Package ‘degradr’

May 8, 2026

Type Package

Title Estimating Remaining Useful Life with Linear Mixed Effects Models

Description Provides tools for estimating the Remaining Useful Life (RUL) of degrading systems using linear mixed-effects models and creating a health index. It supports both univariate and multivariate degradation signals. For multivariate inputs, the signals are merged into a univariate health index prior to modeling. Linear and exponential degradation trajectories are supported (the latter using a log transformation). Remaining Useful Life (RUL) distributions are estimated using Bayesian updating for new units, enabling on-site predictive maintenance. Based on the methodology of Liu and Huang (2016) <doi:10.1109/TASE.2014.2349733>.

Version 1.0.2

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.1

Imports dplyr, nlme, quadprog, ggplot2

Maintainer Pedro Abraham Montoya Calzada <pedroabraham.montoya@gmail.com>

Depends R (>= 4.5)

LazyData true

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Pedro Abraham Montoya Calzada [aut, cre, cph] (ORCID: <https://orcid.org/0009-0002-3497-210X>),
Rogelio Salinas Gutiérrez [aut, cph] (ORCID: <https://orcid.org/0000-0002-1669-4460>),
Silvia Rodríguez-Narciso [aut, cph] (ORCID: <https://orcid.org/0000-0001-5429-5914>),
Netzahualcóyotl Castañeda-Leyva [aut, cph] (ORCID: <https://orcid.org/0000-0001-9414-3923>)

Repository CRAN

Date/Publication 2026-03-26 07:00:02 UTC

Contents

`compute_healthindex` *Constructing the Health Index on New Data Using Trained Weights*

Description

Given a fitted "`healthindex`" object, this function constructs the univariate health index for new multivariate sensor data by applying the stored projection (weights and offsets).

Usage

```
compute_healthindex(model, data)
```

Arguments

`model`	An object of class "`healthindex`" returned by `fit_healthindex`.
`data`	A data frame with new sensor readings over time. Must include the columns `t` (time) and `unit` (unit identifier), plus the same set of degradation signals used at training. All degradation signals must follow an upward trend.

Details

This function applies the projection learned in the first stage of `fit_healthindex` to new data:

Value

A data frame (tibble) with the columns:

`unit`	Unit identifier.
`t`	Time index.
`x`	Constructed health index at each `(unit, t)`.

See Also

fit_healthindex for learning the health index and mixed-effects model, predict_rul for RUL prediction based on the fitted model.

Examples

```
library(degradr)
library(dplyr)
# Load example data
data(train_FD001)
data(test_FD001)
data <- train_FD001 %>%
  select(unit,t,T24,T50,P30,
         Nf,Ps30,phi, NRf,
         BPR,htBleed,
         W31, W32) %>%
  mutate(across(c(P30,phi,W31,W32), ~ . * -1))

test <- test_FD001 %>%
  select(unit,t,T24,T50,P30,
         Nf,Ps30,phi, NRf,
         BPR,htBleed,
         W31, W32) %>%
  mutate(across(c(P30,phi,W31,W32), ~ . * -1))

# Fit a health index model (exponential trajectory of degree 2)
model <- fit_healthindex(data = data, type = "exponential",
                         degree = 2, r = 0.8)

# Construct the health index on new data using stored weights/offsets
hi_new <- compute_healthindex(model = model, data = test)
head(hi_new)
```

---

filter_test — *Test data for filter degradation and RUL prediction*

Description

Test data from a real-world degradation process involving the clogging of gas filters. The dataset includes right-censored lifetimes due to a preventive maintenance policy. Failure is defined when pressure differential exceeds 600 Pa.

Usage

```
data(filter_test)
```

Format

A data frame with 50 degradation trajectories. Variables include:

Time Time

Differential_pressure Differential pressure across the filter in Pascal (Pa).

Data_No Unique identifier for each filter test unit. Integer from 1 to 50.

RUL Remaining Useful Life in hours. Represents the time left until failure (threshold = 600 Pa).

