Package 'highMLR'

May 23, 2026

Title Machine Learning Feature Selection for High Dimensional Survival Data

Version 1.0.1

Date 2026-05-23

Description A unified, flexible framework for high dimensional feature selection in the presence of a survival outcome. Provides multiple machine learning approaches (Cox elastic net, random survival forest, accelerated oblique random survival forest, gradient-boosted Cox, stability selection, classical univariate Cox screening, pseudo-observation bridging to arbitrary regression learners, and Fine-Gray competing risks selection) under a single interface. Adds causal survival forest estimation of heterogeneous treatment effects on survival (experimental), conformal survival prediction with finite-sample coverage guarantees, and time-dependent 'SHAP' explanations via 'SurvSHAP(t)'. Methodology is based on regularised Cox regression (2011) <doi:10.18637/jss.v039.i05>, random survival forests (2008) <doi:10.1214/08-AOAS169>, oblique random survival forests (2024) <doi:10.1080/10618600.2023.2231048>, stability selection (2010) <doi:10.1111/j.1467-9868.2010.00740.x>, causal survival forests (2023) <doi:10.1111/rssb.12538>, time-dependent survival explanations (2023) <doi:10.1016/j.knosys.2022.110234>, conformal survival prediction (2023) <doi:10.1093/biomet/asad043>, the Fine-Gray model for competing risks (1999) <doi:10.1080/01621459.1999.10474144>, and pseudo-observation regression (2010) <doi:10.1177/0962280209105020>.

Depends R (>= 4.1.0)

Imports survival, glmnet, ranger, aorsf, xgboost, stabs, survex, grf, prodlim, cmprsk, future, future.apply, tibble, ggplot2, rlang, stats, utils

Suggests knitr, rmarkdown, testthat (>= 3.0.0), mice, riskRegression

License GPL-3

Encoding UTF-8

Language en-GB

LazyData true

LazyDataCompression xz

RoxygenNote 7.3.3

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Atanu Bhattacharjee [aut, cre]

Maintainer Atanu Bhattacharjee <atanustat@gmail.com>

Repository CRAN

Date/Publication 2026-05-23 12:30:02 UTC

Contents

`coef.highmlr_fit` *Coefficients from a highmlr_fit*

Description

Coefficients from a highmlr_fit

Usage

```
## S3 method for class 'highmlr_fit'
coef(object, ...)
```

Arguments

object	A ‘highmlr_fit‘ object.
...	Unused.

Value

A named numeric vector of coefficients (where defined) or importance scores otherwise.

---

highmlr *Machine learning feature selection for high dimensional survival data*

Description

Fits one of several survival ML methods and returns a unified ‘highmlr_fit‘ object summarising the selected features, their importance/coefficients, and (optionally) out-of-sample performance.

Usage

```
highmlr(
  data,
  time,
  status,
  features = NULL,
 method = c("coxnet", "rsf", "aorsf", "xgboost", "stability", "univariate", "pseudo",
    "finegray"),
  engine = NULL,
  recipe = NULL,
  resampling = c("cv", "bootstrap", "holdout", "none"),
  folds = 5L,
  tune = FALSE,
  top_n = 50L,
  parallel = FALSE,
  seed = NULL,
  ...
)
```

Arguments

data	A data frame containing ‘time‘, ‘status‘, and the candidate features (or a superset). Rows with missing time/status are dropped.
time	Character scalar: name of the survival time column.
status	Character scalar: name of the event indicator column. For right-censored methods: 1 = event, 0 = censored. For Fine-Gray (method = "finegray"): 0 = censored, 1 = event of interest, 2+ = competing event(s).
features	Character vector of candidate feature column names. If ‘NULL‘ (default), all columns except ‘time‘ and ‘status‘ are used.

method	One of ‘"coxnet"‘, ‘"rsf"‘, ‘"aorsf"‘, ‘"xgboost"‘, ‘"stability"‘, ‘"univariate"‘, ‘"pseudo"‘, ‘"finegray"‘.
engine	Optional engine override.
recipe	Optional preprocessing recipe object (currently accepted for forward compatibility; not yet applied).
resampling	One of ‘"cv"‘, ‘"bootstrap"‘, ‘"holdout"‘, ‘"none"‘.
folds	Integer, number of CV folds (default 5).
tune	Logical. Internal tuning (currently coxnet only).
top_n	Integer. For ranking-based methods, keep this many top features (default 50).
parallel	Logical. Use future-based parallelism for the embarrassingly parallel parts.
seed	Optional integer for reproducibility.
...	Additional arguments passed to the method-specific fitter.

