casebase: An Alternative Framework For Survival Analysis and Comparison of Event Rates

Sahir Rai Bhatnagar*
McGill University

Maxime Turgeon*
University of Manitoba

Jesse Islam
McGill University

James A. Hanley
McGill University

Olli Saarela
University of Toronto

Abstract

The abstract of the article. *SRB and MT contributed equally to this work.

Keywords: survival analysis, absolute risk, data visualization.

1. Introduction

Survival analysis and the comparison of event rates has been greatly influenced over the last 50 years by the partial likelihood approach of the Cox proportional hazard model (Cox 1972). This approach provides a flexible way of assessing the influence of covariates on the hazard function, without the need to specify a parametric survival model. This flexibility comes at the cost of decoupling the baseline hazard from the effect of the covariates. To recover the whole survival curve—or the cumulative incidence function (CIF)—we then need to separately estimate the baseline hazard (Breslow 1972). This in turn leads to stepwise estimates of the survival function.

From the perspective of clinicians and their patients, the most relevant quantity is often the 5- or 10-year risk of having a certain event given the patient's particular circumstances, and not the hazard ratio between a treatment and control group. Therefore, to make sound clinical decisions, it is important to accurately estimate the *full* hazard function, which can then be used to estimate the cumulative incidence function (CIF). Using a parametric estimator of the hazard function leads to a smooth function of time; as a consequence, the CIF and the survival function estimates also vary smoothly over time.

With the goal of fitting smooth-in-time hazard functions, Hanley & Miettinen (2009) proposed a general framework for estimating fully parametric hazard models via logistic regression. Their approach provides users that are familiar with generalized linear models with a natural way of fitting parametric survival models. Moreover, their framework is very flexible: general functions of time can be estimated (e.g. using splines or general additive models), and hence these models retain some of the flexibility of Cox's partial likelihood approach.

Using the framework of logistic regression for analysing time-to-event data opens the door to an extensive array of powerful modeling tools. Indeed, lasso and elastic-net regression can then be used to select variables that are associated with the hazard function. Logistic regression can also be replaced by multinomial regression to analyse data with multiple competing events.

In this article, we present an R package that combines the ideas of Hanley & Miettinen into a simple interface. The purpose of the **casebase** package is to provide practitioners with an easy-to-use software tool to compute a patient's risk (or cumulative incidence) of an event, conditional on a particular patient's covariate profile. Our package retains the flexibility of case-base sampling and the familiar interface of the `glm` function. It also provides tools for variable selection and competing-risk analysis. In addition, we provide extensive visualization tools.

In what follows, we first recall some theoretical details on case-base sampling and its use for estimating parametric hazard functions. We then give a short review of existing R packages that implement comparable features as **casebase**. Next, we provide some details about the implementation of case-base sampling in our package, and we give a brief survey of its main functions. This is followed by four case studies that illustrate the flexibility and capabilities of **casebase**. We show how the same framework can be used for competing risk analyses, penalized estimation, and for studies with time-dependent exposures. Finally, we end the article with a discussion of the results and of future directions.

2. Theoretical details

As discussed in Hanley & Miettinen (2009), the key idea behind case-base sampling is to sample from the study base a finite amount of *person moments*. These person moments are indexed by both an individual in the study and a time point, and therefore each person moment has a covariate profile, an exposure status and an outcome status attached to it. We note that there is only a finite number of person moments associated with the event of interest (what Hanley & Miettinen call the *case series*). Case-base sampling refers to the sampling from the base of a representative finite sample called the *base series*.

To describe the theoretical foundations of case-base sampling, we use the framework of counting processes. In what follows, we abuse notation slightly and omit any mention of σ-algebras. Instead, following Aalen *et al* (2008), we use the placeholder past to denote the past history of the corresponding process. The reader interested in more details can refer to Saarela & Arjas (2015) and Saarela (2016). First, let $N_i(t) \in \{0, 1\}$ be counting processes corresponding to the event of interest for individual $i = 1, \ldots, n$. For simplicity, we will consider Type I censoring due to the end of follow-up at time τ (the general case of non-informative censoring is treated in Saarela (2016)). We assume a continuous time model, which implies that the counting process jumps are less than or equal to one. We are interested in modeling the

hazard functions $\lambda_i(t)$ of the processes $N_i(t)$, and which satisfy

$$\lambda_i(t)dt = E[dN_i(t) \mid \text{past}].$$

Next, we model the base series sampling mechanism using non-homogeneous Poisson processes $R_i(t) \in \{0, 1, 2, \ldots\}$, with the person-moments where $dR_i(t) = 1$ constituting the base series. The process $Q_i(t) = R_i(t) + N_i(t)$ then counts both the case and base series person-moments contributed by individual i. This process is typically defined by the user via its intensity function $\rho_i(t)$. The process $Q_i(t)$ is characterized by $E[dQ_i(t) \mid \text{past}] = \lambda_i(t)dt + \rho_i(t)dt$.

If the hazard function $\lambda_i(t; \theta)$ is parametrized in terms of θ, we could define an estimator $\hat{\theta}$ by maximization of the likelihood expression

$$L_0(\theta) = \prod_{i=1}^n \exp\left\{ -\int_0^{\min(t_i, \tau)} \lambda_i(t; \theta)dt \right\} \prod_{i=1}^n \prod_{t \in [0, \tau)} \lambda_i(t; \theta)^{dN_i(t)},$$

where $\prod_{t \in [0, u)}$ represents a product integral from 0 to u, and where t_i is the event time for individual i. However, the integral over time makes the computation and maximization of $L_0(\theta)$ challenging.

Case-base sampling allows us to avoid this integral. By conditioning on a sampled person-moment, we get individual likelihood contributions of the form

$$P(dN_i(t) \mid dQ_i(t) = 1, \text{past}) \overset{\theta}{\propto} \frac{\lambda_i(t; \theta)^{dN_i(t)}}{\rho_i(t) + \lambda_i(t; \theta)}.$$

Therefore, we can define an estimating function for θ as follows:

$$L(\theta) = \prod_{i=1}^n \prod_{t \in [0, \tau)} \left(\frac{\lambda_i(t; \theta)^{dN_i(t)}}{\rho_i(t) + \lambda_i(t; \theta)} \right)^{dQ_i(t)}.$$

When a logarithmic link function is used for modeling the hazard function, the above expression is of a logistic regression form with an offset term $\log(1/\rho_i(t))$. Note that the sampling units selected in the case-base sampling mechanism are person-moments, rather than individuals, and the parameters to be estimated are hazards or hazard ratios rather than odds or odds ratios. Generally, an individual can contribute more than one person-moment, and thus the terms in the product integral are not independent. Nonetheless, Saarela (2016) showed that the logarithm of this estimating function has mean zero at the true value $\theta = \theta_0$, and that the resulting estimator $\hat{\theta}$ is asymptotically normally distributed.

In Hanley & Miettinen (2009), the authors suggest sampling the base series *uniformly* from the study base. In terms of Poisson processes, their sampling strategy corresponds essentially to a time-homogeneous Poisson process with hazard equal to $\rho_i(t) = b/B$, where b is the number of sampled observations in the base series, and B is the total population-time for the study base (e.g. the sum of all individual follow-up times). More complex examples are also possible; see for example Saarela & Arjas (2015), where the intensity functions for the sampling mechanism are proportional to the cardiovascular disease event rate given by the Framingham score. With this sampling mechanism, Saarela & Arjas are able to increase the efficiency of their estimators, when compared to uniform sampling.

Let $g(t; X)$ be the linear predictor such that $\log(\lambda(t; X)) = g(t; X)$. Different functions of t lead to different parametric hazard models. The simplest of these models is the one-parameter exponential distribution which is obtained by taking the hazard function to be constant over the range of t:

$$\log(\lambda(t; X)) = \beta_0 + \beta_1 X. \tag{1}$$

In this model, the instantaneous failure rate is independent of t.[1]

The Gompertz hazard model is given by including a linear term for time:

$$\log(\lambda(t; X)) = \beta_0 + \beta_1 t + \beta_2 X. \tag{2}$$

Use of $\log(t)$ yields the Weibull hazard which allows for a power dependence of the hazard on time (Kalbfleisch and Prentice 2011):

$$\log(\lambda(t; X)) = \beta_0 + \beta_1 \log(t) + \beta_2 X. \tag{3}$$

Case-base sampling can also be used in the context of competing-risk analyses. Assuming there are J competing events, we can show that each person-moment's contribution to the likelihood is of the form

$$\frac{\lambda_j(t)^{dN_j(t)}}{\rho(t) + \sum_{j=1}^{J} \lambda_j(t)},$$

where $N_j(t)$ is the counting process associated with the event of type j and $\lambda_j(t)$ is the corresponding cause-specific hazard function. As may be expected, this functional form is similar to the terms appearing in the likelihood function for multinomial regression.[2]

3. Existing packages

Survival analysis is an important branch of applied statistics and epidemiology. Accordingly, there is already a vast ecosystem of R packages implementing different methodologies. In this section, we describe how the functionalities of **casebase** compare to these packages.

At the time of writing, a cursory examination of CRAN's task view on survival analysis reveals that there are over 250 packages related to survival analysis (2019). For the purposes of this article, we restricted our review to packages that implement at least one of the following features: parametric modeling, non-proportional hazard models, competing risk analysis, penalized estimation, and CIF estimation. By searching for appropriate keywords in the DESCRIPTION file of these packages, we found 60 relevant packages. These 60 packages were

---

[1] The conditional chance of failure in a time interval of specified length is the same regardless of how long the individual has been in the study. This is also known as the *memoryless property* (Kalbfleisch and Prentice 2011).

[2] Specifically, it corresponds to the following parametrization:

$$\log\left(\frac{P(Y = j \mid X)}{P(Y = J \mid X)}\right) = X^T \beta_j, \qquad j = 1, \ldots, J - 1.$$

then manually examined to determine which ones are comparable to **casebase**. In particular, we excluded packages that were focused on a different set of problems, such as frailty and multi-state models. The remaining 14 packages appear in Table 1, along with some of the functionalities they offer.