Source

Adapted from:
Hagmeyer, S., Mauthe, F., & Zeiler, P. (2021). *Creation of Publicly Available Data Sets for Prognostics and Diagnostics Addressing Data Scenarios Relevant to Industrial Applications*. International Journal of Prognostics and Health Management, 12(2). doi:10.36001/ijphm.2021.v12i2.3087

References

See full description in the dataset repository on Kaggle by "Prognostics @ HSE".

`filter_train` *Training Data for Filter Degradation and RUL Prediction*

Description

Training data from a real-world degradation process involving the clogging of gas filters. The dataset includes right-censored lifetimes due to a preventive maintenance policy. Failure is defined when pressure differential exceeds 600 Pa.

Usage

```
data(filter_train)
```

Format

A data frame with 49 degradation trajectories. Variables include:

Time Time

Differential_pressure Differential pressure across the filter in Pascal (Pa).

Data_No Unique identifier for each filter test unit. Integer from 1 to 50.

Source

Adapted from:
Hagmeyer, S., Mauthe, F., & Zeiler, P. (2021). *Creation of Publicly Available Data Sets for Prognostics and Diagnostics Addressing Data Scenarios Relevant to Industrial Applications*. International Journal of Prognostics and Health Management, 12(2). doi:10.36001/ijphm.2021.v12i2.3087

References

See full description in the dataset repository on Kaggle by "Prognostics @ HSE".

`fit_healthindex` *Fitting a Health Index Model from Multivariate Signals*

Description

Fits a health index–based degradation model by projecting multivariate sensor signals into a univariate health index and modeling its evolution using a linear mixed-effects model.

Usage

```
fit_healthindex(data,
  type = "exponential",
  method = "lm",
  degree = 2,
  phi = NULL,
  r = 0.5)
```

Arguments

`data`	A data frame containing sensor readings over time. Must include the columns `t` (time), `unit` (unit identifier), and multiple degradation signals. All degradation signals must have an upward trend.
`type`	Model type. Either `"linear"` or `"exponential"`. The exponential model applies a logarithmic transformation to `x - phi`. Default is `"exponential"`.
`method`	Estimation method. Either `"nlme"` to fit a nonlinear mixed-effects model using `nlme::lme()`, or `"lm"` to fit separate linear models per unit and estimate fixed and random effects from the set of coefficients. Default is `"lm"`.
`degree`	Degree of the polynomial model. Default is `2`. The fixed and random effects will include powers of time up to the specified degree.
`phi`	Initial degradation level for non-defective units. Used in the exponential model as a fixed offset to ensure that the logarithmic transformation is valid and interpretable. If `NULL`, it is automatically estimated as a value slightly below the minimum observed degradation level. Ignored when `type = "linear"`.
`r`	parameter that controls the relative importance of the threshold variance and the weighted residual sum of squares in the index-fitted degradation model.

Details

This function implements a two-stage modeling strategy. In the first stage, a univariate health index is constructed as a weighted linear combination of the input signals, using correlation-based shrinkage. In the second stage, the resulting health index is modeled over time with a linear mixed-effects model (on the log scale for exponential models).

The exponential model uses a log transformation of `x - phi`, where `phi` ensures positivity and interpretability. The `phi` parameter can be supplied by the user or estimated automatically.

The resulting object stores both the projection (health index definition) and the fitted model used for RUL prediction.

Value

Returns an object of class `"healthindex"`, which contains:

`index`	A list with components: weights `w`, offset `phi`, and raw projections.
`model`	A fitted mixed-effects model of the health index over time.

References

Liu, K. and Huang, S. (2016). *Integration of Data Fusion Methodology and Degradation Modeling Process to Improve Prognostics*. IEEE Transactions on Automation Science and Engineering, 13(1), 344–354.doi:10.1109/TASE.2014.2349733

