

Simulation

Jose Tamez

2023-05-18

Contents

1	Poisson Events	1
1.1	Libraries	1
1.2	Parameters and risk	2
1.3	RRplots()	5

1 Poisson Events

1.1 Libraries

```
library(survival)
library(FRESA.CAD)

## Loading required package: Rcpp
## Loading required package: stringr
## Loading required package: miscTools
## Loading required package: Hmisc
##
## Attaching package: 'Hmisc'
## The following objects are masked from 'package:base':
##
##   format.pval, units
## Loading required package: pROC
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
## The following objects are masked from 'package:stats':
##
##   cov, smooth, var
##source("~/GitHub/FRESA.CAD/R/RRPlot.R")
##source("~/GitHub/FRESA.CAD/R/PoissonEventRiskCalibration.R")
op <- par(no.readonly = TRUE)
pander::panderOptions('digits', 3)
#pander::panderOptions('table.split.table', 400)
```

```
pander::panderOptions('keep.trailing.zeros',TRUE)
layout(matrix(1:1, nrow=1))
```

1.2 Parameters and risk

```
censoredProb <- 0.05
timeSpan <- 10
timeInterval = 0.1
InitialPopulatoin <- 1000
ContBetaRate_1 <- 0.5
ContBetaRate_2 <- 0.025
BinBetaRate_1 <- 1.0
BinBetaRate_2 <- 2.0
betaRates <- c(ContBetaRate_1,ContBetaRate_2,BinBetaRate_1,BinBetaRate_2)
BaselineHazard <- 0.02
ContVar_1 <- runif(InitialPopulatoin)
summary(ContVar_1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## 0.0002768 0.2324736 0.4853391 0.4910489 0.7425847 0.9977289
```

```
ContVar_2 <- rnorm(InitialPopulatoin,50,10)
ContVar_2[ContVar_2 < 1] <- 1
summary(ContVar_2)
```

```
##      Min. 1st Qu.  Median     Mean 3rd Qu.     Max.
##   18.04  43.07   49.36   49.68  56.53   89.58
```

```
BinVar_1 <- rbinom(InitialPopulatoin,1,0.3)
table(BinVar_1)
```

```
## BinVar_1
##    0    1
## 696 304
```

```
BinVar_2 <- rbinom(InitialPopulatoin,1,0.25)
table(BinVar_2)
```

```
## BinVar_2
##    0    1
## 744 256
```

```
dataFeatures <- as.matrix(cbind(ContVar_1,ContVar_2,BinVar_1,BinVar_2))
hazardRate <- as.numeric(BaselineHazard*(dataFeatures %*% betaRates))
summary(hazardRate)
```

```
##      Min. 1st Qu.  Median     Mean 3rd Qu.     Max.
## 0.01337 0.02966 0.03933 0.04607 0.06058 0.10642
```

