Colon Cancer

Jose Tamez

2023-05-15

Contents

1 RRPlot and the Colon data set

1.0.1 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
```

```
#library(corrplot)
#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)
```

1.1 The data set

```
data(cancer)
colon <- subset(colon,etype==1)
colon$etype <- NULL
rownames(colon) <- colon$id
colon$id <- NULL
colon <- colon[complete.cases(colon),]
time <- colon$time
status <- colon$status
data <- colon
data$time <- NULL
data$study <- NULL
table(data$status)
```

0 1 442 446

```
dataColon <- as.data.frame(model.matrix(status~.*.,data))
dataColon$`(Intercept)` <- NULL
dataColon$time <- time/365
dataColon$status <- status
colnames(dataColon) <-str_replace_all(colnames(dataColon),":","_")
colnames(dataColon) <-str_replace_all(colnames(dataColon),"\\.","_")
colnames(dataColon) <-str_replace_all(colnames(dataColon),"\\+","_")
data <- NULL

trainsamples <- sample(nrow(dataColon),0.7*nrow(dataColon))
dataColonTrain <- dataColon[trainsamples,]
dataColonTest <- dataColon[-trainsamples,]


pander::pander(table(dataColonTrain$status))
```

0	1
306	315

```
pander::pander(table(dataColonTest$status))
```

0	1
136	131

1.2 Modeling

```
ml <- BSWiMS.model(Surv(time,status)~1,data=dataColonTrain,NumberofRepeats = 3)
```

[++++++++-++++++++-++++++++-]..

```
sm <- summary(m1)
pander::pander(sm$coefficients)
```

Table 3: Table continues below

	Estimate	lower	HR	upper	u.Accuracy
nodes	0.018711	1.013	1.019	1.024	0.614
nodes_node4	0.012480	1.009	1.013	1.016	0.609
differ_node4	0.066230	1.047	1.068	1.090	0.607
extent_node4	0.031074	1.021	1.032	1.042	0.607
age_nodes	0.000127	1.000	1.000	1.000	0.604
node4	0.058374	1.037	1.060	1.083	0.607
age_node4	0.001203	1.001	1.001	1.002	0.607
nodes_extent	0.003370	1.002	1.003	1.005	0.622
rxLev_5FU_age	-0.001451	0.998	0.999	0.999	0.576
nodes_differ	0.001619	1.001	1.002	1.003	0.615
sex_nodes	-0.009527	0.986	0.991	0.995	0.509
rxLev_5FU_differ	-0.023466	0.965	0.977	0.989	0.576
rxLev_5FU_extent	-0.030572	0.955	0.970	0.985	0.576
rxLev_5FU	-0.045898	0.932	0.955	0.978	0.576
rxLev_5FU_sex	-0.065777	0.903	0.936	0.971	0.560
differ_extent	0.015186	1.006	1.015	1.024	0.546
extent	0.078339	1.031	1.081	1.135	0.551
rxLev_5FU_node4	0.010141	1.003	1.010	1.017	0.515

Table 4: Table continues below

	r.Accuracy	full.Accuracy	u.AUC	r.AUC	full.AUC
nodes	0.529	0.618	0.616	0.524	0.620
nodes_node4	0.507	0.609	0.612	0.500	0.612
differ_node4	0.588	0.620	0.611	0.586	0.623
extent_node4	0.574	0.610	0.611	0.571	0.613
age_nodes	0.510	0.604	0.606	0.503	0.606
node4	0.596	0.609	0.611	0.594	0.613
age_node4	0.596	0.612	0.611	0.595	0.615
nodes_extent	0.587	0.628	0.624	0.586	0.629
rxLev_5FU_age	0.618	0.620	0.574	0.622	0.623
nodes_differ	0.602	0.607	0.617	0.602	0.608
sex_nodes	0.611	0.627	0.511	0.610	0.628
rxLev_5FU_differ	0.602	0.612	0.574	0.604	0.615
rxLev_5FU_extent	0.618	0.627	0.574	0.621	0.628
rxLev_5FU	0.606	0.612	0.574	0.609	0.615
rxLev_5FU_sex	0.607	0.607	0.555	0.611	0.611
differ_extent	0.604	0.612	0.550	0.607	0.615
extent	0.609	0.620	0.546	0.612	0.623
rxLev_5FU_node4	0.599	0.609	0.521	0.599	0.608

	IDI	NRI	z.IDI	z.NRI	Delta.AUC	Frequency
nodes	0.03760	0.4472	6.72	6.083	0.096424	1.000

	IDI	NRI	z.IDI	z.NRI	Delta.AUC	Frequency
nodes_node4	0.03676	0.4491	6.56	6.679	0.112278	1.000
differ_node4	0.04932	0.4424	6.41	6.527	0.036928	1.000
extent_node4	0.04407	0.4424	6.20	6.527	0.041515	1.000
age_nodes	0.03284	0.4209	6.20	5.707	0.103268	1.000
node4	0.03141	0.4255	5.44	6.305	0.018459	1.000
age_node4	0.03153	0.4358	5.39	6.464	0.019286	1.000
nodes_extent	0.02817	0.3831	5.35	5.152	0.042958	1.000
rxLev_5FU_age	0.02174	0.2960	4.27	3.974	0.001821	1.000
nodes_differ	0.01698	0.2833	3.93	3.767	0.005854	1.000
sex_nodes	0.01040	0.1824	3.92	2.329	0.017805	1.000
rxLev_5FU_differ	0.01817	0.2960	3.90	3.974	0.010498	1.000
rxLev_5FU_extent	0.01839	0.3002	3.80	4.079	0.007530	1.000
rxLev_5FU	0.01716	0.2960	3.73	3.974	0.006348	1.000
rxLev_5FU_sex	0.01704	0.2215	3.58	3.862	0.000032	1.000
differ_extent	0.01586	0.1155	3.31	1.448	0.007547	1.000
extent	0.01386	-0.0138	3.19	-0.213	0.011111	1.000
rxLev_5FU_node4	0.00426	0.1059	2.77	1.359	0.008777	0.333

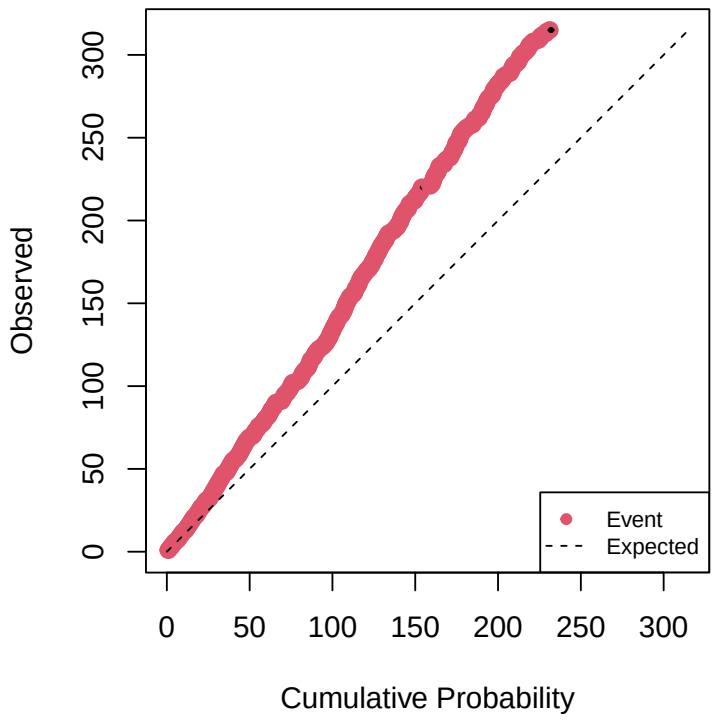
1.3 Cox Model Performance

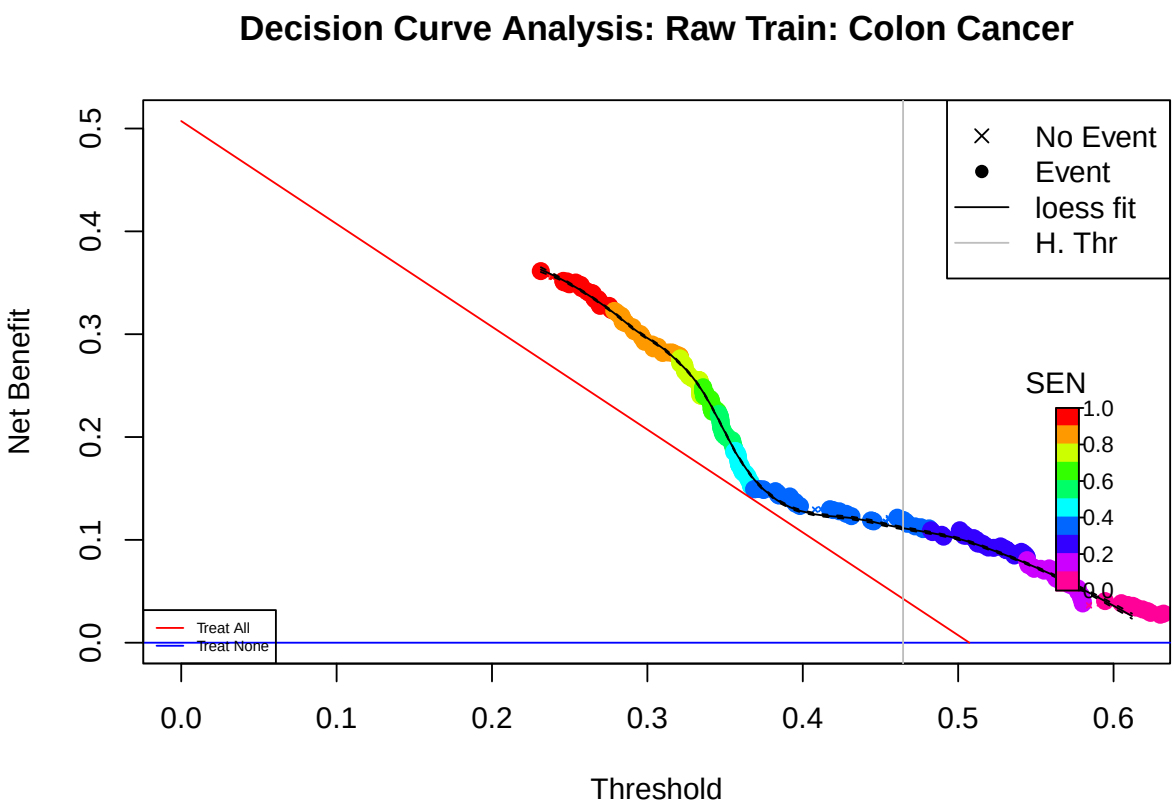
Here we evaluate the model using the RRPlot() function.

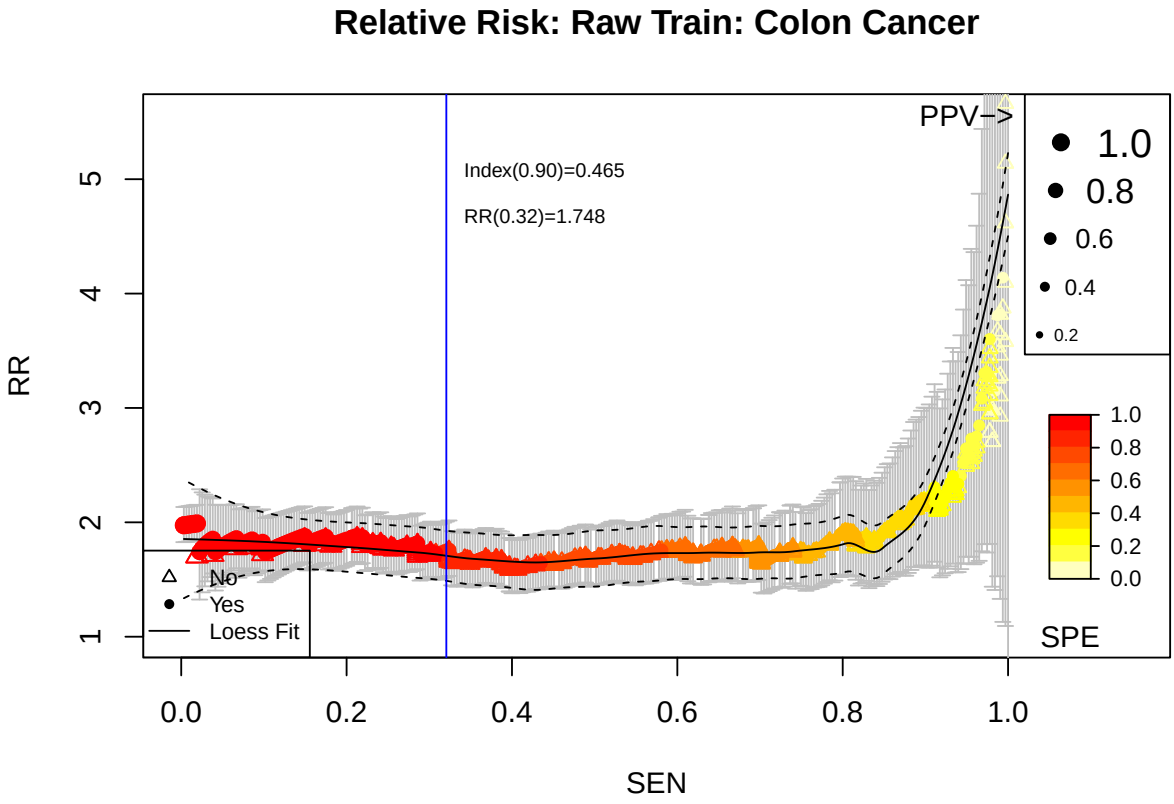
1.3.1 The evaluation of the raw Cox model with RRPlot()

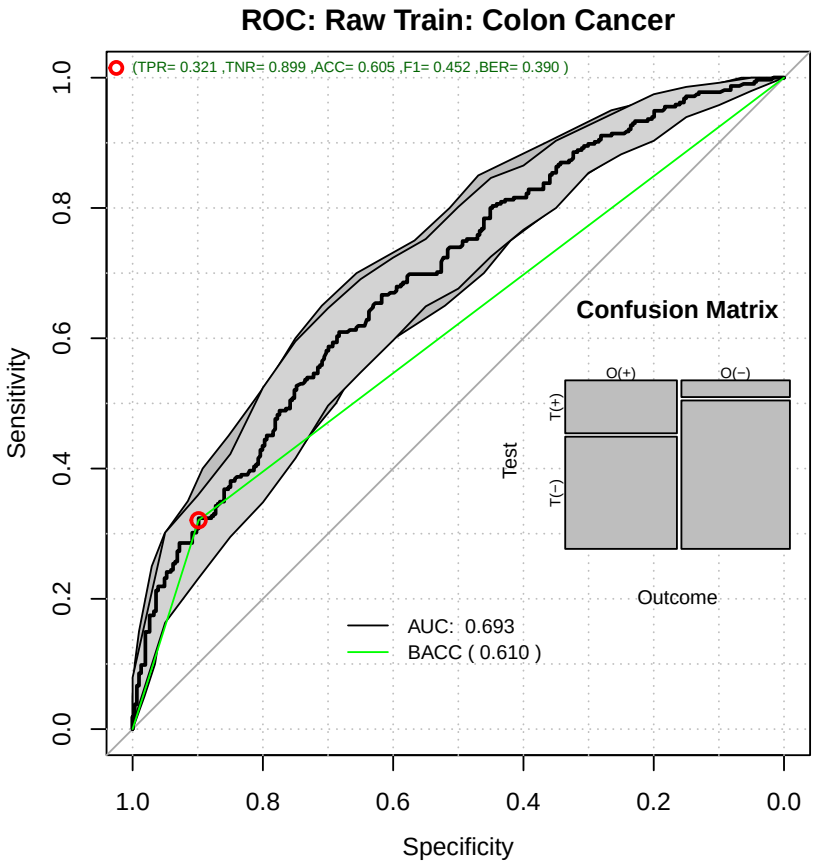
Here we will use the predicted event probability assuming a baseline hazard for events withing 5 years

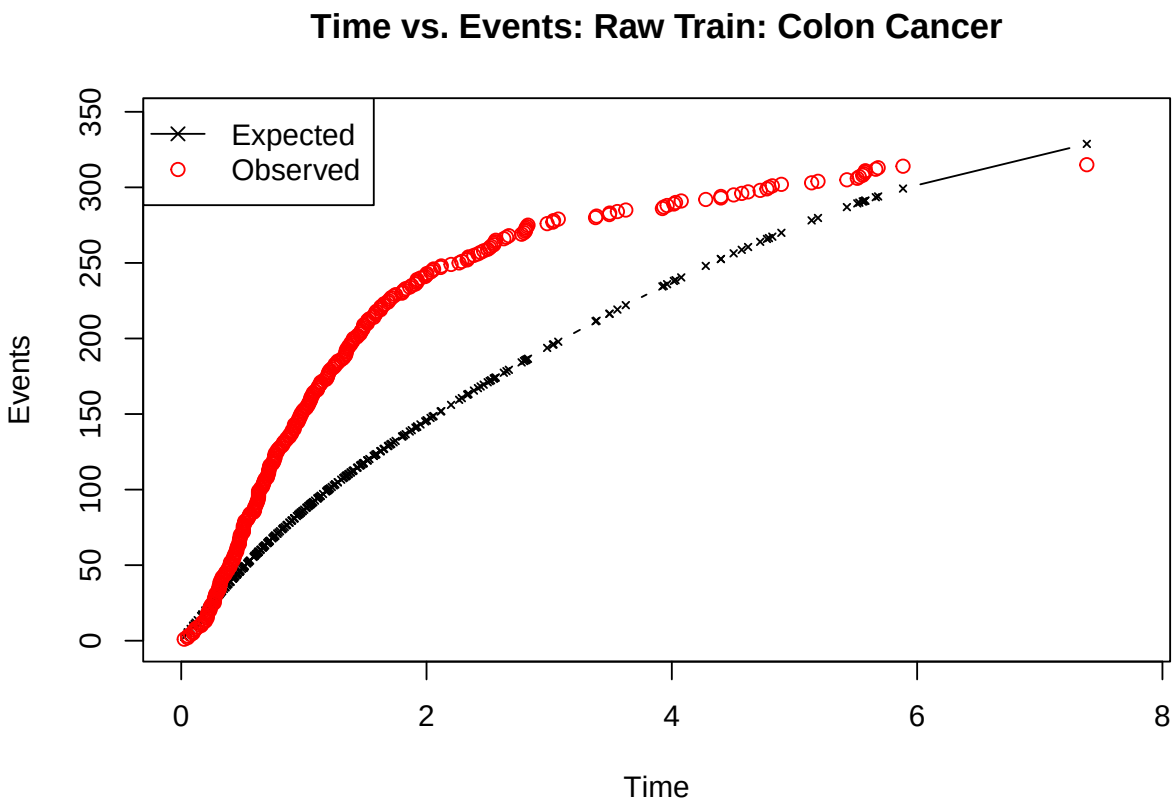
```
index <- predict(ml,dataColonTrain)
timeinterval <- round(2*mean(subset(dataColonTrain,status==1)$time),0)