Parametric survival models are implemented in a handful of packages: **CFC** (2019), **flexsurv** (2016), **SmoothHazard** (2017), **rsptm2** (2019), **mets** (2014), and **survival** (2015). The types of models they allow vary for each package. For example, **SmoothHazard** is limited to Weibull distributions (2017), whereas both **flexsurv** and **survival** allow users to supply any distribution of their choice. Also, **flexsurv**, **smoothhazard**, **mets** and **rstpm2** also have the ability to model the effect of time using splines, which allows flexible modeling of the hazard function. Moreover, **flexsurv** has the ability to estimate both scale and shape parameters for a variety of parametric families. As discussed above, **casebase** can model any parametric family whose log-hazard can be expressed as a linear combination of covariates (including time). Therefore, our package allows the user to model the effect of time using splines. Also, by including interaction terms between covariates and time, it also allows users to fit (non-proportional) time-varying coefficient models. However, we do not explicitly model any shape parameter, unlike **flexsurv**.

Several packages implement penalized estimation for the Cox model: **glmnet** (2011), **glmpath** (2018), **penalized** (2010), **RiskRegression** (2019). Moreover, some packages also include penalized estimation in the context of Cox models with time-varying coefficients: elastic-net penalization with **CoxRidge** (2015) and **rstpm2** (2019), while **survival** (2015) has an implementation of ridge-penalized estimation. On the other hand, our package **casebase** provides penalized estimation of the hazard function. To our knowledge, **casebase** and **rsptm2** are the only packages to offer this functionality.

Next, several R packages implement methodologies for competing risk analysis; for a different perspective on this topic, see Mahani & Sharabiani (2019). The package **cmprsk** provides methods for cause-specific subdistribution hazards, such as in the Fine-Gray model (1999). On the other hand, the package **CFC** estimates cause-specific CIFs from unadjusted, non-parametric survival functions. Our package **casebase** also provides functionalities for competing risk analysis by estimating parametrically the cause-specific hazards. From these quantities, we can then estimate the cause-specific CIFs.

Finally, several packages include functions to estimate the CIF. The corresponding methods generally fall into two categories: transformation of the estimated hazard function, and semi-parametric estimation of the baseline hazard. The first category broadly corresponds to parametric survival models, where the full hazard is explicitly modeled. Using this estimate, the survival function and the CIF can be obtained using their functional relationships (see Equations 4 and 5 below). Packages providing this functionality include **CFC**, **Flexsurv**, **mets**, and **survival**. Our package **casebase** also follows this approach for both single-event and competing-risk analyses. The second category outlined above broadly corresponds to semi-parametric models. These models do not model the full hazard function, and therefore the baseline hazard needs to be estimated separately in order to estimate the survival function. This is achieved using semi-parametric estimators (e.g. Breslow's estimator) or parametric estimators (e.g. spline functions). Packages that implement this approach include **RiskRegression**, **rstpm2**, and **survival**. As mentioned in the introduction, a key distinguishing factor between these two approaches is that the first category leads to smooth estimates of the cumulative incidence function, whereas the second category often produces estimates in

the form of stepwise functions. Providing smooth estimates of the CIF was one of the main motivations for introducing case-base sampling in survival analysis.

Package	Competing Risks	Allows Non PH	Penalized Likelihood	Splines	Parametric	Semi Parametric	Interval/Left Censoring	Absolute Risk
casebase	x	x	x	x	x			x
CFC	x	x			x			x
cmprsk	x					x		x
coxRidge		x	x			x		
crrp	x		x					
fastcox			x			x		
Flexrsurv		x		x	x			x
Flexsurv	x	x		x	x			x
glmnet			x			x		
glmpath			x			x		
mets	x			x		x		x
penalized			x			x		
RiskRegression			x			x		x
Rstpm2		x	x	x	x	x	x	x
SmoothHazard		x		x	x		x	
Survival	x	x			x	x	x	x

Table 1: Comparison of various R packages for survival analysis based on several defining features.

4. Implementation details

The functions in the **casebase** package can be divided into two categories: 1) exploratory data analysis, in the form of population-time plots; and 2) parametric modeling of the hazard function. We strove for compatibility with both `data.frame`s and `data.table`s; this can be seen in the coding choices we made and the unit tests we wrote.

4.1. Population-time plots

The case-base sampling approach described in Section 2 can be visualized in the form of a population time plot. These plots are extremely informative graphical displays of survival data and should be one of the first steps in an exploratory data analysis. The `popTime` function and `plot` method facilitate this task:

1. The `casebase::popTime` function takes as input the original dataset along with the column names corresponding to the timescale, the event status and an exposure group of interest (optional). This will create an object of class `popTime`.

2. The corresponding `plot` method for the object created in Step 1 can be called to create the population time plot with several options for customizing the aesthetics.

By splitting these tasks, we give flexibility to the user. While the method call in Step 2 allows further customization by using the **ggplot2** (Wickham 2016) family of functions, users can use for the graphics system of their choice to create population-time plots from the object created in Step 1.

To illustrate these functions, we will use the European Randomized Study of Prostate Cancer Screening (ERSPC) data (Schröder, Hugosson, Roobol, Tammela, Ciatto, Nelen, Kwiatkowski, Lujan, Lilja, Zappa *et al.* 2009) which was extracted using the approach described in Liu *et al.* (2014) This dataset is available through the **casebase** package. It contains the individual observations for 159,893 men from seven European countries, who were between the ages of 55 and 69 years when recruited for the trial.

We first create the necessary dataset for producing the population time plot using the `popTime` function. In this example, we stratify the plot by treatment group. The resulting object inherits from class `popTime` and stores the exposure variable as an attribute:

```
R> pt_object <- casebase::popTime(ERSPC,
R+                                 time = "Follow.Up.Time",
R+                                 event = "DeadOfPrCa",
R+                                 exposure = "ScrArm")
R> inherits(pt_object, "popTime")

#> [1] TRUE

R> attr(pt_object, "exposure")

#> [1] "ScrArm"
```

We then pass this object to the corresponding `plot` method:

```
R> plot(pt_object, add.base.series = TRUE)
```

Figure 1 is built sequentially by first adding a layer for the area representing the population time in gray, with subjects having the least amount of observation time plotted at the top of the y-axis. We immediately notice a distinctive *stepwise shape* in the population time area. This is due to the randomization of the Finnish cohorts which were carried out on January 1 of each of year from 1996 to 1999. Coupled with the uniform December 31 2006 censoring date, this led to large numbers of men with exactly 11, 10, 9 or 8 years of follow-up. Tracked backwards in time (i.e. from right to left), the population-time plot shows the recruitment pattern from its beginning in 1991, and the January 1 entries in successive years. Tracked forwards in time (i.e. from left to right), the plot for the first three years shows attrition due entirely to death (mainly from other causes). Since the Swedish and Belgian centres were the last to complete recruitment–in December 2003–the minimum potential follow-up is three years. Tracked further forwards in time (i.e. after year 3) the attrition is a combination of deaths and staggered entries. As we can see, population-time plots summarise a wealth of information about the study into a simple graph.

Next, layers for the case series and base series are added. The case series is sampled at random vertically on the plot to avoid having all points along the upper edge of the gray area. By randomly distributing the cases, we can get a better sense of the incidence density. In Figure 1, we see that more events are observed at later follow-up times. Therefore, a constant hazard model would not be appropriate in this instance as it would overestimate the cumulative incidence earlier on in time, and underestimate it later on. Finally, the base
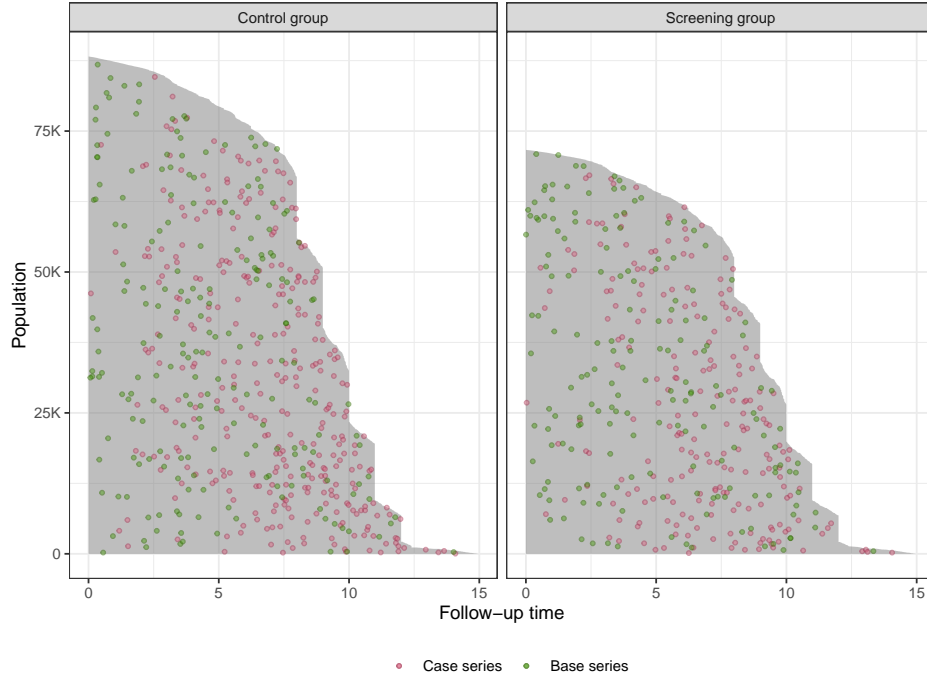
Figure 1: Population time plot for ERSPC data.

series is sampled horizontally with sampling weight proportional to their follow-up time. The reader should refer to the package vignettes for more examples and a detailed description of how to modify the aesthetics of a population-time plot.

4.2. Parametric modeling

The parametric modeling step was separated into three parts:

1. case-base sampling;
2. estimation of the smooth hazard function;
3. estimation of the CIF.

By separating the sampling and estimation functions, we allow the possibility of users implementing more complex sampling scheme (as described in Saarela (2016)), or more complex study designs (e.g. time-varying exposure).

The sampling scheme selected for `sampleCaseBase` was described in Hanley & Miettinen (2009): we first sample along the "person" axis, proportional to each individual's total follow-up time, and then we sample a moment uniformly over their follow-up time. This sampling scheme is equivalent to the following picture: imagine representing the total follow-up time of all individuals in the study along a single dimension, where the follow-up time of the next individual would start exactly when the follow-up time of the previous individual ends. Then the base series could be sampled uniformly from this one-dimensional representation of the overall follow-up time. In any case, the output is a dataset of the same class as the input, where each row corresponds to a person-moment. The covariate profile for each such person-

moment is retained, and an offset term is added to the dataset. This output could then be used to fit a smooth hazard function, or for visualization of the base series.

