

calibmsm: An R package for calibration plots of a multistate model using inverse probability of censoring weights

Alexander Pate
University of Manchester

Glen P. Martin
University of Manchester

Abstract

This vignette is a replica of the article published in ... **ADD REF XXXX**.

Multistate models, which allow users to answer a wide range of clinical questions, are becoming a more commonly used tool for clinical prediction. It is paramount to evaluate the calibration (as well as other metrics) of a risk prediction model before implementation of the model in practice. Currently no software exists to aid in assessing the calibration of a multistate model. **calibmsm** has been developed to simplify this process for practicing model developers. Calibration of the transition probabilities at time t , when in state j at time s are assessed. This is done using a combination of landmarking, calibration methods for binary logistic and multinomial logistic regression models and inverse probability of censoring weights.

This article details the methodology and provides a comprehensive example on how to assess the calibration of a model developed to predict recovery, adverse events, relapse and survival in patients with blood cancer after a transplantation. This model of interest is developed identically as in the comprehensive demonstration on how to develop a multistate model given by the developers of **mstate**.

Keywords: —!!!—at least one keyword is required—!!!—.

— vignette: > —

1. Introduction

Risk prediction models enable the prediction of clinical events in either diagnostic or prognostic settings (?) and are used widely to inform clinical practice. Multistate models are becoming more commonly developed in a risk prediction setting (????). Using a multistate model for prediction is important when the development of an intermediate condition occurring post index date may have an impact on the risk of future outcomes of interest. All risk prediction models developed for use in clinical practice should be validated in a relevant cohort prior to implementation (??). A key part of the validation process is assessment of the calibration of the model (?). Ideally calibration curves should be produced which allow evaluation of the calibration over the entire distribution of predicted risk, corresponding to moderate assessment of calibration (?).

The R package **mstate** (?) provides a comprehensive set of tools to develop a multistate model for a continuously observed multistate survival process. However, currently no software

exists to aid researchers in assessing the calibration of a multistate model that has been developed. **calibmsm** has been developed to enable researchers to estimate calibration curves and scatter plots using the BLR-IPCW and MLR-IPCW approaches outlined by Pate et al **REF XXXX**, which focus on assessing the calibration of the transition probabilities out of the starting state. The work in this paper extends the framework to assess the calibration of transition probabilities out of any state j at any time s using landmarking (??), gives details on estimation of the weights, and demonstrates the process for estimating confidence intervals. **calibmsm** is available from the Comprehensive R Archive Network at **XXXX**.

? used data from the European Society for Blood and Marrow Transplantation (?) to showcase how to develop, and the benefits of developing, a multistate model for clinical prediction. In this study, we show how to assess the calibration of a model developed on the same EBMT data. Details on the methodology are given in section 2. The clinical setting for our example and steps for data preparation and are described in section 3. In section 4, we show how to estimate calibration curves and scatter plots using **calibmsm**. In section 5 we demonstrate how to implement a bootstrapping procedure to estimate confidence intervals for the calibration curves.

2. Methodology

2.1. Setup

Let $X(t) \in \{1, \dots, K\}$ be a multistate survival process with K states. We assume a multistate model has already been developed and we want to assess the calibration of the predicted transition probabilities, $\hat{p}_{j,k}(s, t)$, in a cohort of interest. The transition probabilities are the probability of being in state k at time t , if in state j at time s . The aim is to estimate observed event probabilities:

$$o_{j,k}(s, t) = P[X(t) = k | X(s) = j, \hat{p}_{j,k}(s, t)].$$

These can be estimated by fitting an appropriate model in a landmark cohort of individuals in state j at time s . We propose doing this using cross sectional calibration techniques in the group of landmarked individuals who are also uncensored at time t (i.e. methods to assess the calibration of models predicting binary or polytomous outcomes). Unless censoring is non-informative and there are no absorbing states, calibration must be assessed in a cohort reweighted using inverse probability of censoring weights. Appropriate calibration models and the process for calculating the weights are detailed in sections 2.2 - 2.4.