h0 <- sum(dataColonTrain$status & dataColonTrain$time <= timeinterval)
h0 <- h0/sum((dataColonTrain$time > timeinterval) | (dataColonTrain$status==1))

rdata <- cbind(dataColonTrain$status,ppoisGzero(index,h0))

rrAnalysisTrain <- RRPlot(rdata,atRate=c(0.90),
                      timetoEvent=dataColonTrain$time,
                      title="Raw Train: Colon Cancer",
                      ysurvlim=c(0.00,1.0),
                      riskTimeInterval=timeinterval)
```

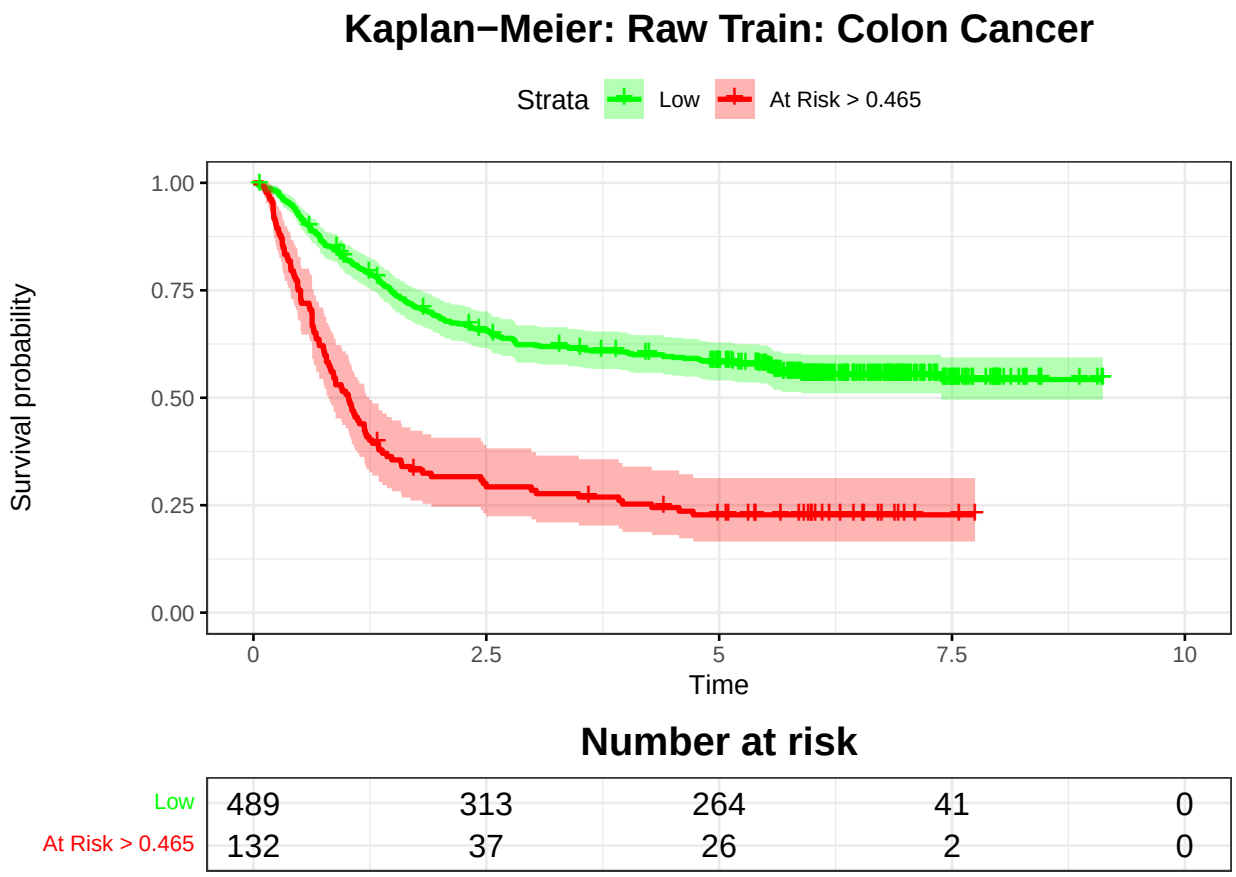

Cumulative vs. Observed: Raw Train: Colon Cancer
Observed
0
50
100
150
200
250
300
Cumulative Probability
Event
Expected


Decision Curve Analysis: Raw Train: Colon Cancer
Net Benefit
0.0
0.1
0.2
0.3
0.4
0.5
Threshold
0.0
0.1
0.2
0.3
0.4
0.5
0.6
No Event
Event
loess fit
H. Thr
SEN
1.0
0.8
0.6
0.4
0.2
0.0
Treat All
Treat None


Relative Risk: Raw Train: Colon Cancer
PPV–>
Index(0.90)=0.465
RR(0.32)=1.748
1.0
0.8
0.6
0.4
0.2
1.0
0.8
0.6
0.4
0.2
0.0
SPE
No
Yes
Loess Fit
RR
SEN
1
2
3
4
5
0.0
0.2
0.4
0.6
0.8
1.0


ROC: Raw Train: Colon Cancer
(TPR= 0.321 ,TNR= 0.899 ,ACC= 0.605 ,F1= 0.452 ,BER= 0.390)
Sensitivity
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
Confusion Matrix
O(+)
O(−)
T(+)
T(−)
Test
Outcome
AUC: 0.693
BACC (0.610)


Time vs. Events: Raw Train: Colon Cancer
Expected
Observed
Events
0
50
100
150
200
250
300
350
0
2
4
6
8
Time



1.3.2 Time to event

