Dynamic random survival forests using functional principal component analysis for the prediction of survival outcomes from time-varying predictors

Corentin Ségalas, Robin Genuer, Cécile Proust-Lima
csegalas

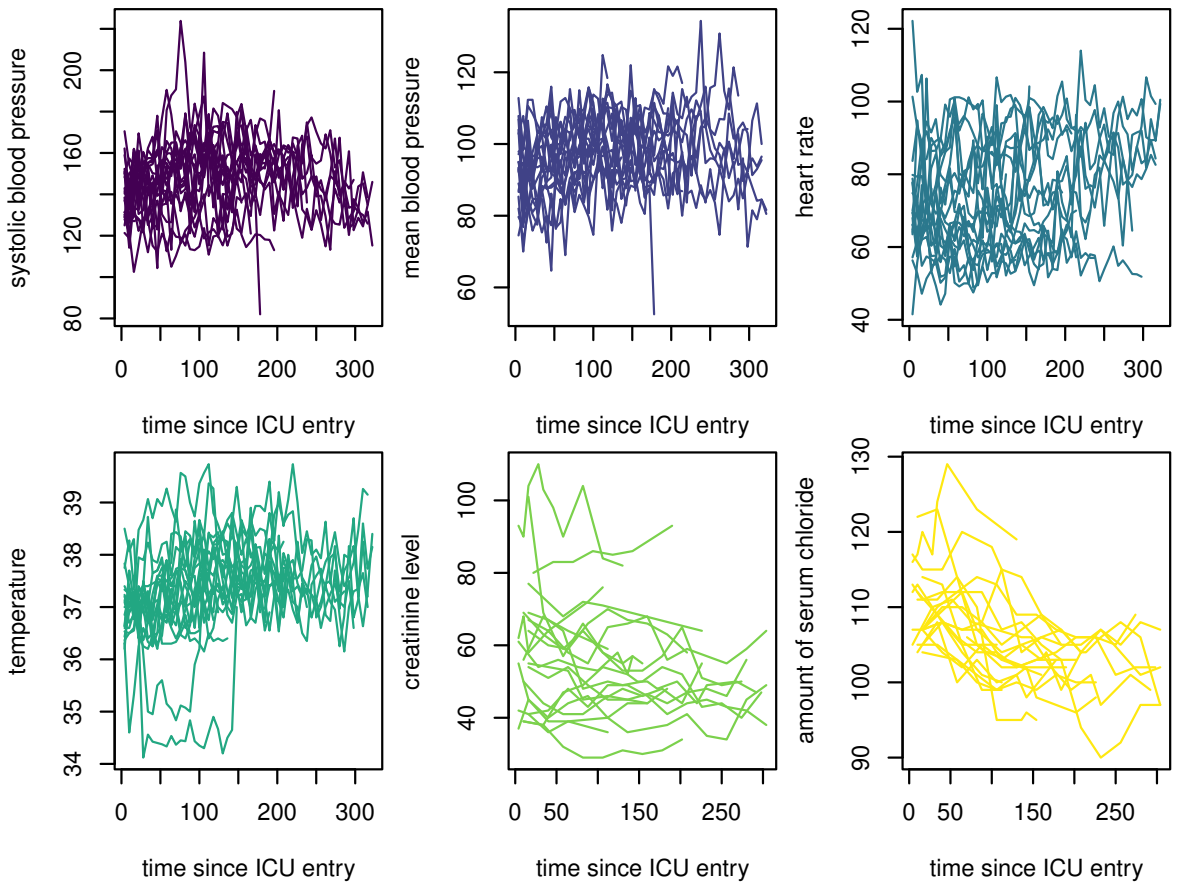
ISCB 2024

BORDEAUX POPULATION HEALTH Research Center - U1219
université de BORDEAUX
Inria

Statistical context

Longitudinal biomarkers

$y_{ij} = y_i^\star(t_{ij}) + \varepsilon_{ij}$ with $i = 1, \ldots, n$ and $j = 1, \ldots, n_i$