1.2.1 Getting the events and time to event

```
aliveSet <- c(1:InitialPopulatoin)
eventSet <- numeric(InitialPopulatoin)
timetoEvent <- numeric(InitialPopulatoin)

for (time in c(1:(timeSpan/timeInterval)))
```

```
{
  randProb <- runif(length(aliveSet))
  Isensored <- randProb <= censoredProb
  Isevent <- randProb <= (1.0-exp(-hazardRate[aliveSet]))
  Isensored <- Isensored & !Isevent
  eventSet[aliveSet] <- Isevent
  timetoEvent[aliveSet] <- time*timeInterval-timeInterval/2
  isCensoredOrEvent <- Isensored | Isevent
  aliveSet <- aliveSet[!isCensoredOrEvent]
  cat(length(aliveSet), "(", sum(isCensoredOrEvent), ", ", sum(Isevent), ", ", sum(Isensored), ") \n")
}
```

```
## 934 ( 66 , 54 , 12 )
## 886 ( 48 , 33 , 15 )
## 843 ( 43 , 33 , 10 )
## 799 ( 44 , 35 , 9 )
## 750 ( 49 , 36 , 13 )
## 710 ( 40 , 32 , 8 )
## 662 ( 48 , 36 , 12 )
## 618 ( 44 , 31 , 13 )
## 582 ( 36 , 29 , 7 )
## 554 ( 28 , 26 , 2 )
## 522 ( 32 , 24 , 8 )
## 492 ( 30 , 21 , 9 )
## 470 ( 22 , 16 , 6 )
## 436 ( 34 , 27 , 7 )
## 412 ( 24 , 14 , 10 )
## 383 ( 29 , 24 , 5 )
## 360 ( 23 , 21 , 2 )
## 337 ( 23 , 16 , 7 )
## 317 ( 20 , 18 , 2 )
## 294 ( 23 , 22 , 1 )
## 279 ( 15 , 12 , 3 )
## 259 ( 20 , 15 , 5 )
## 234 ( 25 , 23 , 2 )
## 224 ( 10 , 10 , 0 )
## 210 ( 14 , 10 , 4 )
## 199 ( 11 , 9 , 2 )
## 187 ( 12 , 10 , 2 )
## 176 ( 11 , 9 , 2 )
## 172 ( 4 , 2 , 2 )
## 161 ( 11 , 9 , 2 )
## 156 ( 5 , 5 , 0 )
## 146 ( 10 , 5 , 5 )
## 139 ( 7 , 5 , 2 )
## 133 ( 6 , 4 , 2 )
## 129 ( 4 , 2 , 2 )
## 118 ( 11 , 9 , 2 )
## 109 ( 9 , 8 , 1 )
## 104 ( 5 , 4 , 1 )
## 97 ( 7 , 7 , 0 )
## 95 ( 2 , 2 , 0 )
## 89 ( 6 , 5 , 1 )
## 78 ( 11 , 8 , 3 )
```

```

## 77 ( 1 , 1 , 0 )
## 69 ( 8 , 7 , 1 )
## 67 ( 2 , 2 , 0 )
## 60 ( 7 , 5 , 2 )
## 55 ( 5 , 4 , 1 )
## 53 ( 2 , 2 , 0 )
## 49 ( 4 , 3 , 1 )
## 46 ( 3 , 1 , 2 )
## 44 ( 2 , 0 , 2 )
## 44 ( 0 , 0 , 0 )
## 43 ( 1 , 1 , 0 )
## 42 ( 1 , 1 , 0 )
## 39 ( 3 , 3 , 0 )
## 36 ( 3 , 3 , 0 )
## 34 ( 2 , 1 , 1 )
## 33 ( 1 , 1 , 0 )
## 32 ( 1 , 0 , 1 )
## 30 ( 2 , 2 , 0 )
## 29 ( 1 , 0 , 1 )
## 29 ( 0 , 0 , 0 )
## 29 ( 0 , 0 , 0 )
## 27 ( 2 , 1 , 1 )
## 26 ( 1 , 0 , 1 )
## 25 ( 1 , 0 , 1 )
## 23 ( 2 , 2 , 0 )
## 22 ( 1 , 1 , 0 )
## 20 ( 2 , 2 , 0 )
## 19 ( 1 , 0 , 1 )
## 18 ( 1 , 0 , 1 )
## 18 ( 0 , 0 , 0 )
## 16 ( 2 , 2 , 0 )
## 16 ( 0 , 0 , 0 )
## 16 ( 0 , 0 , 0 )
## 16 ( 0 , 0 , 0 )
## 16 ( 0 , 0 , 0 )
## 14 ( 2 , 1 , 1 )
## 12 ( 2 , 1 , 1 )
## 12 ( 0 , 0 , 0 )
## 11 ( 1 , 1 , 0 )
## 11 ( 0 , 0 , 0 )
## 11 ( 0 , 0 , 0 )
## 11 ( 0 , 0 , 0 )
## 11 ( 0 , 0 , 0 )
## 10 ( 1 , 0 , 1 )
## 10 ( 0 , 0 , 0 )
## 10 ( 0 , 0 , 0 )
## 10 ( 0 , 0 , 0 )
## 10 ( 0 , 0 , 0 )
## 9 ( 1 , 0 , 1 )
## 9 ( 0 , 0 , 0 )
## 9 ( 0 , 0 , 0 )
## 9 ( 0 , 0 , 0 )
## 9 ( 0 , 0 , 0 )
## 9 ( 0 , 0 , 0 )

```

```
## 8 ( 1 , 0 , 1 )
## 8 ( 0 , 0 , 0 )
## 7 ( 1 , 0 , 1 )
## 7 ( 0 , 0 , 0 )

timetoEvent[aliveSet] <- time*timeInterval + timeInterval/2
summary(timetoEvent)

##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
##    0.050  0.525   1.150   1.680   2.250  10.050

table(eventSet)

## eventSet
##      0      1
## 231 769

pevent <- (1.0-exp(-hazardRate))
summary(pevent)

##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## 0.01328 0.02923 0.03857 0.04483 0.05878 0.10095

simulatedDataFrame <- as.data.frame(cbind(status=eventSet,time=timetoEvent,pevent=pevent,dataFeatures))
```

