seed(3)
rand_inds_pbc <- sample(1:nrow(pbc_use), 3)

# Obtain predicted values for RSF
pbc_pred <- predict(pbc_rf, pbc_use)

# obtain time values
Time <- pbc_rf$time.interest

matplot(Time, t(pbc_pred$survival[rand_inds_pbc,]), ylab = "Survival", col = c("blue", "green", "orange
        type = "l", lty = 1, main = "RSF: PBC Predicted Survival Curves for 3 Individuals")
legend(x = "topright",           # Position
       legend = c("261", "186", "140"),  # Legend texts
       title = "id",
       col = c("blue", "green", "orange"),
       lwd = 2)
```



```
# Finding the Median Survival Probabilty
# obtain the index for which the survival probability = 0.5
t_id1 <- which(round(pbc_pred$survival[rand_inds_pbc[1],], 2) == .5)
t_id2 <- which(round(pbc_pred$survival[rand_inds_pbc[2],], 2) == .5)
t_id3 <- which(round(pbc_pred$survival[rand_inds_pbc[3],], 2) == .5)
# obtain the times at which survival probability = 0.5 occurs
Time[t_id1] # 3853
```

```
## [1] 4079
```

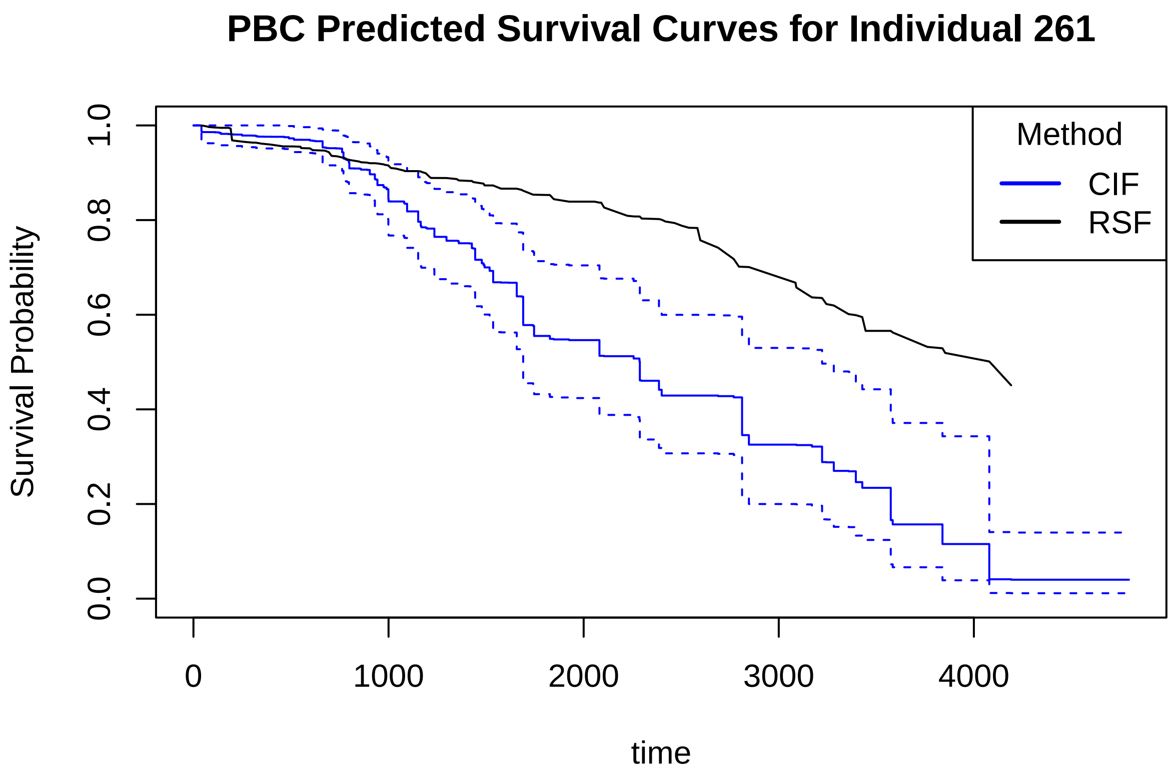
```
Time[t_id2] # 1012
```