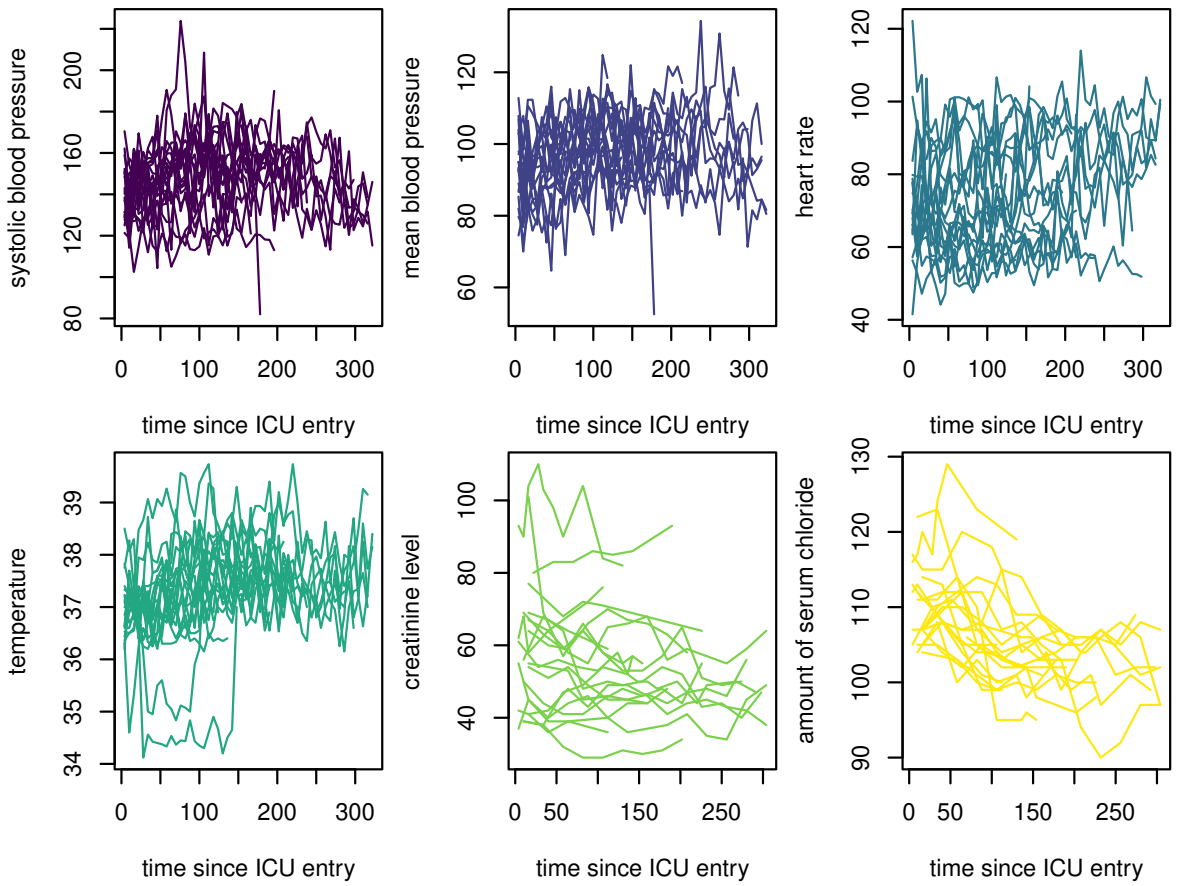
Statistical context

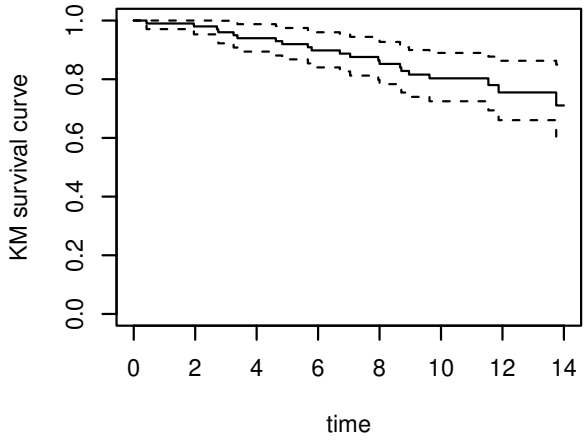
Longitudinal biomarkers

$$y_{ij} = y_i^\star(t_{ij}) + \varepsilon_{ij} \quad \text{with } i = 1, \ldots, n \text{ and } j = 1, \ldots, n_i$$

Time-to-event outcome

$$\lambda(t_{ij})$$





Dynamic prediction: limits of existing approaches

Landmark approach (Van Houwelingen, 2007)

- Easy to implement
- Information loss
- Individual prediction only at landmark times used to build the model

Dynamic prediction: limits of existing approaches

Landmark approach (Van Houwelingen, 2007)

- Easy to implement
- Information loss
- Individual prediction only at landmark times used to build the model

Shared random effect joint models (Rizopoulos, 2012)

- Huge numerical integration
- Number of predictors limited

$\longrightarrow$ Calibration-regression (but bias)

Random Survival Forest

Random Forest framework (Breiman, 2001)

- Aggregation of binary trees (classification/regression)

Random Survival Forest

Random Forest framework (Breiman, 2001)

- Aggregation of binary trees (classification/regression)
- A tree is built for each of the B bootstrap samples
- At each node, only a subset of predictors as candidate to split

Random Survival Forest

Random Forest framework (Breiman, 2001)

- Aggregation of binary trees (classification/regression)
- A tree is built for each of the B bootstrap samples
- At each node, only a subset of predictors as candidate to split
- Can model complex relation between many predictors

Random Survival Forest

Random Forest framework (Breiman, 2001)

- Aggregation of binary trees (classification/regression)
- A tree is built for each of the B bootstrap samples
- At each node, only a subset of predictors as candidate to split
- Can model complex relation between many predictors

Random Survival Forest

Random Forest framework (Breiman, 2001)

- Aggregation of binary trees (classification/regression)
- A tree is built for each of the B bootstrap samples
- At each node, only a subset of predictors as candidate to split
- Can model complex relation between many predictors

$\longrightarrow$ Out-Of-Bag error, variable importance

Random Survival Forest

Random Forest framework (Breiman, 2001)

- Aggregation of binary trees (classification/regression)
- A tree is built for each of the B bootstrap samples
- At each node, only a subset of predictors as candidate to split
- Can model complex relation between many predictors

$\longrightarrow$ Out-Of-Bag error, variable importance

Random Survival Forest (Ishwaran et al., 2008)