2.2. Estimation of the inverse probability of censoring weights

The estimand for the weights is $w_j(s, t)$, the inverse of the probability of being uncensored at time t if in state j at time s :

$$w_j(s, t) = \frac{1}{P[t_{cens} > t | t > s, X(s) = j, \mathbf{Z}, \mathbf{X}(\mathbf{t})]},$$

where $\mathbf{X}(\mathbf{t})$ denotes the history of the multistate survival process up to time t , including the transition times. First the estimator $\hat{P}[t_{cens} > t | t > s, X(s) = j, \mathbf{Z}]$ is calculated by

developing an appropriate survival model. The outcome in this model is the time until censoring occurs. Moving into an absorbing state acts as the censoring mechanism in this model. $\mathbf{X}(t)$ is explicitly conditioned on when defining $w_j(s, t)$ because the weights must reflect that censoring can no longer be observed for an individual if they enter an absorbing state at some time $t_{abs} < t$. Therefore

$$\hat{P}[t_{cens} > t | t > s, X(s) = j, \mathbf{Z}, \mathbf{X}(t)] = \hat{P}[t_{cens} > t_{abs} | t > s, X(s) = j, \mathbf{Z}]$$

if $X(t)$ is in an absorbing state, and

$$\hat{P}[t_{cens} > t | t > s, X(s) = j, \mathbf{Z}, \mathbf{X}(t)] = \hat{P}[t_{cens} > t | t > s, X(s) = j, \mathbf{Z}]$$

otherwise. In **calibmsm**, $\hat{P}[t_{cens} > t | t > s, X(s) = j, \mathbf{Z}]$ is estimated using a cox proportional hazards model where all predictors \mathbf{Z} are assumed to have a linear effect on the log-hazard. However users can also input their own vector of weights. Given the BLR-IPCW and MLR-IPCW approaches are both reliant on correct estimation of the weights, we encourage users to take the time to carefully estimate weights themselves using non-linear models of \mathbf{Z} and interaction terms when appropriate.

Stabilised weights can be estimated by multiplying by the weights $w_j(s, t)$ by the mean probability of being uncensored:

$$w_j^{stab}(s, t) = \frac{P[t_{cens} > t | t > s, X(s) = j]}{P[t_{cens} > t | t > s, X(s) = j, \mathbf{Z}, \mathbf{X}(t)]}.$$

The numerator can be estimated using an intercept only model, and note there is no dependence on $\mathbf{X}(t)$.

Another option is to estimate $w(s, t)$, which is the inverse of the probability of being uncensored at time t if uncensored at time s :

$$w(s, t) = \frac{1}{P[t_{cens} > t | t > s, \mathbf{Z}, \mathbf{X}(t)]}.$$

This can be estimated using the same approach as for $w_j(s, t)$, except there is no requirement to be in state j when landmarking at time s . If the censoring mechanism is non-informative after conditioning on the predictors \mathbf{Z} , then $w(s, t) = w_j(s, t)$, and any consistent estimator for $w(s, t)$ will be a consistent estimator of $w_j(s, t)$. The advantage is that $\hat{w}(s, t)$ is calculated by developing a model in the cohort of individuals uncensored at time s , which is a larger cohort than those uncensored and in state j at time s . Therefore $\hat{w}(s, t)$ will be a more precise estimator than $\hat{w}_j(s, t)$. On the contrary, if the assumption of non-informative censoring after conditioning on \mathbf{Z} , there is a risk of bias.

2.3. BLR-IPCW calibration curves

The first approach produces calibration curves using a framework for binary logistic regression models with inverse probability of censoring weights (BLR-IPCW). Let $I_k(t)$ be an indicator for whether an individual is in state k at time t . $I_k(t)$ is then modeled using a flexible approach with $\hat{p}_{j,k}(s, t)$ as the sole predictor. This model is fit in the reweighted landmark cohort (in state j at time s) of individuals uncensored at time t . We suggest using a loess smoother:

$$I_k(t) = \text{loess}(\hat{p}_{j,k}(s, t)),$$

or a logistic regression model with restricted cubic splines:

$$\text{logit}(I_k(t)) = \text{rcs}(\text{logit}(\hat{p}_{j,k}(s, t))).$$

Any flexible model for binary outcomes could be used, but these are the most common and are implemented in this package. Observed event probabilities $\hat{o}_{j,k}(s, t)$ are then estimated as fitted values from these models. The calibration curve is plotted using the set of points $\{\hat{p}_{j,k}(s, t), \hat{o}_{j,k}(s, t)\}$.

