



pencal: Dynamic Prediction with Numerous Longitudinal Covariates

Mirko Signorelli

 [mirkosignorelli.github.io](https://github.com/mirkosignorelli)

 [in/signorelli](https://www.linkedin.com/in/signorelli)

Mathematical Institute
Leiden University

July 10, 2024 - UseR! 2024 conference



Universiteit
Leiden

Vignette: bit.ly/pencal-UseR

 [in/signorelli](https://www.linkedin.com/in/signorelli)



Dynamic prediction and the R package `pencal`

Input data

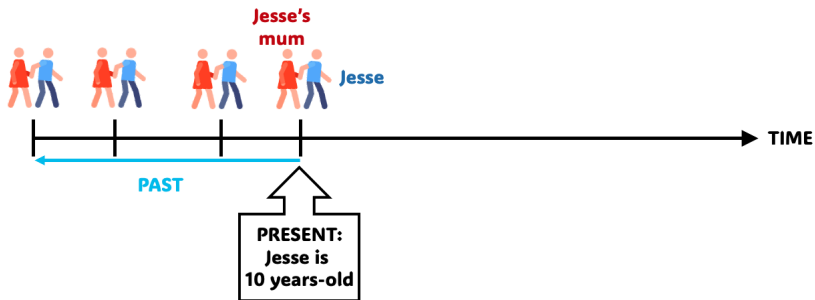
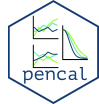
Model estimation

Prediction

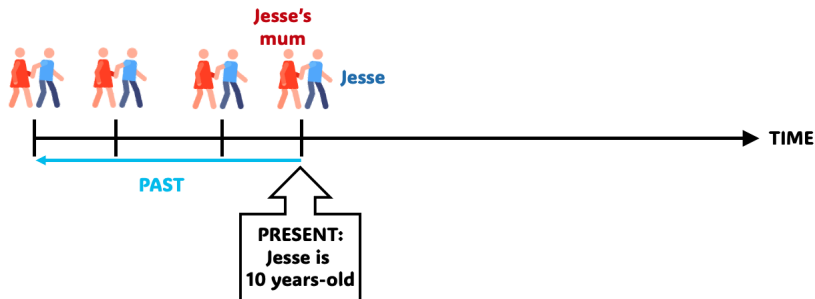
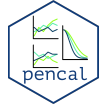
Evaluation of predictive performance

Wrap-up

Dynamic prediction 101

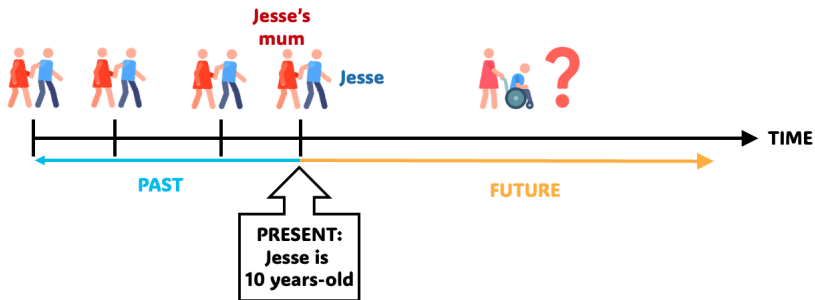
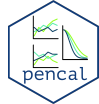


Dynamic prediction 101



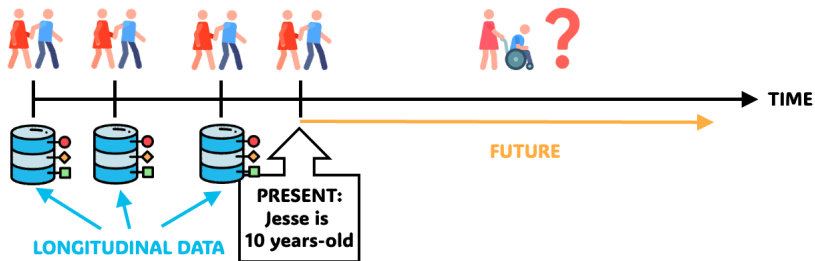
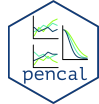
- ▶ Jesse suffers from Duchenne muscular dystrophy (DMD)
- ▶ Usually: **loss of ambulation** during adolescence

Dynamic prediction 101

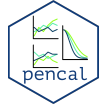


- ▶ Mum: what's the probability that in 3 year's time, Jesse will need a wheelchair?