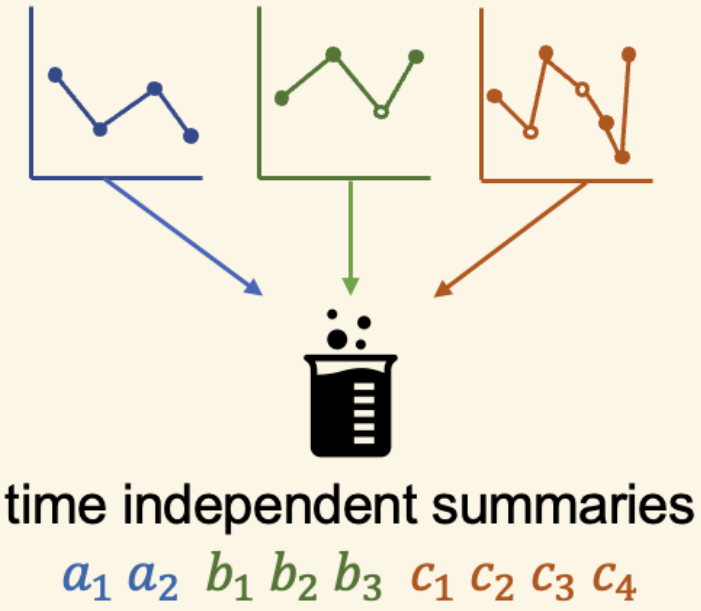
- Extension of RF suited to survival outcome

$\surd$ **time-independent** $\times$ **time-dependent**

Dynamic Random Survival Forest

Core idea

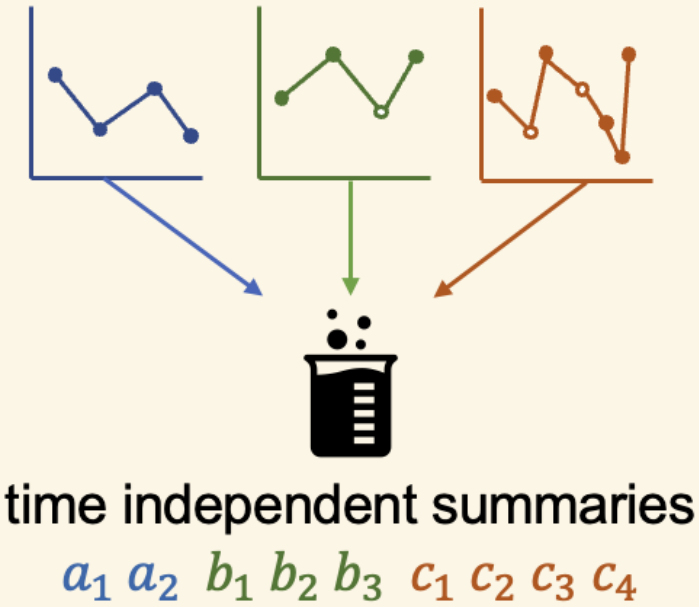
Inside each node, summarize time-dependent predictors by time-independent summaries



Dynamic Random Survival Forest

Core idea

Inside each node, summarize time-dependent predictors by time-independent summaries



DynForest (Devaux et al., 2023)

Time-independent summaries: random effects from a mixed model

$\longrightarrow$ **Parametric assumptions needed**

$\longrightarrow$ **Computational limitations**

Functional Principal Component Analysis

Karhunen-Loève decomposition

We assume $y_i^\star(t)$ a random process with mean function $\mu(t)$ and covariance $G(s,t) = \sum_{k=1}^{\infty} \lambda_k \psi_k(s)\psi_k(t)$ with eigenvalues λ_k and eigenfunctions ψ_k.

$$y_i^\star(t) = \mu(t) + \sum_{k=1}^{\infty} \xi_{ik}\psi_k(t) \quad i = 1, \ldots, N, \quad t \in \mathbb{R}$$

with ξ_{ik} principal component scores, $E(\xi_{ik}) = 0$ and $Var(\xi_{ik}) = \lambda_k$.

Functional Principal Component Analysis

Karhunen-Loève decomposition

We assume $y_i^\star(t)$ a random process with mean function $\mu(t)$ and covariance $G(s,t) = \sum_{k=1}^{\infty} \lambda_k \psi_k(s)\psi_k(t)$ with eigenvalues λ_k and eigenfunctions ψ_k.

$$y_i^\star(t) = \mu(t) + \sum_{k=1}^{\infty} \xi_{ik}\psi_k(t) \quad i = 1, \ldots, N, \quad t \in \mathbb{R}$$

with ξ_{ik} principal component scores, $E(\xi_{ik}) = 0$ and $Var(\xi_{ik}) = \lambda_k$.

Functional Principal Component Analysis

Karhunen-Loève decomposition - truncated

We assume $y_i^\star(t)$ a random process with mean function $\mu(t)$ and covariance $G(s,t) = \sum_{k=1}^{\infty} \lambda_k \psi_k(s)\psi_k(t)$ with eigenvalues λ_k and eigenfunctions ψ_k.

$$y_i^\star(t) = \mu(t) + \sum_{k=1}^{K} \xi_{ik}\psi_k(t) \quad i = 1, \ldots, N, \quad t \in \mathbb{R}$$

with ξ_{ik} principal component scores, $E(\xi_{ik}) = 0$ and $Var(\xi_{ik}) = \lambda_k$.