Examples

```
library(degradr)
library(dplyr)
# Load example data
data(train_FD001)
data(test_FD001)
data <- train_FD001 %>%
  select(unit,t,T24,T50,P30,
         Nf,Ps30,phi, NRf,
         BPR,htBleed,
         W31, W32) %>%
  mutate(across(c(P30,phi,W31,W32), ~ . * -1))

test <- test_FD001 %>%
  select(unit,t,T24,T50,P30,
         Nf,Ps30,phi, NRf,
         BPR,htBleed,
         W31, W32) %>%
  mutate(across(c(P30,phi,W31,W32), ~ . * -1))

# Fit a health index model (exponential trajectory of degree 2)
model <- fit_healthindex(data = data, type = "exponential",
                         degree = 2, r = 0.8)
rul <- predict_rul(data = test, model = model)
head(rul)
```

`fit_model` *Fitting the Linear Mixed Effects Model*

Description

Fits a linear or exponential mixed-effects model of degree \(p\) for the degradation process.

Usage

```
fit_model(data,
  type = "exponential",
  method = "lm",
  degree = 2,
  phi = NULL)
```

Arguments

`data`	A data frame with three columns: `t` for time, `x` for the degradation signal, and `unit` as the unit identifier. At least two distinct units are required.
`type`	Model type. Either `"linear"` or `"exponential"`. The exponential model applies a logarithmic transformation to `x - phi`. Default is `"exponential"`.
`method`	Estimation method. Either `"nlme"` to fit a nonlinear mixed-effects model using `nlme::lme()`, or `"lm"` to fit separate linear models per unit and estimate fixed and random effects from the set of coefficients. Default is `"lm"`.
`degree`	Degree of the polynomial model. Default is `2`. The fixed and random effects will include powers of time up to the specified degree.
`phi`	Initial degradation level for non-defective units. Used in the exponential model as a fixed offset to ensure that the logarithmic transformation is valid and interpretable. If `NULL`, it is automatically estimated as a value slightly below the minimum observed degradation level. Ignored when `type = "linear"`.

Details

This function fits a linear or exponential polynomial mixed-effects model of degree `p` to degradation data collected over time from multiple units. The model captures both fixed effects (population-level degradation trends) and random effects (unit-specific deviations).

The exponential model applies a logarithmic transformation with an offset parameter `phi`. The offset `phi` can be provided or automatically estimated from the data.

At least two distinct units are required to estimate random effects. The degree parameter controls the polynomial order for the time terms in both fixed and random effects.

Value

Returns a list with the estimated model and prior distributions.

References

Liu, K. and Huang, S. (2016). *Integration of Data Fusion Methodology and Degradation Modeling Process to Improve Prognostics*. IEEE Transactions on Automation Science and Engineering, 13(1), 344–354.doi:10.1109/TASE.2014.2349733

Examples

```
library(degradr)

# Load example data sets
data(filter_train)
data(filter_test)

# Show the original column names
colnames(filter_train)

# Rename the columns to match the expected format: t, x, unit
colnames(filter_train) <- c("t", "x", "unit")
colnames(filter_test)  <- c("t", "x", "unit", "RUL")

# Plot the training set
plot_degradr(data = filter_train, D = 600)

# Fit an exponential mixed-effects model of degree 1
model <- fit_model(data = filter_train, type = "exponential", degree = 1)

# Predict the remaining useful life (RUL) for the test units,
# assuming a fixed failure threshold D = 600
predict_rul(data = filter_test, model = model, D = 600)
```

`plot_degradr` *Plot Degradation Trajectories for Multiple Units*

Description

Generates a line plot of degradation signals over time for each unit, optionally overlaying a failure threshold line. This function is useful for visualizing degradation paths across multiple components or systems.

Usage

```
plot_degradr(data, D = NULL)
```

Arguments

`data`	A data frame with three columns: `t` for time, `x` for the degradation signal, and `unit` as the unit identifier.
`D`	Optional numeric value indicating the failure threshold.

Details

The function is designed to work with degradation datasets where each row represents an observation of a unit at a particular time. The plot shows how the degradation variable x evolves over time t for each unit. This is especially useful for visual inspection before model fitting or threshold analysis.

Value

Returns a ggplot object that can be further customized or directly printed.