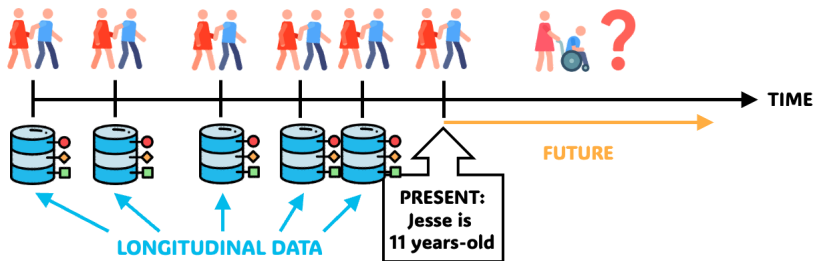
Dynamic prediction 101



Dynamic prediction 101



► One year later:



Goals of dynamic prediction:

1. predict future survival $S(t|\ell_1) = P(T > t | T > \ell_1)$, $t > \ell_1$, using repeated measurements over $[0, \ell_1]$
2. update predictions when newer information becomes available, i.e. update $S(t|\ell_2)$ given repeated measurements over $[0, \ell_2]$, $t > \ell_2 > \ell_1$

- ▶ Modern biomedical datasets can feature **tens / hundreds / thousands of longitudinal predictors**
- ▶ Traditional methods for dynamic prediction (joint modelling / landmarking) struggle with such datasets
- ▶ Problem: **how to do dynamic prediction with "many" longitudinal predictors?**

The method: Penalized Regression Calibration (PRC)



- ▶ Statistical method: Penalized Regression Calibration (PRC, Signorelli et al. (2021))



RESEARCH ARTICLE | [Open Access](#) |

Penalized regression calibration: A method for the prediction of survival outcomes using complex longitudinal and high-dimensional data

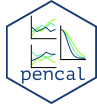
Mirko Signorelli , Pietro Spitali, Cristina Al-Khalili Szigyarto, The MARK-MD Consortium, Roula Tsonaka

First published: 31 August 2021 | <https://doi.org/10.1002/sim.9178>

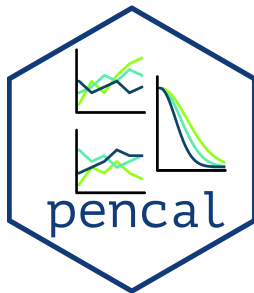
Vignette: bit.ly/pencal-UseR

: [in/signorelli](https://in.linkedin.com/in/signorelli)

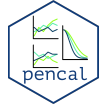
The R package: `pencal`




- ▶ PRC implemented in the R package `pencal`




The R package: `pencal`



► Available on  **CRAN**:

`pencal`: Penalized Regression Calibration (PRC) for the Dynamic Prediction of Survival

Computes penalized regression calibration (PRC), a statistical method for the dynamic prediction of survival when many longitudinal predictors are available. PRC is described in Signorelli (2024) <[doi:10.48550/arXiv.2309.15600](https://doi.org/10.48550/arXiv.2309.15600)> and in Signorelli et al. (2021) <[doi:10.1002/sim.9178](https://doi.org/10.1002/sim.9178)>.

Version: 2.2.2
Depends: R (≥ 4.1.0)
Imports: [doParallel](#), [dplyr](#), [foreach](#), [glmnet](#), [lcm](#), [magic](#), [MASS](#), [Matrix](#), methods, [nlme](#), [purrr](#), [riskRegression](#),
stats, [survcomp](#), [survival](#), [survivalROC](#)
Suggests: [knitr](#), [ptmixed](#), [rmarkdown](#), [survminer](#)
Published: 2024-06-12
Author: Mirko Signorelli  [aut, cre, cph], Pietro Spitali [ctb], Roula Tsonaka [ctb], Barbara Vreede [ctb]
Maintainer: Mirko Signorelli <mssignorelli.rpackages@gmail.com>
License: [GPL \(≥ 3\)](#)
URL: <https://mirkosignorelli.github.io/r>
NeedsCompilation: no
Citation: [pencal citation info](#)
Materials: [NEWS](#)
CRAN checks: [pencal results](#)

Documentation:


Reference manual: [pencal.pdf](#)