Value

An object of class ‘highmlr_fit‘. See [new_highmlr_fit()].

Examples

```
if (requireNamespace("glmnet", quietly = TRUE)) {
  data(hnscc)
  fit <- highmlr(hnscc, time = "OS", status = "Death",
                 method = "coxnet", resampling = "cv", folds = 5)
  print(fit)
}
```

---

highmlr_causal *Causal survival forest for heterogeneous treatment effects (experimental)*

Description

Estimates patient-level conditional average treatment effects (CATEs) on a survival outcome using ‘grf::causal_survival_forest‘. Unlike the rest of ‘highMLR‘, this function answers a different question: not "which features predict survival?" but "for which patients does treatment T extend (or shorten) survival, and which features modify that effect?".

Usage

```
highmlr_causal(
  data,
  time,
  status,
  treatment,
  covariates = NULL,
  horizon = NULL,
  num.trees = 2000L,
  target = c("RMST", "survival.probability"),
  honesty = TRUE,
  seed = NULL,
  ...
)

## S3 method for class 'highmlr_causal'
print(x, n = 10, ...)

## S3 method for class 'highmlr_causal'
plot(x, ...)
```

Arguments

data	A data frame.
time	Character: name of the survival time column.
status	Character: name of the event indicator (0/1).
treatment	Character: name of the binary treatment column (0 = control, 1 = treated). Must be exactly two levels.
covariates	Character vector of covariate column names. If ‘NULL‘, all columns other than ‘time‘, ‘status‘, ‘treatment‘.
horizon	Numeric. The time horizon at which the treatment effect on the survival probability is estimated. Defaults to the median observed time.
num.trees	Number of trees in the forest (default 2000).
target	One of ‘"RMST"‘ (restricted mean survival time difference up to ‘horizon‘) or ‘"survival.probability"‘ (difference in survival probability at ‘horizon‘).
honesty	Logical (default TRUE) – honest splitting per ‘grf‘.
seed	Optional integer seed.
...	Passed to ‘grf::causal_survival_forest‘.
x	A ‘highmlr_causal‘ object.
n	Number of top covariates to print (default 10).

Value

An object of class ‘highmlr_causal‘ containing the fitted forest, per-patient CATE estimates with standard errors, and covariate importance.

‘print()‘ invisibly returns ‘x‘; ‘plot()‘ returns a ‘ggplot‘ object showing the distribution of estimated CATEs.

Experimental

This function is marked experimental. The signature, defaults, and return shape may change in a future release. Use with care in published analyses, and report the package version.

Examples

```
## Not run:
set.seed(1)
n <- 500; p <- 10
X <- matrix(rnorm(n*p), n, p); colnames(X) <- paste0("V", 1:p)
W <- rbinom(n, 1, 0.5)
t <- rexp(n, rate = exp(0.3*W + 0.5*X[,1]*W))
c <- rexp(n, rate = 0.05)
d <- data.frame(OS = pmin(t,c), Death = as.integer(t<=c),
                arm = W, X)
cf <- highmlr_causal(d, "OS", "Death", treatment = "arm",
                     covariates = paste0("V", 1:p))
print(cf); plot(cf)

## End(Not run)
```

`highmlr_compare` *Compare multiple highMLR methods on the same data*

Description

Runs several methods and returns a side-by-side comparison of selected features and performance.

Usage

```
highmlr_compare(
  data,
  time,
  status,
  features = NULL,
  methods = c("coxnet", "rsf", "univariate"),
  ...
)
```

Arguments

`data`, `time`, `status`, `features`	As in [highmlr()].
`methods`	Character vector of methods to compare.
`...`	Passed to each call to ‘highmlr()‘.

Value

A list with two elements: ‘fits‘ (named list of ‘highmlr_fit‘ objects) and ‘summary‘ (a tibble of method, n_selected, key metric).

Examples

```
## Not run:
data(hnscc)
cmp <- highmlr_compare(hnscc, "OS", "Death",
                       methods = c("coxnet", "rsf", "univariate"))
cmp$summary

## End(Not run)
```

---

highmlr_conformal *Conformal prediction intervals for survival times*

Description

Computes calibrated lower bounds on survival time for each new subject using a split-conformal procedure with inverse probability of censoring weights (Candes, Lei and Ren, 2023). The returned lower bound satisfies a marginal coverage guarantee approximately equal to one minus alpha under standard conformal assumptions and a consistent censoring model.