```
toinclude <- rdata[,1]==1
obstiemToEvent <- dataColonTrain[,"time"]
tmin<-min(obstiemToEvent)
sum(toinclude)
```

[1] 315

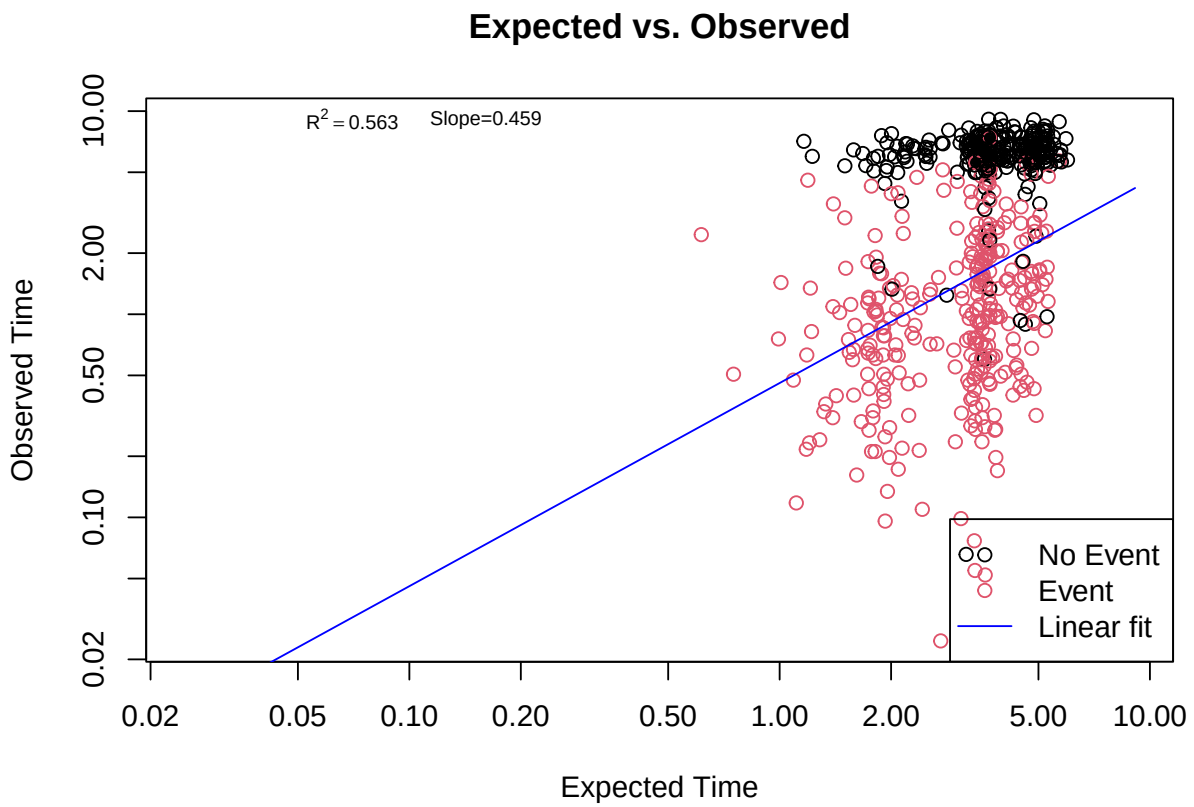
```
timetoEvent <- meanTimeToEvent(rdata[,2],timeinterval)
tmax<-max(c(obstiemToEvent,timetoEvent))
lmfit <- lm(obstiemToEvent[toinclude]~0+timetoEvent[toinclude])
sm <- summary(lmfit)
pander::pander(sm)
```

	Estimate	Std. Error	t value	Pr(>|t|)
timetoEvent[toinclude]	0.459	0.0228	20.1	1.84e-58

Table 7: Fitting linear model: obstiemToEvent[toinclude] ~ 0 + timetoEvent[toinclude]

Observations	Residual Std. Error	R^2	Adjusted R^2
315	1.35	0.563	0.562

```
plot(timetoEvent,obstiemToEvent,
     col=1+rdata[,1],
     xlab="Expected Time",
     ylab="Observed Time",
     main="Expected vs. Observed",
     xlim=c(tmin,tmax),
     ylim=c(tmin,tmax),
     log="xy")
lines(x=c(tmin,tmax),y=lmfit$coefficients*c(tmin,tmax),lty=1,col="blue")
txt <- bquote(paste(R^2 == .(round(sm$r.squared,3))))
text(tmin+0.005*(tmax-tmin),tmax,txt,cex=0.7)
text(tmin+0.015*(tmax-tmin),tmax,sprintf("Slope=%4.3f",sm$coefficients[1]),cex=0.7)
legend("bottomright",legend=c("No Event","Event","Linear fit"),
              pch=c(1,1,-1),
              col=c(1,2,"blue"),
              lty=c(-1,-1,1)
              )
```



```
MADerror2 <- mean(abs(timetoEvent[toinclude]-obstiemToEvent[toinclude]))
pander::pander(MADerror2)
```

1.99

1.3.3 Uncalibrated Performance Report

```
pander::pander(t(rrAnalysisTrain$keyPoints),caption="Threshold values")
```

Table 8: Threshold values

	@:0.9	@MAX_BACC	@MAX_RR	@SPE100	p(0.5)
Thr	0.465	0.343	0.254	2.31e-01	0.499
RR	1.748	1.793	3.603	2.56e+01	1.798
RR_LCI	1.523	1.524	1.811	5.53e-02	1.571
RR_UCI	2.007	2.109	7.166	1.18e+04	2.058
SEN	0.321	0.610	0.978	1.00e+00	0.286
SPE	0.899	0.683	0.131	1.63e-02	0.925
BACC	0.610	0.646	0.554	5.08e-01	0.605
NetBenefit	0.119	0.227	0.350	3.61e-01	0.108

```
pander::pander(t(rrAnalysisTrain$OERatio$estimate),caption="O/E Ratio")
```

Table 9: O/E Ratio

O/E	Low	Upper	p.value
0.958	0.855	1.07	0.473

```
pander::pander(t(rrAnalysisTrain$OE95ci),caption="O/E Mean")
```

Table 10: O/E Mean

mean	50%	2.5%	97.5%
1.57	1.57	1.55	1.6

```
pander::pander(t(rrAnalysisTrain$OAcum95ci),caption="O/Acum Mean")
```

Table 11: O/Acum Mean

mean	50%	2.5%	97.5%
1.38	1.38	1.38	1.39

```
pander::pander(rrAnalysisTrain$c.index$cstatCI,caption="C. Index")
```

mean.C Index	median	lower	upper
0.665	0.665	0.636	0.694

```
pander::pander(t(rrAnalysisTrain$ROCAnalysis$aucs),caption="ROC AUC")
```

Table 13: ROC AUC

est	lower	upper
0.693	0.653	0.734

```
pander::pander((rrAnalysisTrain$ROCAnalysis$sensitivity),caption="Sensitivity")
```

Table 14: Sensitivity

est	lower	upper
0.321	0.269	0.375

```
pander::pander((rrAnalysisTrain$ROCAnalysis$specificity),caption="Specificity")
```

Table 15: Specificity

est	lower	upper
0.899	0.859	0.93

```
pander::pander(t(rrAnalysisTrain$thr_atP),caption="Probability Thresholds")
```

Table 16: Probability Thresholds

90%
0.465

```
pander::pander(rrAnalysisTrain$surdif,caption="Logrank test")
```

Table 17: Logrank test Chisq = 81.105923 on 1 degrees of freedom, p = 0.000000

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
class=0	489	214	269.7	11.5	81.1
class=1	132	101	45.3	68.6	81.1

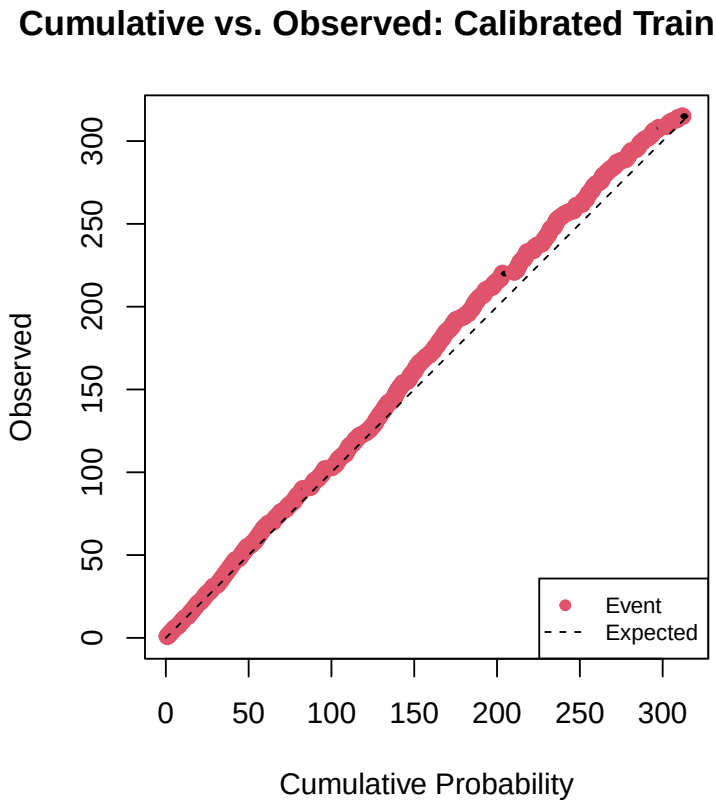
1.3.4 Cox Calibration

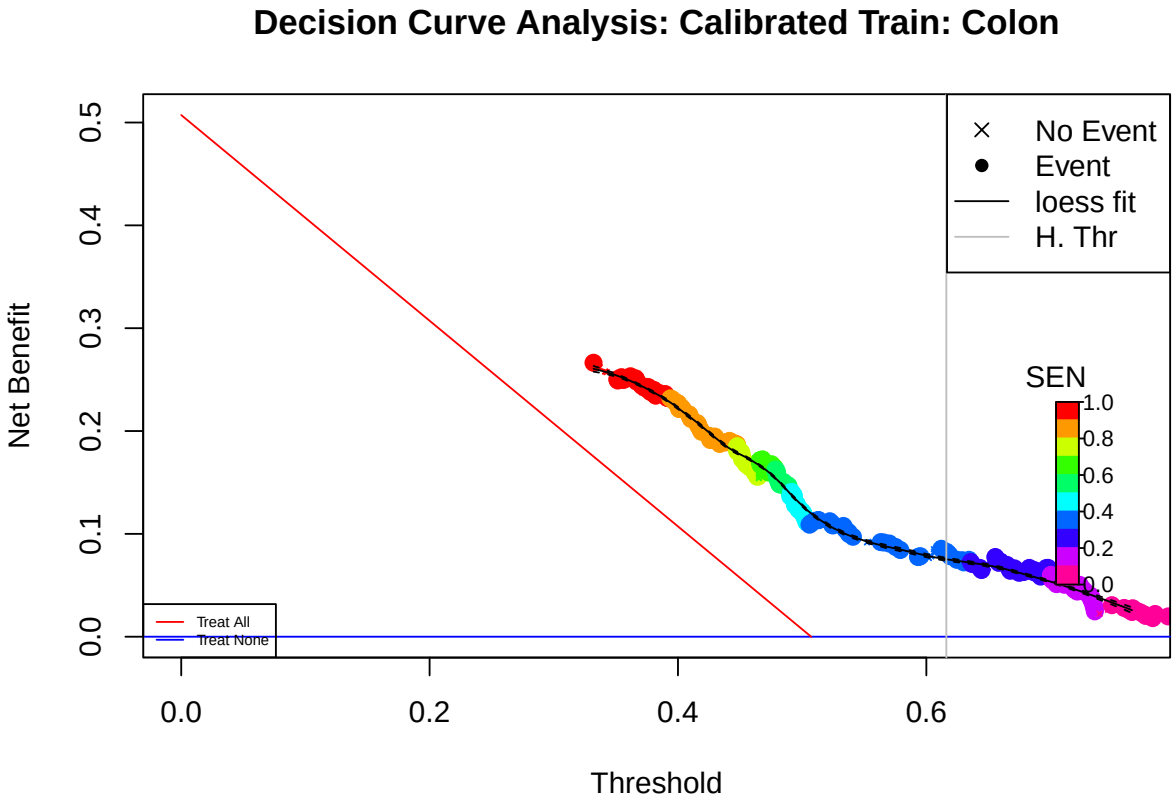
```
op <- par(no.readonly = TRUE)


calprob <- CoxRiskCalibration(ml,dataColonTrain,"status","time",timeInterval=timeinterval)