Vignettes: [pencal: an R Package for the Dynamic Prediction of Survival with Many Longitudinal Predictors](#)

Vignette: bit.ly/pencal-UseR

 [in/signorelli](https://in.linkedin.com/in/signorelli)

- Methodology, R package and example code illustrated in Signorelli (2024), available at [arXiv:2309.15600](https://arxiv.org/abs/2309.15600) (and on CRAN):


stat > arXiv:2309.15600

All fields

[Help](#) | [Advanced Search](#)

Statistics > Methodology

[Submitted on 27 Sep 2023]

pencal: an R Package for the Dynamic Prediction of Survival with Many Longitudinal Predictors

[Mirko Signorelli](#)

In survival analysis, longitudinal information on the health status of a patient can be used to dynamically update the predicted probability that a patient will experience an event of interest. Traditional approaches to dynamic prediction such as joint models become computationally unfeasible with more than a handful of longitudinal covariates, warranting the development of methods that can handle a larger number of longitudinal covariates. We introduce the R package `pencal`, which implements a Penalized Regression Calibration approach that makes it possible to handle many longitudinal covariates as predictors of survival. `pencal` uses mixed-effects models to summarize the trajectories of the longitudinal covariates up to a prespecified landmark time, and a penalized Cox model to predict survival based on both baseline covariates and summary measures of the longitudinal covariates. This article illustrates the structure of the R package, provides a step by step example showing how to estimate PRC, compute dynamic predictions of survival and validate performance, and shows how parallelization can be used to significantly reduce computing time.

Access Paper:

- [Download PDF](#)
- [PostScript](#)
- [Other Formats](#)

(view license)

Current browse context:

stat.ME

[< prev](#) | [next >](#)

[new](#) | [recent](#) | [2309](#)

Change to browse by:

[stat](#)



[stat.CO](#)

References & Citation

- [NASA ADS](#)
- [Google Scholar](#)
- [Semantic Scholar](#)

[Export BibTeX Citation](#)


Bookmark

Subjects: **Methodology** (stat.ME); Computation (stat.CO)

Cite as: [arXiv:2309.15600](#) [stat.ME]

(or [arXiv:2309.15600v1](#) [stat.ME] for this version)

<https://doi.org/10.48550/arXiv.2309.15600> 



Dynamic prediction and the R package `pencal`

Input data

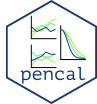
Model estimation

Prediction

Evaluation of predictive performance

Wrap-up

Example (toy) dataset



- ▶ Data from the PBC2 clinical trial (1974-1984)
 - ▶ $n = 312$
 - ▶ Outcome (in this example): time to death

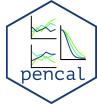
```
library(pencal)
data(pbc2data)
sdata = pbc2data$baselineInfo
ldata = pbc2data$longitudinalInfo
```

NB: `pencal` designed to deal with larger p (and n), but I cannot share more recent datasets with larger p ☹

- ▶ ℓ = **landmark** time (“PRESENT”)
 - ▶ $i \in \{1, \dots, n_\ell\}$ subjects still at risk at time $t = \ell$
- ▶ Observation window (“PAST”) $t \in [0, \ell]$:
 - ▶ k **baseline covariates** (x_{1i}, \dots, x_{ki}) measured at $t_i = 0$ (study entry)
 - ▶ p **longitudinal covariates** (y_{1ij}, \dots, y_{pij}) measured at $t_{i1}, \dots, t_{im_i} \in [0, \ell]$
- ▶ Prediction window (“FUTURE”) $t \in (\ell, +\infty)$:
 - ▶ $T_i = \min(T_i^*, C_i)$ **observed survival time**¹
 - ▶ $\delta_i = I(T_i = T_i^*)$ **event indicator**

¹ T_i^* true survival time, C_i censoring time

Data preparation steps



1. Choose the landmark. Here: $\ell = 2$

```
lmark = 2
```

2. Select the correct subjects and corresponding data

```
# remove subjects with event / censoring before landmark:
sdata = subset(sdata, time > lmark)
ldata = subset(ldata, id %in% sdata$id)
# remove repeated measurements taken after landmark:
ldata = subset(ldata, fuptime <= lmark)
```