Examples

```
library(degradr)

# Load example data sets
data(filter_train)
data(filter_test)

# Show the original column names
colnames(filter_train)

# Rename the columns to match the expected format: t, x, unit
colnames(filter_train) <- c("t", "x", "unit")
plot_degradr(data = filter_train, D = 600)
```

---

predict_rul — *Predicting Remaining Useful Life (RUL) from Degradation Signals*

Description

Estimates the Remaining Useful Life (RUL) for one or more partially observed degradation signals based on a previously fitted linear or exponential mixed-effects model.

Usage

```
predict_rul(data, model, D = NULL, upper = NULL)
```

Arguments

data	A data frame with columns t (time), x (degradation measurement), and unit (unit identifier) or a data frame containing sensor readings over time: must include the columns t (time), unit (unit identifier), and multiple degradation signals. Multiple units can be passed simultaneously.
model	An object of class "degradation_model" produced by the fit_model function.
D	(Optional) Critical degradation threshold. If provided, it will be used to compute the RUL via a fixed-threshold model. If NULL (default), a random-threshold model will be used based on training data statistics.

upper — Optional upper bound for the search interval when solving for the quantiles of the RUL distribution. If NULL, the function will use the maximum observed time in the training data.

Details

This function applies Bayesian updating to compute the posterior distribution of the degradation model parameters for each unit, conditional on its observed signal. Then, it computes the Remaining Life Distribution (RLD) and returns the estimated Remaining Useful Life.

It supports both linear and exponential degradation models, matching the formulation used in fit_model. The posterior updating follows the methodology of Liu and Huang (2016).

Value

A data frame with one row per unit and the following columns:

unit Unit identifier.

RUL Estimated RUL.

References

Liu, K. and Huang, S. (2016). *Integration of Data Fusion Methodology and Degradation Modeling Process to Improve Prognostics*. IEEE Transactions on Automation Science and Engineering, 13(1), 344–354. doi:10.1109/TASE.2014.2349733

Examples

```
library(degradr)

# Load example data sets
data(filter_train)
data(filter_test)

# Show the original column names
colnames(filter_train)

# Rename the columns to match the expected format: t, x, unit
colnames(filter_train) <- c("t", "x", "unit")
colnames(filter_test)  <- c("t", "x", "unit", "RUL")

# Plot the training set
plot_degradr(data = filter_train, D = 600)

# Fit an exponential mixed-effects model of degree 1
model <- fit_model(data = filter_train, type = "exponential", degree = 1)

# Predict the remaining useful life (RUL) for the test units,
# assuming a fixed failure threshold D = 600
predict_rul(data = filter_test, model = model, D = 600)
```

prul — *Probability of Remaining Useful Life (RUL) Falling Within a Time Horizon*

Description

Evaluates the cumulative probability that the Remaining Useful Life (RUL) of a unit is less than or equal to a specified time `t`. The computation is based on a fitted degradation model and the observed degradation signal for the unit.

Usage

```
prul(t, data, model, D = NULL)
```

Arguments

`t`	Time at which to evaluate the RUL cumulative distribution function.
`data`	A data frame with columns `t` (time), `x` (degradation measurement), and `unit` (unit identifier) or a data frame containing sensor readings over time: must include the columns `t` (time), `unit` (unit identifier), and multiple degradation signals. Multiple units can be passed simultaneously.
`model`	An object of class `"degradation_model"` produced by `fit_model` or a `"healthindex"` object returned by `fit_healthindex`.
`D`	Optional critical degradation threshold. If provided, a fixed-threshold model is used; otherwise a random-threshold model is assumed. For exponential models the threshold is automatically transformed to the log scale.

Details

For a fixed threshold model (`D` supplied), the function computes the Remaining Life Distribution (RLD) using the specified failure threshold. If `D` is `NULL`, the distribution is computed under a random-threshold formulation based on the training data.

Value

Numeric value between 0 and 1 giving $P(\mathrm{RUL} \leq t)$.

