Package: zeitzeiger (via r-universe)

October 21, 2024

Type Package

Title Regularized Supervised Learning for Data from Rhythmic Systems

Version 2.1.3

Description Method for predicting the value of a periodic variable from a high-dimensional observation. See Hughey et al. (2016) <doi:10.1093/nar/gkw030> and Hughey (2017) <doi:10.1186/s13073-017-0406-4>.

URL https://zeitzeiger.hugheylab.org, https://github.com/hugheylab/zeitzeiger

License GPL-2

Encoding UTF-8

LazyData TRUE

RoxygenNote 7.2.2

Roxygen list(markdown = TRUE)

Depends R (>= 3.2)

Imports abind (>= 1.4-3), bbmle (>= 1.0.17), data.table (>= 1.14.2), foreach (>= 1.4.3), limma (>= 3.38.2), PMA (>= 1.0.9), statmod (>= 1.4.30), sva (>= 3.18.0)

Suggests doParallel (>= 1.0.10), ggplot2 (>= 3.1.0), knitr (>= 1.20), rmarkdown (>= 1.10)

VignetteBuilder knitr

Repository https://hugheylab.r-universe.dev

RemoteUrl https://github.com/hugheylab/zeitzeiger

RemoteRef HEAD

RemoteSha 2b8619997583e58d7733090f34c76ef8fd47afb2

Contents

getCircDiff . . . 2
predictIntensity . . . 3
zeitzeiger . . . 3
zeitzeigerBatch . . . 5
zeitzeigerEnsembleLikelihood . . . 7
zeitzeigerEnsembleMean . . . 7
zeitzeigerFit . . . 8
zeitzeigerFitCv . . . 9
zeitzeigerPredict . . . 9
zeitzeigerPredictCv . . . 10
zeitzeigerPredictGroup . . . 11
zeitzeigerPredictGroupCv . . . 13
zeitzeigerSpc . . . 14
zeitzeigerSpcCv . . . 15

Index **16**

`getCircDiff` *Calculate circular difference*

Description

Calculate circular difference.

Usage

```
getCircDiff(x, y, period = 1, towardZero = TRUE)
```

Arguments

`x`	Numeric vector or matrix.
`y`	Numeric vector or matrix.
`period`	Period of the periodic variable.
`towardZero`	If `TRUE`, returned values will be between `-period / 2` and `period / 2`. If `FALSE`, returned values will be between 0 and `period`.

Value

Vector or matrix corresponding to `x - y`.

predictIntensity *Calculate time-dependent mean*

Description

Calculate the expected value of each feature.

Usage

```
predictIntensity(fitCoef, time, period = 1, knots = NULL)
```

Arguments

fitCoef	Matrix of coefficients from the spline fits, where rows correspond to features and columns correspond to variables in the model.
time	Vector of values of the periodic variable for the observations, where 0 corresponds to the lowest possible value and 1 corresponds to the highest possible value.
period	Period for the periodic variable.
knots	Optional vector of knots. This argument is designed for internal use.

Value

Matrix of predicted measurements, where rows correspond to time-points and columns correspond to features.

See Also

zeitzeigerFit()

zeitzeiger *Train and test a ZeitZeiger predictor*

Description

Train and test a ZeitZeiger predictor, calling the necessary functions.

Usage

```
zeitzeiger(
  xTrain,
  timeTrain,
  xTest,
  nKnots = 3,
  nTime = 10,
  useSpc = TRUE,
  sumabsv = 2,
  orth = TRUE,
  nSpc = 2,
  timeRange = seq(0, 1 - 0.01, 0.01)
)
```

Arguments

xTrain	Matrix of measurements for training data, observations in rows and features in columns.
timeTrain	Vector of values of the periodic variable for training observations, where 0 corresponds to the lowest possible value and 1 corresponds to the highest possible value.
xTest	Matrix of measurements for test data, observations in rows and features in columns.
nKnots	Number of internal knots to use for the periodic smoothing spline.
nTime	Number of time-points by which to discretize the time-dependent behavior of each feature. Corresponds to the number of rows in the matri for which the SPCs will be calculated.
useSpc	Logical indicating whether to use `PMA::SPC()` (default) or `base::svd()`.
sumabsv	L1-constraint on the SPCs, passed to `PMA::SPC()`.
orth	Logical indicating whether to require left singular vectors be orthogonal to each other, passed to `PMA::SPC()`.
nSpc	Vector of the number of SPCs to use for prediction. If `NA` (default), `nSpc` will become `1:K`, where `K` is the number of SPCs in `spcResult`. Each value in `nSpc` will correspond to one prediction for each test observation. A value of 2 means that the prediction will be based on the first 2 SPCs.
timeRange	Vector of values of the periodic variable at which to calculate likelihood. The time with the highest likelihood is used as the initial value for the MLE optimizer.