3. Further preprocessing (if desirable / needed)

```
# let's log-transform some highly-skewed predictors:
ldata$logSerBil = log(ldata$serBilir)
ldata$logSerChol = log(ldata$serChol)
ldata$logAlk = log(ldata$alkaline)
ldata$logSGOT = log(ldata$SGOT)
ldata$logProthr = log(ldata$prothrombin)
```

1. A dataset (ldata) with the **longitudinal covariates** measured up to the landmark time:

##	id	age	fuptime	logSerBil	logSerChol	albumin	logAlk
## 3	2	56.45	0.00	0.10	5.71	4.14	8.91
## 4	2	56.95	0.50	-0.22	NA	3.60	7.65
## 5	2	57.45	1.00	0.00	NA	3.55	7.44
## 16	4	54.74	0.00	0.59	5.50	2.54	8.72
## 17	4	55.26	0.51	0.47	NA	2.88	7.07
## 18	4	55.76	1.02	0.53	NA	2.80	7.05
## 19	4	56.74	2.00	1.16	NA	2.92	7.07

##	logSGOT	platelets	logProthr
## 3	4.73	221	2.36
## 4	4.94	188	2.40
## 5	4.97	161	2.45
## 16	4.10	183	2.33
## 17	5.13	240	2.94
## 18	5.11	251	2.45
## 19	5.12	220	2.38

2. A dataset (sdata) with the survival outcome, and baseline covariates:

##	id	time	event	baselineAge	sex	treatment
## 3	2	14.152338	0	56.44782	female	D-penicil
## 12	3	2.770781	1	70.07447	male	D-penicil
## 16	4	5.270507	1	54.74209	female	D-penicil
## 23	5	4.120578	0	38.10645	female	placebo
## 29	6	6.853028	1	66.26054	female	placebo
## 35	7	6.847552	0	55.53609	female	placebo



Dynamic prediction and the R package `pencal`

Input data

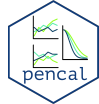
Model estimation

Prediction

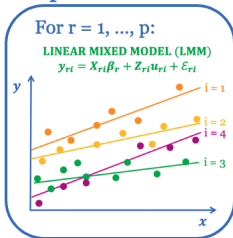
Evaluation of predictive performance

Wrap-up

PRC: modelling steps



Step 1



Step 2

Compute the
predicted random effects

$$\hat{u}_{ri} = \hat{W}_{ri}(y_{ri} - X_{ri}\hat{\beta}_r)$$

where

$$\hat{W}_{ri} = \hat{D}_r Z_{ri} \hat{V}_{ri}^{-1}$$

Step 3

Estimate

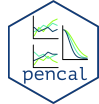
$$h(t|\eta_i) = h_0(t)e^{\eta_i},$$

where

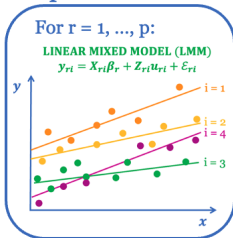
$$\eta_i = x_i^T \tau + \hat{u}_i^T \gamma,$$

using **penalized maximum likelihood** (ridge / lasso / elasticnet)

Step 1: model the longitudinal data



Step 1



Step 2

Compute the
predicted random effects

$$\hat{u}_{ri} = \hat{W}_{ri}(y_{ri} - X_{ri}\hat{\beta}_r)$$

where

$$\hat{W}_{ri} = \hat{D}_r Z_{ri} \hat{V}_{ri}^{-1}$$

Step 3

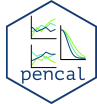
Estimate
 $h(t|\eta_i) = h_0(t)e^{\eta_i}$,
where
 $\eta_i = x_i^T \tau + \hat{u}_i^T \gamma$,
using **penalized maximum likelihood** (ridge / lasso / elasticnet)

Step 1: model the longitudinal data



- Step 1: fit to each longitudinal Y_r a LMM: $y_{ri} = X_{ri}\beta_r + Z_{ri}u_{ri} + \varepsilon_{ri}$

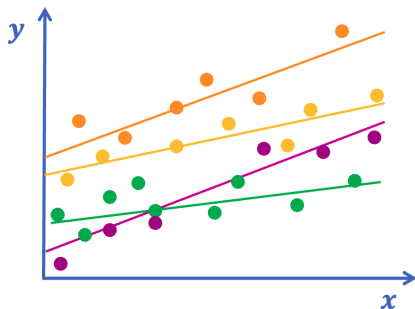
Step 1: model the longitudinal data



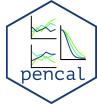
► Example:

$$y_{rij} = \beta_{r0} + u_{r0i} + (\beta_{r1} + u_{r1i})a_{ij} + \varepsilon_{rij},$$

where $u_{ri} = (u_{r0i}, u_{r1i}) \sim N(0, D_r)$ and $\varepsilon_{ri} \sim N(0, \sigma_r^2 I_{m_i})$