1.3 RRplots()

```
plotTimeInterval <- 10.0

hazard <- -log(1.0-simulatedDataFrame$pevent)
hboost <- plotTimeInterval/timeInterval
pvalue <- 1.0-exp(-hboost*hazard)

rdata <- cbind(simulatedDataFrame$status,pvalue)
summary(rdata[,2])

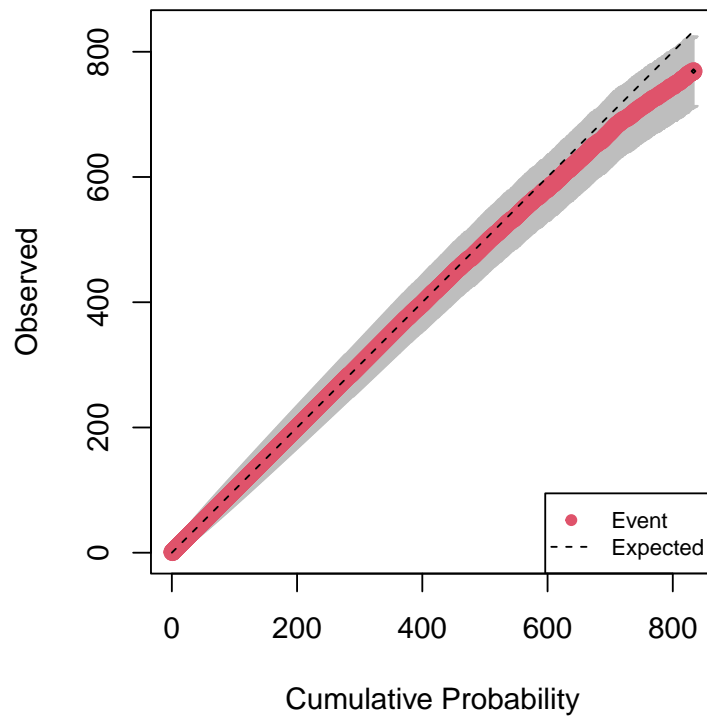
Min. 1st Qu.  Median      Mean 3rd Qu.      Max. 0.7374 0.9485 0.9804 0.9673 0.9977 1.0000

table(simulatedDataFrame$status)

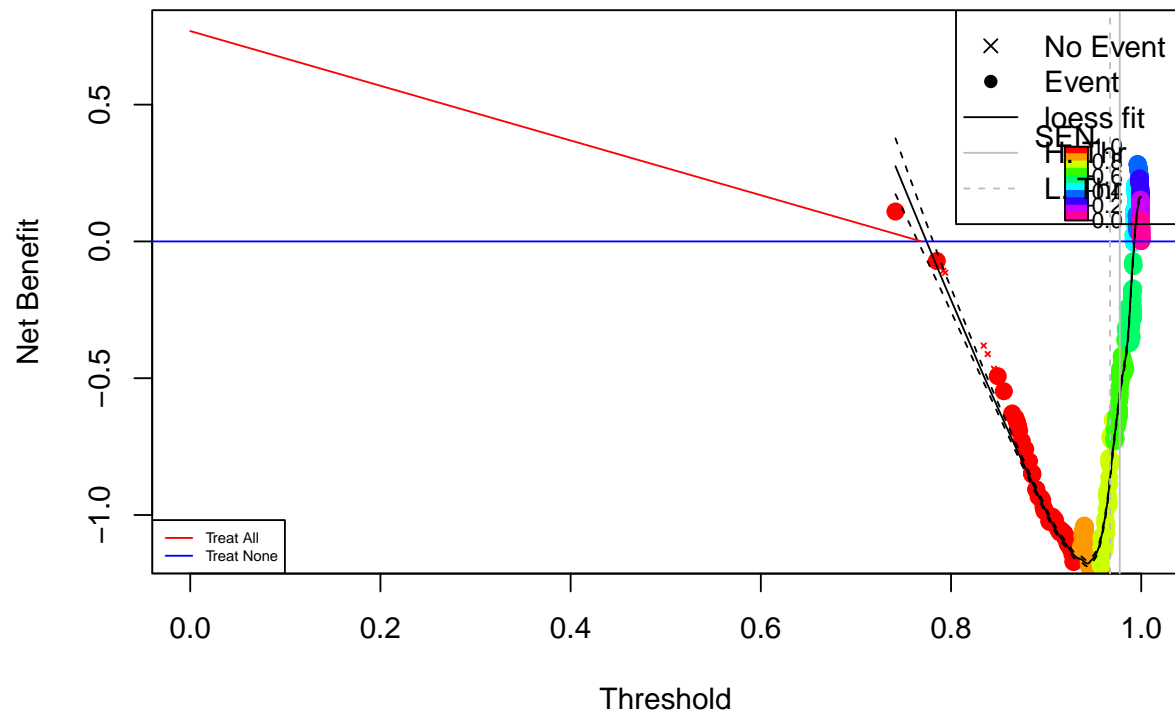
0 1 231 769

RRAnalysisCI <- RRPlot(rdata,atRate=c(0.90,0.80),
                       timetoEvent=simulatedDataFrame$time,
                       title="Simulation",
                       ysurvlim=c(0.00,1.0),
                       riskTimeInterval=plotTimeInterval)
```

