Package: riskscores (via r-universe)

October 28, 2024

Title Optimized Integer Risk Score Models

Version 1.1.1

Description Implements an optimized approach to learning risk score models, where sparsity and integer constraints are integrated into the model-fitting process.

URL https://github.com/hjeglinton/riskscores

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports dplyr, foreach, ggplot2, magrittr, stats

Suggests knitr, kableExtra, rmarkdown, doParallel

VignetteBuilder knitr, kableExtra

Depends R (>= 2.10)

LazyData true

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

RemoteUrl https://github.com/hjeglinton/riskscores

RemoteRef HEAD

RemoteSha 59d1a893db549eb6d885183ccd48d2e82d4ccbb7

Contents

`breastcancer` *Breast tissue biopsy data*

Description

The Breast Cancer Wisconsin dataset from the UCI machine learning repository records the measurements from breast tissue biopsies. The outcome of interest is whether the sample was benign or malignant.

Usage

```
breastcancer
```

Format

`breastcancer`**:**

A data frame with 683 rows and 10 columns:

Benign 1 for malignant, 0 for benign

ClumpThickness Clump thickness on an integer scale from 1 to 10

UniformityOfCellSize Uniformity of cell size on an integer scale from 1 to 10

UniformityofCellShape Uniformity of cell shape on an integer scale from 1 to 10

MarginalAdhesion Marginal adhesion on an integer scale from 1 to 10

SingleEpithelialCellSize Single epithelial cell size on an integer scale from 1 to 10

BareNuclei Bare nuclei on an integer scale from 1 to 10

BlandChromatin Bland chromatin on an integer scale from 1 to 10

NormalNucleoli Normal nucleoli on an integer scale from 1 to 10

Mitosis Mitosis on an integer scale from 1 to 10

Source

https://archive.ics.uci.edu/dataset/15/breast+cancer+wisconsin+original

clip_exp_vals *Clip Values*

Description

Clip values prior to exponentiation to avoid numeric errors.

Usage

```
clip_exp_vals(x)
```

Arguments

x	Numeric vector.

Value

Input vector x with all values between -709.78 and 709.78.

Examples

```
clip_exp_vals(710)
```

coef.risk_mod *Extract Model Coefficients*

Description

Extracts a vector of model coefficients (both nonzero and zero) from a "risk_mod" object. Equivalent to accessing the beta attribute of a "risk_mod" object.

Usage

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

Arguments

object	An object of class "risk_mod", usually a result of a call to risk_mod().
...	Additional arguments.

Value

Numeric vector with coefficients.

Examples

```
y <- breastcancer[[1]]
X <- as.matrix(breastcancer[,2:ncol(breastcancer)])

mod <- risk_mod(X, y, lambda0 = 0.01)
coef(mod)
```

`cv_risk_mod` *Run Cross-Validation to Tune Lambda0*

Description

Runs k-fold cross-validation on a grid of λ_0 values. Records class accuracy and deviance for each λ_0. Returns an object of class "cv_risk_mod".

Usage

```
cv_risk_mod(
  X,
  y,
  weights = NULL,
  beta = NULL,
  a = -10,
  b = 10,
  max_iters = 100,
  tol = 1e-05,
  nlambda = 25,
  lambda_min_ratio = ifelse(nrow(X) < ncol(X), 0.01, 1e-04),
  lambda0 = NULL,
  nfolds = 10,
  foldids = NULL,
  parallel = FALSE,
  shuffle = TRUE,
  seed = NULL
)
```

Arguments

`X`	Input covariate matrix with dimension $n \times p$; every row is an observation.
`y`	Numeric vector for the (binomial) response variable.
`weights`	Numeric vector of length n with weights for each observation. Unless otherwise specified, default will give equal weight to each observation.
`beta`	Starting numeric vector with p coefficients. Default starting coefficients are rounded coefficients from a logistic regression model.
`a`	Integer lower bound for coefficients (default: -10).

b	Integer upper bound for coefficients (default: 10).
max_iters	Maximum number of iterations (default: 100).
tol	Tolerance for convergence (default: 1e-5).
nlambda	Number of lambda values to try (default: 25).
lambda_min_ratio	Smallest value for lambda, as a fraction of lambda_max (the smallest value for which all coefficients are zero). The default depends on the sample size (n) relative to the number of variables (p). If $n > p$, the default is 0.0001, close to zero. If $n < p$, the default is 0.01.
lambda0	Optional sequence of lambda values. By default, the function will derive the lambda0 sequence based on the data (see lambda_min_ratio).
nfolds	Number of folds, implied if foldids provided (default: 10).
foldids	Optional vector of values between 1 and nfolds.
parallel	If TRUE, parallel processing (using foreach) is implemented during cross-validation to increase efficiency (default: FALSE). User must first register parallel backend with a function such as doParallel::registerDoParallel.
shuffle	Whether order of coefficients is shuffled during coordinate descent (default: TRUE).
seed	An integer that is used as argument by set.seed() for offsetting the random number generator. Default is to not set a particular randomization seed.