Step 1: fit_lmms



```
long_covs = c('logSerBil', 'logSerChol', 'albumin',  
              'logAlk', 'logSGOT', 'platelets',  
              'logProthr')  
  
step1 = fit_lmms(y.names = long_covs,  
                fixeefs = ~ age, ranefs = ~ age | id,  
                long.data = ldata, surv.data = sdata,  
                t.from.base = fuptime)
```

- ▶ `ranefs` argument follows the `nlme` package notation for the random effects formula (random intercept: `~ 1 | id`, random slope: `~ x | id, ...`)

Extracting output from the fitted LMMs



```
summary(step1, yname = 'logSerBil', what = 'betas') |> round(6)
```

```
## (Intercept)      age
##    0.518320   -0.001045
```

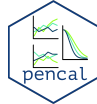
```
summary(step1, yname = 'logSerBil', what = 'tTable') |> round(4)
```

```
##           Value Std.Error  DF t-value p-value
## (Intercept) 0.5183    0.2788 566  1.8590  0.0636
## age         -0.0010    0.0055 566 -0.1884  0.8506
```

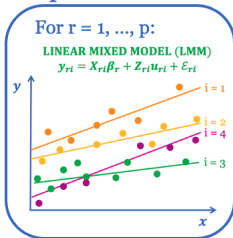
```
summary(step1, yname = 'logSerBil', what = 'variances')
```

```
## id = pdLogChol(age)
##           Variance      StdDev      Corr
## (Intercept) 7.332118e-01 0.856277849 (Intr)
## age         4.731627e-05 0.006878682 0.103
## Residual    1.437622e-01 0.379159888
```

Step 2: summarize the longitudinal predictors



Step 1



Step 2

Compute the
predicted random effects

$$\hat{u}_{ri} = \hat{W}_{ri}(y_{ri} - X_{ri}\hat{\beta}_r)$$

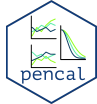
where

$$\hat{W}_{ri} = \hat{D}_r Z_{ri} \hat{V}_{ri}^{-1}$$

Step 3

Estimate
 $h(t|\eta_i) = h_0(t)e^{\eta_i}$,
where
 $\eta_i = x_i^T \tau + \hat{u}_i^T \gamma$,
using **penalized maximum likelihood** (ridge / lasso / elasticnet)

Step 2: summarize the longitudinal predictors



- ▶ Derive **subject-specific summaries of the longitudinal trajectories** from the mixed-effects models
- ▶ We use the **predicted random effects** from the LMM:

$$\hat{u}_{ri} = E(u_{ri} | Y_{ri} = y_{ri}) = \hat{D}_r Z_i^T \hat{V}_{ri}^{-1} (y_{ri} - X_i \hat{\beta}_r),$$

where $V_{ri} = Z_i D_r Z_i^T + \sigma_r^2 I_{m_i}$ is the marginal covariance matrix of subject i

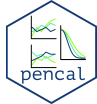
Step 2: summarize_lmms



```
step2 = summarize_lmms(step1)
```

- ▶ Handy: `summarize_lmms` automatically inherits relevant arguments from `fit_lmms` 😊

Step 2: sample output



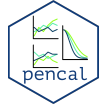
```
summary(step2)
```

```
## Number of predicted random effect variables: 14
## Sample size: 278
```

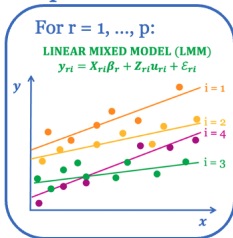
```
round(step2$ranef.orig[1:4, 1:6], 6)
```

```
##   logSerBil_b_int logSerBil_b_age logSerChol_b_int
## 2      -0.382988      -0.001661      -0.071154
## 3      -0.117107      -0.000584      -0.598453
## 4       0.168600       0.000922      -0.370434
## 5       0.380035       0.001170      -0.291031
##   logSerChol_b_age albumin_b_int albumin_b_age
## 2       0.000660       0.179725       3e-06
## 3       0.004916       0.018124       1e-06
## 4       0.003468      -0.529776      -7e-06
## 5       0.002886      -0.148329       8e-06
```