Functional Principal Component Analysis

Karhunen-Loève decomposition - truncated

We assume $y_i^\star(t)$ a random process with mean function $\mu(t)$ and covariance $G(s,t) = \sum_{k=1}^{\infty} \lambda_k \psi_k(s)\psi_k(t)$ with eigenvalues λ_k and eigenfunctions ψ_k.

$$y_i^\star(t) = \mu(t) + \sum_{k=1}^{K} \xi_{ik}\psi_k(t) \quad i = 1, \ldots, N, \quad t \in \mathbb{R}$$

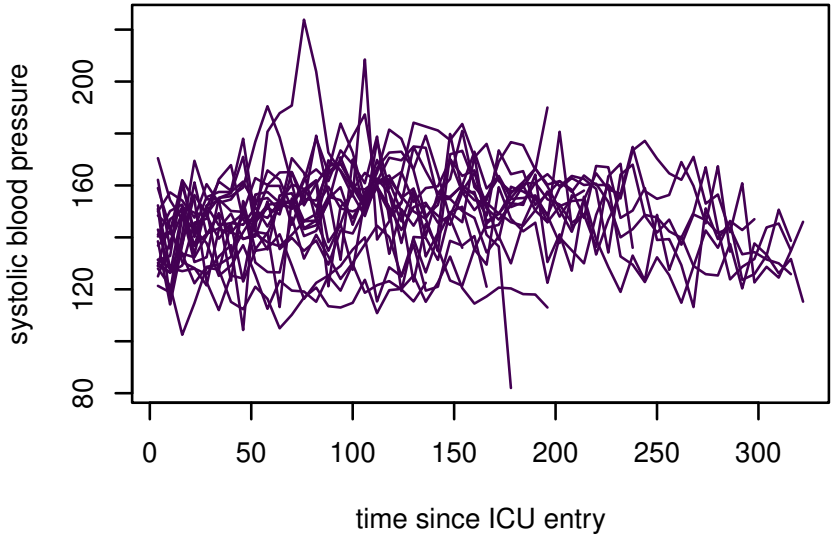
with ξ_{ik} principal component scores, $E(\xi_{ik}) = 0$ and $Var(\xi_{ik}) = \lambda_k$.

PACE algorithm (Yao et al., 2005)

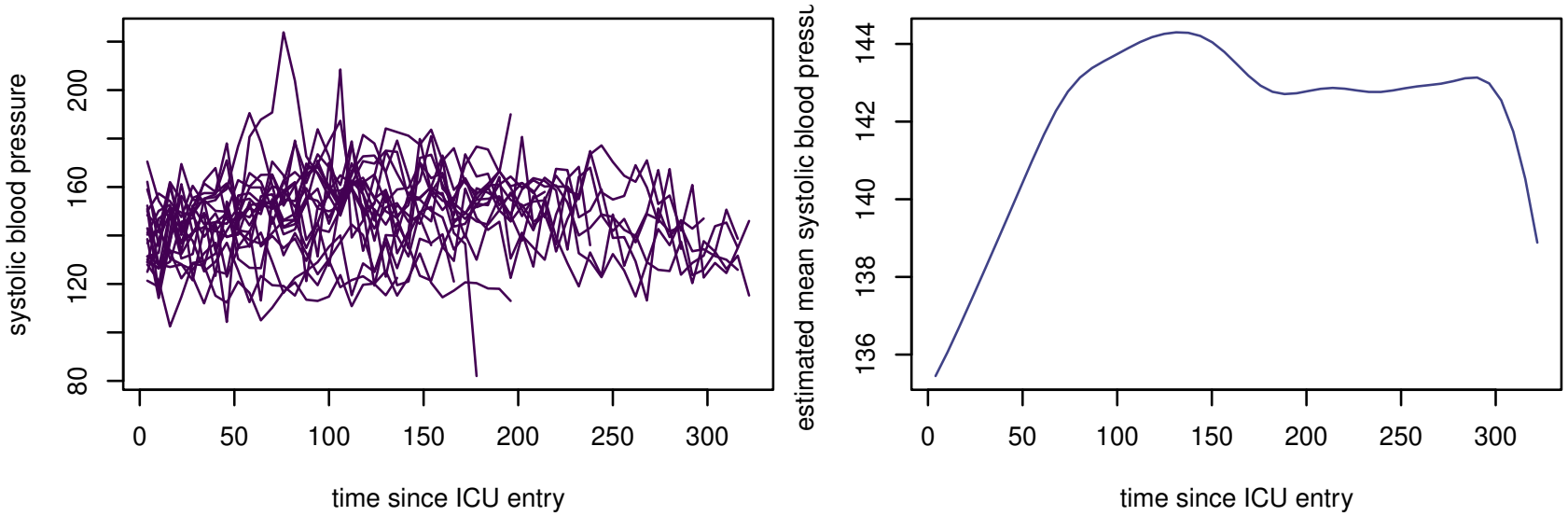
Fit for sparse and irregular functional data. For a chosen K:

- $\hat{\mu}(t)$ and $\hat{\psi}_k(t)$ over a time grid
- $\hat{\xi}_{ik}$ for $k = 1, \ldots, K$ and for all i