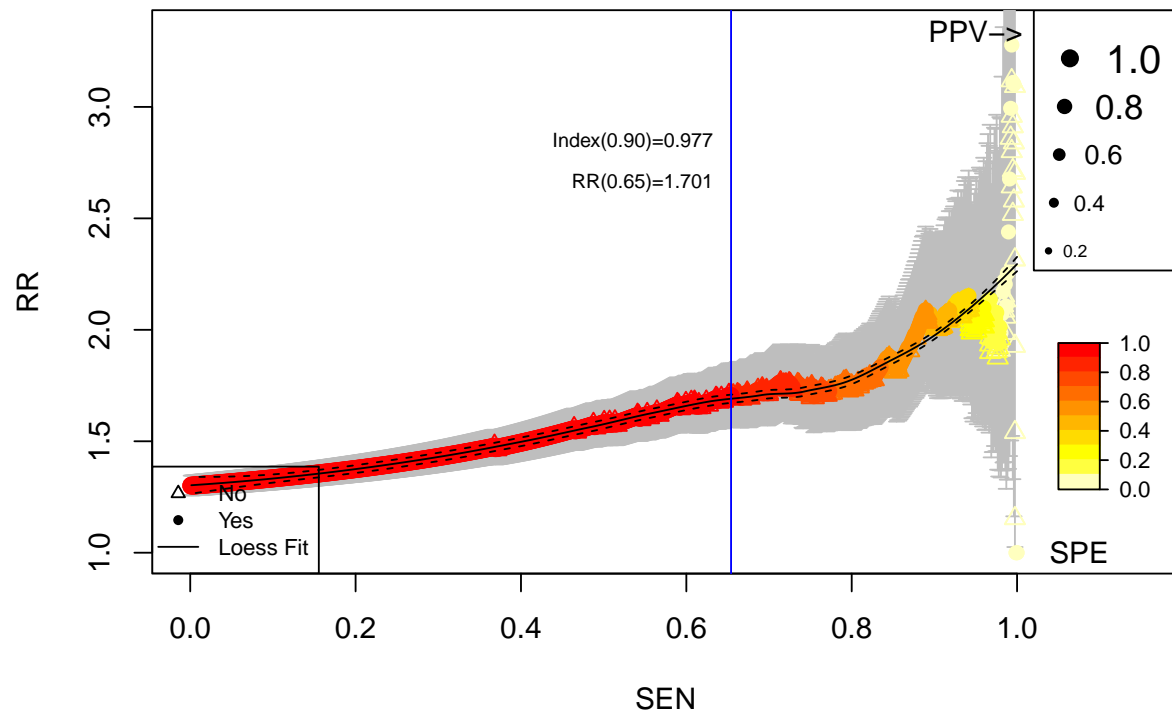
Cumulative vs. Observed: Simulation

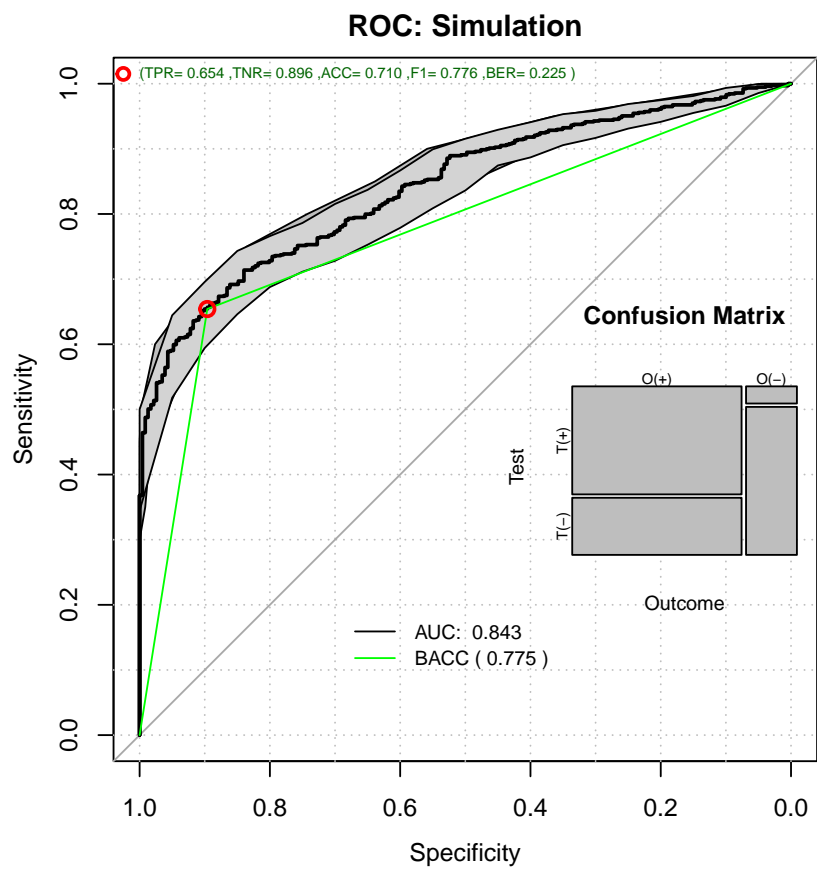


Decision Curve Analysis: Simulation

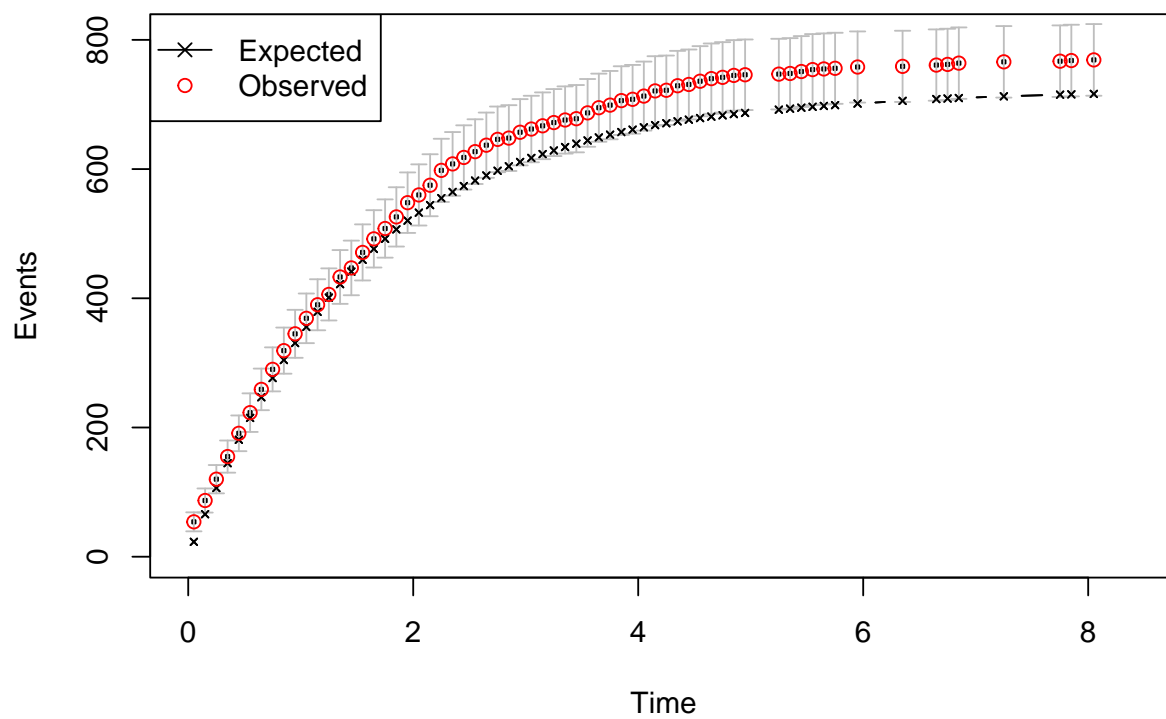


Relative Risk: Simulation

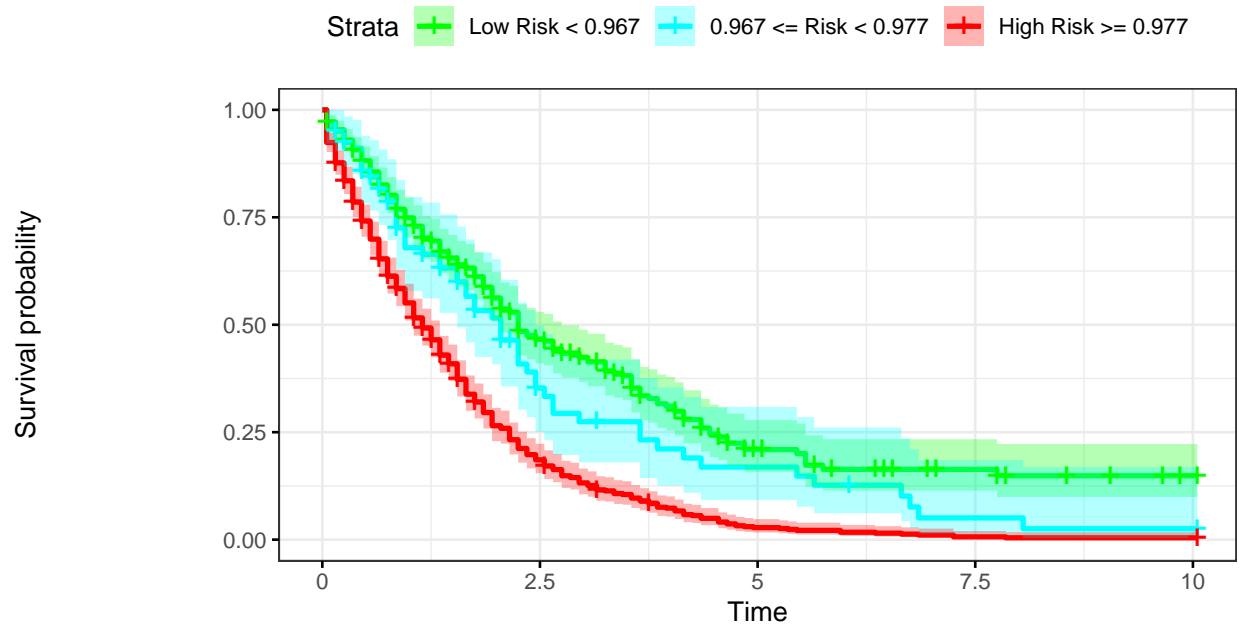




Time vs. Events: Simulation



Kaplan–Meier: Simulation



Number at risk

Low Risk < 0.967	393	101	25	11	4
0.967 <= Risk < 0.977	80	18	8	2	1
High Risk >= 0.977	527	91	13	3	2

```
par(op)
```

1.3.1 Risk Calibration

```
op <- par(no.readonly = TRUE)

crdata <- cbind(simulatedDataFrame$status, pvalue, simulatedDataFrame$time)

#calprob <- CalibrationProbPoissonRisk(crdata, timeInterval=plotTimeInterval)
calprob <- CalibrationProbPoissonRisk(crdata)

pander::pander(c(h0=calprob$h0,
                  Gain=calprob$hazardGain,
                  DeltaTime=calprob$timeInterval),
               caption="Cox Calibration Parameters")
```