Step 3: model the survival outcome



Step 1



Step 2

Compute the
predicted random effects

$$\hat{u}_{ri} = \hat{W}_{ri}(y_{ri} - X_{ri}\hat{\beta}_r)$$

where

$$\hat{W}_{ri} = \hat{D}_r Z_{ri} \hat{V}_{ri}^{-1}$$

Step 3

Estimate

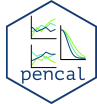
$$h(t|\eta_i) = h_0(t)e^{\eta_i},$$

where

$$\eta_i = x_i^T \tau + \hat{u}_i^T \gamma,$$

using **penalized maximum likelihood** (ridge / lasso / elasticnet)

Step 3: model the survival outcome



- ▶ Cox model linking **survival outcome** to **baseline covariates** and **summaries of longitudinal covariates**:

$$h(t_i | x_i, \hat{u}_{0i}, \hat{u}_{1i}) = h_0(t_i) \exp(\eta_i), \quad (1)$$

$$\eta_i = \sum_{q=1}^k \theta_q x_{qi} + \sum_{r=1}^p \gamma_r \hat{u}_{r0i} + \sum_{r=1}^p \delta_r \hat{u}_{r1i}$$

- ▶ Estimated using **penalized** maximum likelihood

$$\max_{\xi, \gamma, \delta} \ell(\xi, \gamma, \delta) - \lambda p(\xi, \gamma, \delta; \alpha)$$

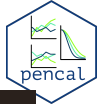
- ▶ Penalty functions: ridge (ℓ^2 , recommended), lasso (ℓ^1), elastic net

Step 3: fit_prclmm



```
step3 = fit_prclmm(step2, surv.data = sdata,  
  baseline.covs = ~ baselineAge + sex + treatment,  
  penalty = 'ridge', standardize = T)
```

Step 3: fitted model



```
summary(step3)
```

```
## Fitted model: PRC-LMM
## Penalty function used: ridge
## Tuning parameters:
##      lambda alpha
## 1 0.2126761      0
## Sample size: 278
## Number of events: 107
## Bootstrap optimism correction: not computed
## Penalized likelihood estimates (rounded to 4 digits):
##   baselineAge sexfemale treatmentD-penicil logSerBil_b_int
## 1      0.0476   -0.2872             -0.0157           0.4341
##   logSerBil_b_age logSerChol_b_int logSerChol_b_age
## 1      111.3935             0.0986          -10.5311
##   albumin_b_int albumin_b_age logAlk_b_int logAlk_b_age
## 1      -1.1361      23070.92           0.0874      -12.5617
##   logSGOT_b_int logSGOT_b_age platelets_b_int
## 1         0.238       272.246           -0.0011
##   platelets_b_age logProthr_b_int logProthr_b_age
## 1         -0.2046           2.8114          -573.3093
```



Dynamic prediction and the R package `pencal`

Input data

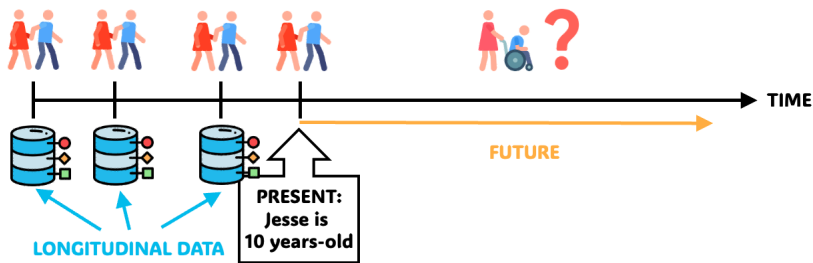
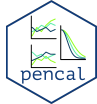
Model estimation

Prediction

Evaluation of predictive performance

Wrap-up

Back to our goal: predicting $P(T > t | T > \ell)$



Prediction of survival probabilities



- ▶ Predicted survival probabilities: $\hat{S}(h|\ell) = \hat{S}(h) = e^{-\int_0^h \hat{h}_0(s) e^{\hat{\eta}_i} ds}$

```
Shat = survpred_prc1mm(step1, step2, step3, times = 3:5)
```