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

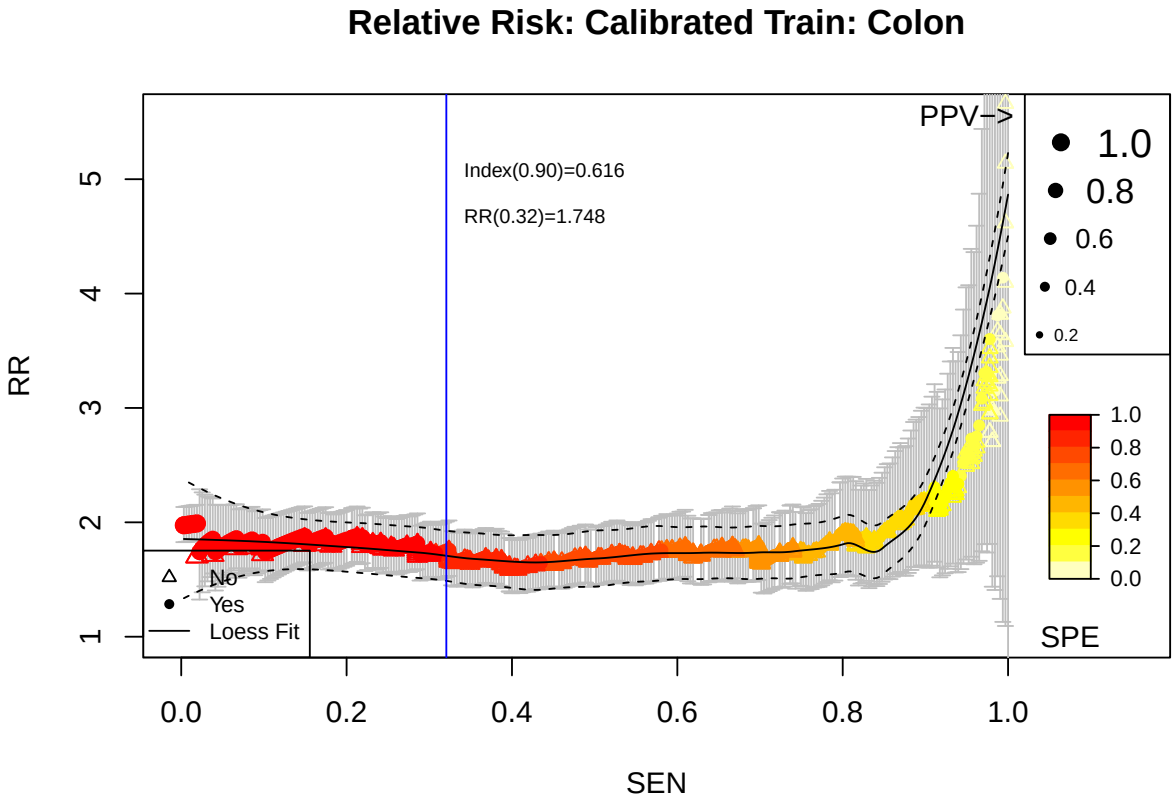
h0	Gain	TimeInterval
0.696	1.53	2.86

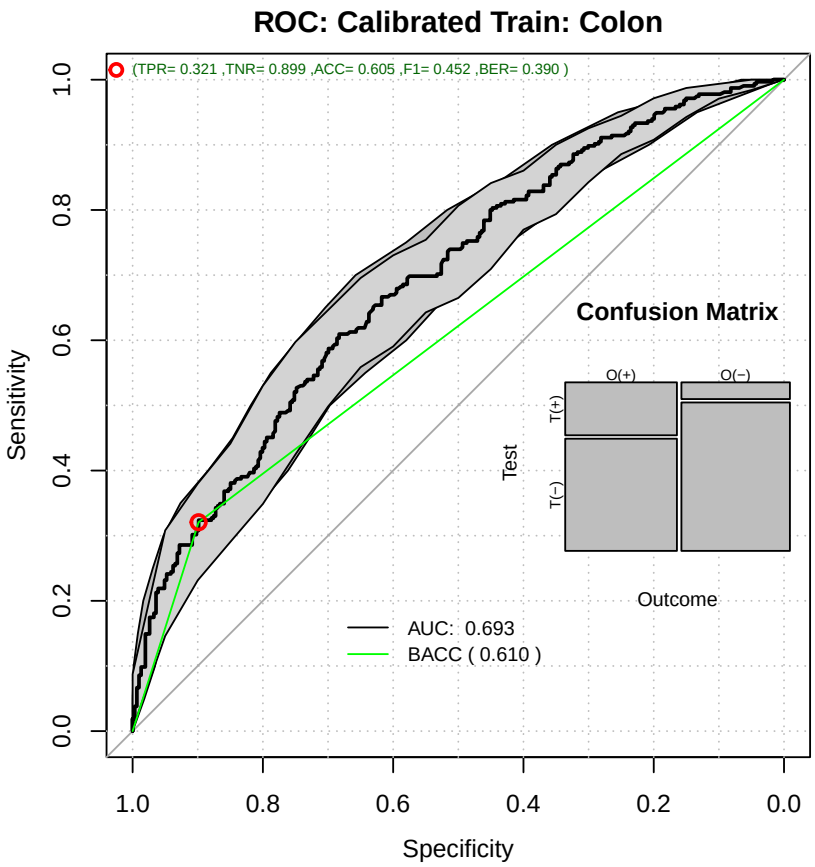
1.3.5 The RRplot() of the calibrated model

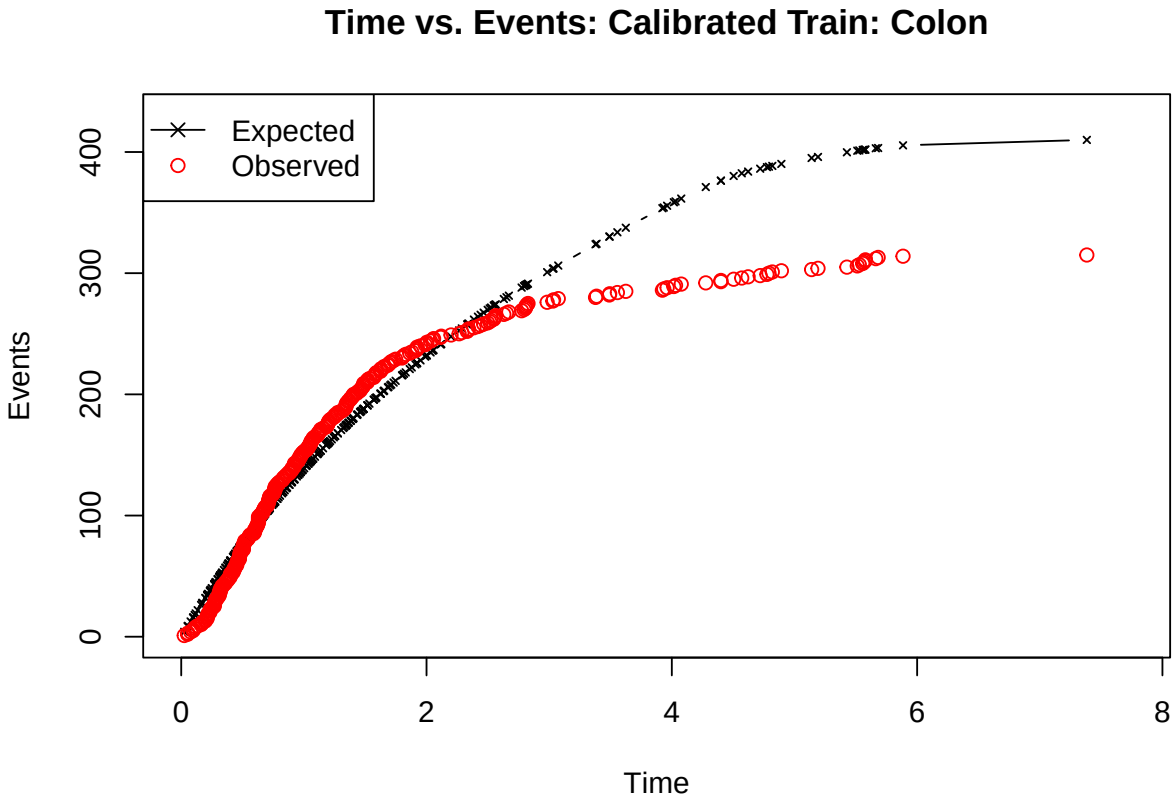
```
h0 <- calprob$h0
timeinterval <- calprob$timeInterval;

rdata <- cbind(dataColonTrain$status,calprob$prob)


rrAnalysisTrain <- RRPlot(rdata,atRate=c(0.90),
                      timetoEvent=dataColonTrain$time,
                      title="Calibrated Train: Colon",
                      ysurvlim=c(0.00,1.0),
                      riskTimeInterval=timeinterval)
```

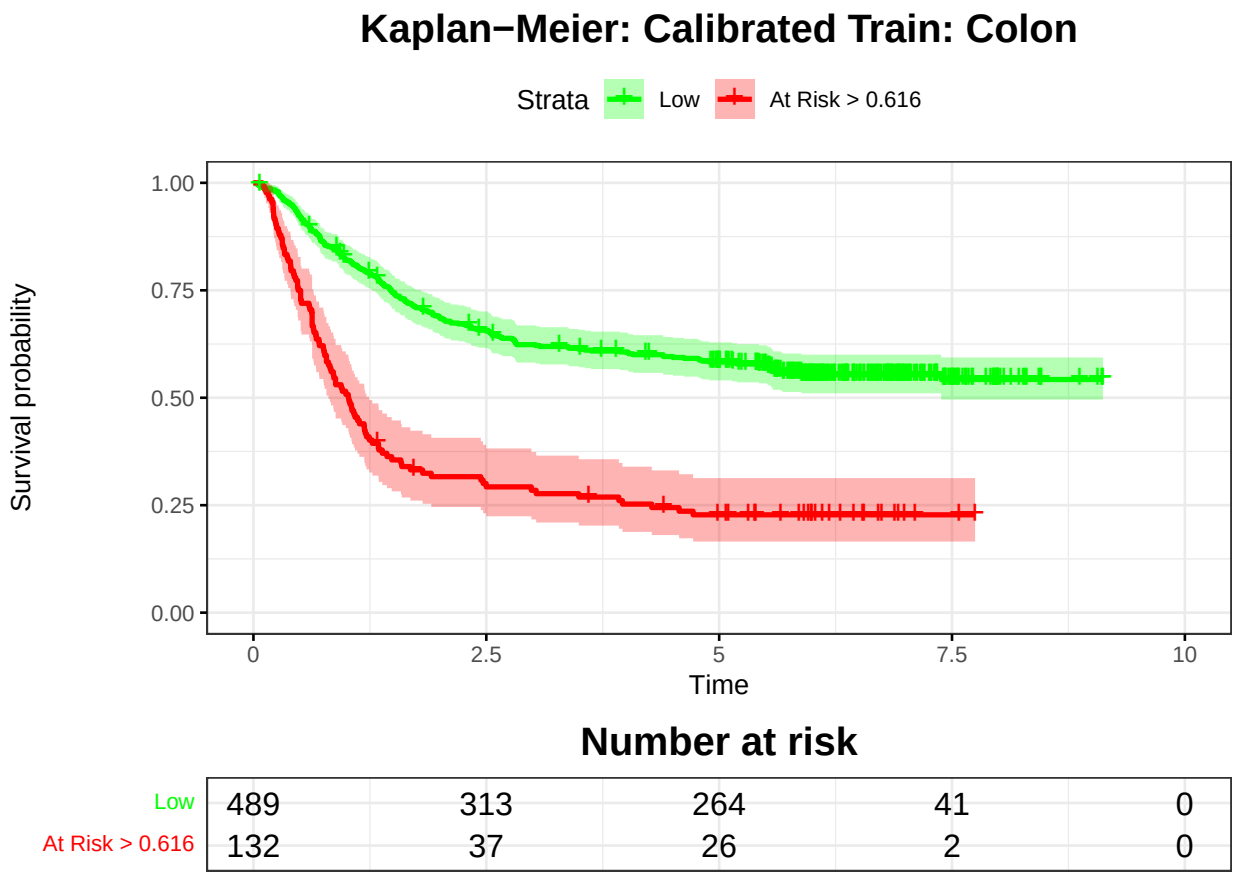



Decision Curve Analysis: Calibrated Train: Colon
Net Benefit
0.0
0.1
0.2
0.3
0.4
0.5
0.0
0.2
0.4
0.6
Threshold
No Event
Event
loess fit
H. Thr
SEN
1.0
0.8
0.6
0.4
0.2
0.0
Treat All
Treat None


Relative Risk: Calibrated Train: Colon
PPV–>
Index(0.90)=0.616
RR(0.32)=1.748
1.0
0.8
0.6
0.4
0.2
1.0
0.8
0.6
0.4
0.2
0.0
SPE
No
Yes
Loess Fit
RR
SEN
1
2
3
4
5
0.0
0.2
0.4
0.6
0.8
1.0


ROC: Calibrated Train: Colon
(TPR= 0.321 ,TNR= 0.899 ,ACC= 0.605 ,F1= 0.452 ,BER= 0.390)
Sensitivity
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
Confusion Matrix
O(+)
O(−)
T(+)
T(−)
Test
Outcome
AUC: 0.693
BACC (0.610)


Time vs. Events: Calibrated Train: Colon
Expected
Observed
Events
0
100
200
300
400
0
2
4
6
8
Time



1.3.6 Time to event after calibration

```
timetoEvent <- meanTimeToEvent(rdata[,2],timeinterval)
tmax<-max(c(obstiemToEvent,timetoEvent))
lmfit <- lm(obstiemToEvent[toinclude]~0+timetoEvent[toinclude])
sm <- summary(lmfit)
pander::pander(sm)
```