See Also

`qrul`, `predict_rul`

Examples

```
library(degradr)
library(dplyr)
# Load example data
data(train_FD001)
```

```
data(test_FD001)
data <- train_FD001 %>%
  select(unit,t,T24,T50,P30,
         Nf,Ps30,phi, NRf,
         BPR,htBleed,
         W31, W32) %>%
  mutate(across(c(P30,phi,W31,W32), ~ . * -1))

test <- test_FD001 %>%
  select(unit,t,T24,T50,P30,
         Nf,Ps30,phi, NRf,
         BPR,htBleed,
         W31, W32) %>%
  mutate(across(c(P30,phi,W31,W32), ~ . * -1))

# Fit a health index model (exponential trajectory of degree 2)
model <- fit_healthindex(data = data, type = "exponential",
                         degree = 2, r = 0.8)
#Probability that the run length will be less than or equal to 86 cycles
head(prul(t = 86, data = test, model = model))
```

`qrul` *Quantiles of the Remaining Useful Life (RUL) Distribution*

Description

Returns quantiles of the Remaining Life Distribution for one or more units based on their observed degradation signals and a fitted model.

Usage

```
qrul(prob = 0.05, data, model, D = NULL, upper = NULL)
```

Arguments

`prob`	Probability at which to evaluate the quantile of the RUL distribution.
`data`	A data frame with columns `t` (time), `x` (degradation measurement), and `unit` (unit identifier) or a data frame containing sensor readings over time: must include the columns `t` (time), `unit` (unit identifier), and multiple degradation signals. Multiple units can be passed simultaneously.
`model`	An object of class `"degradation_model"` returned by `fit_model` or a `"healthindex"` object from `fit_healthindex`.
`D`	Optional critical degradation threshold. If `NULL`, a random-threshold model is used; otherwise the provided threshold is used. For exponential models the threshold is internally log-transformed.
`upper`	Optional upper bound for the numerical search when computing quantiles. If `NULL`, the maximum time observed during training is used.

Details

For each unit in `data` the function computes the posterior distribution of the model parameters and evaluates the specified quantile of the Remaining Life Distribution. Units that result in computational errors return `NA`.

Value

A data frame with one row per unit containing:

`unit`	Unit identifier.
`RUL`	The requested quantile of the RUL distribution.

See Also

`prul`, `predict_rul`

Examples

```
library(degradr)
library(dplyr)
# Load example data
data(train_FD001)
data(test_FD001)
data <- train_FD001 %>%
  select(unit,t,T24,T50,P30,
         Nf,Ps30,phi, NRf,
         BPR,htBleed,
         W31, W32) %>%
  mutate(across(c(P30,phi,W31,W32), ~ . * -1))

test <- test_FD001 %>%
  select(unit,t,T24,T50,P30,
         Nf,Ps30,phi, NRf,
         BPR,htBleed,
         W31, W32) %>%
  mutate(across(c(P30,phi,W31,W32), ~ . * -1))

# Fit a health index model (exponential trajectory of degree 2)
model <- fit_healthindex(data = data, type = "exponential",
                         degree = 2, r = 0.8)
head(qrul(prob = 0.05, data = test, model = model))
```

`RUL_FD001` *Remaining Useful Life (RUL) Ground Truth for FD001 Dataset*

Description

Ground truth values of the Remaining Useful Life (RUL) for each engine unit in the FD001 subset of the C-MAPSS dataset. These values correspond to the last observed cycle in the test set and are used for evaluation purposes in prognostics models.

Usage

```
data("RUL_FD001")
```

Format

A data frame with 100 observations on the following variable:

RUL Numeric vector indicating the true Remaining Useful Life (in cycles) for each unit in the test set.

Details

This dataset is part of the C-MAPSS (Commercial Modular Aero-Propulsion System Simulation) benchmark and is used as ground truth for performance evaluation of predictive maintenance models, particularly for models estimating Remaining Useful Life (RUL) under the FD001 operating condition scenario.

References

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). *Damage propagation modeling for aircraft engine run-to-failure simulation*. In *2008 International Conference on Prognostics and Health Management* (pp. 1–9). IEEE. doi:10.1109/PHM.2008.4711414

Examples

```
data(RUL_FD001)
```

`test_FD001` *NASA Turbofan Engine Degradation Simulation Test Data (Subset FD001)*

Description

Truncated time series data from turbofan engine degradation simulations, generated using the C-MAPSS (Commercial Modular Aero-Propulsion System Simulation) model. This test dataset contains operational sensor data up to a point before failure, intended for validating prognostic algorithms that estimate Remaining Useful Life (RUL).