Usage

```
highmlr_conformal(
  fit,
  new_data,
  calibration_data = NULL,
  alpha = 0.1,
  calibration_split = 0.3,
  time = NULL,
  status = NULL,
  seed = NULL
)
```

Arguments

fit	A highmlr_fit object whose predict() method returns a linear predictor or risk score.
new_data	Data frame on which to compute prediction intervals.
calibration_data	Data frame on which to compute conformity scores. If NULL, a random calibration_split fraction of new_data is held out for calibration and the rest is used as the test set (split-conformal).

alpha	Miscoverage level; default 0.1 (so 90 percent coverage).
calibration_split	Fraction of new_data to use for calibration when calibration_data is NULL. Default 0.3.
time	Name of the survival time column in calibration data. Defaults to the column used in fit.
status	Name of the event column in calibration data.
seed	Optional integer seed for the split.

Value

An object of class highmlr_conformal containing per-subject point predictions and lower confidence bounds for survival time.

Examples

```
## Not run:
fit  <- highmlr(d_train, "OS", "Death", method = "coxnet")
intv <- highmlr_conformal(fit, new_data = d_test, alpha = 0.1)
print(intv)
plot(intv)

## End(Not run)
```

---

highmlr_explain *Time-dependent SHAP explanations for a highmlr_fit (SurvSHAP(t))*

Description

Computes SurvSHAP(t) attributions (Krzyzinski et al., 2023) – SHAP values that vary with follow-up time – for the top features in a fitted ‘highmlr_fit‘. Returns the survex explainer, per-feature aggregated importance, and a plotting helper.

Usage

```
highmlr_explain(
  fit,
  new_data = NULL,
  top_n = 10L,
  times = NULL,
  method = c("survshap", "permutation", "break_down"),
  n_explain = 25L,
  seed = NULL,
  ...
)
```

```
## S3 method for class 'highmlr_explain'
print(x, n = 10, ...)

## S3 method for class 'highmlr_explain'
plot(x, top_n = 10, ...)
```

Arguments

fit	A ‘highmlr_fit‘ object with a stored model.
new_data	Data on which to compute explanations.
top_n	Number of top features to explain (default 10).
times	Optional numeric vector of time points at which SHAP values are computed. Defaults to a 20-point grid spanning the observed time range.
method	SHAP method passed through to ‘survex‘. Default ‘"survshap"‘ (time-dependent). Other options: ‘"permutation"‘, ‘"break_down"‘.
n_explain	How many test rows to compute SHAP for. Default 25 (SHAP is expensive; full-cohort computation is rarely needed).
seed	Optional integer for reproducibility of subsampling.
...	Passed to ‘survex::model_survshap()‘ or ‘survex::explain_survival()‘.
x	A ‘highmlr_explain‘ object.
n	Number of top features to print (default 10).

Value

A list with class ‘highmlr_explain‘ containing: * ‘explainer‘ – the ‘survex‘ explainer object * ‘survshap‘ – the time-dependent SHAP object (if applicable) * ‘top_features‘ – the top features table from the fit * ‘aggregated‘ – tibble of mean absolute SHAP per feature, averaged across time and explained rows

---

highmlr_report *Generate a Quarto/Rmd report skeleton for a highmlr_fit*

Description

Writes a self-contained Rmd file that, when rendered, produces a standard biomarker report (selected features, hazard ratios where available, performance, forest plot).

Usage

```
highmlr_report(fit, file = "highmlr_report.Rmd", render = FALSE)
```

Arguments

fit	A ‘highmlr_fit‘ object.
file	Output ‘.Rmd‘ path (default ‘"highmlr_report.Rmd"‘).
render	Logical: if ‘TRUE‘, also render via ‘rmarkdown::render()‘.

Value

Invisibly, the path to the written file.

`highmlr_screen` *Pre-screen features when p is very large*

Description

Lightweight filter before the main pipeline (e.g. to drop features with low variance or low marginal association).

Usage

```
highmlr_screen(
  data,
  time,
  status,
  features = NULL,
  filter = c("variance", "univariate_p", "none"),
  keep = 1000L
)
```

Arguments

`data`, `time`, `status`, `features`	As in [highmlr()].
`filter`	One of '"variance"', '"univariate_p"', '"none"'.
`keep`	Integer, how many features to retain (default 1000).

Value

Character vector of retained feature names.

