

survAccuracyMeasures

This R package computes non-parametric (NP) and semi-parametric (SP) estimates of common accuracy measures for risk prediction markers from survival data. It consists of the function `survAM.estimate` which estimates the AUC , $TPR(c)$, $FPR(c)$, $PPV(c)$, and $NPV(c)$ for a specific prediction time and marker cutoff value c .

NP estimates are calculated using inverse probability weighting, while SP estimates are based on a Cox proportional hazards model. For detailed information regarding estimation methods, see references below.

Standard errors for estimates can be obtained by bootstrapping. Asymptotic standard error calculations are also available for semi-parametric Cox estimates. Confidence intervals using a normal approximation are computed.

Tutorial

```
#install the package from github
# download the package from github
if (!require("devtools")) install.packages("devtools")
devtools::install_github("survAccuracyMeasures", "mdbrown")
```

```
library(survAccuracyMeasures)
```

```
## Loading required package: survival
## Loading required package: splines
## Loading required package: Rcpp
## Loading required package: RcppArmadillo
```

```
#simulated data for illustration
data(SimData)
head(SimData)
```

```
##   survTime status      Y
## 1    0.1197      1 1.49310
## 2    1.0231      0 -0.73260
## 3    0.8282      0 -0.50211
## 4    2.0875      1  0.65758
## 5    4.6827      1  1.57806
## 6    0.3001      1  0.02419
```

Estimate all measures, using the bootstrap to estimate standard errors. First we obtain non-parametric estimates using inverse probability weighting:

```
#Estimate all the measures at prediction time 2, with a marker cutpoint at 0.

#non-parametric estimates with bootstrap standard errors
##in practice 'bootstraps' should be set to a larger value
survAM.estimate(time =survTime,
```

```

event = status,
marker = Y,
data = SimData,
estimation.method = "IPW",
se.method = "bootstrap",
predict.time = 1,
marker.cutpoint = 0,
bootstraps = 50)

```

```

##
## Non-Parametric IPW estimates of accuracy measures:
##   (SE's calculated using the bootstrap)
##
##      estimate      se      lower 0.95  upper 0.95
## AUC          0.786    0.029      0.723    0.837
## TPR(c)       0.769    0.049      0.661    0.851
## FPR(c)       0.410    0.024      0.364    0.458
## PPV(c)       0.231    0.029      0.178    0.293
## NPV(c)       0.941    0.015      0.903    0.965
##
## marker cutpoint: c = 0

```

Alternatively, we can calculate semi-parametric estimates based on a Cox proportional hazards model:

```

#semi-parametric estimates assuming a cox model
##in practice 'bootstraps' should be set to a larger value
survAM.estimate(time = survTime,
event = status,
marker = Y,
data = SimData,
estimation.method = "Cox",
se.method = "bootstrap",
predict.time = 1,
marker.cutpoint = 0,
bootstraps = 50)

```

```

##
## Semi-Parametric Cox estimates of accuracy measures:
##   (SE's calculated using the bootstrap)
##
##      estimate      se      lower 0.95  upper 0.95
## coef          1.010    0.084      0.845    1.175
## AUC           0.768    0.020      0.727    0.805
## TPR(c)       0.788    0.026      0.732    0.834
## FPR(c)       0.412    0.028      0.359    0.468
## PPV(c)       0.241    0.030      0.188    0.304
## NPV(c)       0.943    0.009      0.923    0.959
##
## marker cutpoint: c = 0

```

```

## here we calculate the SE's based on the asymptotic properties of the Cox estimator.
survAM.estimate(time = survTime,

```

```

event = status,
marker = Y,
data = SimData,
estimation.method = "Cox",
se.method = "asymptotic",
predict.time = 1,
marker.cutpoint = 0)

```

```

##
## Semi-Parametric Cox estimates of accuracy measures:
##   (SE's calculated using asymptotic variance)
##
##      estimate      se      lower 0.95  upper 0.95
## coef      1.010    0.085      0.842    1.177
## AUC        0.768    0.020      0.727    0.805
## TPR(c)     0.788    0.027      0.731    0.836
## FPR(c)     0.412    0.023      0.368    0.459
## PPV(c)     0.241    0.028      0.191    0.300
## NPV(c)     0.943    0.009      0.924    0.958
##
## marker cutpoint: c = 0

```

For more information see `?survAM.estimate`.

References

- Liu D, Cai T, Zheng Y. Evaluating the predictive value of biomarkers with stratified case-cohort design. *Biometrics* 2012, 4: 1219-1227.
- Pepe MS, Zheng Y, Jin Y. Evaluating the ROC performance of markers for future events. *Lifetime Data Analysis*. 2008, 14: 86-113.
- Zheng Y, Cai T, Pepe MS, Levy, W. Time-dependent predictive values of prognostic biomarkers with failure time outcome. *JASA* 2008, 103: 362-368.