2.4. MLR-IPCW calibration plots

The second approach produces calibration scatter plots using a framework for multinomial logistic regression models with inverse probability of censoring weights (MLR-IPCW). Let $I_X(t)$ be an polytomous indicator variable taking values $I_X(t) \in \{1, \dots, K\}$ such that $I_X(t) = k$ if an individual is in state k at time t . The nominal recalibration framework of ?? is then applied in the reweighted landmark cohort of individuals uncensored at time t . First calculate the log-ratios of the predicted transition probabilities:

$$LP_k = \ln \left(\frac{\hat{p}_{j,k}(s, t)}{\hat{p}_{j,k_{ref}}(s, t)} \right),$$

Then fit the following multinomial logistic regression model:

$$\ln \left(\frac{P[I_X(t) = k]}{P[I_X(t) = k_{ref}]} \right) = \alpha_k + \sum_{h=2}^K \beta_{k,h} * s_k(LP_h),$$

where k_{ref} is an arbitrary reference category which can be reached from state j , $k \neq k_{ref}$ takes values in the set of states that can be reached from state j , and where s is a vector spline smoother (?). Observed event probabilities $\hat{o}_{j,k}(s, t)$ are then estimated as fitted values from this model. This results in a calibration scatter plot rather than a curve due to all states being modeled simultaneously, as opposed to a "one vs all" approach. The scatter occurs because the observed event probabilities for state k vary depending on the predicted transition probabilities of the other states. This is a stronger (?) form of calibration than that evaluated by BLR-IPCW, and will also result in observed event probabilities which sum to 1.

3. Clinical setting and data preperation

We utilise data from the European Society for Blood and Marrow Transplantation (?), containing multistate survival data after a transplant for patients with blood cancer. The start of follow up is the day of the transplant and the initial state is alive and in remission. There are three intermediate events (2: recovery, 3: adverse event, or 4: recovery + adverse event), and two absorbing states (5: relapse and 6: death). This data is available from the **mstate** package (?).

Two datasets are provided to enable assessment of a multistate model fitted to these data. The first is `ebmtcal`, which is the same as the `ebmt` dataset provided in `mstate`, with two extra variables derived: time until censoring (`dtcens`) and an indicator for whether censoring was observed (`dtcens.s = 1`) or an absorbing state was entered (`dtcens.s = 0`). This dataset contains baseline information on year of transplant (`year`), age at transplant (`age`), prophylaxis given (`proph`), and whether the donor was gender matched (`match`). The second dataset provided is `msebmcal`, which is the `ebmt` dataset converted into `msdata` format using the process outlined in ?. It contains all transition times, an event indicator for each transition, as well as a `trans` attribute containing the transition matrix. The package `magrittr` is also loaded to enable the use of pipes.

```
R> library("calibmsm")
R> library("magrittr")
R> data("ebmtcal")
R> head(ebmtcal)
```

| | id | rec | rec.s | ae | ae.s | recae | recae.s | rel | rel.s | srv | srv.s |
|---|----|------|-------|------|------|-------|---------|------|-------|------|-------|
| 1 | 1 | 22 | 1 | 995 | 0 | 995 | 0 | 995 | 0 | 995 | 0 |
| 2 | 2 | 29 | 1 | 12 | 1 | 29 | 1 | 422 | 1 | 579 | 1 |
| 3 | 3 | 1264 | 0 | 27 | 1 | 1264 | 0 | 1264 | 0 | 1264 | 0 |
| 4 | 4 | 50 | 1 | 42 | 1 | 50 | 1 | 84 | 1 | 117 | 1 |
| 5 | 5 | 22 | 1 | 1133 | 0 | 1133 | 0 | 114 | 1 | 1133 | 0 |
| 6 | 6 | 33 | 1 | 27 | 1 | 33 | 1 | 1427 | 0 | 1427 | 0 |