Next, the fitting function `fitSmoothHazard` starts by looking at the class of the dataset: if it was generated from `sampleCaseBase`, it automatically inherited the class `cbData`. If the dataset supplied to `fitSmoothHazard` does not inherit from `cbData`, then the fitting function starts by calling `sampleCaseBase` to generate the base series. In other words, users can bypass `sampleCaseBase` altogether and only worry about the fitting function `fitSmoothHazard`.

The fitting function retains the familiar formula interface of `glm`. The left-hand side of the formula should be the name of the column corresponding to the event type. The right-hand side can be any combination of the covariates, along with an explicit functional form for the time variable. Note that non-proportional hazard models can be achieved at this stage by adding an interaction term involving time (cf. Case Study 4 below). The offset term does not need to be specified by the user, as it is automatically added to the formula before calling `glm`.

To fit the hazard function, we provide several approaches that are available via the `family` parameter. These approaches are:

- `glm`: This is the familiar logistic regression.
- `glmnet`: This option allows for variable selection using Lasso or elastic-net (cf. Case Study 3). This functionality is provided through the **glmnet** package (Friedman, Hastie, and Tibshirani 2010).
- `gam`: This option provides support for *Generalized Additive Models* via the **mgcv** package (Hastie and Tibshirani 1987).
- `gbm`: This option provides support for *Gradient Boosted Trees* via the **gbm** package. This feature is still experimental.

In the case of multiple competing events, the hazard is fitted via multinomial regression as performed by the **VGAM** package. We selected this package for its ability to fit multinomial regression models with an offset.

Once a model-fit object has been returned by `fitSmoothHazard`, all the familiar summary and diagnostic functions are available: `print`, `summary`, `predict`, `plot`, etc. Our package provides one more functionality: it computes risk functions from the model fit. For the case of a single event, it uses the familiar identity

$$S(t) = \exp\left(-\int_0^t \lambda(u; X)du\right). \tag{4}$$

The integral is computed using either the `stats::integrate` function or Monte-Carlo integration. The risk function (or cumulative incidence function) is then defined as

$$CI(t) = 1 - S(t). \tag{5}$$

For the case of a competing-event analysis, the event-specific risk is computed using the following procedure: first, we compute the overall survival function (i.e. for all event types):

$$S(t) = \exp\left(-\int_0^t \lambda(u; X)du\right), \qquad \lambda(t; X) = \sum_{j=1}^{J} \lambda_j(t; X).$$

From this, we can derive the event-specific subdensities:

$$f_j(t) = \lambda_j(t)S(t).$$

Finally, by integrating these subdensities, we obtain the event-specific cumulative incidence functions:

$$CI_j(t) = \int_0^t f_j(u)du.$$

The integrals are computed using either numerical integration (via the trapezoidal rule) or Monte Carlo integration. This option is controlled by the argument `method` of the `absoluteRisk` function.

In the following sections, we illustrate these functionalities in the context of four case studies.

5. Case study 1—European Randomized Study of Prostate Cancer Screening

For our first case study, we return to the ERSPC study and investigate the differences in survival between the control and screening arms. We fit four models that differ in which functional form of time is used: 1) excluded from the linear predictor, 2) linear function, 3) log function, and 4) a smooth function using cubic B-splines. The models are fit using `fitSmoothHazard`. Since the output object from `fitSmoothHazard` inherits from the `glm` class, we can directly use the function `summary`:

```
R> fmla <- list(exponential = as.formula(DeadOfPrCa ~ ScrArm),
R>              gompertz = as.formula(DeadOfPrCa ~ Follow.Up.Time + ScrArm),
R>              weibull = as.formula(DeadOfPrCa ~ log(Follow.Up.Time) + ScrArm),
R>              splines = as.formula(DeadOfPrCa ~ bs(Follow.Up.Time) + ScrArm))
R>
R> fits <- lapply(fmla, function(i) {
R>   casebase::fitSmoothHazard(i, data = ERSPC, ratio = 100)
R> })
R>
R> summary(fits[["gompertz"]])

#>
#> Call:
#> glm(formula = formula, family = binomial, data = sampleData)
#>
#> Deviance Residuals:
#>    Min      1Q  Median      3Q     Max
#> -0.378  -0.162  -0.122  -0.094   3.468
#>
#> Coefficients:
#>                   Estimate Std. Error z value            Pr(>|z|)
#> (Intercept)        -9.0335     0.1122  -80.54 <0.0000000000000002 ***
#> Follow.Up.Time      0.2228     0.0145   15.32 <0.0000000000000002 ***
```

```
#> ScrArmScreening group  -0.2340    0.0886   -2.64                  0.0083 **
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> (Dispersion parameter for binomial family taken to be 1)
#>
#>     Null deviance: 6059.0  on 54539  degrees of freedom
#> Residual deviance: 5810.1  on 54537  degrees of freedom
#> AIC: 5816
#>
#> Number of Fisher Scoring iterations: 8
```

Next, the `absoluteRisk` function takes as input the `fitSmoothHazard` object and returns
a matrix where each column corresponds to the covariate profiles specified in the `newdata`
argument, and each row corresponds to a specified time point:

```
R> new_data <- data.frame(ScrArm = c("Control group", "Screening group"))
R>
R> risks <- lapply(fits, function(i) {
R>   casebase::absoluteRisk(object = i, time = seq(0,15,0.1), newdata = new_data)
R> })
R>
R> head(risks[["gompertz"]])
```

In Figure 2, we overlay the estimated CIFs from **casebase** on the Cox model CIF. The CIF
estimates for the exponential model in panel (1) overestimate the cumulative incidence earlier
on in time, and underestimate it later on. Based on our earlier discussion of the population-
time plot, this poor fit for the exponential hazard was expected.

In Figure 2 (panels 2–4), we notice a better fit with increasing complexity of our model
for time. As noted above, the usual asymptotic results hold for likelihood ratio tests built
using case-base sampling models. Therefore, we can easily test the null hypothesis that the
exponential model is just as good as the larger (in terms of number of parameters) splines
model:

```
#> Analysis of Deviance Table
#>
#> Model 1: DeadOfPrCa ~ ScrArm + offset(offset)
#> Model 2: DeadOfPrCa ~ bs(Follow.Up.Time) + ScrArm + offset(offset)
#>   Resid. Df Resid. Dev Df Deviance          Pr(>Chi)
#> 1     54538       6053
#> 2     54535       5787  3      266 <0.0000000000000002 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As expected, we see that splines model provides a better fit. Similarly, we can compare the
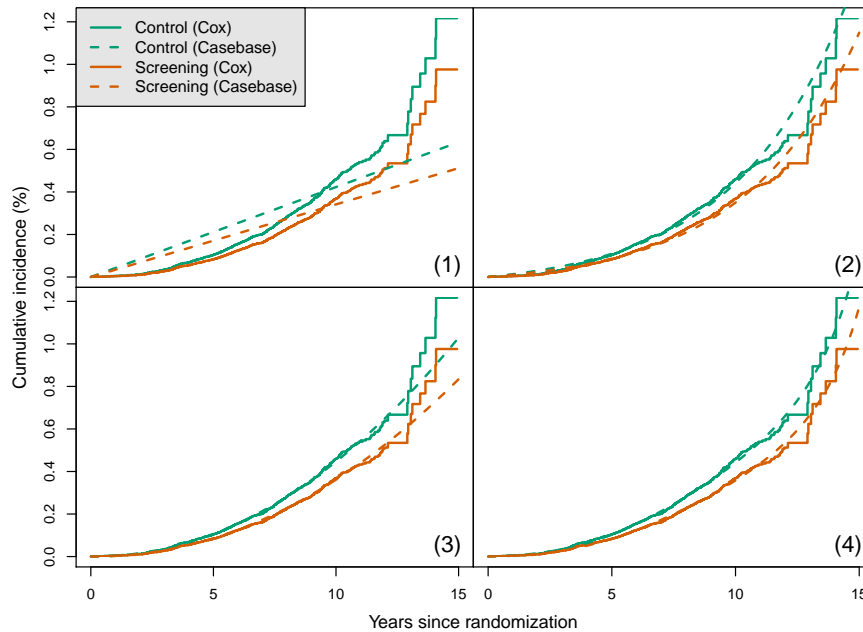AIC for all four case-base models:

Figure 2: CIFs for control and screening groups in the ERSPC data. In each of the panels, we plot the CIF from the Cox model using `survival::survfit` (solid line) and the CIF from the case-base sampling scheme (dashed line) with different functional forms of time. (1) The time variable is excluded (exponential). (2) Linear function of time (Gompertz). (3) The natural logarithm (Weibull). (4) Cubic B-spline expansion of time.

```
#>    Exp.   Gomp.   Weib. Splines
#>    6057    5816    6057    5797
```

Once again, we have evidence that the splines model provides the best fit.

In the following table, we present a side-by-side comparison of the hazard ratios and confidence intervals estimated from `fitSmoothHazard` and the corresponding parametric model using `survival::survreg`, as well as the Cox model estimate. The hazard ratio estimates and confidence intervals are similar across all four models. This reinforces the idea that, under proportional hazards, we do not need to model the full hazard to obtain reliable estimates of the hazard ratio. Nevertheless, Figure 2 shows that different parametric models can still give rise to qualitatively different estimates for the CIF.

With this first case study, we explored how **casebase** allows us to fit different parametric survival models, and how we can compare the fit of each model with tools from GLMs.

6. Case study 2—Bone-marrow transplant

In the next case study, we show how case-base sampling can be used in the context of a competing risk analysis. For illustrative purposes, we will use the same data that was used in Scrucca *et al* (2010). The data was downloaded from the first author's website, and it is

Table 2: Comparison of estimated hazard ratios and 95% confidence intervals for ERSPC data.

Model	casebase::fitSmoothHazard	survival::survreg
Exponential	0.81 (0.68, 0.96)	0.81 (0.68, 0.96)
Gompertz	0.79 (0.66, 0.94)	0.80 (0.67, 0.95)
Weibull	0.81 (0.68, 0.96)	0.80 (0.65, 0.96)
Splines	0.81 (0.68, 0.96)	–

Cox model estimate: HR (95% CI) = 0.80 (0.67, 0.95)

now available as part of the **casebase** package.

The data contains information on 177 patients who received a stem-cell transplant for acute leukemia. The event of interest is relapse, but other competing causes (e.g. death, progression, graft failure, graft-versus-host disease) were also recorded. Several covariates were captured at baseline: sex, disease type (acute lymphoblastic or myeloblastic leukemia, abbreviated as ALL and AML, respectively), disease phase at transplant (Relapse, CR1, CR2, CR3), source of stem cells (bone marrow and peripheral blood, coded as BM+PB, or only peripheral blood, coded as PB), and age.

First, we can look at a population-time plot to visualize the incidence density of both relapse and the competing events. In Figure 3, failure times are highlighted on the plot using red dots for the event of interest (panel A) and blue dots for competing events (panel B). In both panels, we see evidence of a non-constant hazard function: the density of points is larger at the beginning of follow-up than at the end.

Our main objective is to compute the cumulative incidence of relapse for a given set of covariates. We start by fitting a smooth hazard to the data using a linear term for time:

We will compare our hazard ratio estimates to that obtained from a Cox regression. To do so, we need to treat the competing event as censoring.