Value

An object of class "cv_risk_mod" with the following attributes:

results	Dataframe containing a summary of deviance and accuracy for each value of lambda0 (mean and SD). Also includes the number of nonzero coefficients that are produced by each lambda0 when fit on the full data.
lambda_min	Numeric value indicating the lambda0 that resulted in the lowest mean deviance.
lambda_1se	Numeric value indicating the largest lamdba0 that had a mean deviance within one standard error of lambda_min.

---

cv_risk_mod_random_start

Run Cross-Validation to Tune Lambda0 with Random Start

Description

Runs k-fold cross-validation on a grid of λ_0 values using random warm starts (see risk_mod_random_start. Records class accuracy and deviance for each λ_0. Returns an object of class "cv_risk_mod".

Usage

```
cv_risk_mod_random_start(
  X,
  y,
  weights = NULL,
  a = -10,
  b = 10,
  max_iters = 100,
  tol = 1e-05,
  nlambda = 25,
  lambda_min_ratio = ifelse(nrow(X) < ncol(X), 0.01, 1e-04),
  lambda0 = NULL,
  nfolds = 10,
  foldids = NULL,
  parallel = FALSE,
  seed = NULL,
  nstart = 5
)
```

Arguments

`X`	Input covariate matrix with dimension $n \times p$; every row is an observation.
`y`	Numeric vector for the (binomial) response variable.
`weights`	Numeric vector of length n with weights for each observation. Unless otherwise specified, default will give equal weight to each observation.
`a`	Integer lower bound for coefficients (default: -10).
`b`	Integer upper bound for coefficients (default: 10).
`max_iters`	Maximum number of iterations (default: 100).
`tol`	Tolerance for convergence (default: 1e-5).
`nlambda`	Number of lambda values to try (default: 25).
`lambda_min_ratio`	Smallest value for lambda, as a fraction of lambda_max (the smallest value for which all coefficients are zero). The default depends on the sample size (n) relative to the number of variables (p). If $n > p$, the default is 0.0001, close to zero. If $n < p$, the default is 0.01.
`lambda0`	Optional sequence of lambda values. By default, the function will derive the lambda0 sequence based on the data (see `lambda_min_ratio`).
`nfolds`	Number of folds, implied if `foldids` provided (default: 10).
`foldids`	Optional vector of values between 1 and `nfolds`.
`parallel`	If `TRUE`, parallel processing (using foreach) is implemented during cross-validation to increase efficiency (default: `FALSE`). User must first register parallel backend with a function such as doParallel::registerDoParallel.
`seed`	An integer that is used as argument by `set.seed()` for offsetting the random number generator. Default is to not set a particular randomization seed.
`nstart`	Number of different random starts to try (default: 5).

get_metrics *Get Model Metrics*

Description

Calculates a risk model's accuracy, sensitivity, and specificity given a set of data.

Usage

```
get_metrics(
  mod,
  X = NULL,
  y = NULL,
  weights = NULL,
  threshold = NULL,
  threshold_type = c("response", "score")
)
```

Arguments

mod	An object of class risk_mod, usually a result of a call to risk_mod().
X	Input covariate matrix with dimension $n \times p$; every row is an observation.
y	Numeric vector for the (binomial) response variable.
weights	Numeric vector of length n with weights for each observation. Unless otherwise specified, default will give equal weight to each observation.
threshold	Numeric vector of classification threshold values used to calculate the accuracy, sensitivity, and specificity of the model. Defaults to a range of risk probability thresholds from 0.1 to 0.9 by 0.1.
threshold_type	Defines whether the threshold vector contains risk probability values ("response") or threshold values expressed as scores from the risk score model ("score"). Default: "response".

Value

Data frame with accuracy, sensitivity, and specificity for each threshold.

Examples

```
y <- breastcancer[[1]]
X <- as.matrix(breastcancer[,2:ncol(breastcancer)])

mod <- risk_mod(X, y)
get_metrics(mod, X, y)

get_metrics(mod, X, y, threshold = c(150, 175, 200), threshold_type = "score")
```

`get_metrics_internal` *Get Model Metrics for a Single Threshold*

Description

Calculates a risk model's deviance, accuracy, sensitivity, and specificity given a set of data and a threshold value.