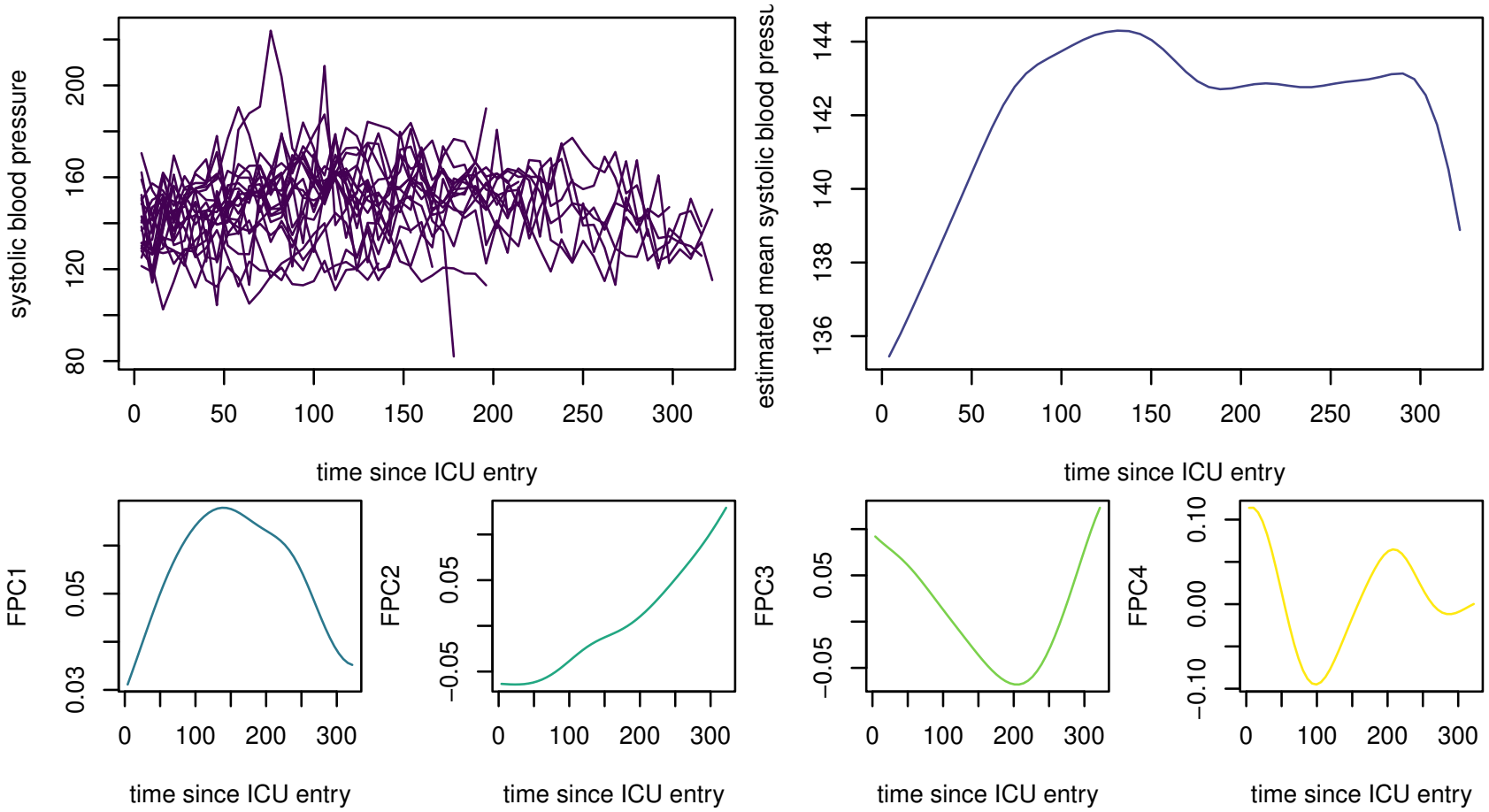
FPCA to summarise longitudinal data



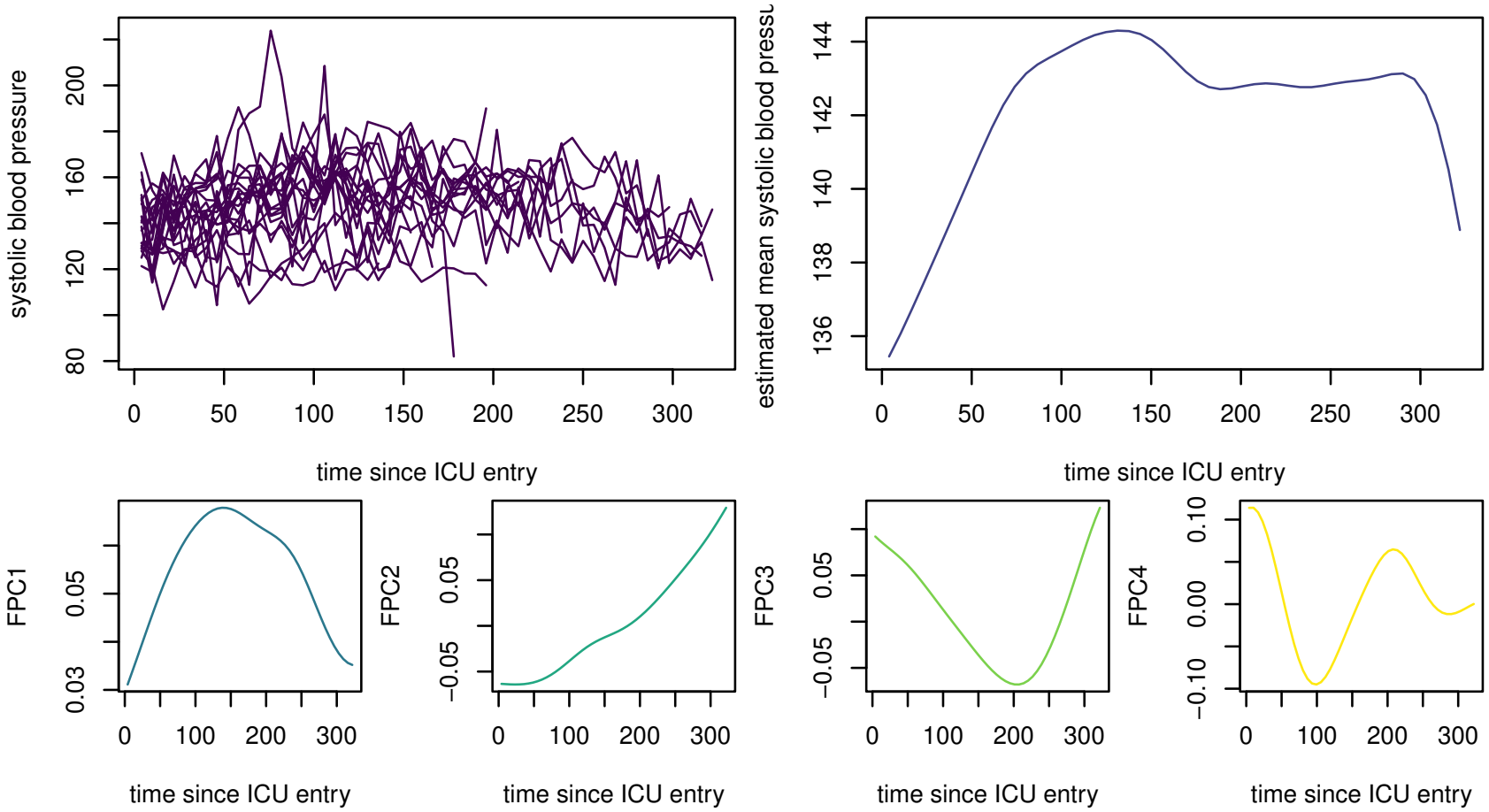
FPCA to summarise longitudinal data



FPCA to summarise longitudinal data



FPCA to summarise longitudinal data



i	$\hat{\xi}_{i1}$	$\hat{\xi}_{i2}$	$\hat{\xi}_{i3}$	$\hat{\xi}_{i4}$
1	278.79	−32.57	5.09	−23.44
2	176.60	−85.12	−67.40	9.80

Is FPCA robust to missing data?

Is FPCA robust to missing data?

Functional principal component analysis as an alternative to mixed-effect models for describing sparse repeated measures in presence of missing data

Corentin Ségalas[*,1], Catherine Helmer[2], Robin Genuer[†,1] and Cécile Proust-Lima[†,2]

[1]Univ. Bordeaux, INSERM, INRIA, BPH, U1219, F-33000 Bordeaux, France
[2]Univ. Bordeaux, INSERM, BPH, U1219, F-33000 Bordeaux, France

Is FPCA robust to missing data?

Functional principal component analysis as an alternative to mixed-effect models for describing sparse repeated measures in presence of missing data

Corentin Ségalas*,[1], Catherine Helmer[2], Robin Genuer†,[1] and Cécile Proust-Lima†,[2]

[1]Univ. Bordeaux, INSERM, INRIA, BPH, U1219, F-33000 Bordeaux, France
[2]Univ. Bordeaux, INSERM, BPH, U1219, F-33000 Bordeaux, France

Simulation study

FPCA is robust to MAR data under non pathological scenarios

Simulation code

Available on github `@csegalas`

Functional `DynForest` in R

Functional DynForest in R

```
timeVarModel <- list(timeVar1 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar2 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar3 = list(PVEfpca = 0.99, nRegGrid = 30),
                     timeVar4 = list(PVEfpca = 0.99, nRegGrid = 30))
```

Functional DynForest in R

```
timeVarModel <- list(timeVar1 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar2 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar3 = list(PVEfpca = 0.99, nRegGrid = 30),