| | year | agecl | proph | match | dtcens | dtcens.s |
|---|-----------|-------|-------|-----------------|--------|----------|
| 1 | 1995-1998 | 20-40 | no no | gender mismatch | 995 | 1 |
| 2 | 1995-1998 | 20-40 | no no | gender mismatch | 422 | 0 |
| 3 | 1995-1998 | 20-40 | no no | gender mismatch | 1264 | 1 |
| 4 | 1995-1998 | 20-40 | no | gender mismatch | 84 | 0 |
| 5 | 1995-1998 | >40 | no | gender mismatch | 114 | 0 |
| 6 | 1995-1998 | 20-40 | no no | gender mismatch | 1427 | 1 |

```
R> data("msebmcal")
R> head(msebmcal)
```

| | id | from | to | trans | Tstart | Tstop | time | status |
|---|----|------|----|-------|--------|-------|------|--------|
| 1 | 1 | 1 | 2 | 1 | 0 | 22 | 22 | 1 |
| 2 | 1 | 1 | 3 | 2 | 0 | 22 | 22 | 0 |
| 3 | 1 | 1 | 5 | 3 | 0 | 22 | 22 | 0 |
| 4 | 1 | 1 | 6 | 4 | 0 | 22 | 22 | 0 |
| 5 | 1 | 2 | 4 | 5 | 22 | 995 | 973 | 0 |
| 6 | 1 | 2 | 5 | 6 | 22 | 995 | 973 | 0 |

In the work of ?, the focus is on predicting transition probabilities at times $s = 0$ and $s = 100$, and comparing prognosis for patients in different states j and with different predictor variables. We therefore focus on assessing the calibration of the transition probabilities made at these times. We estimate transition probabilities for each individual by developing a model

as demonstrated in ?. The predicted transitions probabilities from each state j at times $s = 0$ and $s = 100$ are contained in stacked datasets `tps0` and `tps100` respectively. A leave-one-out approach was used when estimating these transition probabilities. This allows validation to be assessed in the same dataset that the model was developed and any assessment of model performance will be free from in-sample optimism.

```
R> data("tps0")
```

```
R> head(tps0)
```

| | id | pstate1 | pstate2 | pstate3 | pstate4 | pstate5 | pstate6 | |
|---|------------|------------|------------|------------|------------|------------|-----------|---|
| 1 | 1 | 0.1139726 | 0.2295006 | 0.08450376 | 0.2326861 | 0.1504855 | 0.1888514 | |
| 2 | 2 | 0.1140189 | 0.2316569 | 0.08442692 | 0.2328398 | 0.1481977 | 0.1888598 | |
| 3 | 3 | 0.1136646 | 0.2317636 | 0.08274331 | 0.2325663 | 0.1504787 | 0.1887834 | |
| 4 | 4 | 0.1383878 | 0.1836189 | 0.07579429 | 0.2179331 | 0.1538475 | 0.2304185 | |
| 5 | 5 | 0.1233226 | 0.1609740 | 0.05508100 | 0.1828176 | 0.1425950 | 0.3352099 | |
| 6 | 6 | 0.1136646 | 0.2317636 | 0.08462424 | 0.2305854 | 0.1505534 | 0.1888087 | |
| | | se1 | se2 | se3 | se4 | se5 | se6 | j |
| 1 | 0.01291133 | 0.02369584 | 0.01257251 | 0.02323376 | 0.01648630 | 0.01601795 | 1 | |
| 2 | 0.01291552 | 0.02374329 | 0.01256056 | 0.02324869 | 0.01632797 | 0.01603703 | 1 | |
| 3 | 0.01289444 | 0.02375770 | 0.01245752 | 0.02322375 | 0.01647890 | 0.01601525 | 1 | |
| 4 | 0.01857439 | 0.03004447 | 0.01462570 | 0.03018673 | 0.02124071 | 0.02416121 | 1 | |
| 5 | 0.01944967 | 0.03419721 | 0.01367768 | 0.03423941 | 0.02329644 | 0.03688586 | 1 | |
| 6 | 0.01289444 | 0.02375770 | 0.01257276 | 0.02317348 | 0.01649531 | 0.01602438 | 1 | |

```
R> data("tps100")
```

```
R> head(tps100)
```