Usage

```
get_metrics_internal(
  mod,
  X = NULL,
  y = NULL,
  weights = NULL,
  threshold = 0.5,
  threshold_type = c("response", "score")
)
```

Arguments

`mod`	An object of class `risk_mod`, usually a result of a call to `risk_mod()`.
`X`	Input covariate matrix with dimension $n \times p$; every row is an observation.
`y`	Numeric vector for the (binomial) response variable.
`weights`	Numeric vector of length n with weights for each observation. Unless otherwise specified, default will give equal weight to each observation.
`threshold`	Numeric vector of classification threshold values used to calculate the accuracy, sensitivity, and specificity of the model. Defaults to a range of risk probability thresholds from 0.1 to 0.9 by 0.1.
`threshold_type`	Defines whether the `threshold` vector contains risk probability values ("response") or threshold values expressed as scores from the risk score model ("score"). Default: "response".

Value

List with deviance (dev), accuracy (acc), sensitivity (sens), and specificity (spec).

get_risk *Calculate Risk Probability from Score*

Description

Returns the risk probabilities for the provided score value(s).

Usage

```
get_risk(object, score)
```

Arguments

object	An object of class "risk_mod", usually a result of a call to risk_mod().
score	Numeric vector with score value(s).

Value

Numeric vector with the same length as score.

Examples

```
y <- breastcancer[[1]]
X <- as.matrix(breastcancer[,2:ncol(breastcancer)])

mod <- risk_mod(X, y)
get_risk(mod, score = c(1, 10, 20))
```

get_score *Calculate Score from Risk Probability*

Description

Returns the score(s) for the provided risk probabilities.

Usage

```
get_score(object, risk)
```

Arguments

object	An object of class "risk_mod", usually a result of a call to risk_mod().
risk	Numeric vector with probability value(s).

Value

Numeric vector with the same length as `risk`.

Examples

```
y <- breastcancer[[1]]
X <- as.matrix(breastcancer[,2:ncol(breastcancer)])

mod <- risk_mod(X, y)
get_score(mod, risk = c(0.25, 0.50, 0.75))
```

---

plot.cv_risk_mod *Plot Risk Score Cross-Validation Results*

Description

Plots the mean deviance for each $lambda_0$ tested during cross-validation.

Usage

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

Arguments

`x`	An object of class "cv_risk_mod", usually a result of a call to `cv_risk_mod()`.
`...`	Additional arguments affecting the plot produced

Value

Object of class "ggplot".

---

plot.risk_mod *Plot Risk Score Model Curve*

Description

Plots the linear regression equation associated with the integer risk score model. Plots the scores on the x-axis and risk on the y-axis.

Usage

```
## S3 method for class 'risk_mod'
plot(x, score_min = NULL, score_max = NULL, ...)
```

Arguments

x	An object of class "risk_mod", usually a result of a call to risk_mod().
score_min	The minimum score displayed on the x-axis. The default is the minimum score predicted from model's training data.
score_max	The maximum score displayed on the x-axis. The default is the maximum score predicted from model's training data.
...	Additional arguments affecting the plot produced

Value

Object of class "ggplot".

Examples

```
y <- breastcancer[[1]]
X <- as.matrix(breastcancer[,2:ncol(breastcancer)])
mod <- risk_mod(X, y, lambda0 = 0.01)

plot(mod)
```

---

predict.risk_mod *Predict Method for Risk Model Fits*

Description

Obtains predictions from risk score models.

Usage

```
## S3 method for class 'risk_mod'
predict(object, newx = NULL, type = c("link", "response", "score"), ...)
```

Arguments

object	An object of class "risk_mod", usually a result of a call to risk_mod().
newx	Optional matrix of new values for X for which predictions are to be made. If ommited, the fitted values are used.
type	The type of prediction required. The default ("link") is on the scale of the predictors (i.e. log-odds); the "response" type is on the scale of the response variable (i.e. risk probabilities); the "score" type returns the risk score calculated from the integer model.
...	Additional arguments.

Value

Numeric vector of predicted values.

Examples

```
y <- breastcancer[[1]]
X <- as.matrix(breastcancer[,2:ncol(breastcancer)])
mod <- risk_mod(X, y, lambda0 = 0.01)
predict(mod, type = "link")[1]
predict(mod, type = "response")[1]
predict(mod, type = "score")[1]
```

`risk_mod` *Fit an Integer Risk Score Model*

Description

Fits an optimized integer risk score model using a cyclical coordinate descent algorithm. Returns an object of class "risk_mod".