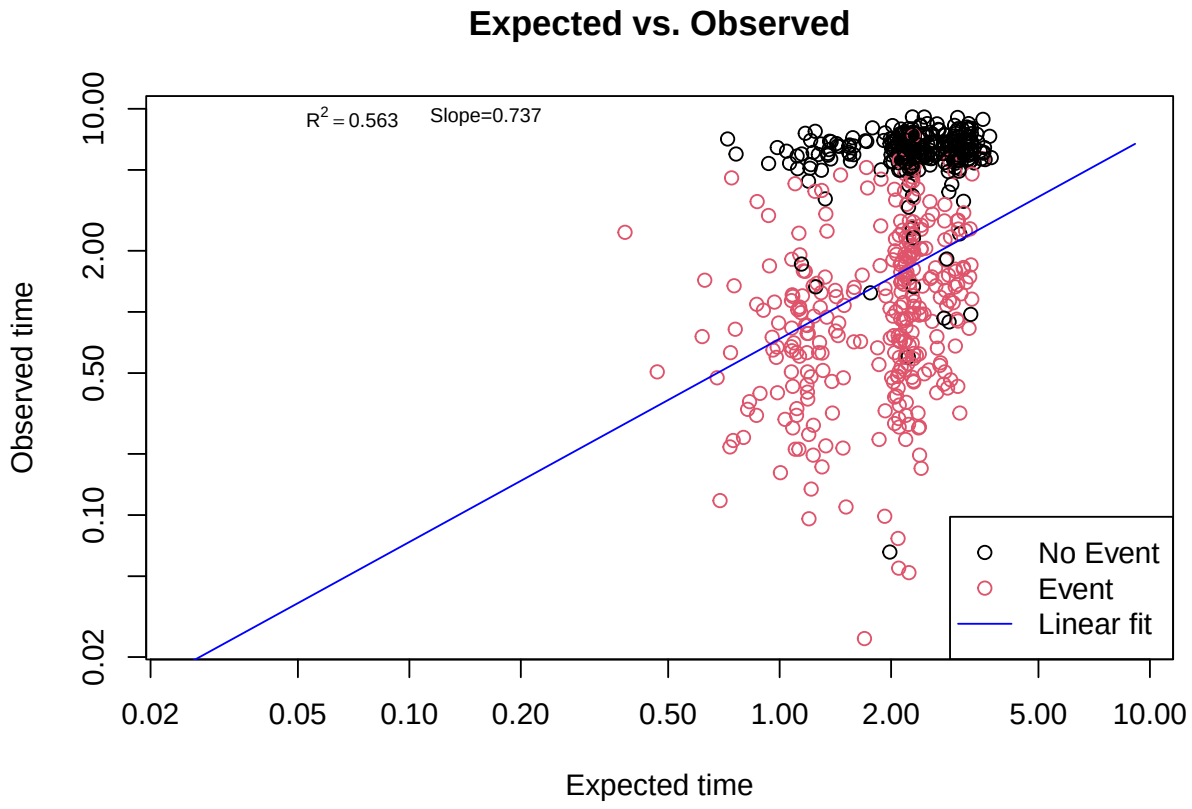
	Estimate	Std. Error	t value	Pr(>|t|)
timetoEvent[toinclude]	0.737	0.0366	20.1	1.84e-58

Table 20: Fitting linear model: obstiemToEvent[toinclude] ~ 0 + timetoEvent[toinclude]

Observations	Residual Std. Error	R^2	Adjusted R^2
315	1.35	0.563	0.562

```
plot(timetoEvent,obstiemToEvent,
     col=1+rdata[,1],
     xlab="Expected time",
     ylab="Observed time",
```

```
     main="Expected vs. Observed",
     xlim=c(tmin,tmax),
     ylim=c(tmin,tmax),
     log="xy")
lines(x=c(tmin,tmax),y=lmfit$coefficients*c(tmin,tmax),lty=1,col="blue")
txt <- bquote(paste(R^2 == .(round(sm$r.squared,3))))
text(tmin+0.005*(tmax-tmin),tmax,txt,cex=0.7)
text(tmin+0.015*(tmax-tmin),tmax,sprintf("Slope=%4.3f",sm$coefficients[1]),cex=0.7)
legend("bottomright",legend=c("No Event","Event","Linear fit"),
             pch=c(1,1,-1),
             col=c(1,2,"blue"),
             lty=c(-1,-1,1)
             )
```




```
MADerror2 <- c(MADerror2,mean(abs(timetoEvent[toinclude]-obstiemToEvent[toinclude])))
pander::pander(MADerror2)
```

1.99 and *1.17*

1.3.7 Calibrated Train Performance

```
pander::pander(t(rrAnalysisTrain$keyPoints),caption="Threshold values")
```

Table 21: Threshold values

	@:0.9	@MAX_BACC	@MAX_RR	@SPE100	p(0.5)
Thr	0.6164	0.475	0.362	3.32e-01	0.500
RR	1.7484	1.793	3.603	2.56e+01	1.626
RR_LCI	1.5231	1.524	1.811	5.53e-02	1.406
RR_UCI	2.0070	2.109	7.166	1.18e+04	1.881
SEN	0.3206	0.610	0.978	1.00e+00	0.429
SPE	0.8987	0.683	0.131	1.63e-02	0.801
BACC	0.6097	0.646	0.554	5.08e-01	0.615
NetBenefit	0.0824	0.168	0.253	2.66e-01	0.119

```
pander::pander(t(rrAnalysisTrain$OERatio$estimate),caption="O/E Ratio")
```

Table 22: O/E Ratio

O/E	Low	Upper	p.value
0.768	0.686	0.858	1.27e-06

```
pander::pander(t(rrAnalysisTrain$OE95ci),caption="O/E Mean")
```

Table 23: O/E Mean

mean	50%	2.5%	97.5%
0.986	0.986	0.972	1

```
pander::pander(t(rrAnalysisTrain$OAcum95ci),caption="O/Acum Mean")
```

Table 24: O/Acum Mean

mean	50%	2.5%	97.5%
1.06	1.06	1.06	1.06

```
pander::pander(rrAnalysisTrain$c.index$cstatCI,caption="C. Index")
```

mean.C Index	median	lower	upper
0.665	0.665	0.636	0.694

```
pander::pander(t(rrAnalysisTrain$ROCAnalysis$aucs),caption="ROC AUC")
```

Table 26: ROC AUC

est	lower	upper
0.693	0.653	0.734

```
pander::pander((rrAnalysisTrain$ROCAnalysis$sensitivity),caption="Sensitivity")
```

Table 27: Sensitivity

est	lower	upper
0.321	0.269	0.375

```
pander::pander((rrAnalysisTrain$ROCAnalysis$specificity),caption="Specificity")
```

Table 28: Specificity

est	lower	upper
0.899	0.859	0.93

```
pander::pander(t(rrAnalysisTrain$thr_atP),caption="Probability Thresholds")
```

Table 29: Probability Thresholds

90%
0.616

```
pander::pander(rrAnalysisTrain$surdif,caption="Logrank test")
```

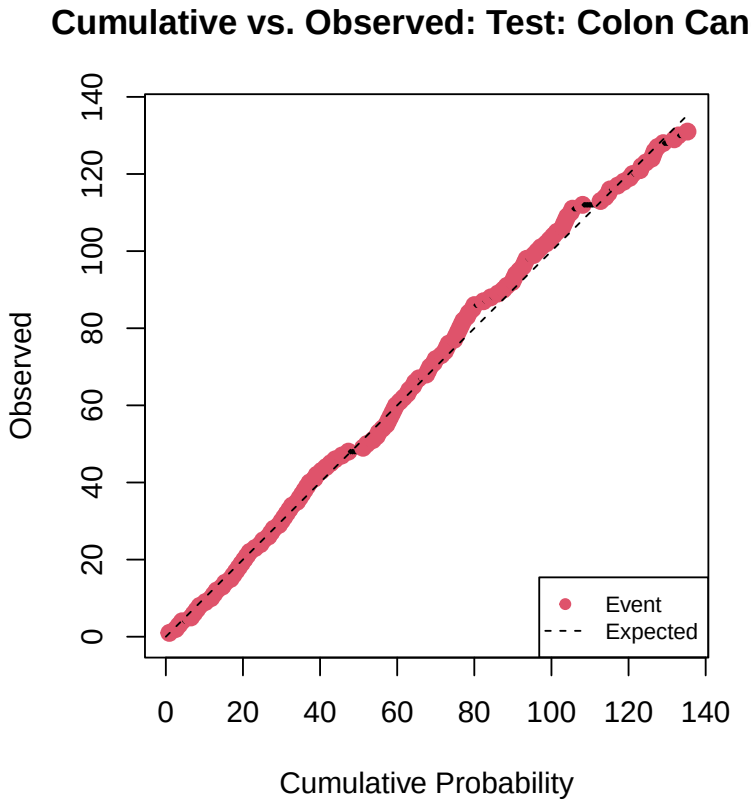
Table 30: Logrank test Chisq = 81.105923 on 1 degrees of freedom, p = 0.000000

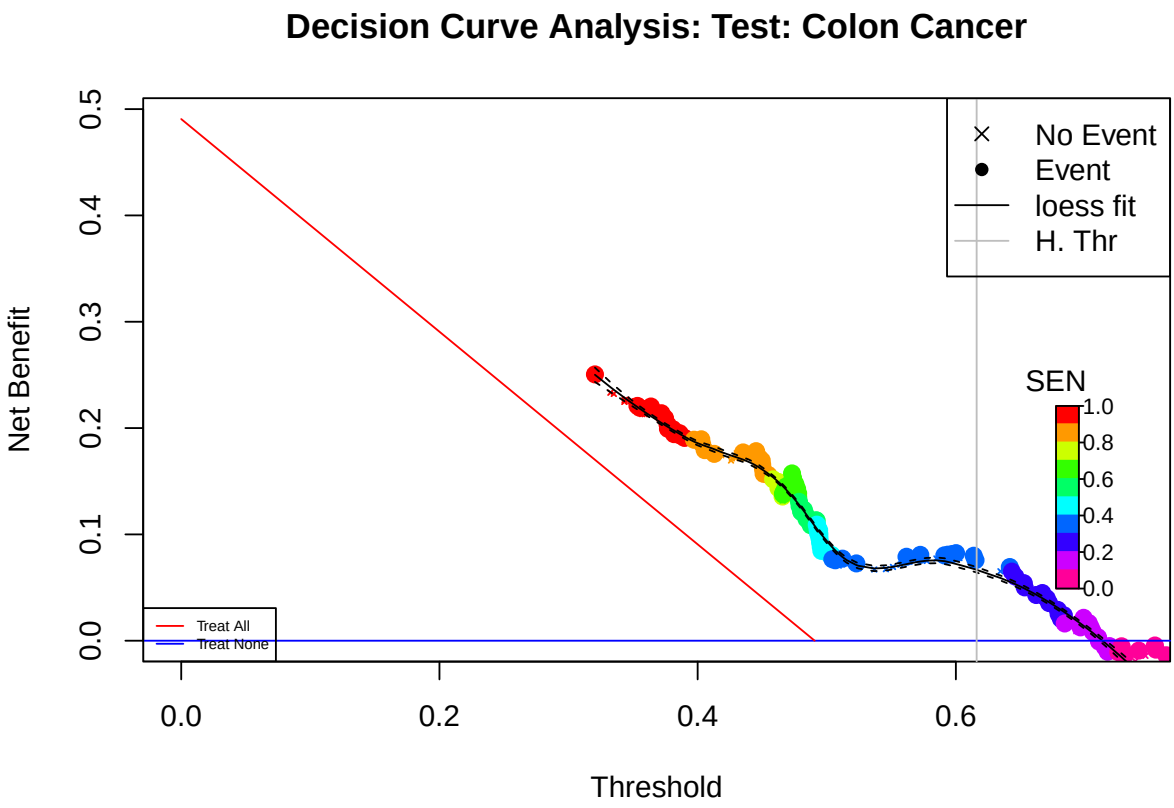
	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
class=0	489	214	269.7	11.5	81.1
class=1	132	101	45.3	68.6	81.1