                     timeVar4 = list(PVEfpca = 0.99, nRegGrid = 30))
```

```
mb_fpcaDF <- DynForest(timeData = timeData_train,
                       fixedData = fixedData_train,
                       timeVar = "timeVariable", idVar = "ID",
                       timeVarModel = timeVarModel, Y = Y,
                       ntree = 500, nodesize = 5, minsplit = 5,
                       cause = 1, ncores = 1, seed = 1234)
```

Functional DynForest in R

```
timeVarModel <- list(timeVar1 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar2 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar3 = list(PVEfpca = 0.99, nRegGrid = 30),
                     timeVar4 = list(PVEfpca = 0.99, nRegGrid = 30))
```

```
mb_fpcaDF <- DynForest(timeData = timeData_train,
                       fixedData = fixedData_train,
                       timeVar = "timeVariable", idVar = "ID",
                       timeVarModel = timeVarModel, Y = Y,
                       ntree = 500, nodesize = 5, minsplit = 5,
                       cause = 1, ncores = 1, seed = 1234)
```

```
OOB_fpcaDF <- compute_OOBerror(mb_fpcaDF)
```

Functional DynForest in R

```
timeVarModel <- list(timeVar1 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar2 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar3 = list(PVEfpca = 0.99, nRegGrid = 30),
                     timeVar4 = list(PVEfpca = 0.99, nRegGrid = 30))
```

```
mb_fpcaDF <- DynForest(timeData = timeData_train,
                       fixedData = fixedData_train,
                       timeVar = "timeVariable", idVar = "ID",
                       timeVarModel = timeVarModel, Y = Y,
                       ntree = 500, nodesize = 5, minsplit = 5,
                       cause = 1, ncores = 1, seed = 1234)
```

```
OOB_fpcaDF <- compute_OOBerror(mb_fpcaDF)
```

```
VIMP_fpccaDF <- compute_VIMP(mb_fpcaDF)
```