| | id | pstate1 | pstate2 | pstate3 | pstate4 | pstate5 | pstate6 | |
|---|------------|------------|------------|---------|------------|------------|-----------|---|
| 1 | 1 | 0.7013881 | 0.05220658 | 0 | 0 | 0.1409981 | 0.1054072 | |
| 2 | 2 | 0.7012745 | 0.05242579 | 0 | 0 | 0.1409480 | 0.1053516 | |
| 3 | 3 | 0.7011368 | 0.05251587 | 0 | 0 | 0.1409486 | 0.1053987 | |
| 4 | 4 | 0.6840325 | 0.04116917 | 0 | 0 | 0.1702800 | 0.1045183 | |
| 5 | 5 | 0.6804049 | 0.04283189 | 0 | 0 | 0.1502868 | 0.1264764 | |
| 6 | 6 | 0.7011368 | 0.05251587 | 0 | 0 | 0.1409486 | 0.1053987 | |
| | | se1 | se2 | se3 | se4 | se5 | se6 | j |
| 1 | 0.04691168 | 0.02070211 | 0 | 0 | 0.03457190 | 0.03081258 | 1 | |
| 2 | 0.04691218 | 0.02075972 | 0 | 0 | 0.03456625 | 0.03079617 | 1 | |
| 3 | 0.04693068 | 0.02080004 | 0 | 0 | 0.03456280 | 0.03081033 | 1 | |
| 4 | 0.05885230 | 0.02151183 | 0 | 0 | 0.04710952 | 0.03673242 | 1 | |
| 5 | 0.06694739 | 0.02471395 | 0 | 0 | 0.04906032 | 0.04628088 | 1 | |
| 6 | 0.04693068 | 0.02080004 | 0 | 0 | 0.03456280 | 0.03081033 | 1 | |

The data has been structured this way for the following reasons. The `ebmtcal` dataset has one row per individual and provides the basis for assessing calibration. The models to estimate the observed event probabilities are fit within this dataset. The predicted risks are provided separately because there are more than one predicted transition probabilities for

each individual (states j and times s). The `msebmtool` dataset is required to identify which individuals are in state j at time s in order to apply the landmarking. The code for deriving all these datasets is provided in the supplementary material **TODO**.

4. Assessing calibration using calibration curves and scatter plots

The procedure for producing calibration plots requires the use of two functions. The first function, `calc_calib_blr` or `calc_calib_mlr`, calculates the data for the calibration plot. The second function, `plot_calib_blr` or `plot_calib_mlr`, produces the plot. This approach allows users the flexibility of producing their own plots.

4.1. Plots out of state $j = 1$ at time $s = 0$

We start by producing calibration curves for the predicted transition probabilities out of state $j = 1$ at time $s = 0$. Given all individuals start in state 1, there is no need to consider the transition probabilities out of states $j \neq 1$. We start by assigning the time at which calibration is assessed to be 5 years (1826 days). We then estimate the observed event probabilities for the transition into each state using the function `calc_calib_blr`. The `data.mstate` argument requires an object of class `msdata`. The `data.raw` argument requires a `data.frame` with variables `dtcens` and `dtcens.s`, plus any baseline predictors \mathbf{Z} used to estimate the weights. The predicted transition probabilities from state $j = 1$ at time $s = 0$ are extracted from the object `tps0`. We choose to estimate the calibration curves using restricted cubic splines, and 3 knots are chosen given the reasonably small size of the dataset. Weights are estimated using the internal estimation procedure and the predictor variables `year`, `agec1`, `proph` and `match`. The `w.landmark.type` argument assigns whether weights are estimated using all individuals uncensored at time s , or only those uncensored and in state j at time s , as discussed in section 2.2. The maximum weight (`w.max = 10`) and stabilisation of weights (`stabilised = FALSE`) are left as default.

```
R> t.eval <- 1826
R> dat.calib.blr <-
+   calc_calib_blr(data.mstate = msebmtool,
+                 data.raw = ebmtol,
+                 j=1,
+                 s=0,
+                 t.eval = t.eval,
+                 tp.pred = tps0 %>%
+                 dplyr::filter(j == 1) %>%
+                 dplyr::select(any_of(paste("pstate", 1:6, sep = ""))),
+                 curve.type = "rcs",
+                 rcs.nk = 3,
+                 w.covs = c("year", "agec1", "proph", "match"))
```