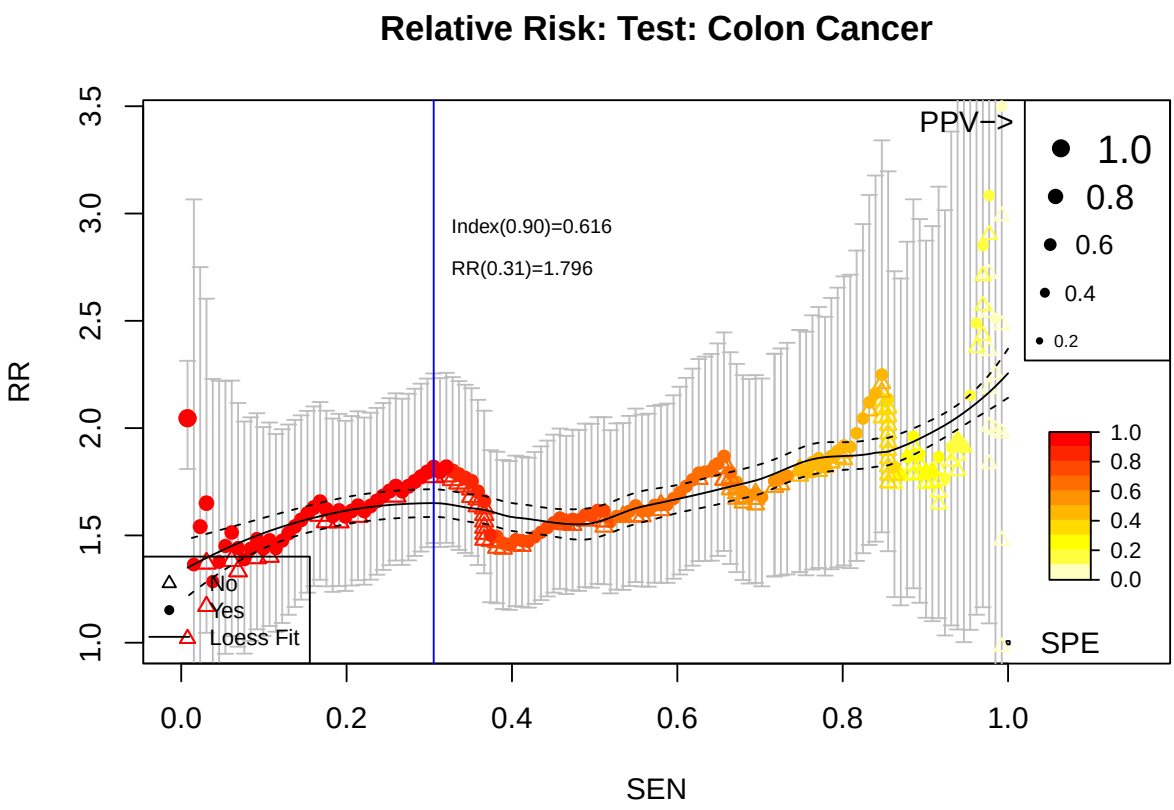
1.3.8 Evaluating on the test set

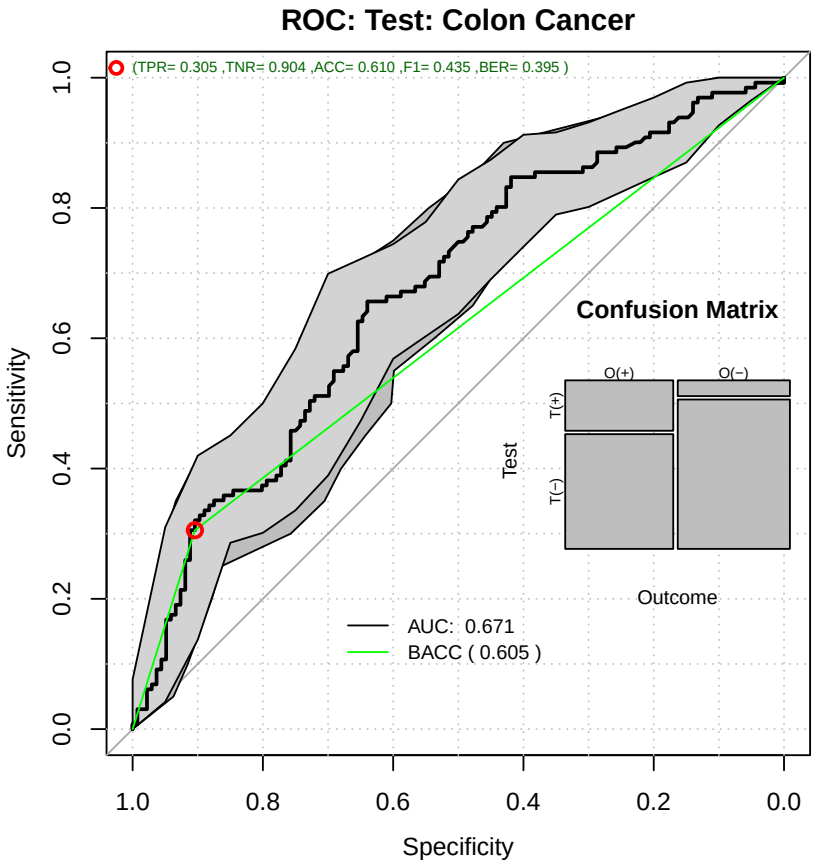
The calibrated h0 and timeinterval were estimated on the training set

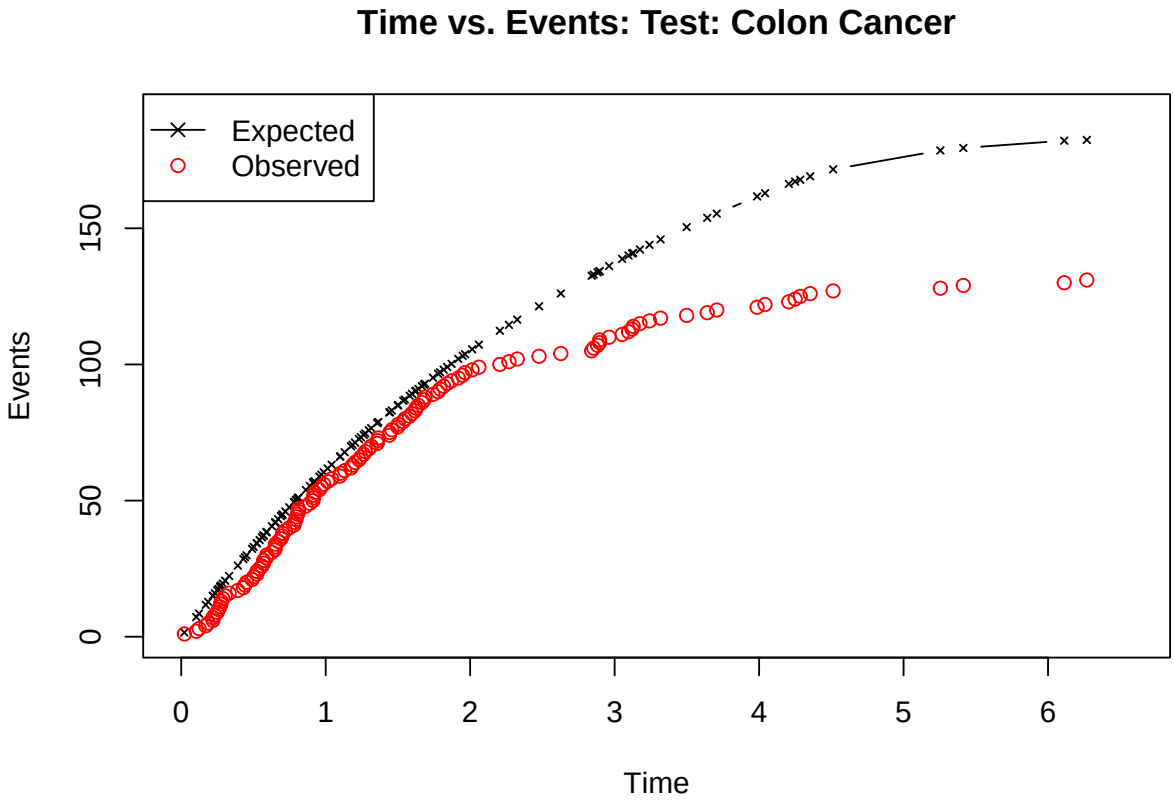
```
index <- predict(ml,dataColonTest)
rdata <- cbind(dataColonTest$status,ppoisGzero(index,h0))

rrAnalysisTest <- RRPlot(rdata,atThr = rrAnalysisTrain$thr_atP,
                    timetoEvent=dataColonTest$time,
                    title="Test: Colon Cancer",
                    ysurvlim=c(0.00,1.0),
                    riskTimeInterval=timeinterval)
```

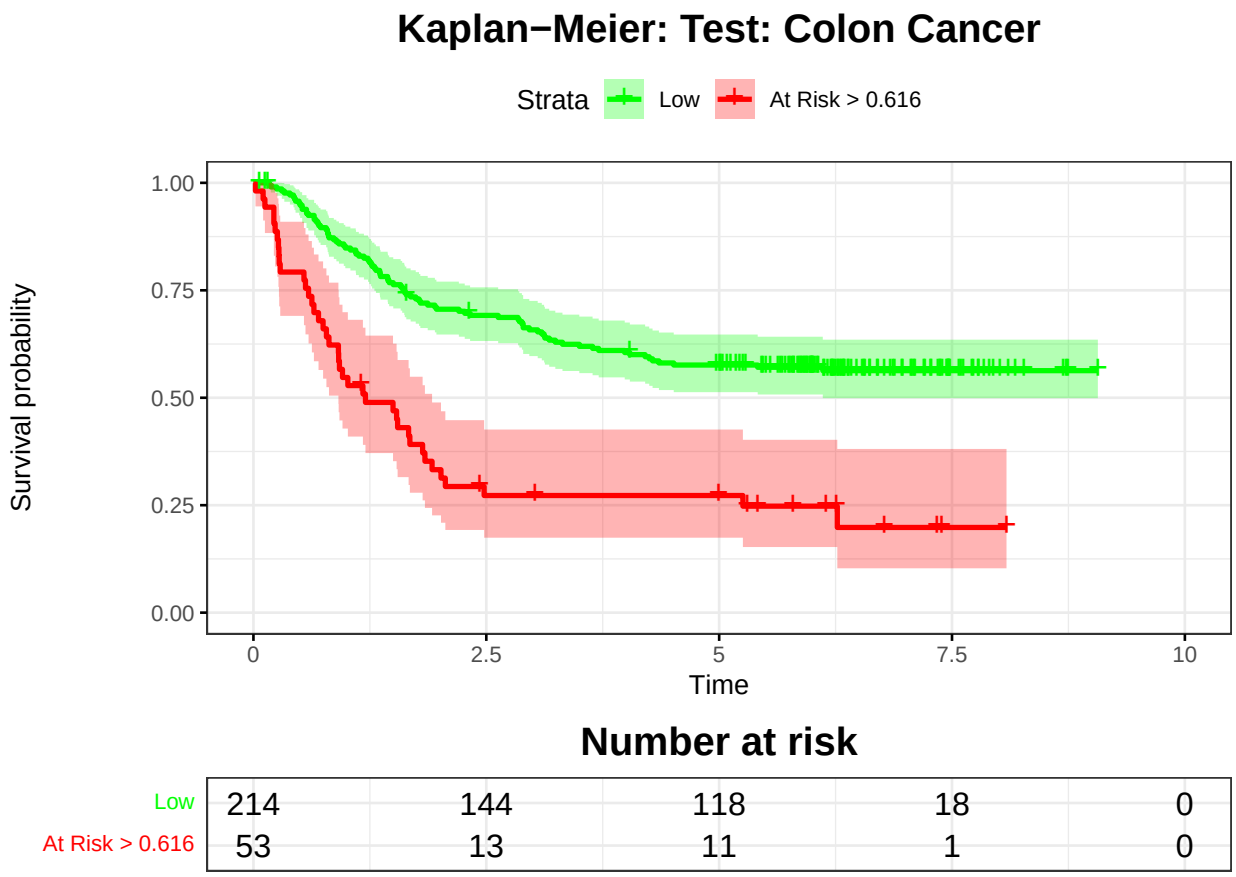

Cumulative vs. Observed: Test: Colon Cancer
Observed
0
20
40
60
80
100
120
140
Event
Expected
0
20
40
60
80
100
120
140
Cumulative Probability


Decision Curve Analysis: Test: Colon Cancer
Net Benefit
0.0
0.1
0.2
0.3
0.4
0.5
0.0
0.2
0.4
0.6
Threshold
No Event
Event
loess fit
H. Thr
SEN
1.0
0.8
0.6
0.4
0.2
0.0
Treat All
Treat None


Relative Risk: Test: Colon Cancer
RR
SEN
PPV–>
SPE
Index(0.90)=0.616
RR(0.31)=1.796
No
Yes
Loess Fit
1.0
0.8
0.6
0.4
0.2
0.0
1.0
1.5
2.0
2.5
3.0
3.5
0.0
0.2
0.4
0.6
0.8
1.0


ROC: Test: Colon Cancer
(TPR= 0.305 ,TNR= 0.904 ,ACC= 0.610 ,F1= 0.435 ,BER= 0.395)
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
Confusion Matrix
O(+)
O(−)
T(+)
T(−)
Test
Outcome
AUC: 0.671
BACC (0.605)


Time vs. Events: Test: Colon Cancer
Expected
Observed
Events
0
50
100
150
0
1
2
3
4
5
6
Time