Value

fitResult	Output of `zeitzeigerFit()`
spcResult	Output of `zeitzeigerSpc()`
predResult	Output of `zeitzeigerPredict()`

See Also

zeitzeigerFit(), zeitzeigerSpc(), zeitzeigerPredict()

---

zeitzeigerBatch *Train and test a ZeitZeiger predictor, accounting for batch effects*

Description

Train and test a predictor on multiple datasets independently, using sva::ComBat() to correct for batch effects prior to running zeitzeiger().

Usage

```
zeitzeigerBatch(
  ematList,
  trainStudyNames,
  sampleMetadata,
  studyColname,
  batchColname,
  timeColname,
  nKnots = 3,
  nTime = 10,
  useSpc = TRUE,
  sumabsv = 2,
  orth = TRUE,
  nSpc = 2,
  timeRange = seq(0, 1 - 0.01, 0.01),
  covariateName = NA,
  featuresExclude = NULL,
  dopar = TRUE
)
```

Arguments

ematList	Named list of matrices of measurements, one for each dataset, some of which will be for training, others for testing. Each matrix should have rownames corresponding to sample names and colnames corresponding to feature names.
trainStudyNames	Character vector of names in ematList corresponding to datasets for training.
sampleMetadata	data.frame containing relevant information for each sample across all datasets. Must have a column named sample.
studyColname	Name of column in sampleMetdata that contains information about which dataset each sample belongs to.
batchColname	Name of column in sampleMetdata that contains information about which dataset each sample belongs to. This should correspond to the names of ematList, and will often be the same as studyColname, but doesn't have to be.

timeColname	Name of column in sampleMetdata that contains the values of the periodic variable.
nKnots	Number of internal knots to use for the periodic smoothing spline.
nTime	Number of time-points by which to discretize the time-dependent behavior of each feature. Corresponds to the number of rows in the matrix for which the SPCs will be calculated.
useSpc	Logical indicating whether to use PMA::SPC() (default) or base::svd().
sumabsv	L1-constraint on the SPCs, passed to PMA::SPC().
orth	Logical indicating whether to require left singular vectors be orthogonal to each other, passed to PMA::SPC().
nSpc	Vector of the number of SPCs to use for prediction. If NA (default), nSpc will become 1:K, where K is the number of SPCs in spcResult. Each value in nSpc will correspond to one prediction for each test observation. A value of 2 means that the prediction will be based on the first 2 SPCs.
timeRange	Vector of values of the periodic variable at which to calculate likelihood. The time with the highest likelihood is used as the initial value for the MLE optimizer.
covariateName	Name of column(s) in sampleMetadata containing information about other covariates for sva::ComBat(), besides batchColname. If NA (default), then there are no other covariates.
featuresExclude	Named list of character vectors corresponding to features to exclude from being used for prediction for the respective test datasets.
dopar	Logical indicating whether to process the folds in parallel. Use doParallel::registerDoParallel() to register the parallel backend.

Value

spcResultList	List of output from zeitzeigerSpc(), one for each test dataset.
timeDepLike	3-D array of likelihood, with dimensions for each test observation (across all datasets), each element of nSpc, and each element of timeRange.
mleFit	List (for each element in nSpc) of lists (for each test observation) of mle2 objects.
timePred	Matrix of predicted times for test observations by values of nSpc.

See Also

zeitzeiger(), sva::ComBat()

zeitzeigerEnsembleLikelihood

Combine predictions into an ensemble using the log-likelihood

Description

Make predictions by finding the maximum of the sum of the log-likelihoods.

Usage

```
zeitzeigerEnsembleLikelihood(timeDepLike, timeRange)
```

Arguments

timeDepLike	List or 3-D array of time-dependent likelihood from zeitzeigerPredict(). If a list, then each element (for each member of the ensemble) should be a matrix in which rows correspond to observations and columns correspond to time-points. If a 3-D array, the three dimensions should correspond to observations, time-points, and members of the ensemble.
timeRange	Vector of time-points at which the likelihood was calculated.

Value

timeDepLike	Matrix of likelihood for observations by time-points.
timePred	Vector of predicted times. Each predicted time will be an element of timeRange.