```
## integer(0)
```

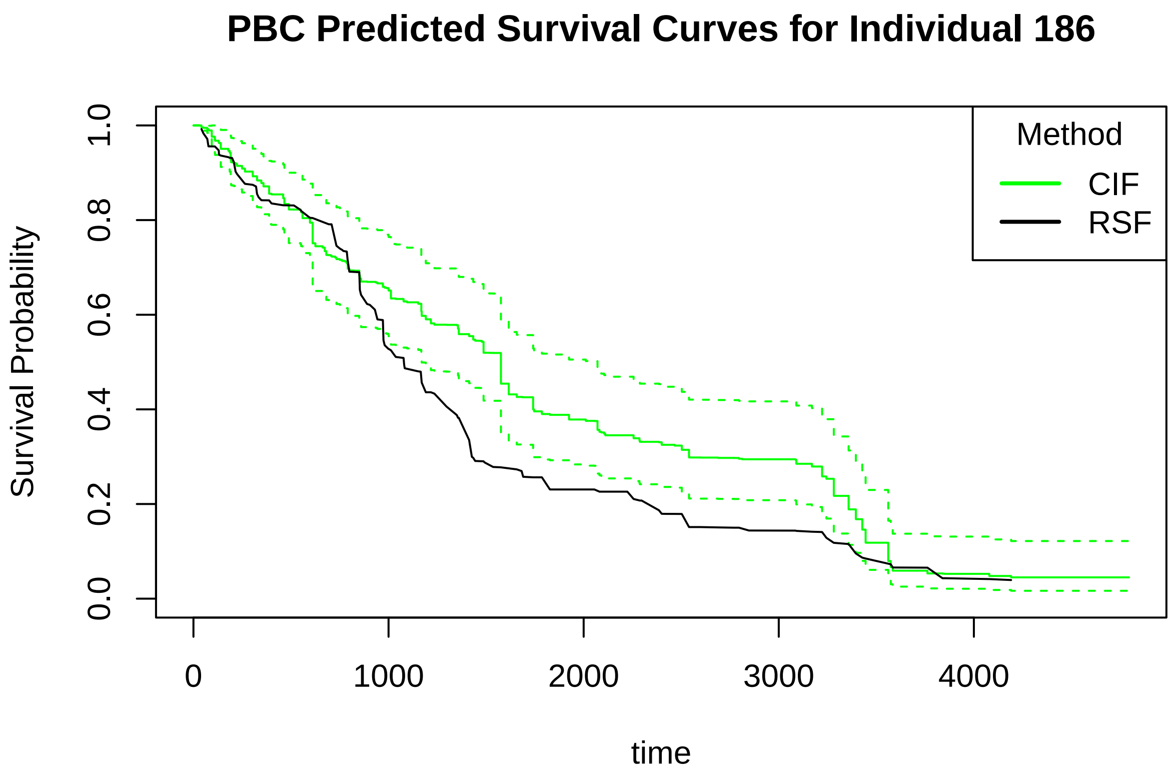
```
Time[t_id3] # does not occur!
```

