Package 'TCRpred'

March 17, 2025

Type Package

Title TCRpred: incorporating T-cell receptor repertoire for clinical outcome prediction

Version 0.1.0

Author Meiling Liu and Qianchuan He

Maintainer Meiling Liu <mliu@fredhutch.org>

Description An analytic tool for incorporating TCR repertoire for clinical outcome prediction.

Depends R (>= 3.5.0), glmnet, Biostrings, verification

License LGPL(>=2.0)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

R topics documented:

eva_metric *Evaluate Prediction Performance*

Description

eva_metric calculates evaluation metrics for predicted values against true values, supporting both binary and continuous outcomes.

Usage

```
eva_metric(Y, Y_pred)
```

Arguments

Y A numeric vector representing the true values of the response variable. If Y contains only 0 and 1, it is treated as a binary classification problem. Otherwise, it is treated as a continuous regression problem.

Y_pred A numeric vector of predicted values. For binary outcomes, values represent predicted probabilities. For continuous outcomes, values represent predicted responses.

Details

- If Y is binary, predictions (Y_pred) are thresholded at 0.5 to classify outcomes as 0 or 1. - If Y is continuous, only the Mean Squared Error (MSE) is returned. - The function handles edge cases where no classification table is generated, returning NA values instead.

Value

A named numeric vector containing evaluation metrics:

For binary outcomes:

PPV Positive Predictive Value (Precision), calculated as TP / (TP + FP).

NPV Negative Predictive Value, calculated as TN / (TN + FN).

clas_error Classification Error Rate, computed as the mean squared error between Y and Y_pred >= 0.5.

auc Area Under the ROC Curve (AUC), using verification::roc.area.

For continuous outcomes (Y is not binary):

MSE Mean Squared Error (MSE).

Examples

```
# Binary classification example
set.seed(123)
Y_true <- sample(0:1, 100, replace = TRUE)
Y_pred_prob <- runif(100)  # Simulated probability scores
eva_metric(Y_true, Y_pred_prob)

# Continuous regression example
Y_cont <- rnorm(100)
Y_pred_cont <- Y_cont + rnorm(100, sd = 0.1)  # Adding noise to simulate prediction
eva_metric(Y_cont, Y_pred_cont)
```

TCRpred *TCRpred function*

Description

TCRpred function

Usage

```
TCRpred(
  Y,
  X = NULL,
  K = NULL,
  Z = NULL,
  sid = NULL,
  aaSeq = NULL,
  abundance = NULL,
  k = NULL,
  refm = NULL,
  ntrain,
  seed = 500,
  maxiter,
  tol
)
```

Arguments

`Y` A response vector representing the outcome variable of interest. It should be either a binary variable (e.g., disease presence/absence) or a continuous variable (e.g., a clinical measurement or biomarker level).

`X` A numeric covariate matrix where each row corresponds to a subject and each column represents a covariate (e.g., age, gender, clinical factors). This matrix provides additional features that may influence the response variable.

`K` A similarity matrix that quantifies the relationships between subjects based on T-cell receptor (TCR) features. If left blank, both `Z` (TCR feature matrix) and `refm` (substitution matrix) must be provided to compute `K`.

`Z` A TCR feature matrix representing extracted sequence-based features. Each row corresponds to a subject, and each column represents a computed TCR feature. If `Z` is left blank, `K` must be explicitly assigned.

`sid` A subject identifier vector, where each element corresponds to a unique subject in the dataset. This ensures proper alignment of features and response values across multiple input matrices.

`aaSeq` A character vector containing amino acid sequences of TCRs. Each entry corresponds to a specific TCR sequence observed in the dataset.

`abundance` A numeric vector representing the abundance of each TCR sequence. Higher values indicate a greater presence of the corresponding sequence within a subject's TCR repertoire.

`k` An integer specifying the value of `k` in `k`-mer analysis. This determines the length of substrings (k-mers) used for sequence-based feature extraction. A higher `k` value captures longer sequence patterns but increases computational complexity.

`refm` A character string specifying the name of the substitution matrix used to compute the similarity matrix `K`. Common options include "BLOSUM62", "PAM250", or other biologically relevant substitution matrices.

`ntrain` An integer defining the number of samples used for training in a predictive model. This specifies how many subjects will be included in the training set before evaluating performance on test data.

`seed`	An integer used for setting the random seed, ensuring that results are reproducible across multiple runs. Setting a fixed seed allows for consistent training/testing splits and stable model behavior.
`maxiter`	An integer specifying the maximum number of iterations before the optimization process stops. This prevents infinite loops and ensures computational efficiency.
`tol`	A numeric value representing the convergence threshold. The algorithm terminates when the change in the objective function is smaller than this cutoff, indicating that further iterations will not significantly improve results.

Value

The function returns a list with the following components:

`Y_true`	The true response values from the dataset.
`Y_pred`	The predicted response values from the model.

Examples

```
# Load example dataset
data("TCRpred.data")

# Run TCRpred function with training data and specified parameters
out = TCRpred(Y = data$Y, X = data$X, K = data$K, Z = data$Z,
              ntrain = 500, maxiter = 10, tol = 0.01)

# Evaluate prediction accuracy
eva_metric(out$Y_true, out$Y_pred)
```

Index