See Also

zeitzeigerPredict(), zeitzeigerEnsembleMean()

zeitzeigerEnsembleMean

Combine predictions into an ensemble using the circular mean

Description

Make predictions by calculating the circular mean of the predictions across members of the ensemble.

Usage

```
zeitzeigerEnsembleMean(timePredInput, timeMax = 1, naRm = TRUE)
```

Arguments

timePredInput	Matrix of predicted times in which rows correspond to observations and columns correspond to members of the ensemble.
timeMax	Maximum value of the periodic variable, i.e., the value that is equivalent to zero.
naRm	Logical indicating whether NA values should be removed from the calculation.

Value

Matrix with a row for each observation and columns for the predicted time and the normalized magnitude of the circular mean. The latter can range from 0 to 1, with 1 indicating perfect agreement among members of the ensemble.

See Also

zeitzeigerPredict(), zeitzeigerEnsembleLikelihood()

---

zeitzeigerFit *Fit a periodic spline for each feature*

Description

Fit a periodic smoothing spline to the measurements for each feature as a function of the periodic variable.

Usage

```
zeitzeigerFit(x, time, nKnots = 3)
```

Arguments

x	Matrix of measurements, with observations in rows and features in columns. Missing values are allowed.
time	Vector of values of the periodic variable for the observations, where 0 corresponds to the lowest possible value and 1 corresponds to the highest possible value.
nKnots	Number of internal knots to use for the periodic smoothing spline.

Value

xFitMean	Matrix of coefficients, where rows correspond to features and columns correspond to variables in the fit.
xFitResid	Vector of root mean square of residuals, same length as x.

See Also

zeitzeigerSpc(), zeitzeigerPredict()

zeitzeigerFitCv *Fit a periodic spline for each feature on cross-validation*

Description

Fit a periodic spline for each feature for each fold of cross-validation.

Usage

```
zeitzeigerFitCv(x, time, foldid, nKnots = 3)
```

Arguments

x	Matrix of measurements, with observations in rows and features in columns.
time	Vector of values of the periodic variable for the observations, where 0 corresponds to the lowest possible value and 1 corresponds to the highest possible value.
foldid	Vector of values indicating the fold to which each observation belongs.
nKnots	Number of internal knots to use for the periodic smoothing spline.

Value

A list consisting of the result from `zeitzeigerFit()` for each fold.

See Also

`zeitzeigerFit()`, `zeitzeigerSpcCv()`, `zeitzeigerPredictCv()`

zeitzeigerPredict *Predict corresponding time for test observations*

Description

Predict the value of the periodic variable for test observations given training data and SPCs.

Usage

```
zeitzeigerPredict(
  xTrain,
  timeTrain,
  xTest,
  spcResult,
  nKnots = 3,
  nSpc = NA,
  timeRange = seq(0, 1 - 0.01, 0.01)
)
```

Arguments

xTrain	Matrix of measurements for training data, observations in rows and features in columns.
timeTrain	Vector of values of the periodic variable for training observations, where 0 corresponds to the lowest possible value and 1 corresponds to the highest possible value.
xTest	Matrix of measurements for test data, observations in rows and features in columns.
spcResult	Output of zeitzeigerSpc().
nKnots	Number of internal knots to use for the periodic smoothing spline.
nSpc	Vector of the number of SPCs to use for prediction. If NA (default), nSpc will become 1:K, where K is the number of SPCs in spcResult. Each value in nSpc will correspond to one prediction for each test observation. A value of 2 means that the prediction will be based on the first 2 SPCs.
timeRange	Vector of values of the periodic variable at which to calculate likelihood. The time with the highest likelihood is used as the initial value for the MLE optimizer.

Value

timeDepLike	3-D array of likelihood, with dimensions for each test observation, each element of nSpc, and each element of timeRange.
mleFit	List (for each element in nSpc) of lists (for each test observation) of mle2 objects.
timePred	Matrix of predicted times for test observations by values of nSpc.

See Also

zeitzeigerFit(), zeitzeigerSpc()

---

zeitzeigerPredictCv *Predict corresponding time for observations on cross-validation*

Description

Make predictions for each observation for each fold of cross-validation.