1.3.9 Test Performance

```
pander::pander(t(rrAnalysisTest$keyPoints),caption="Threshold values")
```

Table 31: Threshold values

	@:0.616	@MAX_BACC	@MAX_RR	@SPE100	p(0.5)
Thr	0.6152	0.473	0.364	0.32	0.5015
RR	1.7969	1.869	3.084	1.00	1.4510
RR_LCI	1.4460	1.428	1.090	0.00	1.1463
RR_UCI	2.2330	2.446	8.727	0.00	1.8365
SEN	0.3130	0.656	0.977	1.00	0.4122
SPE	0.9044	0.640	0.110	0.00	0.7574
BACC	0.6087	0.648	0.544	0.50	0.5848
NetBenefit	0.0757	0.157	0.220	0.25	0.0779

```
pander::pander(t(rrAnalysisTest$OERatio$estimate),caption="O/E Ratio")
```

Table 32: O/E Ratio

O/E	Low	Upper	p.value
0.718	0.6	0.852	7.33e-05

```
pander::pander(t(rrAnalysisTest$OE95ci),caption="O/E Mean")
```

Table 33: O/E Mean

mean	50%	2.5%	97.5%
0.838	0.839	0.823	0.854

```
pander::pander(t(rrAnalysisTest$OAcum95ci),caption="O/Acum Mean")
```

Table 34: O/Acum Mean

mean	50%	2.5%	97.5%
1.01	1.01	1	1.01

```
pander::pander(rrAnalysisTest$c.index$cstatCI,caption="C. Index")
```

mean.C Index	median	lower	upper
0.645	0.647	0.6	0.693

```
pander::pander(t(rrAnalysisTest$ROCAnalysis$aucs),caption="ROC AUC")
```

Table 36: ROC AUC

est	lower	upper
0.671	0.607	0.736

```
pander::pander((rrAnalysisTest$ROCAnalysis$sensitivity),caption="Sensitivity")
```

Table 37: Sensitivity

est	lower	upper
0.305	0.228	0.392

```
pander::pander((rrAnalysisTest$ROCAnalysis$specificity),caption="Specificity")
```

Table 38: Specificity

est	lower	upper
0.904	0.842	0.948

```
pander::pander(t(rrAnalysisTest$thr_atP),caption="Probability Thresholds")
```

Table 39: Probability Thresholds

90%
0.616

```
pander::pander(rrAnalysisTest$surdif,caption="Logrank test")
```

Table 40: Logrank test Chisq = 34.193780 on 1 degrees of freedom, p = 0.000000

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
class=0	214	91	113.6	4.49	34.2
class=1	53	40	17.4	29.32	34.2

1.4 Cross-Validation

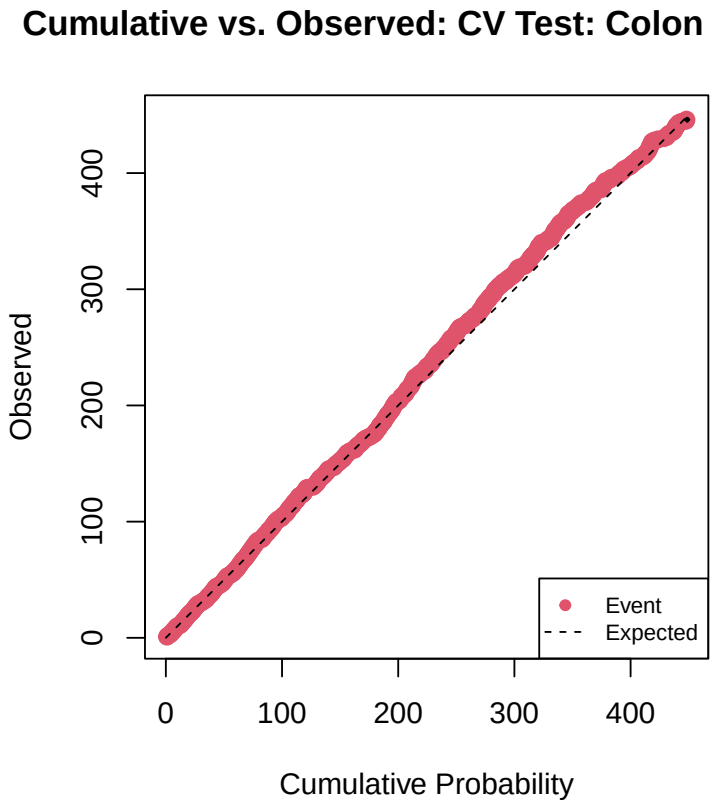
Here we will cross validate the training set and evaluate also on the testing set. The h0 and the timeinterval are the ones estimated on the calibration process

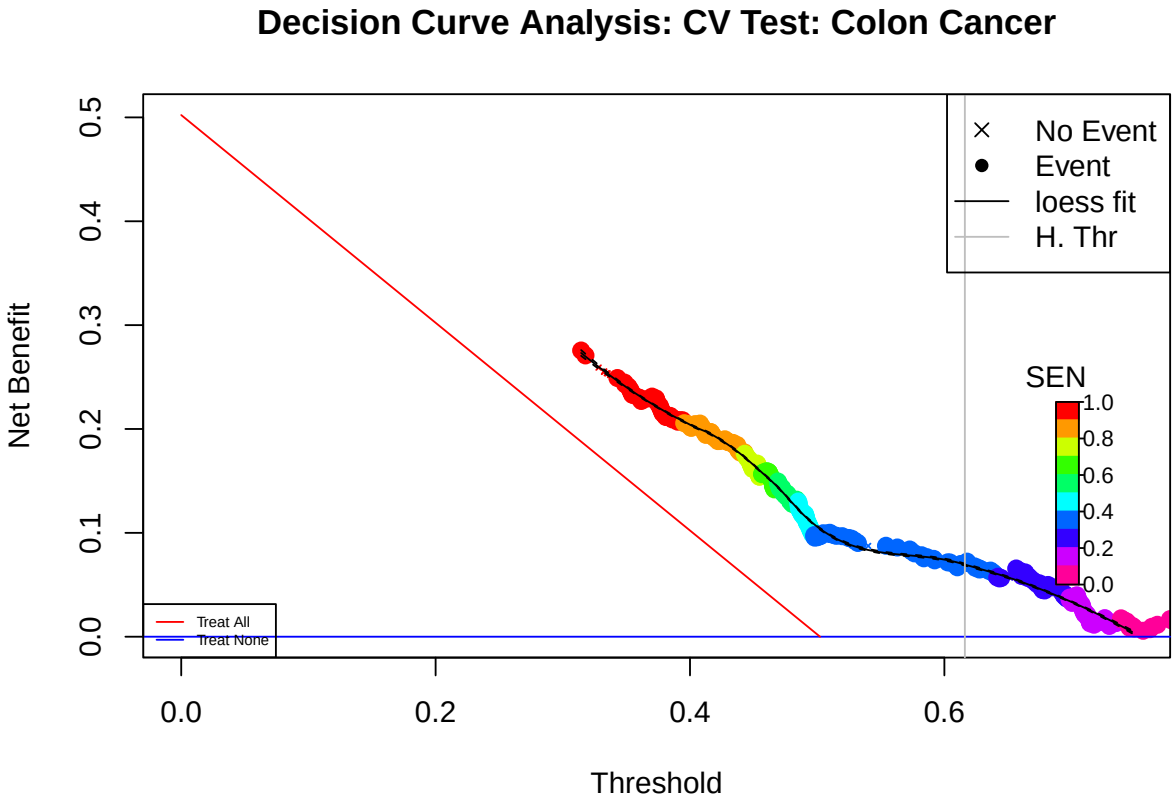
```
rcv <- randomCV(theData=dataColonTrain,
                theOutcome = Surv(time,status)~1,
                fittingFunction=BSWiMS.model,
                trainFraction = 0.75,
                repetitions=50,
                classSamplingType = "Pro",
                testingSet=dataColonTest
          )
```

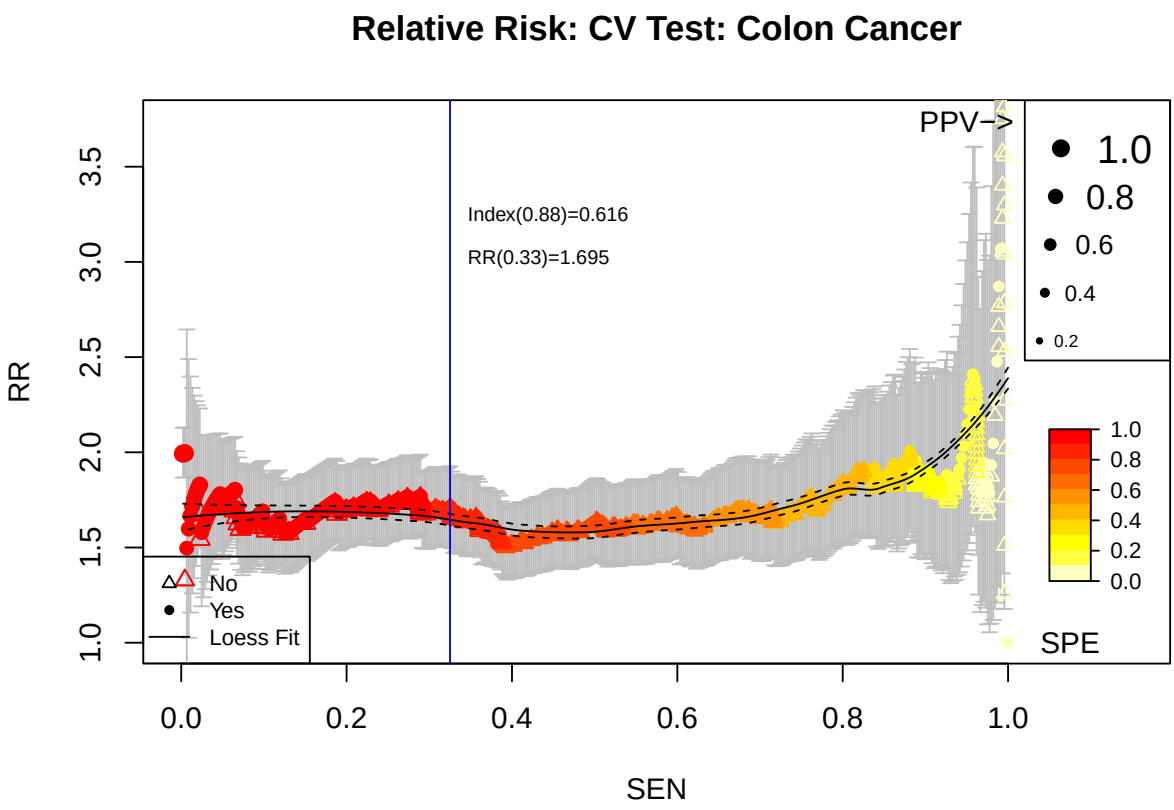
.[++++++++++-].[++++++++++-].[++++++++++-].[++++++++++-].[++++++++-].[++++++++++-].[++++++++++-]..[++++++++++-].[++++++++++-].[++++++++-]10 Tested: 865 Avg. Selected: 19 Min Tests: 1 Max Tests: 10 Mean Tests: 4.890173 . MAD: 0.4746584 .[++++++++++-]..[++++++++++]..[++++++++-].[+++++++++-].[+++++++++-].[+++++++++-].[++++++++++-]..[+++++++++++++]..[++++++++-].[+++++++++-]20 Tested: 885 Avg. Selected: 19.15 Min Tests: 1 Max Tests: 20 Mean Tests: 9.559322 . MAD: 0.4737181 .[+++++++++-].[+++++++++-].[+++++++++-].[++++++++-].[++++++++-].[++++++++++-].[+++++++++-].[++++++++-].[+++++++++-].[++++++++++-]30 Tested: 888 Avg. Selected: 18.86667 Min Tests: 1 Max Tests: 30 Mean Tests: 14.29054 . MAD: 0.4732959 .[+++++++-].[++++++++++-].[+++++++-].[+++++++++-].[++++++++++-]..[++++++++++-].[+++++++-].[+++++++++-].[+++++++++-].[++++++++++-].40 Tested: 888 Avg. Selected: 18.35 Min Tests: 1 Max Tests: 40 Mean Tests: 19.05405 . MAD: 0.4739015 .[++++++-].[++++++++++-].[+++++++++].[+++++++++-].[++++++++++-]..[+++++++++-].[++++++++++-].[+++++++++-].[++++++-].[+++++++++-]50 Tested: 888 Avg. Selected: 18.38 Min Tests: 1 Max Tests: 50 Mean Tests: 23.81757 . MAD: 0.4733696