Usage

```
risk_mod(
  X,
  y,
  gamma = NULL,
  beta = NULL,
  weights = NULL,
  lambda0 = 0,
  a = -10,
  b = 10,
  max_iters = 100,
  tol = 1e-05,
  shuffle = TRUE,
  seed = NULL
)
```

Arguments

`X`	Input covariate matrix with dimension $n \times p$; every row is an observation.
`y`	Numeric vector for the (binomial) response variable.
`gamma`	Starting value to rescale coefficients for prediction (optional).
`beta`	Starting numeric vector with p coefficients. Default starting coefficients are rounded coefficients from a logistic regression model.
`weights`	Numeric vector of length n with weights for each observation. Unless otherwise specified, default will give equal weight to each observation.
`lambda0`	Penalty coefficient for L0 term (default: 0). See `cv_risk_mod()` for `lambda0` tuning.
`a`	Integer lower bound for coefficients (default: -10).

b	Integer upper bound for coefficients (default: 10).
max_iters	Maximum number of iterations (default: 100).
tol	Tolerance for convergence (default: 1e-5).
shuffle	Whether order of coefficients is shuffled during coordinate descent (default: TRUE).
seed	An integer that is used as argument by `set.seed()` for offsetting the random number generator. Default is to not set a particular randomization seed.

Details

This function uses a cyclical coordinate descent algorithm to solve the following optimization problem.

$$\min_{\alpha,\beta} \quad \frac{1}{n}\sum_{i=1}^{n}(\gamma y_i x_i^T \beta - log(1 + exp(\gamma x_i^T \beta))) + \lambda_0 \sum_{j=1}^{p} 1(\beta_j \neq 0)$$

$$l \leq \beta_j \leq u \quad \forall j = 1, 2, ..., p$$

$$\beta_j \in \mathbb{Z} \quad \forall j = 1, 2, ..., p$$

$$\beta_0, \gamma \in \mathbb{R}$$

These constraints ensure that the model will be sparse and include only integer coefficients.

Value

An object of class "risk_mod" with the following attributes:

gamma	Final scalar value.
beta	Vector of integer coefficients.
glm_mod	Logistic regression object of class "glm" (see stats::glm).
X	Input covariate matrix.
y	Input response vector.
weights	Input weights.
lambda0	Imput `lambda0` value.
model_card	Dataframe displaying the nonzero integer coefficients (i.e. "points") of the risk score model.
score_map	Dataframe containing a column of possible scores and a column with each score's associated risk probability.

Examples

```
y <- breastcancer[[1]]
X <- as.matrix(breastcancer[,2:ncol(breastcancer)])

mod1 <- risk_mod(X, y)
mod1$model_card

mod2 <- risk_mod(X, y, lambda0 = 0.01)
mod2$model_card

mod3 <- risk_mod(X, y, lambda0 = 0.01, a = -5, b = 5)
mod3$model_card
```

---

risk_mod_random_start *Run risk model with random start*

Description

Runs `nstart` iterations of `risk_mod()`, each with a different warm start, and selects the best model. Each coefficient start is randomly selected as -1, 0, or 1.

Usage

```
risk_mod_random_start(
  X,
  y,
  weights = NULL,
  lambda0 = 0,
  a = -10,
  b = 10,
  max_iters = 100,
  tol = 1e-05,
  seed = NULL,
  nstart = 5
)
```

Arguments

`X`	Input covariate matrix with dimension $n \times p$; every row is an observation.
`y`	Numeric vector for the (binomial) response variable.
`weights`	Numeric vector of length n with weights for each observation. Unless otherwise specified, default will give equal weight to each observation.
`lambda0`	Penalty coefficient for L0 term (default: 0). See `cv_risk_mod()` for `lambda0` tuning.
`a`	Integer lower bound for coefficients (default: -10).
`b`	Integer upper bound for coefficients (default: 10).

max_iters	Maximum number of iterations (default: 100).
tol	Tolerance for convergence (default: 1e-5).
seed	An integer that is used as argument by set.seed() for offsetting the random number generator. Default is to not set a particular randomization seed.
nstart	Number of different random starts to try (default: 5).

---

stratify_folds *Generate Stratified Fold IDs*

Description

Returns a vector of fold IDs that preserves class proportions.

Usage

```
stratify_folds(y, nfolds = 10, seed = NULL)
```

Arguments

y	Numeric vector for the (binomial) response variable.
nfolds	Number of folds (default: 10).
seed	An integer that is used as argument by set.seed() for offsetting the random number generator. Default is to not set a particular randomization seed.

Value

Numeric vector with the same length as y.

Examples

```
y <- rbinom(100, 1, 0.3)
foldids <- stratify_folds(y, nfolds = 5)
table(y, foldids)
```

summary.risk_mod *Summarize Risk Model Fit*

Description

Prints text that summarizes "risk_mod" objects.

Usage

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

Arguments

object	An object of class "risk_mod", usually a result of a call to risk_mod().
...	Additional arguments affecting the summary produced.

Value

Printed text with intercept, nonzero coefficients, gamma, lambda, and deviance

Examples

```
y <- breastcancer[[1]]
X <- as.matrix(breastcancer[,2:ncol(breastcancer)])

mod <- risk_mod(X, y, lambda0 = 0.01)
summary(mod)
```

Index