Examples

```
## Not run:
data(srdata)
keep <- highmlr_screen(srdata, "OS", "event",
                       filter = "variance", keep = 500)
fit <- highmlr(srdata, "OS", "event", features = keep, method = "coxnet")

## End(Not run)
```

highmlr_stability *Post-hoc stability analysis of a fitted highmlr_fit*

Description

Runs stability selection on the data used in ‘fit‘, returning a selection frequency per feature.

Usage

```
highmlr_stability(fit, B = 100L, cutoff = 0.75, PFER = 1, ...)
```

Arguments

fit	A ‘highmlr_fit‘ object (used only for the data / call).
B	Number of subsamples (default 100).
cutoff	Selection probability threshold (default 0.75).
PFER	Per-family error rate bound (default 1).
...	Passed to [fit_stability()].

Value

A new ‘highmlr_fit‘ with ‘method = "stability"‘.

hnscc *High dimensional head and neck cancer survival and gene expression data*

Description

Survival and gene expression measurements for head and neck squamous cell carcinoma patients, used to demonstrate high-dimensional feature selection.

Usage

```
hnscc
```

Format

A data frame with 565 rows (one per patient) and 104 columns. The first five columns are the identifier and outcome variables: `ID` (patient identifier), `Death` (overall survival event indicator, 1 = death, 0 = censored), `OS` (overall survival time), `PFS` (progression-free survival time), and `Prog` (progression event indicator, 1 = progression, 0 = none). The remaining 99 columns are numeric gene expression features named by gene symbol (for example `GJB1`, `HPN`, `PROM1`).

Source

Bundled with the package since highMLR v0.1.1.

plot.highmlr_conformal *Plot method for highmlr_conformal objects*

Description

Plot method for highmlr_conformal objects

Usage

```
## S3 method for class 'highmlr_conformal'
plot(x, ...)
```

Arguments

x	A highmlr_conformal object.
...	Unused.

Value

A ggplot object.

plot.highmlr_fit *Forest / importance plot for a highmlr_fit*

Description

Forest / importance plot for a highmlr_fit

Usage

```
## S3 method for class 'highmlr_fit'
plot(x, top_n = 20, ...)
```

Arguments

x	A ‘highmlr_fit‘ object.
top_n	Number of top features to plot (default 20).
...	Unused.

Value

A ‘ggplot‘ object.

predict.highmlr_fit *Predict from a highmlr_fit*

Description

Predict from a highmlr_fit

Usage

```
## S3 method for class 'highmlr_fit'
predict(object, new_data, type = c("linear_pred", "survival", "risk"), ...)
```

Arguments

object	A ‘highmlr_fit‘ object.
new_data	A data frame containing the features used in fitting.
type	One of ‘"linear_pred"‘, ‘"survival"‘, or ‘"risk"‘. Availability depends on the underlying model.
...	Passed to the underlying model’s predict method.

Value

Predicted values (vector or tibble depending on ‘type‘).

print.highmlr_conformal *Print method for highmlr_conformal objects*

Description

Print method for highmlr_conformal objects

Usage

```
## S3 method for class 'highmlr_conformal'
print(x, n = 10, ...)
```

Arguments

x	A highmlr_conformal object.
n	Number of rows to display in the preview table (default 10).
...	Unused.

Value

Invisibly returns x.

print.highmlr_fit *Print method for highmlr_fit*

Description

Print method for highmlr_fit

Usage

```
## S3 method for class 'highmlr_fit'
print(x, n = 10, ...)
```

Arguments

x	A ‘highmlr_fit‘ object.
n	Number of top features to display (default 10).
...	Unused.

Value

Invisibly returns ‘x‘.

srdata *High dimensional protein gene expression survival data*

Description

Protein expression measurements with a survival outcome, used to demonstrate high-dimensional feature selection.

Usage

```
srdata
```

Format

A data frame with 288 rows and 250 columns. The first four columns are the identifier and outcome variables: `ID` (subject identifier), `Visit` (visit number), `OS` (overall survival time), and `event` (survival event indicator, 1 = event, 0 = censored). The remaining 246 columns are numeric protein expression features named by protein or marker (for example `C6kine`, `ActivinA`, `Adiponectin`).

Source

Bundled with the package since highMLR v0.1.1.

summary.highmlr_fit *Summary method for highmlr_fit*

Description

Summary method for highmlr_fit

Usage

```
## S3 method for class 'highmlr_fit'
summary(object, ...)
```

Arguments

object	A 'highmlr_fit' object.
...	Unused.

Value

A list with the full selected feature table and performance.

Index