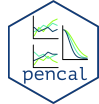
- ▶ This will compute $\hat{S}(t|2)$, $t = 3, 4, 5$:

```
head(Shat$predicted_survival, 4) |> dfround(3)
```

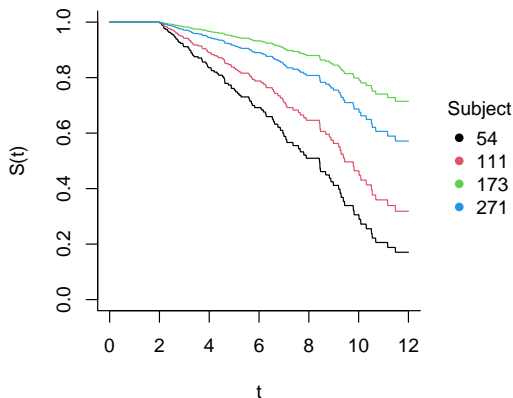
```
##   id  S(3)  S(4)  S(5)
## 2   2 0.940 0.887 0.833
## 3   3 0.856 0.739 0.632
## 4   4 0.814 0.671 0.545
## 5   5 0.946 0.898 0.849
```

- ▶ Prediction for **new** subjects? Possible through additional arguments `new.longdata` and `new.basecovs`

Visualizing predictions



```
survplot_prc(step1, step2, step3,  
             ids = c(54, 111, 173, 271), tmax = 12)
```





Dynamic prediction and the R package `pencal`

Input data

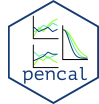
Model estimation

Prediction

Evaluation of predictive performance

Wrap-up

Evaluation of predictive performance



- ▶ Predictive performance estimated through a “**cluster bootstrap optimism correction** procedure” (Signorelli et al., 2021)
- ▶ Performance measures: C index, time-dependent AUC, Brier score

Evaluation of predictive performance with pencal

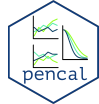


1. Rerun all 3 steps adding `n.boots` argument to step 1:

```
step1 = fit_lmms(y.names = long_covs, fixeefs = ~ age,
                 ranefs = ~ age | id, t.from.base = fuptime,
                 long.data = ldata, surv.data = sdata,
                 n.boots = 100, n.cores = 8, verbose = FALSE)
step2 = summarize_lmms(object = step1, n.cores = 8,
                       verbose = FALSE)
step3 = fit_prclmm(object = step2, surv.data = sdata,
                   baseline.covs = ~ baselineAge + sex + treatment,
                   penalty = 'ridge', n.cores = 8, verbose = FALSE)
```

- Use `n.cores` argument to automatically parallelize computations

Evaluation of predictive performance with pencial



2. Call `performance_prc` to evaluate the predictive performance:

```
cbocp = performance_prc(step2 = step2, step3 = step3,  
  metric = c('tdauc', 'brier'), times = 3:5,  
  n.cores = 8, verbose = FALSE)
```

- `times` indicates the time points at which to evaluate the time-dependent AUC and Brier score

```
cbocp$tdAUC
```

##	pred.time	tdAUC.naive	optimism.correction	tdAUC.adjusted
## 1	3	0.9439	-0.0060	0.9379
## 2	4	0.9351	-0.0152	0.9199
## 3	5	0.9266	-0.0128	0.9138

```
cbocp$Brier
```

##	pred.time	Brier.naive	optimism.correction	Brier.adjusted
## 1	3	0.0571	0.0132	0.0703
## 2	4	0.0699	0.0251	0.0950
## 3	5	0.0844	0.0307	0.1151



Dynamic prediction and the R package `pencal`

Input data

Model estimation

Prediction

Evaluation of predictive performance

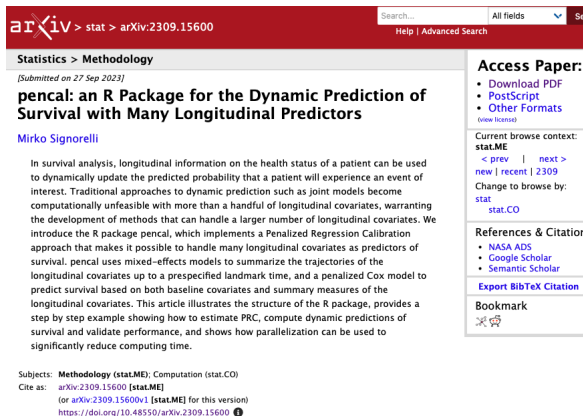
Wrap-up

Summary



- ▶ `pencal` makes it **easy** to do **dynamic prediction** of survival outcomes when you have **numerous longitudinal** predictors
- ▶ `pencal` automates:
 1. model estimation
 2. prediction of survival probabilities
 3. evaluation of predictive performance
- ▶ **Easy parallelization**: just specify `n.cores` as argument inside `pencal`'s functions