Functional DynForest in R

```r
timeVarModel <- list(timeVar1 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar2 = list(PVEfpca = 0.99, nRegGrid = 50),
                     timeVar3 = list(PVEfpca = 0.99, nRegGrid = 30),
                     timeVar4 = list(PVEfpca = 0.99, nRegGrid = 30))
```

```r
mb_fpcaDF <- DynForest(timeData = timeData_train,
                       fixedData = fixedData_train,
                       timeVar = "timeVariable", idVar = "ID",
                       timeVarModel = timeVarModel, Y = Y,
                       ntree = 500, nodesize = 5, minsplit = 5,
                       cause = 1, ncores = 1, seed = 1234)
```

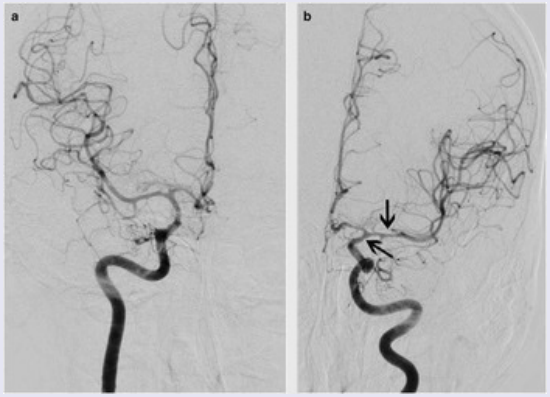
```r
OOB_fpcaDF <- compute_OOBerror(mb_fpcaDF)
```

```r
VIMP_fpccaDF <- compute_VIMP(mb_fpcaDF)
```

```r
pred_fpcaDF <- predict(mb_fpcaDF,
                       timeData = timeData_test,
                       fixedData = fixedData_test,
                       idVar = "ID",
                       timeVar = "timeVariable",
                       t0 = 100)
```

Vasospasm data

Cerebral vasospasm

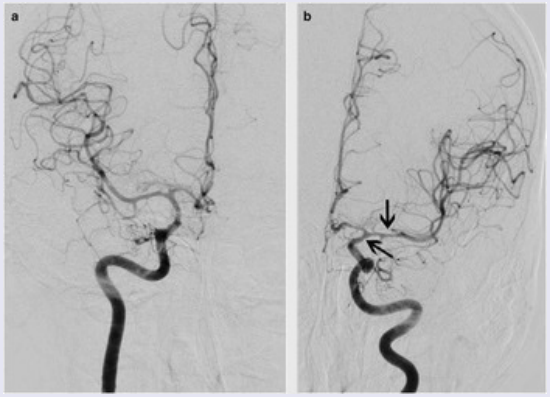


Narrowing of brain blood vessel, complication after a subarachnoid hemorrhage.

Hard to anticipate, and hard to treat if diagnosed too late.

Vasospasm data

Cerebral vasospasm

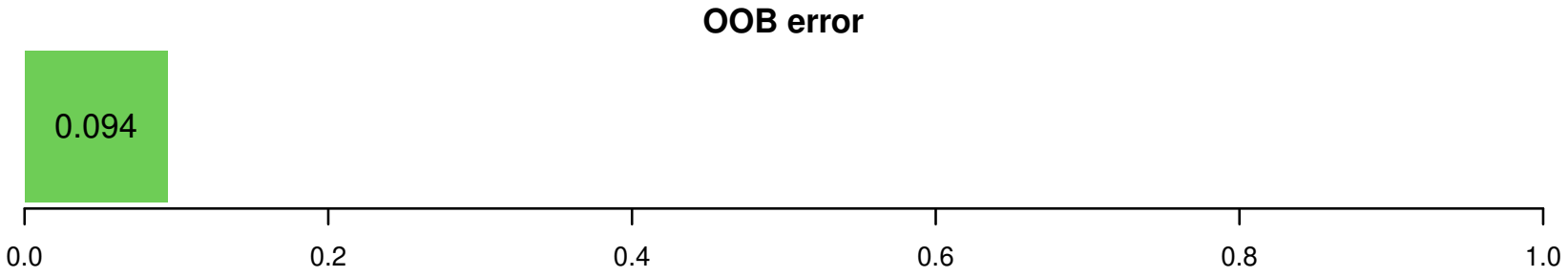


Narrowing of brain blood vessel, complication after a subarachnoid hemorrhage.
Hard to anticipate, and hard to treat if diagnosed too late.

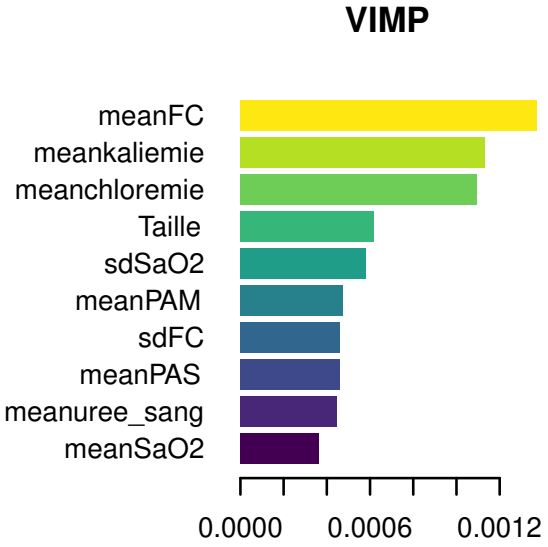
Vasospasm data from CHU de Bordeaux

- 201 patients
- 14 days of hourly follow-up after ICU admission
- 12 longitudinal biomarkers (BP, temperature, heart rate, etc.) + their standard deviation trend + 9 fixed variables (demographic, sex, tobacco, etc.). Some missing data.
- 46 vasospasms