h0	Gain	DeltaTime
0.402	0.49	4.62

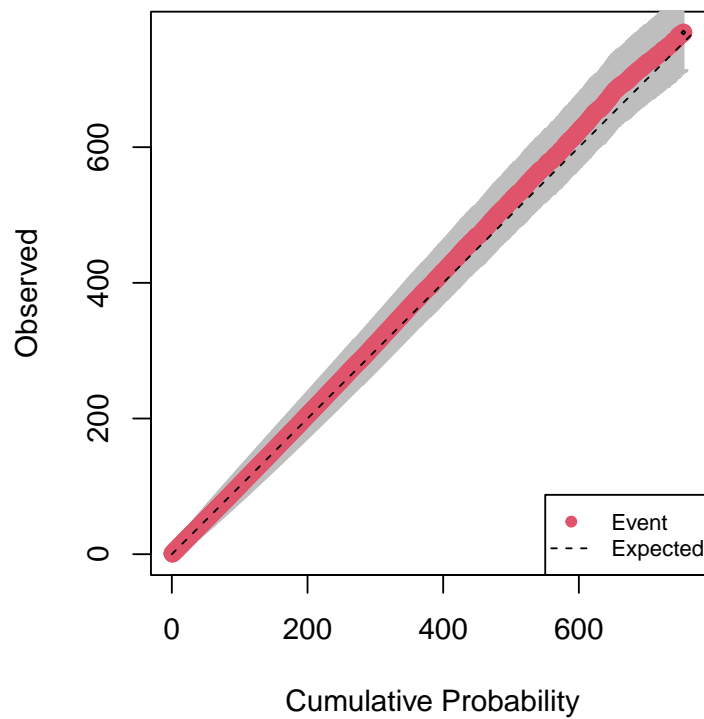
1.3.2 After Calibration

```
h0 <- calprob$h0

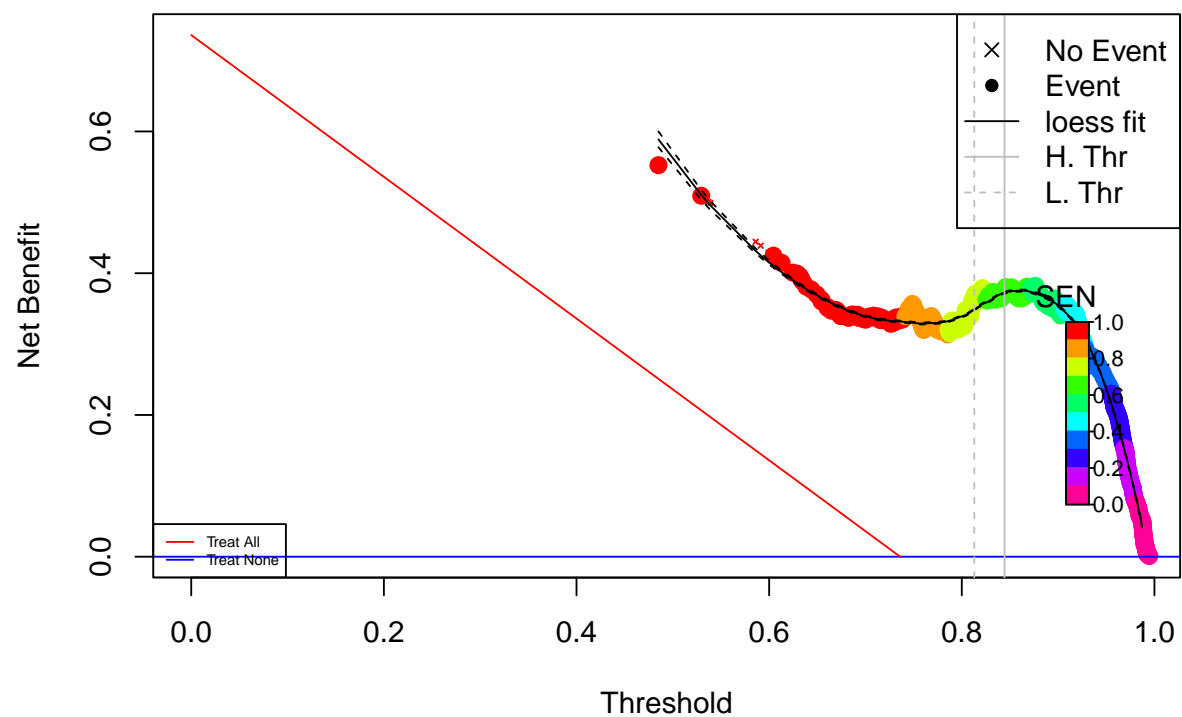
caldata <- cbind(simulatedDataFrame$status, calprob$prob)

rrAnalysisTrain <- RRPlot(caldata, atRate=c(0.90, 0.80),
                           timetoEvent=simulatedDataFrame$time,
                           title="Cal. Simulation",
                           ysurvlim=c(0.00, 1.0),
                           riskTimeInterval=calprob$timeInterval)
```