```
R> library(survival)
R> # Treat competing event as censoring
R> model_cox <- coxph(Surv(ftime, Status == 1) ~ Sex + D + Phase + Source + Age,
R+    data = bmtcrr
R+ )
```

From the fit object, we can extract both the hazard ratios and their corresponding confidence intervals. These quantities appear in Table 3. As we can see, the only significant hazard ratio identified by case-base sampling is the one associated with the phase of the disease at transplant. More precisely, being in relapse at transplant is associated with a hazard ratio of 3.89 when compared to CR1.

Given the estimate of the hazard function obtained using case-base sampling, we can compute the absolute risk curve for a fixed covariate profile. We perform this computation for a 35 year old woman who received a stem-cell transplant from peripheral blood at relapse. We
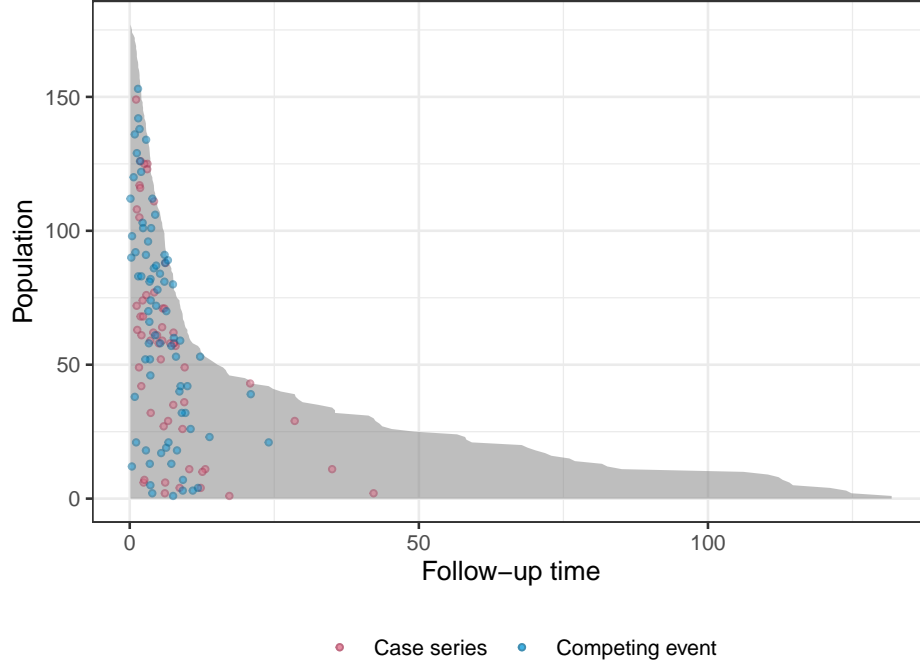
Figure 3: Population-time plot for the stem-cell transplant study with both relapse and competing events.

Table 3: Estimates and confidence intervals for the hazard ratios for each coefficient. Both estimates from case-base sampling and Cox regression are presented.

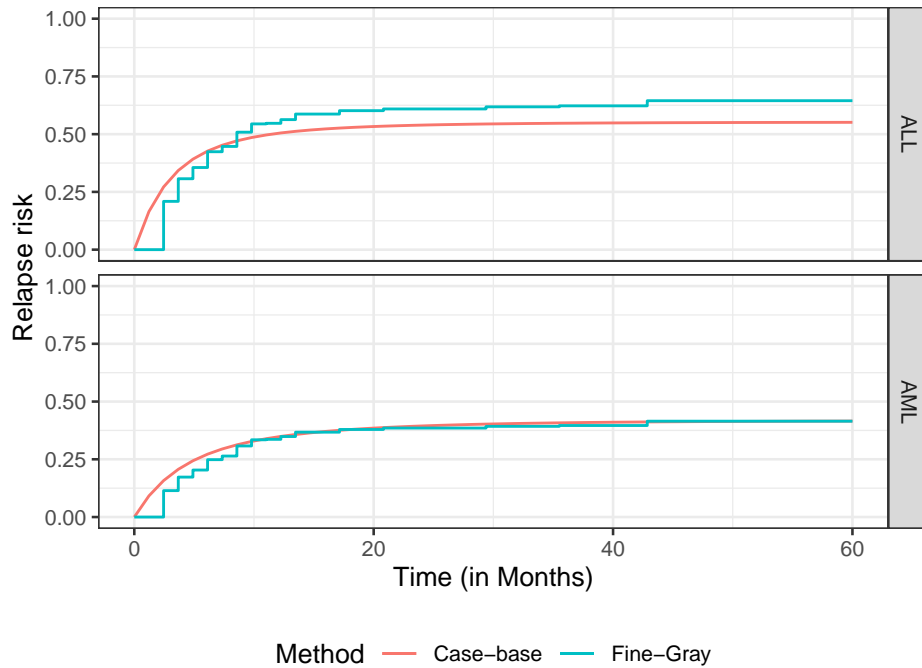| Covariates	Case-Base		Cox	
	HR	95% CI	HR	95% CI
Sex	0.70	(0.4, 1.23)	0.68	(0.39, 1.19)
Disease	0.53	(0.29, 0.95)	0.52	(0.29, 0.93)
Phase (CR2 vs. CR1)	1.12	(0.45, 2.78)	1.21	(0.48, 3.02)
Phase (CR3 vs. CR1)	1.45	(0.37, 5.66)	1.67	(0.43, 6.5)
Phase (Relapse vs. CR1)	4.04	(1.88, 8.68)	4.55	(2.09, 9.9)
Source	1.48	(0.48, 4.54)	1.46	(0.47, 4.54)
Age	0.99	(0.97, 1.01)	0.99	(0.97, 1.02)

Figure 4: Absolute risk curve for a fixed covariate profile and the two disease groups. The estimate obtained from case-base sampling is compared to the Kaplan-Meier estimate.

compared the absolute risk curve for such a woman with ALL with that for a similar woman with AML. We will estimate the curve from 0 to 60 months.

We will compare our estimates to that obtained from a corresponding Fine-Gray model (1999). The Fine-Gray model is a semiparametric model for the cause-specific *subdistribution hazard*, i.e. the function $f_k(t)$ such that

$$CI_k(t) = 1 - \exp\left(-\int_0^t f_k(u)\mathrm{d}u\right),$$

where $CI_k(t)$ is the cause-specific cumulative incidence. The Fine-Gray model allows to directly assess the effect of a covariate on the subdistribution hazard, as opposed to the hazard. For the computation, we will use the **timereg** package (Scheike and Zhang 2011):

```
R> library(timereg)
R> model_fg <- comp.risk(Event(ftime, Status) ~ const(Sex) + const(D) +
R+                          const(Phase) + const(Source) + const(Age),
R+                        data = bmtcrr, cause = 1, model = "fg")
R>
R> # Estimate absolute risk curve
R> risk_fg <- predict(model_fg, newdata, times = time_points)
```

Figure 4 shows the absolute risk curves for both case-base sampling and the Fine-Gray model. As we can see, the two approaches agree quite well for AML; however, there seems to be a

difference of about 5% between the two curves for ALL. This difference does not appear to be significant: the curve from case-base sampling is contained within a 95% confidence band around the Fine-Gray absolute risk curve (figure not shown).

7. Case study 3—SUPPORT Data

In the first two case studies, we described the basic functionalities of the **casebase** package: creating population-time plots, fitting parametric models for hazard functions, and estimating the corresponding cumulative incidence curves. For the third case study, we show how **casebase** can also be used for variable selection through regularized estimation of the hazard function. To do this, we simply replace logistic regression with a penalized counterpart. In **casebase**, we achieve this by using the **glmnet** package (Friedman *et al.* 2010).

To illustrate this functionality, we use the dataset from the Study to Understand Prognoses Preferences Outcomes and Risks of Treatment (SUPPORT) (Knaus, Harrell, Lynn, Goldman, Phillips, Connors, Dawson, Fulkerson, Califf, Desbiens *et al.* 1995). The SUPPORT dataset tracks death in five American hospitals within individuals who are considered seriously ill. The original data is available online from the Department of Biostatistics at Vanderbilt University (Harrell 2020). The cleaned and imputed data consists of 9104 observations and 30 variables, and it is available as part of the **casebase** package. For more information about this dataset, the reader is encouraged to look at the documentation in our package. A description of the variables can be found in Table 4, and a breakdown of each categorical variable appears in Table 5.

In the original study, the authors developed two scores for predicting the outcome. These scores are available as part of the data: SUPPORT day 3 physiology score (`sps`) and APACHE III day 3 physiology score (`aps`). Using these scores, we will explore three Cox regression scenarios as a baseline comparison for **casebase**: `sps` only, `aps` only, and all covariates excluding `sps` and `aps`. We will compare the three models using a time-dependent version of the classical Brier score that is adjusted for censoring (Graf, Schmoor, Sauerbrei, and Schumacher 1999). The Brier score can be used to assess both discrimination and calibration in a single model. Moreover, it provides a metric that can be used to compare parametric and semi-parametric survival models. To account for overfitting, we will use 95% of the observations to train our models, while the remaining 5% will be used to compute absolute risk curves and the Brier scores.

In Figure5 (A), we can see that all Cox models have similar average survival curves. On the other hand, Figure 5 (B) shows that the full model has the lowest Brier score overall. For this reason, we will use this model in our comparison between regularized Cox regression, Kaplan-Meier estimation, and the approach from **casebase**.

For our penalized logistic regression model, we opt for the natural log of time; recall that this corresponds to a Weibull distribution. For fitting the penalized hazard, we use `fitSmoothHazard.fit`, which is a matrix interface to the function `fitSmoothHazard`. In order to use this interface, we first create a matrix `y` containing the time and event variables, and a matrix `x` containing all remaining covariates we would like to include in the model space. We used lasso penalization by setting `alpha = 1`. Moreover, we use the argument `penalty.factor` to force the time variable into the final model.