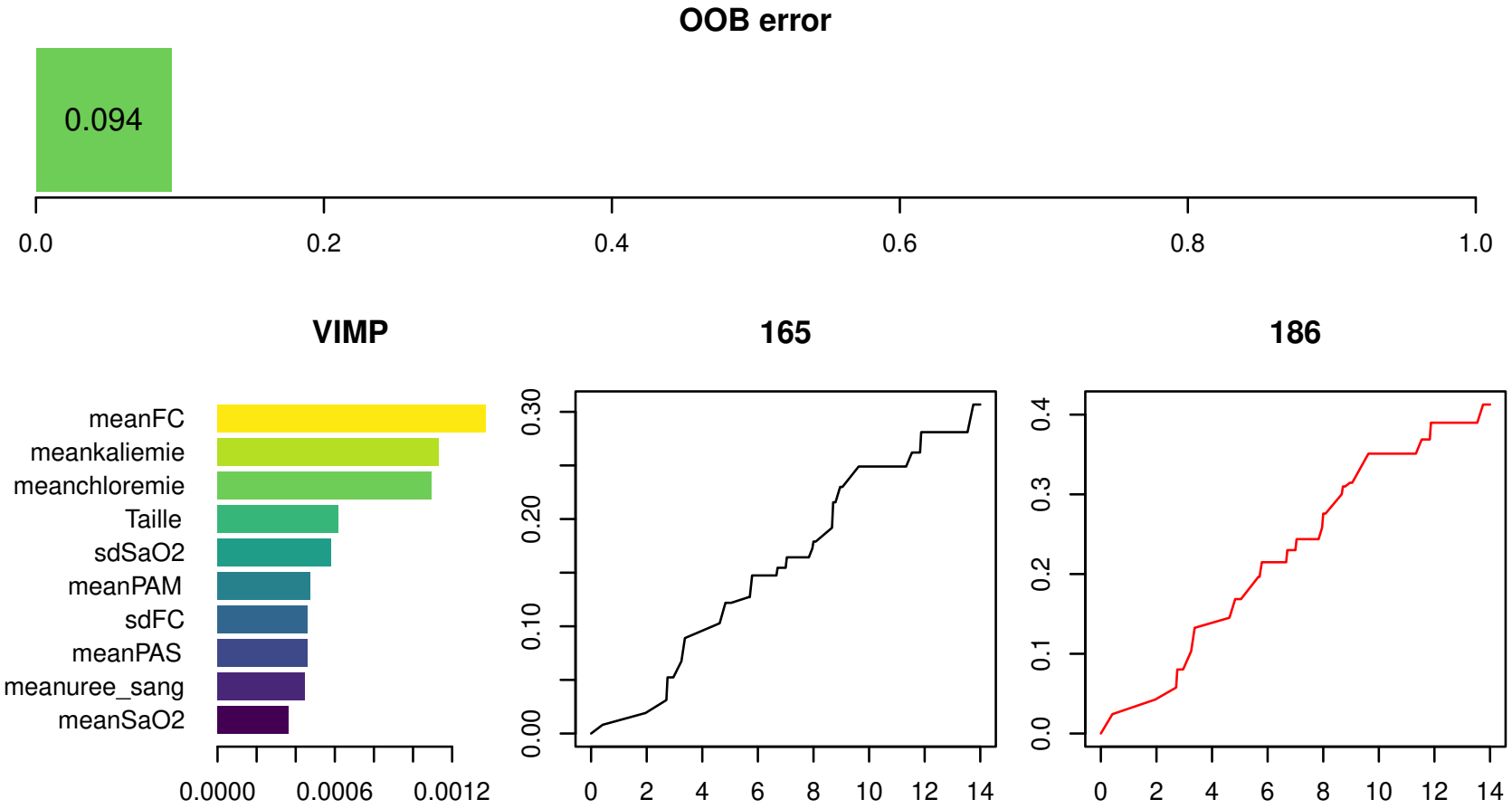
Results on 500 trees



Results on 500 trees



Results on 500 trees



Discussion and perspectives

Functional DynForest

Nonparametric method to predict **time-to-event outcome** from **longitudinal predictors**

- handle informative missingness
- able to open the black box (variable importance)
- flexible: both FPCA and mixed-models; derivatives of longitudinal trajectories

Discussion and perspectives

Functional DynForest

Nonparametric method to predict **time-to-event outcome** from **longitudinal predictors**

- handle informative missingness
- able to open the black box (variable importance)
- flexible: both FPCA and mixed-models; derivatives of longitudinal trajectories

Future work

- a complete simulation study
- logrank assumption
- time computation and code cleaning
- different types of outcome (longitudinal data)

Acknowledgements

The `DynForest` family

Cécile Proust-Lima

Robin Genuer

Anthony Devaux

Funding

CARE project, Innovative Medicines Initiative 2 (No 101005077)

Resources

Bibliography

- Van Houwelingen, 2005, Dynamic Prediction by Landmarking in Event History Analysis, Scandinavian Journal of Statistics
- Rizopoulos, 2008, Joint Models for Longitudinal and Time-to-Event Data: With Applications in R, CRC Press
- Breiman, 2001, Random Forests, Machine Learning
- Ishwaran et al., 2008, Random Survival Forests, The Annals of Applied Statistics
- Devaux et al., 2023, Random survival forests with multivariate longitudinal endogenous covariates, Statistical Methods in Medical Research
- Yao et al., 2005, Functional Data Analysis for Sparse Longitudinal Data, Journal of the American Statistical Association

R Packages

- random survival forests: `DynForest`, `survival`
- functional data: `fdapace`, `FunData`
- data management and plotting: `tidyverse`, `viridis`

Thanks!