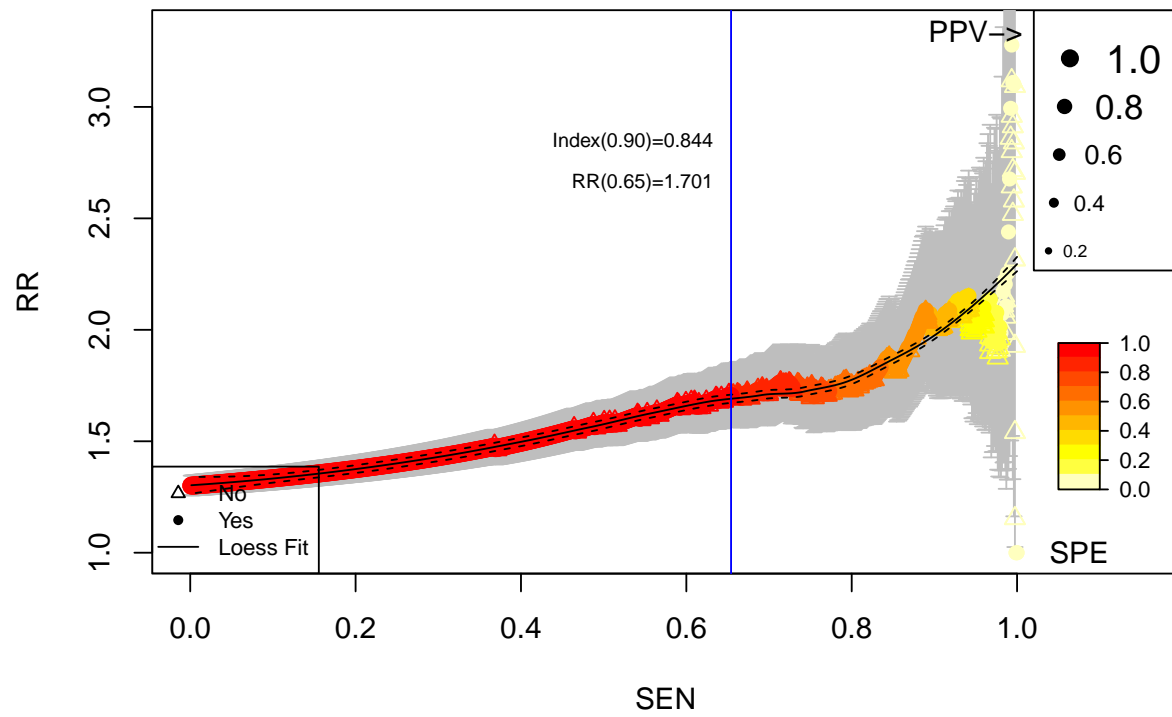
Cumulative vs. Observed: Cal. Simulation