Usage

```
zeitzeigerPredictCv(
  x,
  time,
  foldid,
  spcResultList,
  nKnots = 3,
```

```
  nSpc = NA,
  timeRange = seq(0, 1 - 0.01, 0.01),
  dopar = TRUE
)
```

Arguments

x	Matrix of measurements, observations in rows and features in columns.
time	Vector of values of the periodic variable for observations, where 0 corresponds to the lowest possible value and 1 corresponds to the highest possible value.
foldid	Vector of values indicating the fold to which each observation belongs.
spcResultList	Output of zeitzeigerSpcCv().
nKnots	Number of internal knots to use for the periodic smoothing spline.
nSpc	Vector of the number of SPCs to use for prediction. If NA (default), nSpc will become 1:K, where K is the number of SPCs in spcResult. Each value in nSpc will correspond to one prediction for each test observation. A value of 2 means that the prediction will be based on the first 2 SPCs.
timeRange	Vector of values of the periodic variable at which to calculate likelihood. The time with the highest likelihood is used as the initial value for the MLE optimizer.
dopar	Logical indicating whether to process the folds in parallel. Use doParallel::registerDoParallel() to register the parallel backend.

Value

A list of the same structure as zeitzeigerPredict(), combining the results from each fold of cross-validation.

timeDepLike	3-D array of likelihood, with dimensions for each observation, each element of nSpc, and each element of timeRange.
mleFit	List (for each element in nSpc) of lists (for each observation) of mle2 objects.
timePred	Matrix of predicted times for observations by values of nSpc.

See Also

zeitzeigerPredict(), zeitzeigerFitCv(), zeitzeigerSpcCv()

---

zeitzeigerPredictGroup

Predict corresponding time for groups of test observations

Description

Predict the value of the periodic variable for each group of test observations, where the amount of time between each observation in a group is known.

Usage

```
zeitzeigerPredictGroup(
  xTrain,
  timeTrain,
  xTest,
  groupTest,
  spcResult,
  nKnots = 3,
  nSpc = NA,
  timeRange = seq(0, 1 - 0.01, 0.01)
)
```

Arguments

xTrain	Matrix of measurements for training data, observations in rows and features in columns.
timeTrain	Vector of values of the periodic variable for training observations, where 0 corresponds to the lowest possible value and 1 corresponds to the highest possible value.
xTest	Matrix of measurements for test data, observations in rows and features in columns.
groupTest	data.frame with one row per observation in xTest, and columns for group and timeDiff. Observations in the same group should have the same value of group. Within each group, the value of timeDiff should correspond to the amount of time between that observation and a reference time. Typically, timeDiff will equal zero for one observation per group.
spcResult	Output of zeitzeigerSpc().
nKnots	Number of internal knots to use for the periodic smoothing spline.
nSpc	Vector of the number of SPCs to use for prediction. If NA (default), nSpc will become 1:K, where K is the number of SPCs in spcResult. Each value in nSpc will correspond to one prediction for each test observation. A value of 2 means that the prediction will be based on the first 2 SPCs.
timeRange	Vector of values of the periodic variable at which to calculate likelihood. The time with the highest likelihood is used as the initial value for the MLE optimizer.

Value

A list with the following elements, where the groups will be sorted by their names.

timeDepLike	3-D array of likelihood, with dimensions for each group of test observations, each element of nSpc, and each element of timeRange.
mleFit	List (for each element in nSpc) of lists (for each group of test observations) of mle2 objects.
timePred	Matrix of predicted times for each group of test observations by values of nSpc.

See Also

zeitzeigerPredict()

---

zeitzeigerPredictGroupCv *Predict corresponding time for groups of observations on cross-validation*

Description

Predict corresponding time for each group of observations in cross-validation. Thus, each fold is equivalent to a group.

Usage

```
zeitzeigerPredictGroupCv(
  x,
  time,
  foldid,
  spcResultList,
  nKnots = 3,
  nSpc = NA,
  timeRange = seq(0, 1 - 0.01, 0.01),
  dopar = TRUE
)
```

Arguments

x	Matrix of measurements, observations in rows and features in columns.
time	Vector of values of the periodic variable for observations, where 0 corresponds to the lowest possible value and 1 corresponds to the highest possible value.
foldid	Vector of values indicating the fold to which each observation belongs.
spcResultList	Result from zeitzeigerSpcCv().
nKnots	Number of internal knots to use for the periodic smoothing spline.
nSpc	Vector of the number of SPCs to use for prediction. If NA (default), nSpc will become 1:K, where K is the number of SPCs in spcResult. Each value in nSpc will correspond to one prediction for each test observation. A value of 2 means that the prediction will be based on the first 2 SPCs.
timeRange	Vector of values of the periodic variable at which to calculate likelihood. The time with the highest likelihood is used as the initial value for the MLE optimizer.
dopar	Logical indicating whether to process the folds in parallel. Use doParallel::registerDoParallel() to register the parallel backend.