- More details: vignette (Signorelli, 2024) available at [arXiv:2309.15600](https://arxiv.org/abs/2309.15600):

A screenshot of the arXiv.org page for the paper 'pencial: an R Package for the Dynamic Prediction of Survival with Many Longitudinal Predictors' by Mirko Signorelli. The page has a red header with the arXiv logo and navigation links. The main content area is white with a blue sidebar on the right. The paper title is in bold black text, followed by the author's name. The abstract is a paragraph of text. The sidebar contains links to download the paper in various formats, a 'Current browse context' section with navigation links, a 'References & Citation' section with links to NASA ADS, Google Scholar, and Semantic Scholar, and a 'Bookmark' section with a share icon. The bottom of the page shows the subjects and citation information.

arXiv > stat > arXiv:2309.15600

Search... All fields Help | Advanced Search

Statistics > Methodology

[Submitted on 27 Sep 2023]

pencial: an R Package for the Dynamic Prediction of Survival with Many Longitudinal Predictors

Mirko Signorelli

In survival analysis, longitudinal information on the health status of a patient can be used to dynamically update the predicted probability that a patient will experience an event of interest. Traditional approaches to dynamic prediction such as joint models become computationally unfeasible with more than a handful of longitudinal covariates, warranting the development of methods that can handle a larger number of longitudinal covariates. We introduce the R package pencial, which implements a Penalized Regression Calibration approach that makes it possible to handle many longitudinal covariates as predictors of survival. pencial uses mixed-effects models to summarize the trajectories of the longitudinal covariates up to a prespecified landmark time, and a penalized Cox model to predict survival based on both baseline covariates and summary measures of the longitudinal covariates. This article illustrates the structure of the R package, provides a step by step example showing how to estimate PRC, compute dynamic predictions of survival and validate performance, and shows how parallelization can be used to significantly reduce computing time.

Subjects: **Methodology** (stat.ME); Computation (stat.CO)

Cite as: [arXiv:2309.15600](https://arxiv.org/abs/2309.15600) [stat.ME]
(or [arXiv:2309.15600v1](https://arxiv.org/abs/2309.15600v1) [stat.ME] for this version)
<https://doi.org/10.48550/arXiv.2309.15600>

Access Paper:

- Download PDF
- PostScript
- Other Formats (view license)

Current browse context: **stat.ME**

< prev | next >
new | recent | 2309

Change to browse by:

stat
stat.CO

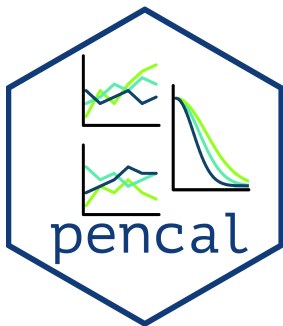
References & Citation

- NASA ADS
- Google Scholar
- Semantic Scholar

Export BibTeX Citation

Bookmark

- Accepted for publication in [The R Journal](https://www.tandfonline.com/journals/tbjr)



 mirkosignorelli.github.io
 m.signorelli@math.leidenuniv.nl
 [in/signorelli](https://www.linkedin.com/in/signorelli)
 [@signormirko](https://twitter.com/signormirko)

Vignette: bit.ly/pencal-UseR

 [in/signorelli](https://www.linkedin.com/in/signorelli)

Signorelli, M. (2024). *pencal*: an R Package for the Dynamic Prediction of Survival with Many Longitudinal Predictors. *To Appear in: The R Journal*. Preprint: *arXiv:2309.15600*.

Signorelli, M., Spitali, P., Al-Khalili Sgyziarto, C., The Mark-MD Consortium, & Tsonaka, R. (2021). Penalized regression calibration: A method for the prediction of survival outcomes using complex longitudinal and high-dimensional data. *Statistics in Medicine*.