```
R> x <- model.matrix(death ~ . - d.time - aps - sps,
```
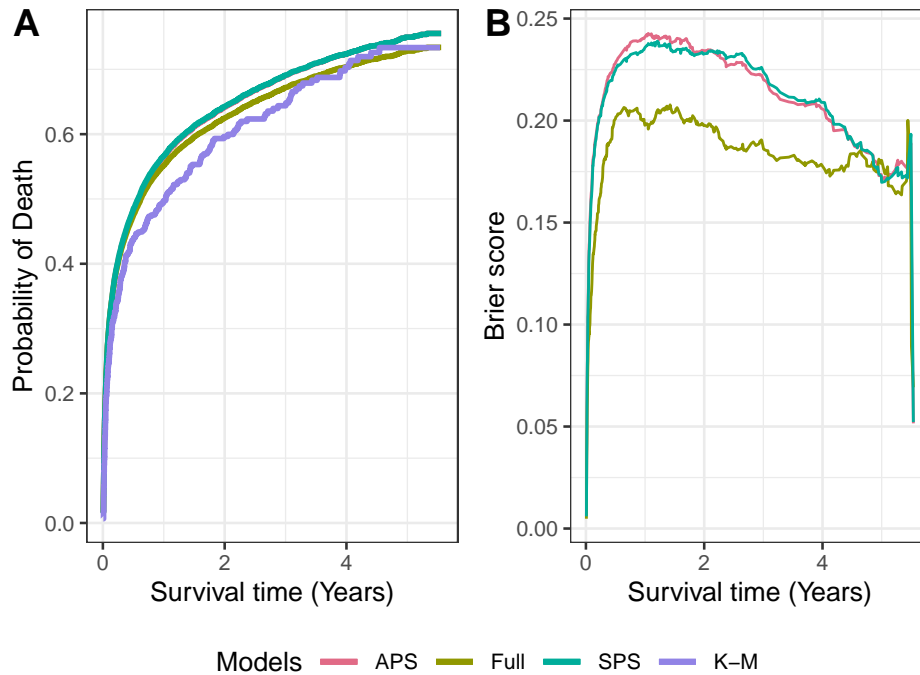
Figure 5: Comparison of three Cox regression models. (A) Cumulative Incidence curves. (B) Time-dependent Brier scores.

```
R+                          data = train)[, -c(1)] # Remove intercept
R> y <- data.matrix(subset(train, select = c(d.time, death)))
R>
R> # Regularized logistic regression to estimate hazard
R> cbFull <- casebase::fitSmoothHazard.fit(x, y,
R+   family = "glmnet",
R+   time = "d.time", event = "death",
R+   formula_time = ~ log(d.time), alpha = 1,
R+   ratio = 10, standardize = TRUE,
R+   penalty.factor = c(0, rep(1, ncol(x)))
R+ )
```

Next, we can use the object `cbFull` to estimate the absolute risk curve on the test data. We first create a matrix `newx` containing the relevant variables. Next, using `times`, we specify the time points at which the cumulative incidence will be computed. Finally, using the parameter `s = "lambda.1se"`, we specify at which value of the tuning parameter we want to retrieve the coefficient estimates.

```
R> # Estimating the absolute risk curve using the newdata parameter
R> newx <- model.matrix(death ~ . - d.time - aps - sps,
R+                      data = test)[, -c(1)]
R> times <- sort(unique(test$d.time))
R>
```
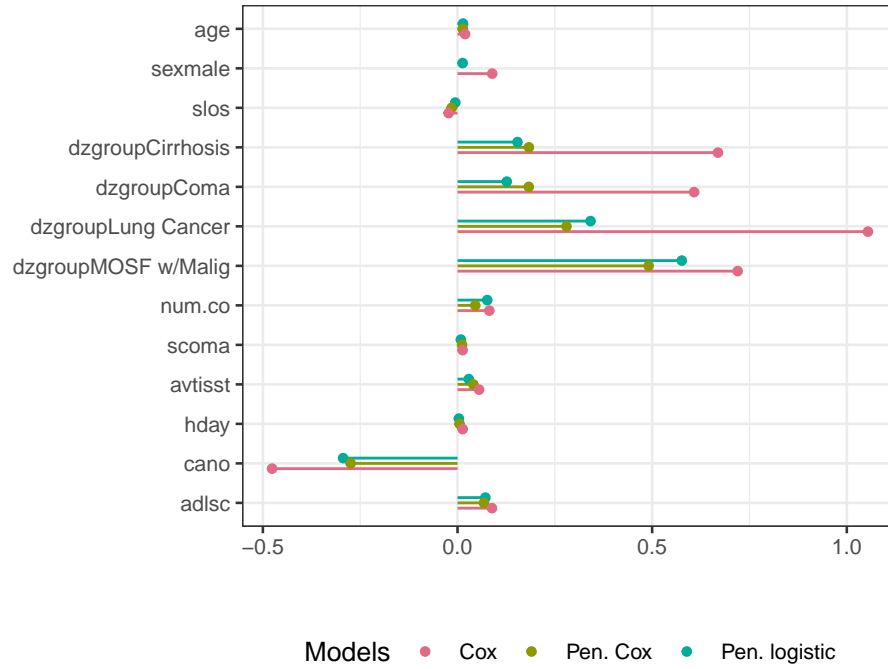
Figure 6:   Lollipop plot showing the coefficient estimates between three models. If a coefficient estimate is 0 in any model, it is removed from the plot for simplicity.

```
R> abCbFull <- casebase::absoluteRisk(cbFull,
R+   time = times,
R+   newdata = newx,
R+   s = "lambda.1se",
R+   method = "numerical"
R+ )
```

Our first comparison plot examines the covariates that were selected between all three hazard functions. Note that Cox regression without penalization selected more covariates than the other two, but only the intersection of coefficients with values greater than 0 are displayed in 6. A visible tendency is for Cox regression to have larger estimated coefficients. This is expected, as there are no restrictions on the size of coefficients. The smallest coefficient estimates are found in penalized logistic model, aside from cancer state (cano).

These values do not inform us of model performance. To do so, we take our hazard functions and use them to predict the absolute risk of everyone in our test set, over time. The probabilities over time for each individual are averaged, resulting in the absolute risk curves in Figure 7 (A). There is separation between the curves, but it is minimal. To better demonstrate the differences in performance between the models, we compare Brier scores over time. In Figure 7 (B), the Brier score is lower in our penalized logistic regression model compared to both Cox regression models. However, this difference is minimal. On the other hand, all three models outperform the Kaplan-Meier model.
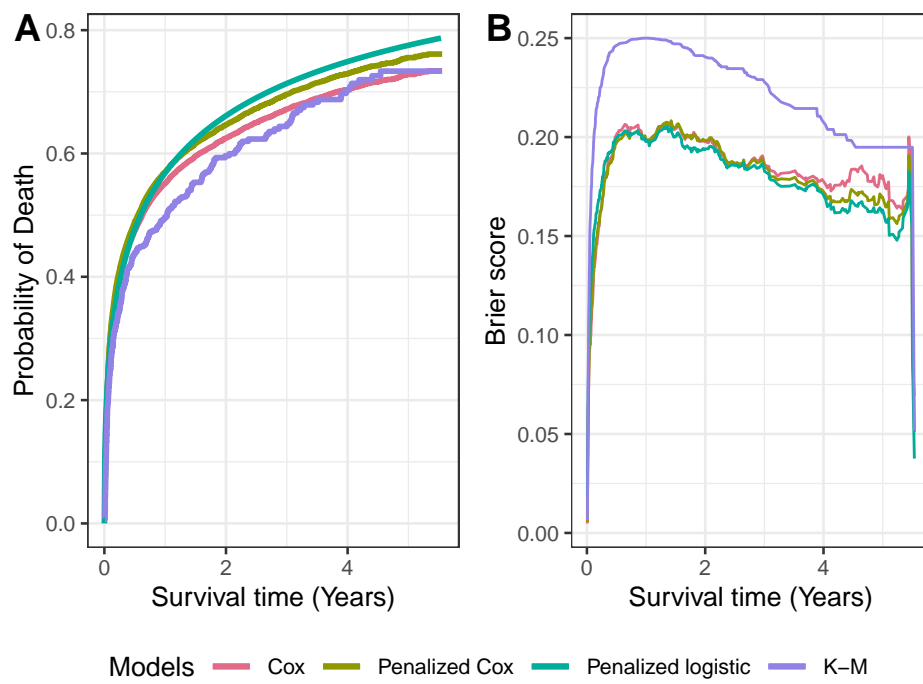
Figure 7: Four curves comparing the Cumulative Incidence (A) and the Brier scores (B) as they change over time between Cox regression, penalized Cox regression, penalized logistic regression, and Kaplan-Meier.

8. Case study 4—Stanford Heart Transplant Data

In the previous case studies, we only considered covariates that were fixed at baseline. In this next case study, we will use the Stanford Heart Transplant data (Clark, Stinson, Griepp, Schroeder, Shumway, and Harrison 1971, Crowley and Hu (1977)) to show how case-base sampling can also be used in the context of time-varying covariates. As an example that already appeared in the literature, case-base sampling was used to study vaccination safety, where the exposure period was defined as the week following vaccination (Saarela and Hanley 2015). Hence, the main covariate of interest, i.e. exposure to the vaccine, was changing over time. In this context, case-base sampling offers an efficient alternative to nested case-control designs or self-matching.

Recall the setting of Stanford Heart Transplant study: patients were admitted to the Stanford program after meeting with their physician and determining that they were unlikely to respond to other forms of treatment. After enrollment, the program searched for a suitable donor for the patient, which could take anywhere between a few days to almost a year. We are interested in the effect of a heart transplant on survival; therefore, the patient is considered exposed only after the transplant has occurred.

As before, we can look at the population-time plot for a graphical summary of the event incidence. Here, we coloured the exposed person-time (i.e. after transplant) in a darker shade of gray. As we can see, most events occur early during the follow-up period, and therefore we do not expect the hazard to be constant.
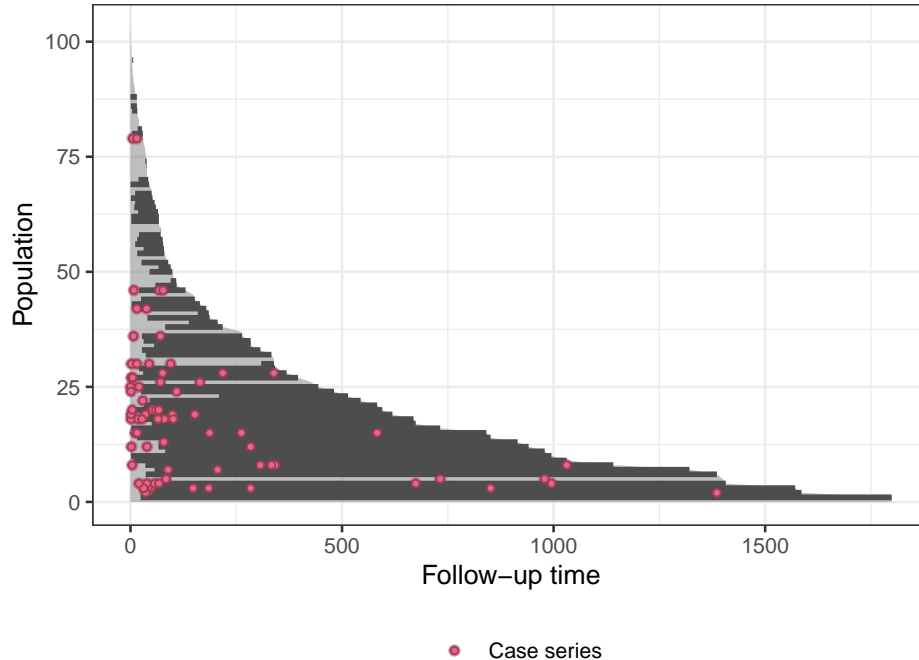
Since the exposure is time-dependent, we need to manually define the exposure variable *after* case-base sampling and *before* fitting the hazard function. For this reason, we will use the `sampleCaseBase` function directly.

Next, we will compute the number of days from acceptance into the program to transplant, and we use this variable to determine whether each population-moment is exposed or not.

Finally, we can fit the hazard using various linear predictors.