```
## integer(0)
```

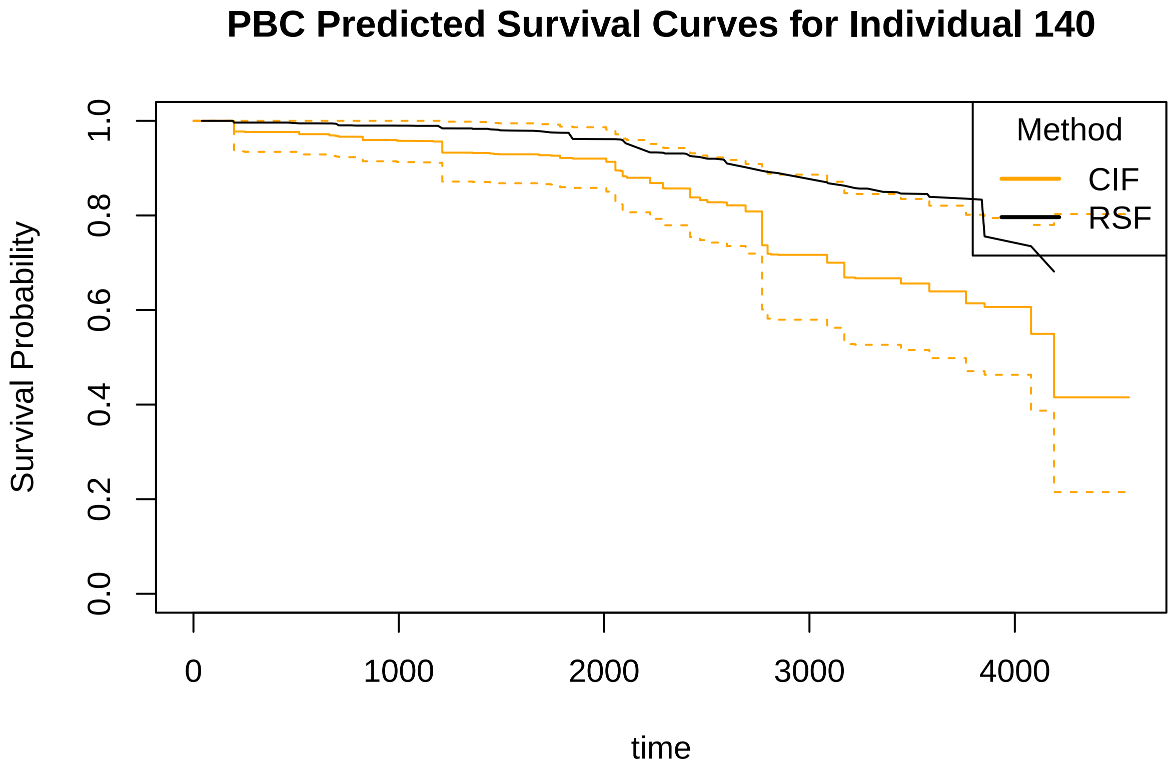
```
# Compare the two methods by plotting
plot(cif.predict3_pbc$`261`, col = "blue",
     main = "PBC Predicted Survival Curves for Individual 261",
     xlab = "time",
     ylab = "Survival Probability")
lines(Time, pbc_pred$survival[rand_inds_pbc[1],])
legend(x = "topright",           # Position
       legend = c("CIF", "RSF"),  # Legend texts
       title = "Method",
       col = c("blue", "black"),
       lwd = 2)
```



```
plot(cif.predict3_pbc$`186`, col = "green",
     main = "PBC Predicted Survival Curves for Individual 186",
     xlab = "time",
     ylab = "Survival Probability")
lines(Time, pbc_pred$survival[rand_inds_pbc[2],])
legend(x = "topright",           # Position
       legend = c("CIF", "RSF"),  # Legend texts
       title = "Method",
       col = c("green", "black"),
       lwd = 2)
```



```
plot(cif.predict3_pbc$`140`, col = "orange",
     main = "PBC Predicted Survival Curves for Individual 140",
     xlab = "time",
     ylab = "Survival Probability")
lines(Time, pbc_pred$survival[rand_inds_pbc[3],])
legend(x = "topright",          # Position
       legend = c("CIF", "RSF"),  # Legend texts
       title = "Method",
       col = c("orange", "black"),
       lwd = 2)
```


PBC Predicted Survival Curves for Individual 140
Survival Probability
0.0
0.2
0.4
0.6
0.8
1.0
0
1000
2000
3000
4000
time
Method
CIF
RSF