The first element of `dat.calib.blr` (named `plotdata`) contains 6 data frames for the calibration curves of the transition probabilities into each of the six states, $k \in \{1, 2, 3, 4, 5, 6\}$. Each data frame contains three columns, `id`: the identifier of each individual; `pred`: the predicted

transition probabilities; `obs`: the observed event probabilities. These data frames have less rows than `ebmtcal` because calibration can only be assessed in individuals uncensored at time `t.eval`. The second element (named `metadata`) is a metadata argument containing a vector of the possible transitions from state j (all states cannot necessarily be reached from state j), the size of the confidence interval (currently `false`), and the type of calibration curve (`rcs` or `loess`).

```
R> str(dat.calib.blr[["plotdata"]])
```

List of 6

```
$ state1:'data.frame':      1778 obs. of  3 variables:
..$ id   : int  [1:1778] 2 4 5 7 10 13 14 16 18 19 ...
..$ pred: num  [1:1778] 0.114 0.1384 0.1233 0.0974 0.1137 ...
..$ obs  : num  [1:1778] 0.11 0.104 0.105 0.124 0.11 ...
$ state2:'data.frame':      1778 obs. of  3 variables:
..$ id   : int  [1:1778] 2 4 5 7 10 13 14 16 18 19 ...
..$ pred: num  [1:1778] 0.232 0.184 0.161 0.212 0.232 ...
..$ obs  : num  [1:1778] 0.17 0.186 0.176 0.179 0.17 ...
$ state3:'data.frame':      1778 obs. of  3 variables:
..$ id   : int  [1:1778] 2 4 5 7 10 13 14 16 18 19 ...
..$ pred: num  [1:1778] 0.0844 0.0758 0.0551 0.0615 0.0844 ...
..$ obs  : num  [1:1778] 0.1249 0.1167 0.0919 0.1001 0.1248 ...
$ state4:'data.frame':      1778 obs. of  3 variables:
..$ id   : int  [1:1778] 2 4 5 7 10 13 14 16 18 19 ...
..$ pred: num  [1:1778] 0.233 0.218 0.183 0.221 0.233 ...
..$ obs  : num  [1:1778] 0.243 0.224 0.185 0.228 0.243 ...
$ state5:'data.frame':      1778 obs. of  3 variables:
..$ id   : int  [1:1778] 2 4 5 7 10 13 14 16 18 19 ...
..$ pred: num  [1:1778] 0.148 0.154 0.143 0.144 0.149 ...
..$ obs  : num  [1:1778] 0.191 0.165 0.222 0.212 0.188 ...
$ state6:'data.frame':      1778 obs. of  3 variables:
..$ id   : int  [1:1778] 2 4 5 7 10 13 14 16 18 19 ...
..$ pred: num  [1:1778] 0.189 0.23 0.335 0.264 0.189 ...
..$ obs  : num  [1:1778] 0.207 0.254 0.316 0.28 0.207 ...
```

```
R> str(dat.calib.blr[["metadata"]])
```

List of 3

```
$ valid.transitions: Named int  [1:6] 1 2 3 4 5 6
..- attr(*, "names")= chr  [1:6] "tp.pred1" "tp.pred2" "tp.pred3" "tp.pred4" ...
$ CI                : logi FALSE
$ curve.type        : chr "rcs"
```