```
R> # Fit several models
R> fit1 <- fitSmoothHazard(fustat ~ exposure,
R+   data = cb_data, time = "futime"
R+ )
R> fit2 <- fitSmoothHazard(fustat ~ exposure + futime,
R+   data = cb_data, time = "futime"
R+ )
R> fit3 <- fitSmoothHazard(fustat ~ exposure * futime,
R+   data = cb_data, time = "futime"
R+ )
```

Note that the third model includes an interaction term between exposure and follow-up time. In other words, this model no longer exhibit proportional hazards. The evidence of non-proportionality of hazards in the Stanford Heart Transplant data has been widely discussed (Arjas 1988).

We can then compare the goodness of fit of these four models using the Akaike Information Criterion (AIC).

```
#> Model1 Model2 Model3
#>   827   790   788
```

As we can see, the best fit is the third model. By visualizing the hazard functions for both exposed and unexposed individuals, we can more clearly see how the hazards are no longer proportional.

```
R> # Compute hazards---
R> # First, create a list of time points for both exposure status
R> hazard_data <- expand.grid(
R+   exposure = c(0, 1),
R+   futime = seq(0, 425,
R+     length.out = 100
R+   )
R+ )
R>
R> # Set the offset to zero
R> hazard_data$offset <- 0
R> # Use predict to get the fitted values, and exponentiate to
R> # transform to the right scale
R> hazard_data$hazard <- exp(predict(fit3,
R+   newdata = hazard_data,
R+   type = "link"
R+ ))
```

```
R> # Add labels for plots
R> hazard_data$Status <- factor(hazard_data$exposure,
R+   labels = c("No Tx", "Tx")
R+ )
R>
R> ggplot(hazard_data, aes(futime, hazard, colour = Status)) +
R+   geom_line() +
R+   paper_gg_theme +
R+   ylab("Hazard") +
R+   xlab("Follow-up time (in days)")
```
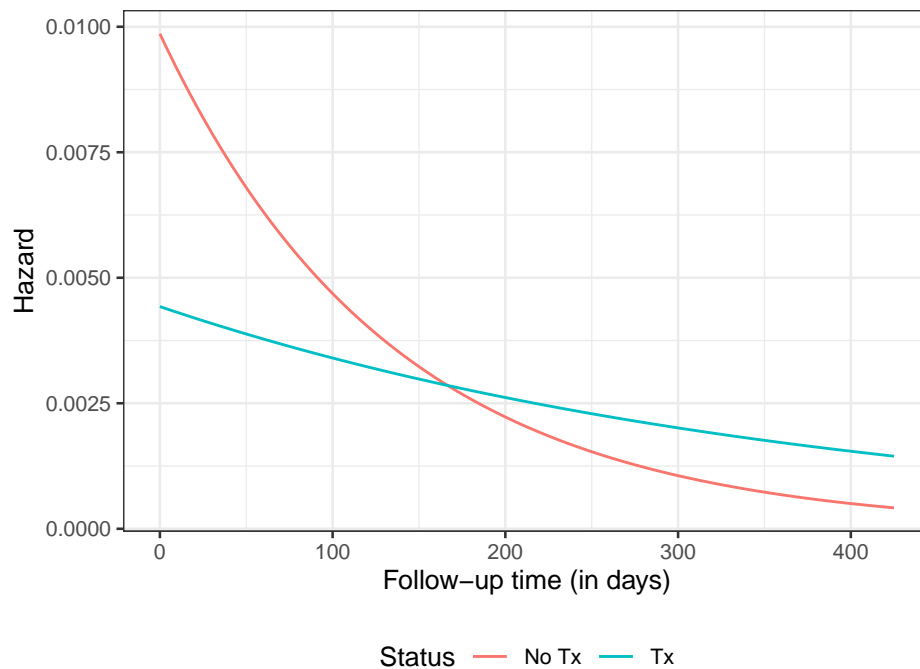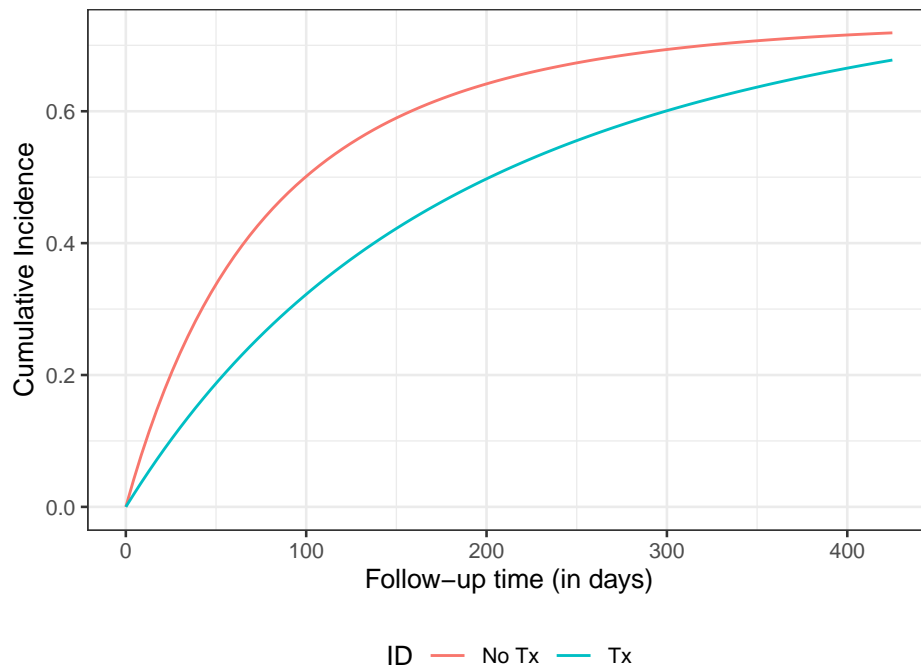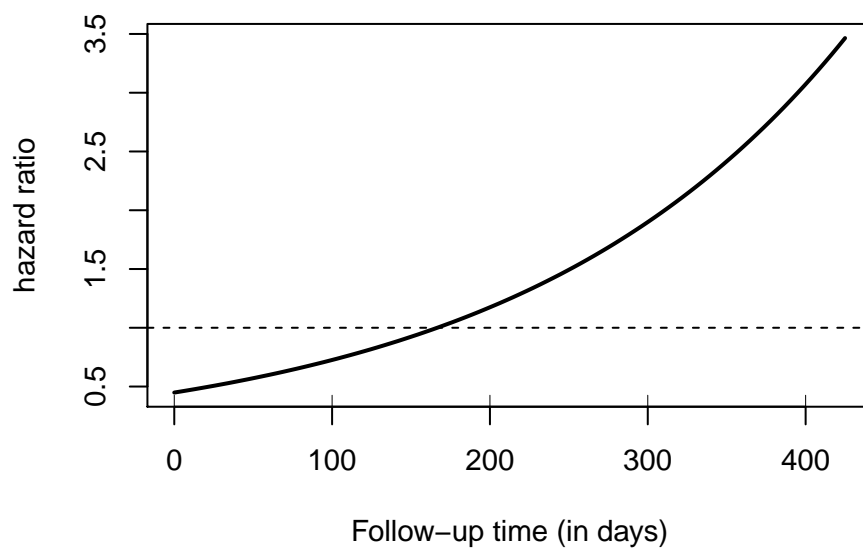

The non-proportionality seems to be more pronounced at the beginning of follow-up than the end. Finally, we can turn these estimates of the hazard function into estimates of the cumulative incidence functions.