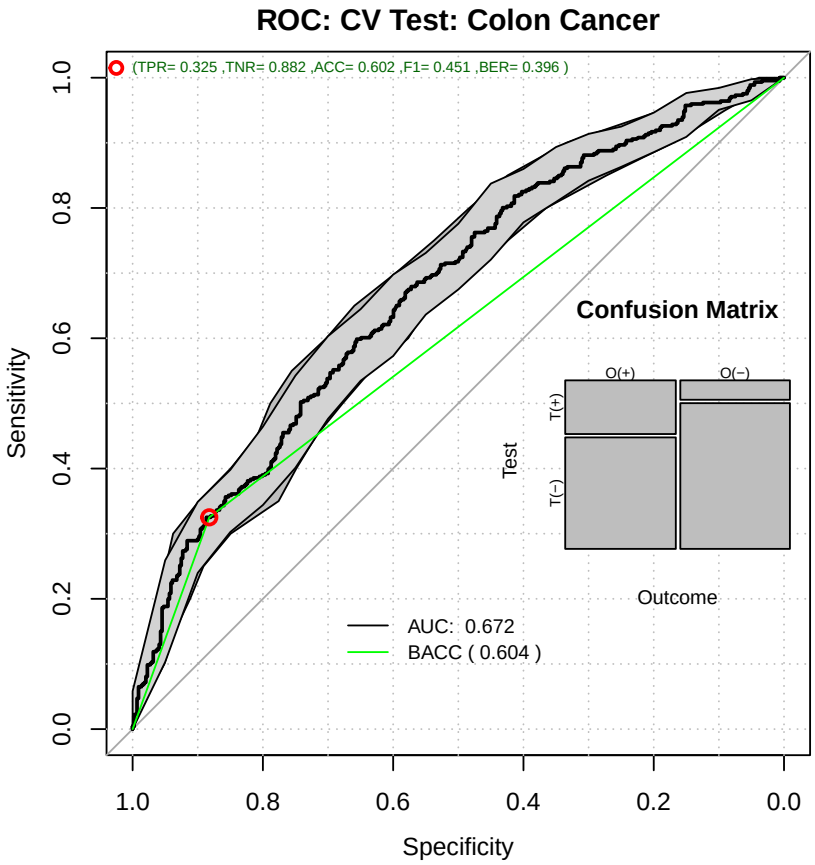
```
stp <- rcv$survTestPredictions
stp <- stp[!is.na(stp[,4]),]

bbx <- boxplot(unlist(stp[,1])~rownames(stp),plot=FALSE)
times <- bbx$stats[3,]
status <- boxplot(unlist(stp[,2])~rownames(stp),plot=FALSE)$stats[3,]
prob <- ppoisGzero(boxplot(unlist(stp[,4])~rownames(stp),plot=FALSE)$stats[3,],h0)

rdatacv <- cbind(status,prob)
rownames(rdatacv) <- bbx$names
```

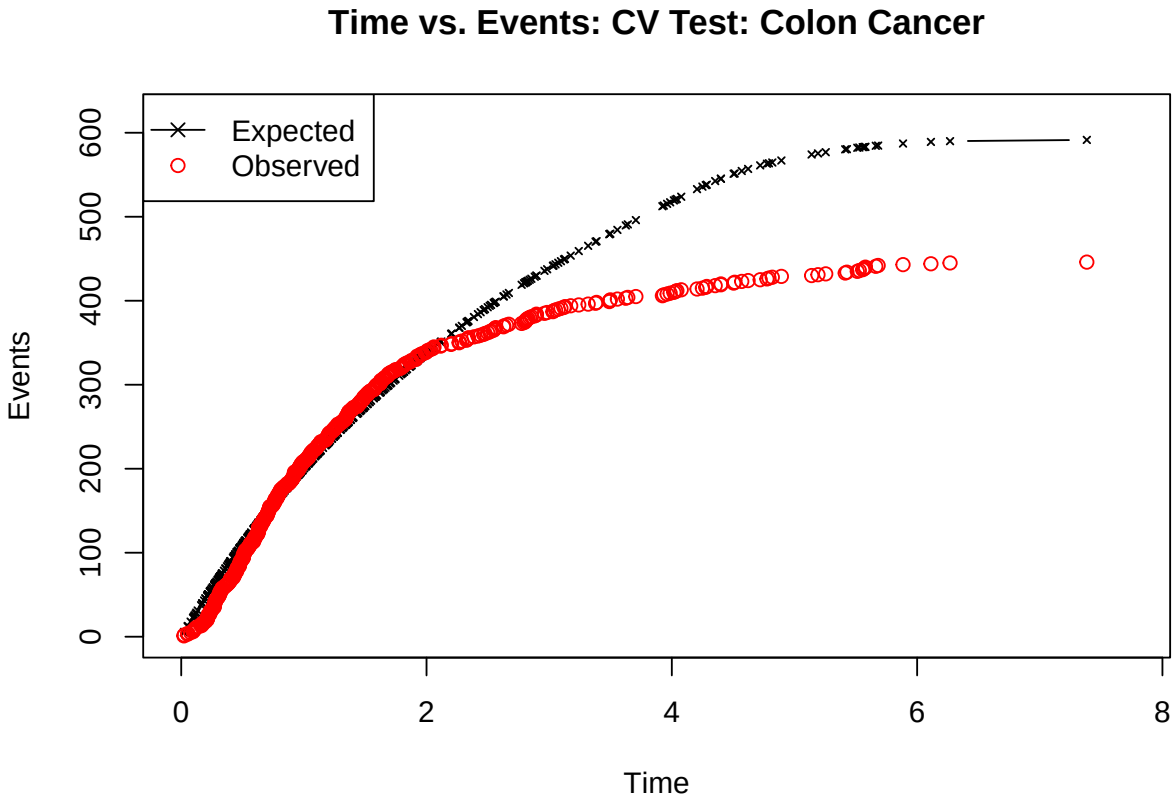
```
names(times) <- bbx$names

rrAnalysisCVTest <- RRPlot(rdatacv,atThr = rrAnalysisTrain$thr_atP,
                      timetoEvent=times,
                      title="CV Test: Colon Cancer",
                      ysurvlim=c(0.00,1.0),
                      riskTimeInterval=timeinterval)
```

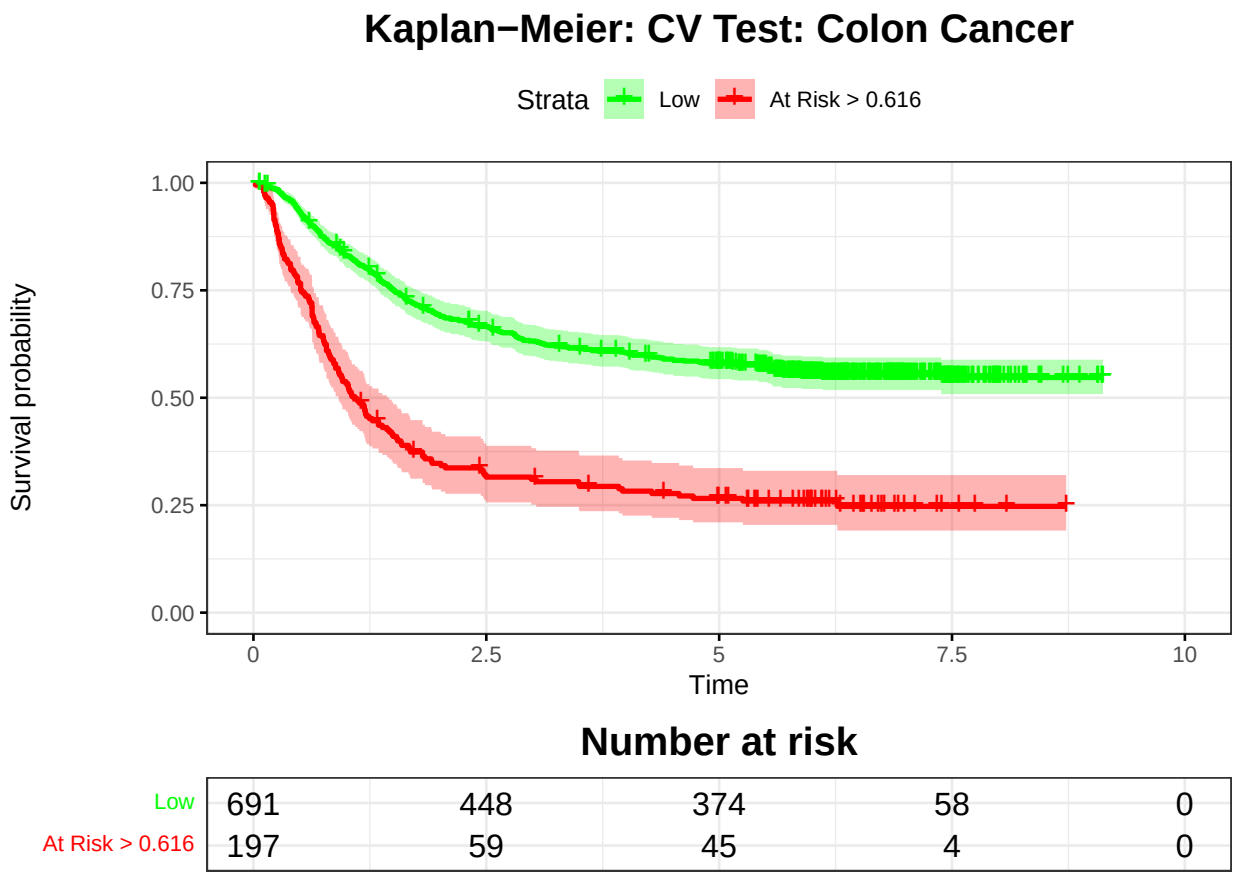



Decision Curve Analysis: CV Test: Colon Cancer
Net Benefit
0.0
0.1
0.2
0.3
0.4
0.5
0.0
0.2
0.4
0.6
Threshold
No Event
Event
loess fit
H. Thr
SEN
1.0
0.8
0.6
0.4
0.2
0.0
Treat All
Treat None


Relative Risk: CV Test: Colon Cancer
PPV–>
1.0
0.8
0.6
0.4
0.2
Index(0.88)=0.616
RR(0.33)=1.695
1.0
0.8
0.6
0.4
0.2
0.0
SPE
No
Yes
Loess Fit
RR
1.0
1.5
2.0
2.5
3.0
3.5
0.0
0.2
0.4
0.6
0.8
1.0
SEN


ROC: CV Test: Colon Cancer
(TPR= 0.325 ,TNR= 0.882 ,ACC= 0.602 ,F1= 0.451 ,BER= 0.396)
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
Confusion Matrix
O(+)
O(−)
T(+)
T(−)
Test
Outcome
AUC: 0.672
BACC (0.604)


Time vs. Events: CV Test: Colon Cancer
Expected
Observed
Events
0
100
200
300
400
500
600
0
2
4
6
8
Time



1.4.1 CV Test Performance

```
pander::pander(t(rrAnalysisCVTest$keyPoints),caption="Threshold values")
```

Table 41: Threshold values

	@:0.616	@MAX_BACC	@MAX_RR	@SPE100	p(0.5)
Thr	0.6166	0.469	0.373	0.31443	0.5000
RR	1.6897	1.670	2.410	1.00000	1.5120
RR_LCI	1.4999	1.457	1.612	0.00000	1.3351
RR_UCI	1.9035	1.913	3.603	0.00000	1.7125
SEN	0.3251	0.599	0.957	1.00000	0.3901
SPE	0.8824	0.656	0.152	0.00679	0.7964
BACC	0.6037	0.627	0.554	0.50339	0.5933
NetBenefit	0.0691	0.149	0.229	0.27555	0.0946

```
pander::pander(t(rrAnalysisCVTest$OERatio$estimate),caption="O/E Ratio")
```

Table 42: O/E Ratio

O/E	Low	Upper	p.value
0.754	0.686	0.827	5.12e-10

```
pander::pander(t(rrAnalysisCVTest$OE95ci),caption="O/E Mean")
```

Table 43: O/E Mean

mean	50%	2.5%	97.5%
0.937	0.936	0.926	0.947

```
pander::pander(t(rrAnalysisCVTest$OAcum95ci),caption="O/Acum Mean")
```

Table 44: O/Acum Mean

mean	50%	2.5%	97.5%
1.03	1.03	1.03	1.03

```
pander::pander(rrAnalysisCVTest$c.index$cstatCI,caption="C. Index")
```

mean.C Index	median	lower	upper
0.649	0.649	0.622	0.673

```
pander::pander(t(rrAnalysisCVTest$ROCAnalysis$aucs),caption="ROC AUC")
```

Table 46: ROC AUC

est	lower	upper
0.672	0.637	0.708

```
pander::pander((rrAnalysisCVTest$ROCAnalysis$sensitivity),caption="Sensitivity")
```

Table 47: Sensitivity

est	lower	upper
0.325	0.282	0.371

```
pander::pander((rrAnalysisCVTest$ROCAnalysis$specificity),caption="Specificity")
```

Table 48: Specificity

est	lower	upper
0.882	0.849	0.911

```
pander::pander(t(rrAnalysisCVTest$thr_atP),caption="Probability Thresholds")
```

Table 49: Probability Thresholds

90%
0.616

```
pander::pander(rrAnalysisCVTest$surdif,caption="Logrank test")
```

Table 50: Logrank test Chisq = 99.248680 on 1 degrees of freedom, p = 0.000000

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
class=0	691	301	376.8	15.3	99.2
class=1	197	145	69.2	83.0	99.2