Format

A data frame with multiple observations (rows) on the following 24 variables (columns):

`unit` Engine unit number (identifier)

`t` Time in cycles

`T2` Total temperature at fan inlet (°R)

`T24` Total temperature at LPC outlet (°R)

`T30` Total temperature at HPC outlet (°R)

`T50` Total temperature at LPT outlet (°R)

`P2` Pressure at fan inlet (psia)

`P15` Total pressure in bypass-duct (psia)

`P30` Total pressure at HPC outlet (psia)

`Nf` Physical fan speed (rpm)

`Nc` Physical core speed (rpm)

`epr` Engine pressure ratio (P50/P2)

`Ps30` Static pressure at HPC outlet (psia)

`phi` Ratio of fuel flow to Ps30 (pps/psi)

`NRf` Corrected fan speed (rpm)

`NRc` Corrected core speed (rpm)

`BPR` Bypass Ratio

`farB` Burner fuel-air ratio

`htBleed` Bleed Enthalpy

`Nf_dmd` Demanded fan speed (rpm)

`PCNfR_dmd` Demanded corrected fan speed (rpm)

`W31` HPT coolant bleed (lbm/s)

`W32` LPT coolant bleed (lbm/s)

Details

Key characteristics of this test dataset:

- Simulates progressive degradation in the High Pressure Compressor (HPC) module
- Time series are truncated prior to failure (true RUL values not included)
- Includes realistic measurement noise and unit-to-unit variability

References

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). *Damage propagation modeling for aircraft engine run-to-failure simulation*. In *2008 International Conference on Prognostics and Health Management* (pp. 1–9). IEEE. doi:10.1109/PHM.2008.4711414

Examples

```
data(test_FD001)
```

train_FD001 *NASA Turbofan Engine Degradation Simulation Data (FD001)*

Description

Run-to-failure simulation data for aircraft turbofan engines generated using C-MAPSS (Commercial Modular Aero-Propulsion System Simulation). This dataset represents engine degradation in the High Pressure Compressor (HPC) module under varying operational conditions.

Usage

```
data("train_FD001")
```

Format

A data frame with multiple observations (rows) on the following 24 variables (columns):

`unit` Engine unit number (identifier)

`t` Time in cycles

`T2` Total temperature at fan inlet (°R)

`T24` Total temperature at LPC outlet (°R)

`T30` Total temperature at HPC outlet (°R)

`T50` Total temperature at LPT outlet (°R)

`P2` Pressure at fan inlet (psia)

`P15` Total pressure in bypass-duct (psia)

`P30` Total pressure at HPC outlet (psia)

`Nf` Physical fan speed (rpm)

`Nc` Physical core speed (rpm)

`epr` Engine pressure ratio (P50/P2)

`Ps30` Static pressure at HPC outlet (psia)

`phi` Ratio of fuel flow to Ps30 (pps/psi)

`NRf` Corrected fan speed (rpm)

`NRc` Corrected core speed (rpm)

`BPR` Bypass Ratio

`farB` Burner fuel-air ratio

`htBleed` Bleed Enthalpy

`Nf_dmd` Demanded fan speed (rpm)

`PCNfR_dmd` Demanded corrected fan speed (rpm)

`W31` HPT coolant bleed (lbm/s)

`W32` LPT coolant bleed (lbm/s)

Details

The data was generated for the Prognostics and Health Management (PHM) 2008 data challenge. Each engine unit starts with some initial wear and progresses to failure as efficiency and flow parameters degrade exponentially. The failure criterion is when the health index (calculated from stall margins and EGT) reaches zero. The dataset includes sensor measurements taken at cruise conditions.

References

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). *Damage propagation modeling for aircraft engine run-to-failure simulation*. In *2008 International Conference on Prognostics and Health Management* (pp. 1–9). IEEE. doi:10.1109/PHM.2008.4711414

Examples

```
data(train_FD001)
```

Index