```
#> [1] "absRiskCB" "matrix"    "array"
```

We can also plot the hazard ratio:

As we can see in the above case-study, the **casebase** package can also be used to model

time-varying exposures and non-proportional hazard functions.

9. Discussion

In this article, we presented the R package **casebase**, which provides functions for fitting smooth parametric hazards and estimating CIFs using case-base sampling. Our package also provide several functions to produce graphical summaries of the data and the results. We outlined the theoretical underpinnings of the approach, we provided details about our implementation, and we illustrated the merits of the approach and the package through four case studies.

As a methodological framework, case-base sampling is very flexible. This flexibility has been explored before in the literature: for example, Saarela and Hanley (2015) used case-base sampling to model a time-dependent exposure variable in a vaccine safety study. As another example, Saarela and Arjas (2015) combined case-base sampling and a Bayesian non-parametric framework to compute individualized risk assessments for chronic diseases. In the case studies above, we further explored this flexibility along two fronts. On the one hand, we showed how splines could be used as part of the linear predictor to model the effect of time on the hazard. This strategy yielded estimates of the survival function that were qualitatively similar to semiparametric estimates derived from Cox regression; however, case-base sampling led to estimates of the survival function that *vary smoothly in time* and are thus easier to interpret. On the other hand, we also displayed the flexibility of case-base sampling by showing how it could be combined with penalized logistic regression to perform variable selection. Even though we did not illustrate it in this article, case-base sampling can also be combined with the framework of *generalized additive models*. This functionality has already been implemented in the package. Similarly, case-base sampling can be combined with quasi-likelihood estimation to fit survival models that can account for the presence of over-dispersion. All of these examples illustrate how the case-base sampling framework in general, and the package **casebase** in particular, allows the user to fit a broad and flexible family of survival functions.

As presented in Hanley & Miettinen (2009), case-base sampling is comprised of three steps: 1) sampling a case series and a base series from the study; 2) fit the log-hazard as a linear function of predictors (including time); and 3) use the fitted hazard to estimate the CIF. Accordingly, our package provides functions for each step. Moreover, the simple interface of the `fittingSmoothHazard` function resembles the `glm` interface. This interface should look familiar to new users. Our modular approach also provides a convenient way to extend our package for new sampling or fitting strategies.

In the case studies above, we compared the performance of case-base sampling with that of Cox regression and Fine-Gray models. In terms of function interface, **casebase** uses a formula interface that is closer to that of `glm`, in that the event variable is the only variable appearing on the left-hand side of the formula. By contrast, both `survival::coxph` and `timereg::comp.risk` use arrays that capture both the event type and time. Both approaches to modeling yield user-friendly code. However, in terms of output, both approaches differ significantly. Case-base sampling produces smooth hazards and smooth cumulative incidence curves, whereas Cox regression and Fine-Gray models produce stepwise CIFs and never explicitly model the hazard function. Qualitatively, we showed that by using splines in the linear predictor, all three models yielded similar curves. However, the smooth nature of

the output of **casebase** provides a more intuitive interpretation for consumers of these predictions. In Table 6, we provide a side-by-side comparison between the Cox model and case-base sampling.

Our choice of modeling the log-hazard as a linear function of covariates allows us to develop a simple computational scheme for estimation. However, as a downside, it does not allow us to model location and scale parameters separately like the package **flexsurv**. For example, if we look at the Weibull distribution as parametrised in `stats::pweibull`, the log-hazard function is given by

$$\log \lambda(t; \alpha, \beta) = [\log(\alpha/\beta) - (\alpha - 1)\log(\beta)] + (\alpha - 1)\log t,$$

where α, β are shape and scale parameters, respectively. Unlike **casebase**, the approach taken by **flexsurv** also allows the user to model the scale parameter as a function of covariates. Of course, this added flexibility comes at the cost of interpretability: by modeling the log-hazard directly, the parameter estimates from **casebase** can be interpreted as estimates of log-hazard ratios. To improve the flexibility of **casebase** at capturing the scale of a parametric family, we could replace the logistic regression with its quasi-likelihood counterpart and therefore model over- and under-dispersion with respect to the logistic likelihood. We defer the study of the properties and performance of such a model to a future article.

Future work will look at some of the methodological extensions of case-base sampling. First, to assess the quality of the model fit, we would like to study the properties of the residuals (e.g. Cox-Snell, martingale). More work needs to be done to understand these residuals in the context of the partial likelihood underlying case-base sampling. The resulting diagnostic tools could then be integrated in this package. Also, we are interested in extending case-base sampling to account for interval censoring. This type of censoring is very common in longitudinal studies, and many packages (e.g. **SmoothHazard**, **survival** and **Rstpm2**) provide functions to account for it. Again, we hope to include any resulting methodology as part of this package.

In future versions of the package, we also want to increase the complement of diagnostic and inferential tools that are currently available. For example, we would like to include the ability to compute confidence intervals for the cumulative incidence curve. The delta method or parametric bootstrap are two different strategies we can use to construct approximate confidence intervals. Furthermore, we would like to include more functions to compute calibration and discrimination statistics (e.g. AUC) for our models. Saarela and Arjas (2015) also describe how to obtain a posterior distribution for the AUC from their model. Their approach could potentially be included in **casebase**. Finally, we want to provide more flexibility in how the case-base sampling is performed. This could be achieved by adding a `hazard` argument to the function `sampleCaseBase`. In this way, users could specify their own sampling mechanism. For example, they could provide a hazard that gives sampling probabilities that are proportional to the cardiovascular disease event rate given by the Framingham score (Saarela and Arjas 2015).

In conclusion, we presented the R package **casebase** which implements case-base sampling for fitting parametric survival models and for estimating smooth cumulative incidence functions using the framework of generalized linear models. We strongly believe that its flexibility and its foundation on the familiar logistic regression model will make it appealing to new and

established practioners.

10. Environment Details

This report was generated on 2020-08-05 11:28:49 using the following computational environment and dependencies:

```
#> R version 4.0.2 (2020-06-22)
#> Platform: x86_64-apple-darwin17.0 (64-bit)
#> Running under: macOS Catalina 10.15.6
#>
#> Matrix products: default
#> BLAS:   /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
#> LAPACK: /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
#>
#> locale:
#> [1] en_CA.UTF-8/en_CA.UTF-8/en_CA.UTF-8/C/en_CA.UTF-8/en_CA.UTF-8
#>
#> attached base packages:
#> [1] splines   stats     graphics  grDevices utils     datasets  methods
#> [8] base
#>
#> other attached packages:
#>  [1] scales_1.1.1              forcats_0.5.0
#>  [3] stringr_1.4.0             dplyr_1.0.0
#>  [5] purrr_0.3.4               readr_1.3.1
#>  [7] tidyr_1.0.2               ggplot2_3.3.2
#>  [9] tidyverse_1.3.0           lubridate_1.7.8
#> [11] visreg_2.6.1              tibble_3.0.3
#> [13] riskRegression_2020.02.05 prodlim_2019.11.13
#> [15] flexsurv_1.1.1            survival_3.1-12
#> [17] pracma_2.2.9              magrittr_1.5
#> [19] kableExtra_1.1.0          glmnet_3.0-2
#> [21] Matrix_1.2-18             data.table_1.12.8
#> [23] cowplot_1.0.0             colorspace_1.4-1
#> [25] casebase_0.9.0.9000
#>
#> loaded via a namespace (and not attached):
#>  [1] TH.data_1.0-10    VGAM_1.1-2        ellipsis_0.3.1
#>  [4] htmlTable_1.13.3  base64enc_0.1-3  fs_1.4.1
#>  [7] rstudioapi_0.11   MatrixModels_0.4-1 fansi_0.4.1
#> [10] mvtnorm_1.1-0     xml2_1.3.2        codetools_0.2-16
#> [13] knitr_1.29        Formula_1.2-3     jsonlite_1.6.1
#> [16] broom_0.5.6       cluster_2.1.0     dbplyr_1.4.3
#> [19] png_0.1-7         compiler_4.0.2    httr_1.4.1
#> [22] backports_1.1.8   assertthat_0.2.1  cli_2.0.2
```

```
#> [25] acepack_1.4.1     htmltools_0.5.0    quantreg_5.55
#> [28] tools_4.0.2       gtable_0.3.0       glue_1.4.1
#> [31] Rcpp_1.0.5        cellranger_1.1.0   vctrs_0.3.2
#> [34] nlme_3.1-148      iterators_1.0.12   xfun_0.15
#> [37] rvest_0.3.5       lifecycle_0.2.0    polspline_1.1.17
#> [40] muhaz_1.2.6.1     MASS_7.3-51.6      zoo_1.8-8
#> [43] hms_0.5.3         sandwich_2.5-1     SparseM_1.78
#> [46] RColorBrewer_1.1-2 rticles_0.14      yaml_2.2.1
#> [49] gridExtra_2.3     rms_5.1-4          rpart_4.1-15
#> [52] latticeExtra_0.6-29 stringi_1.4.6    foreach_1.5.0
#> [55] checkmate_2.0.0   lava_1.6.7         shape_1.4.4
#> [58] rlang_0.4.7       pkgconfig_2.0.3    evaluate_0.14
#> [61] lattice_0.20-41   htmlwidgets_1.5.1  cmprsk_2.2-9
#> [64] tidyselect_1.1.0  deSolve_1.28       R6_2.4.1
#> [67] generics_0.0.2    Hmisc_4.4-0        multcomp_1.4-13
#> [70] DBI_1.1.0         withr_2.2.0        pillar_1.4.6
#> [73] haven_2.2.0       foreign_0.8-80     mgcv_1.8-31
#> [76] nnet_7.3-14       mstate_0.2.12      modelr_0.1.6
#> [79] crayon_1.3.4      rmarkdown_2.1      timereg_1.9.4
#> [82] jpeg_0.1-8.1      grid_4.0.2         readxl_1.3.1
#> [85] reprex_0.3.0      digest_0.6.25      webshot_0.5.2
#> [88] numDeriv_2016.8-1.1 stats4_4.0.2     munsell_0.5.0
#> [91] viridisLite_0.3.0 quadprog_1.5-8
```

The current Git commit details are:

```
#> Local:    editMax /Users/maxturgeon/Documents/git_repos/cbpaper
#> Head:     [d30ba8c] 2020-07-22: Merge pull request #23 from sahirbhatnagar/editMax
```

References

Aalen O, Borgan O, Gjessing H (2008). *Survival and event history analysis: a process point of view.* Springer Science & Business Media.

Allignol A, Latouche A (2019). "CRAN Task View: Survival Analysis." URL https://cran.r-project.org/web/views/Survival.html.

Arjas E (1988). "A graphical method for assessing goodness of fit in Cox's proportional hazards model." *Journal of the American Statistical Association*, **83**(401), 204–212.

Breslow N (1972). "Discussion of the paper by DR Cox cited below." *Journal of the Royal Statistical Society, Series B*, **34**, 187–220.

Clark DA, Stinson EB, Griepp RB, Schroeder JS, Shumway NE, Harrison D (1971). "Cardiac transplantation in man." *Annals of Internal Medicine*, **75**(1), 15–21.

Clements M, Liu XR, Lambert P, Jakobsen LH, Gasparini A, Smyth G, Alken P, Wood S, Ulerich R (2019). "Smooth Survival Models, Including Generalized Survival Models [R package rstpm2 version 1.5.1]." URL https://cran.r-project.org/web/packages/rstpm2/index.html.

Cox DR (1972). "Regression models and life-tables." *Journal of the Royal Statistical Society: Series B (Methodological)*, **34**(2), 187–202.

Crowley J, Hu M (1977). "Covariance analysis of heart transplant survival data." *Journal of the American Statistical Association*, **72**(357), 27–36.

Fine JP, Gray RJ (1999). "A proportional hazards model for the subdistribution of a competing risk." *Journal of the American statistical association*, **94**(446), 496–509.

Friedman J, Hastie T, Tibshirani R (2010). "Regularization Paths for Generalized Linear Models via Coordinate Descent." *Journal of Statistical Software*, **33**(1). ISSN 1548-7660. doi:10.18637/jss.v033.i01.

Gerds TA, Blanche P, Mortersen R, Tollenaar N, Mogensen UB, Ozenne B (2019). "Risk Regression Models and Prediction Scores for Survival Analysis with Competing Risks [R package riskRegression version 2019.11.03]." URL https://CRAN.R-project.org/package=riskRegression.

Goeman JJ (2010). "L1 penalized estimation in the Cox proportional hazards model." *Biometrical Journal*, (52), –14.

Graf E, Schmoor C, Sauerbrei W, Schumacher M (1999). "Assessment and comparison of prognostic classification schemes for survival data." *Statistics in medicine*, **18**(17-18), 2529–2545.

Hanley JA, Miettinen OS (2009). "Fitting smooth-in-time prognostic risk functions via logistic regression." *The International Journal of Biostatistics*, **5**(1).

Harrell F (2020). "SupportDesc < Main < Vanderbilt Biostatistics Wiki." URL http://biostat.mc.vanderbilt.edu/wiki/Main/SupportDesc.

Hastie T, Tibshirani R (1987). "Generalized additive models: some applications." *Journal of the American Statistical Association*, **82**(398), 371–386.

Jackson C (2016). "flexsurv: A Platform for Parametric Survival Modeling in R." *Journal of Statistical Software*, **70**(8), 1–33. doi:10.18637/jss.v070.i08.

Kalbfleisch JD, Prentice RL (2011). *The statistical analysis of failure time data*, volume 360. John Wiley & Sons.

Knaus WA, Harrell FE, Lynn J, Goldman L, Phillips RS, Connors AF, Dawson NV, Fulkerson WJ, Califf RM, Desbiens N, *et al.* (1995). "The SUPPORT prognostic model: objective estimates of survival for seriously ill hospitalized adults." *Annals of internal medicine*, **122**(3), 191–203.

Liu Z, Rich B, Hanley JA (2014). "Recovering the raw data behind a non-parametric survival curve." *Systematic reviews*, **3**(1), 151.

Mahani A, Sharabiani M (2019). "Bayesian, and Non-Bayesian, Cause-Specific Competing-Risk Analysis for Parametric and Nonparametric Survival Functions: The R Package CFC." *Journal of Statistical Software, Articles*, **89**(9), 1–29. ISSN 1548-7660. doi:10.18637/jss.v089.i09. URL https://www.jstatsoft.org/v089/i09.

Park MY, Hastie T (2018). "Package glmpath." URL https://CRAN.R-project.org/package=glmpath.

Perperoglou A (2015). "Package CoxRidge." URL https://CRAN.R-project.org/package=CoxRidge.

Saarela O (2016). "A case-base sampling method for estimating recurrent event intensities." *Lifetime data analysis*, **22**(4), 589–605.

Saarela O, Arjas E (2015). "Non-parametric Bayesian Hazard Regression for Chronic Disease Risk Assessment." *Scandinavian Journal of Statistics*, **42**(2), 609–626.

Saarela O, Hanley JA (2015). "Case-base methods for studying vaccination safety." *Biometrics*, **71**(1), 42–52.

Scheike TH, Holst KK, Hjelmborg JB (2014). "Estimating twin concordance for bivariate competing risks twin data." *Statistics in medicine*, **33**(7), 1193–1204.

Scheike TH, Zhang MJ (2011). "Analyzing Competing Risk Data Using the R timereg Package." *Journal of Statistical Software*, **38**(2), 1–15. URL http://www.jstatsoft.org/v38/i02/.

Schröder FH, Hugosson J, Roobol MJ, Tammela TL, Ciatto S, Nelen V, Kwiatkowski M, Lujan M, Lilja H, Zappa M, *et al.* (2009). "Screening and prostate-cancer mortality in a randomized European study." *New England Journal of Medicine*, **360**(13), 1320–1328.

Scrucca L, Santucci A, Aversa F (2010). "Regression modeling of competing risk using R: an in depth guide for clinicians." *Bone marrow transplantation*, **45**(9), 1388.

Simon N, Friedman J, Hastie T, Tibshirani R (2011). "Regularization Paths for Cox's Proportional Hazards Model via Coordinate Descent." *Journal of Statistical Software*, **39**(5), 1–13. URL http://www.jstatsoft.org/v39/i05/.

Therneau TM (2015). *A Package for Survival Analysis in S.* Version 2.38, URL https://CRAN.R-project.org/package=survival.

Touraine C, Gerds TA, Joly P (2017). "SmoothHazard: An R Package for Fitting Regression Models to Interval-Censored Observations of Illness-Death Models." *Journal of Statistical Software*, **79**(7), 1–22. doi:10.18637/jss.v079.i07.

Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York. ISBN 978-3-319-24277-4. URL https://ggplot2.tidyverse.org.

Affiliation:

Sahir Rai Bhatnagar*
McGill University
1020 Pine Avenue West Montreal, QC, Canada H3A 1A2
E-mail: sahir.bhatnagar@mail.mcgill.ca
URL: http://sahirbhatnagar.com/

Maxime Turgeon*
University of Manitoba
186 Dysart Road Winnipeg, MB, Canada R3T 2N2
E-mail: max.turgeon@umanitoba.ca
URL: https://maxturgeon.ca/

Jesse Islam
McGill University
1020 Pine Avenue West Montreal, QC, Canada H3A 1A2
E-mail: jesse.islam@mail.mcgill.ca

James A. Hanley
McGill University
1020 Pine Avenue West Montreal, QC, Canada H3A 1A2
E-mail: james.hanley@mcgill.ca
URL: http://www.medicine.mcgill.ca/epidemiology/hanley/

Olli Saarela
University of Toronto
Dalla Lana School of Public Health, 155 College Street, 6th floor, Toronto, Ontario M5T 3M7, Canada
E-mail: olli.saarela@utoronto.ca
URL: http://individual.utoronto.ca/osaarela/

Name	Labels	Levels	Storage	NAs
age	Age		double	0
death	Death at any time up to NDI date:31DEC94		double	0
sex		2	integer	0
slos	Days from Study Entry to Discharge		double	0
d.time	Days of Follow-Up		double	0
dzgroup		8	integer	0
dzclass		4	integer	0
num.co	number of comorbidities		double	0
edu	Years of Education		double	202
income		4	integer	349
scoma	SUPPORT Coma Score based on Glasgow D3		double	0
avtisst	Average TISS, Days 3-25		double	6
race		5	integer	5
sps	support physiology score day 3		double	1
aps	APS III no coma, imp bun,uout for ph1,D3		double	1
hday	Day in Hospital at Study Admit		double	0
diabetes	Diabetes (Com 27-28, Dx 73)		double	0
dementia	Dementia (Comorbidity 6)		double	0
ca	Cancer State	3	integer	0
meanbp	Mean Arterial Blood Pressure Day 3		double	0
wblc	White Blood Cell Count Day 3		double	24
hrt	Heart Rate Day 3		double	0
resp	Respiration Rate Day 3		double	0
temp	Temperature (Celsius) Day 3		double	0
pafi	PaO2/(.01*FiO2) Day 3		double	253
alb	Serum Albumin Day 3		double	378
bili	Bilirubin Day 3		double	297
crea	Serum creatinine Day 3		double	3
sod	Serum sodium Day 3		double	0
ph	Serum pH (arterial) Day 3		double	250
glucose	Glucose Day 3		double	470
bun	BUN Day 3		double	455
urine	Urine Output Day 3		double	517
adlp	Activiites of Daily Living (ADL) Patient Day 3		double	634
adlsc	Imputed ADL Calibrated to Surrogate		double	0

Table 4: Description of each variable in the SUPPORT dataset.

Variable	Levels
sex	female
	male
dzgroup	ARF/MOSF w/Sepsis
	COPD
	CHF
	Cirrhosis
	Coma
	Colon Cancer
	Lung Cancer
	MOSF w/Malig
dzclass	ARF/MOSF
	COPD/CHF/Cirrhosis
	Coma
	Cancer
	>$50k
race	white
	black
	asian
	other
	hispanic
ca	Yes
	No
	Metastatic

Table 5: Description of each level within each categorical variable in the SUPPORT dataset.

Table 6: Comparison between the Cox model and case-base sampling

Feature	Cox model	Case-base sampling
Model type	Semi-parametric	Fully parametric
Time	Left hand side of the formula	Right hand side (allows flexible modeling of time)
Cumulative incidence	Step function	Smooth-in-time curve
Non-proportional hazards	Interaction of covariates with time	Interaction of covariates with time
Model testing		Use GLM framework (e.g. LRT, AIC, BIC)
Competing risks	Difficult	Cause-specific CIFs