Calibration curves can then be generated using the S3 generic `plot` function. The calibration curves (Figure 1) indicate the level of calibration is different for the transition probabilities into each of the different states. The calibration into states 4 and 6 looks quite good, although


```
R> plot(dat.calib.blr, combine = TRUE, nrow = 2, ncol = 3)
```

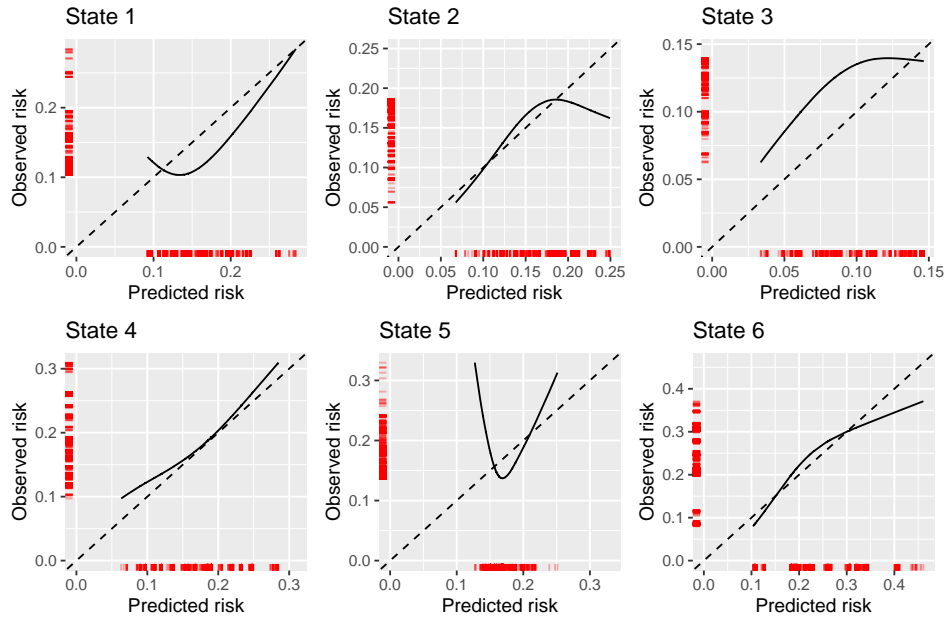


Figure 1: BLR-IPCW calibration curves out of state $j = 1$ at time $s = 0$.

there is under prediction and over prediction among individuals with the lowest and highest predicted risks respectively. State 2 has good calibration over the majority of the predicted risks but over predicts a lot for individuals with the highest predicted risks. Transition probabilities into state 1 are over predicted for almost the entire range of predicted risks, whereas state 3 is under predicted over the entire range of predicted risk. Most importantly the calibration of the transition probabilities into state 5 (Relapse), a key clinical outcome in this clinical settings, is extremely poor.

TO DO

1. Check the rug plots are being implemented with transparency, and add a input parameter to change the transparency
2. Check the MLR-IPCW plots are being implemented with transparency, and add a input parameter to change the transparency
3. Add some proper warnings into the functions in the package for stuff I have listed somewhere

Computational details

WILL UPDATE THIS SECTION ONCE PACKAGE HAS BEEN FINALISED AND UPLOADED TO CRAN

The results in this paper were obtained using R 4.3.0 with the **dplyr** 1.1.2, **tidyr** 1.3.0, **ggplot2** 3.4.2, **ggpubr** 0.6.0, **Hmisc** 5.0.1, **rms** 6.6.0, **VGAM** 1.1.8, **boot** 1.3.28.1, **survival** 3.5.5, **stats** 4.3.0, **magrittr** 2.0.3. R itself and all packages used are available from the Comprehensive R Archive Network (CRAN) at <https://CRAN.R-project.org/>.

Acknowledgments

References

Comparison of calibration curves for transition probabilities generated using pseudo-v
PCW calibration curves for competing risks sub-models with those generated using g

Affiliation:

Alexander Pate
Division of Imaging, Informatics and Data Science
Faculty of Biology, Medicine and Health
University of Manchester M139PR, UK
E-mail: alexander.pate@manchester.ac.uk