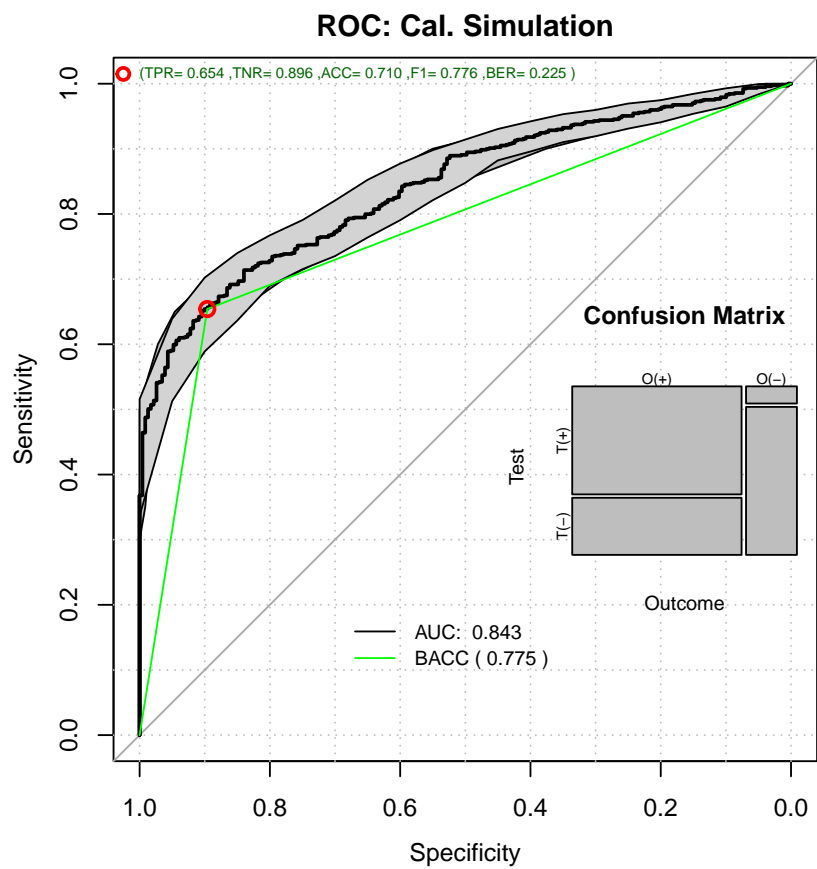


Decision Curve Analysis: Cal. Simulation

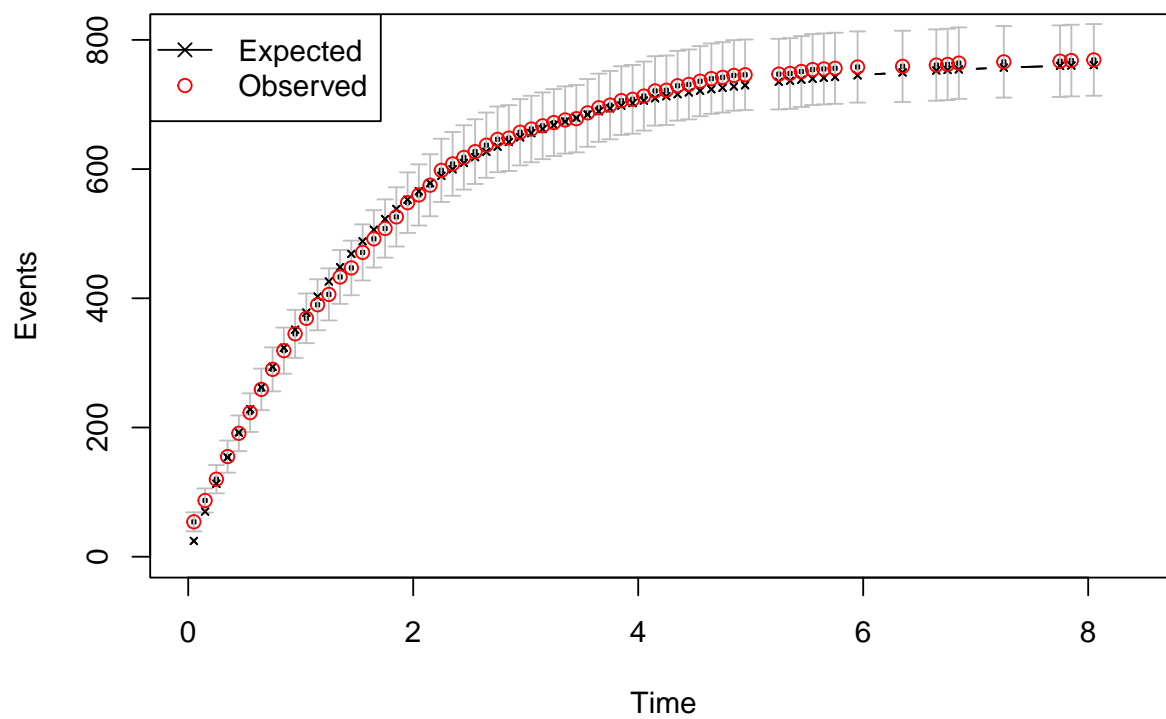


Relative Risk: Cal. Simulation

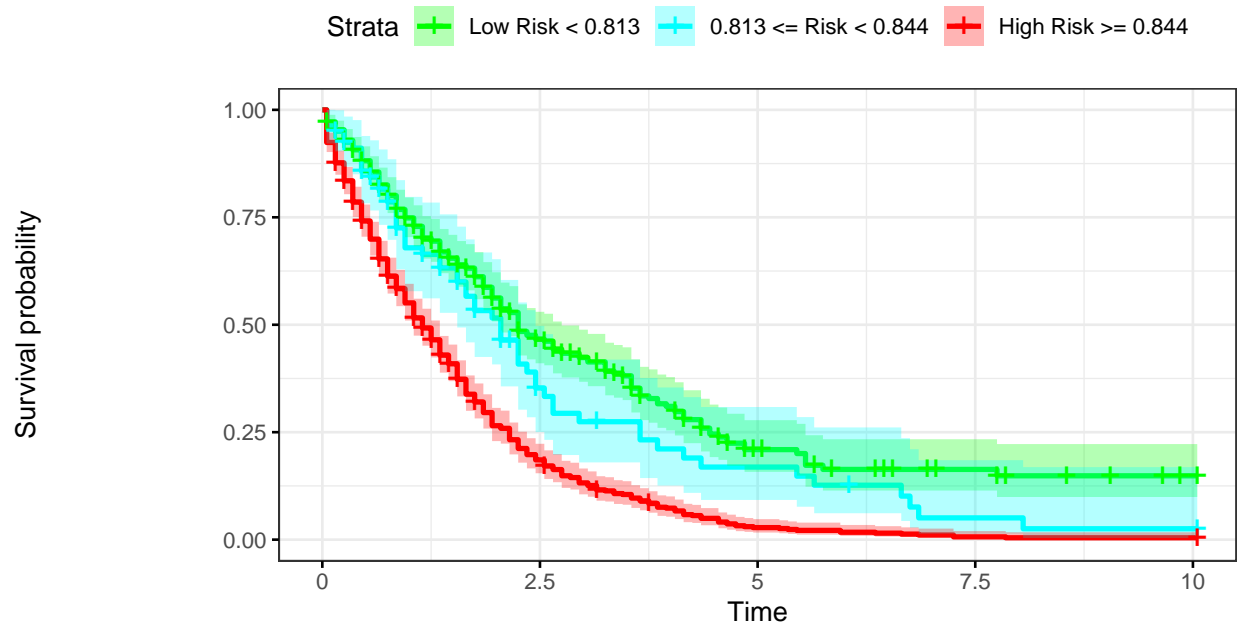




Time vs. Events: Cal. Simulation



Kaplan–Meier: Cal. Simulation



Number at risk

Low Risk < 0.813	393	101	25	11	4
0.813 <= Risk < 0.844	80	18	8	2	1
High Risk >= 0.844	527	91	13	3	2