Value

A list of the same structure as `zeitzeigerPredictGroup`, combining the results from each fold of cross-validation. Folds (i.e, groups) will be sorted by foldid.

`timeDepLike`	3-D array of likelihood, with dimensions for each fold, each element of `nSpc`, and each element of `timeRange`.
`mleFit`	List (for each element in `nSpc`) of lists (for each fold) of `mle2` objects.
`timePred`	Matrix of predicted times for folds by values of `nSpc`.

See Also

`zeitzeigerFitCv()`, `zeitzeigerSpcCv()`, `zeitzeigerPredictGroup()`

`zeitzeigerSpc` *Calculate sparse principal components of time-dependent variation*

Description

Calculate the SPCs given the time-dependent means and the residuals from `zeitzeigerFit()`.

Usage

```
zeitzeigerSpc(
  xFitMean,
  xFitResid,
  nTime = 10,
  useSpc = TRUE,
  sumabsv = 1,
  orth = TRUE,
  ...
)
```

Arguments

`xFitMean`	List of bigsplines, length is number of features.
`xFitResid`	Matrix of residuals, dimensions are observations by features.
`nTime`	Number of time-points by which to discretize the time-dependent behavior of each feature. Corresponds to the number of rows in the matrix for which the SPCs will be calculated.
`useSpc`	Logical indicating whether to use `PMA::SPC()` (default) or `base::svd()`.
`sumabsv`	L1-constraint on the SPCs, passed to `PMA::SPC()`.
`orth`	Logical indicating whether to require left singular vectors be orthogonal to each other, passed to `PMA::SPC()`.
`...`	Other arguments passed to `PMA::SPC()`.

Value

Output of `PMA::SPC()`, unless `useSpc` is `FALSE`, then output of `base::svd()`.

See Also

`zeitzeigerFit()`, `zeitzeigerPredict()`

`zeitzeigerSpcCv` *Calculate sparse principal components of time-dependent variation on cross-validation*

Description

Calculate SPCs for each fold of cross-validation.

Usage

```
zeitzeigerSpcCv(
  fitResultList,
  nTime = 10,
  useSpc = TRUE,
  sumabsv = 1,
  orth = TRUE,
  dopar = TRUE
)
```

Arguments

`fitResultList`	Output of `zeitzeigerFitCv()`.
`nTime`	Number of time-points by which to discretize the time-dependent behavior of each feature. Corresponds to the number of rows in the matrix for which the SPCs will be calculated.
`useSpc`	Logical indicating whether to use `SPC` (default) or `svd`.
`sumabsv`	L1-constraint on the SPCs, passed to `SPC`.
`orth`	Logical indicating whether to require left singular vectors be orthogonal to each other, passed to `SPC`.
`dopar`	Logical indicating whether to process the folds in parallel. Use `doParallel::registerDoParallel()` to register the parallel backend.

Value

A list consisting of the result from `zeitzeigerSpc()` for each fold.

See Also

`zeitzeigerSpc()`, `zeitzeigerFitCv()`, `zeitzeigerPredictCv()`

Index

base::svd(), *4*, *6*, *14, 15*

doParallel::registerDoParallel(), *6*, *11*, *13*, *15*

getCircDiff, 2

PMA::SPC(), *4*, *6*, *14, 15*
predictIntensity, 3

sva::ComBat(), *5, 6*

zeitzeiger, 3
zeitzeiger(), *5, 6*
zeitzeigerBatch, 5
zeitzeigerEnsembleLikelihood, 7
zeitzeigerEnsembleLikelihood(), *8*
zeitzeigerEnsembleMean, 7
zeitzeigerEnsembleMean(), *7*
zeitzeigerFit, 8
zeitzeigerFit(), *3–5*, *9, 10*, *14, 15*
zeitzeigerFitCv, 9
zeitzeigerFitCv(), *11*, *14, 15*
zeitzeigerPredict, 9
zeitzeigerPredict(), *4, 5*, *7, 8*, *11*, *13*, *15*
zeitzeigerPredictCv, 10
zeitzeigerPredictCv(), *9*, *15*
zeitzeigerPredictGroup, 11
zeitzeigerPredictGroup(), *14*
zeitzeigerPredictGroupCv, 13
zeitzeigerSpc, 14
zeitzeigerSpc(), *4–6*, *8*, *10*, *12*, *15*
zeitzeigerSpcCv, 15
zeitzeigerSpcCv(), *9*, *11*, *13, 14*