

Package ‘caplime’

October 29, 2026

Type Package

Title Conic Adaptive Piecewise Local Explanations

Version 0.1.0

Description Provides educational, base-R implementations of a ridge-penalized piecewise hinge regression model and local surrogate explanations for numerical prediction functions. Includes simplified variants inspired by local interpretable model-agnostic explanations (LIME) and a simplified bandwidth selection variant (OPTILIME), plus exploratory stability diagnostics.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.1.0)

NeedsCompilation no

Author Nurgül Gökgöz Küçüksakallı [aut, cre] (affiliation: Department of Biomedical Engineering, Middle East Technical University),
Vilda Puruçuoğlu [aut] (affiliation: Department of Statistics, Middle East Technical University)

Maintainer Nurgül Gökgöz Küçüksakallı <nurgul.gokgoz@gmail.com>

Repository CRAN

Date/Publication 2026-10-29 19:20:14 UTC

Contents

CAP_LIME	2
CAP_stability	3
fit_CAP_model	4
plot_CAP_LIME	5
predict_CAP_model	5

Index	7
--------------	----------

CAP_LIME

*Explain one observation with a local surrogate***Description**

Explain one observation with a local surrogate.

Usage

```
CAP_LIME(x_train, instance, predict_fn, method = "LIME",
         num_samples = 2000, lambda = 1, kernel_width = NULL,
         target_r2 = 0.70, seed = 42)
```

Arguments

<code>x_train</code>	Numeric predictor matrix with observations in rows.
<code>instance</code>	Numeric vector containing one observation to explain.
<code>predict_fn</code>	Function accepting a numeric matrix and returning one finite prediction per row.
<code>method</code>	Either "LIME" or "OPTILIME".
<code>num_samples</code>	Number of simulated neighbors, at least 10.
<code>lambda</code>	Nonnegative ridge penalty.
<code>kernel_width</code>	Positive neighborhood weighting bandwidth, or NULL for a default.
<code>target_r2</code>	Target local fit for the simplified OPTILIME search.
<code>seed</code>	Random seed for repeatability.

Details

The OPTILIME mode is a simplified educational bandwidth search and not a full reproduction of the published procedure. LIME local coefficients depend on the predictor units. This function is model-agnostic and does not itself fit a conic optimization problem.

Value

A list with `method`, `instance_prediction`, `local_r2`, `kernel_width`, `effects` .

Examples

```
x <- matrix(rnorm(120), ncol = 2)
colnames(x) <- c("a", "b")
f <- function(z) z[, 1] + 0.5 * z[, 2]^2
result <- CAP_LIME(x, x[1, ], f, num_samples = 80)
head(result$effects)
```

CAP_stability	<i>Explore repeatability of local explanations</i>
---------------	--

Description

Explore repeatability of local explanations.

Usage

```
CAP_stability(x_train, instance, predict_fn, method = "LIME",
              runs = 10, top_k = 4, num_samples = 1000, seed = 42, ...)
```

Arguments

<code>x_train</code>	Numeric predictor matrix with observations in rows.
<code>instance</code>	Numeric vector containing one observation to explain.
<code>predict_fn</code>	Function accepting a numeric matrix and returning one finite prediction per row.
<code>method</code>	Either "LIME" or "OPTILIME".
<code>runs</code>	Number of independent neighborhood draws, at least 2.
<code>top_k</code>	Number of top-ranked terms compared between explanations.
<code>num_samples</code>	Neighborhood size per run.
<code>seed</code>	Starting integer random seed.
<code>...</code>	Additional parameters supplied to CAP_LIME.

Details

VSI here is mean pairwise Jaccard overlap of selected feature terms. CSI is a descriptive table of coefficient mean, standard deviation, and coefficient of variation, not the standard confidence-interval based CSI from published LIME-stability literature.

Value

A list containing VSI, CSI and explanations.

Examples

```
x <- matrix(rnorm(120), ncol = 2)
colnames(x) <- c("x1", "x2")
f <- function(z) rowSums(z)
s <- CAP_stability(x, x[1, ], f, runs = 3, top_k = 2,
                  num_samples = 80)
s$VSI
```

fit_CAP_model

*Fit ridge-penalized piecewise hinge regression***Description**

Fit ridge-penalized piecewise hinge regression.

Usage

```
fit_CAP_model(x, y, lambda = 5, num_knots = 5, degree = 1)
```

Arguments

x	Numeric matrix of predictors.
y	Numeric outcome vector.
lambda	Nonnegative ridge penalty.
num_knots	Positive integer number of evenly spaced knots per feature.
degree	Either 1 for main-effect hinge terms or 2 for pairwise hinge products.

Details

A simple piecewise ridge model fitted by linear algebra. Here conic refers to the source project name; this implementation has no explicit conic constraint. Large degrees and knot counts can create very large design matrices.

Value

A CAP_model object with knots, coefficients, predictor names and interaction degree.

Examples

```
x <- matrix(rnorm(80), ncol = 2)
colnames(x) <- c("a", "b")
y <- x[, 1]^2 + x[, 2]
model <- fit_CAP_model(x, y, num_knots = 2)
head(predict_CAP_model(model, x))
```

plot_CAP_LIME	<i>Draw ranked local-effect coefficients</i>
---------------	--

Description

Draw ranked local-effect coefficients.

Usage

```
plot_CAP_LIME(explanation, top_n = 15)
```

Arguments

explanation	Result returned by CAP_LIME.
top_n	Number of highest-ranked features to display.

Details

Signed regression coefficients are shown on their original feature scales.

Value

Draws a base graphics bar plot; returns the result from barplot invisibly or as a graphics side effect.

Examples

```
x <- matrix(rnorm(100), ncol = 2)
colnames(x) <- c("a", "b")
f <- function(z) rowSums(z)
result <- CAP_LIME(x, x[1, ], f, num_samples = 80)
plot_CAP_LIME(result)
```

predict_CAP_model	<i>Predict from a fitted piecewise model</i>
-------------------	--

Description

Predict a numeric outcome from a CAP model.

Usage

```
predict_CAP_model(model, newdata, probability = FALSE)
## S3 method for class 'CAP_model'
predict(object, newdata, ...)
```

Arguments

<code>model</code>	Fitted CAP_model object.
<code>object</code>	Fitted CAP_model object.
<code>newdata</code>	Numeric matrix with predictor columns in their training order.
<code>probability</code>	If TRUE, clip predictions to the range zero to one.
<code>...</code>	Additional arguments forwarded to <code>predict_CAP_model</code> , such as <code>probability</code> .

Details

Clipping values to zero and one does not guarantee probability calibration.

Value

A numeric vector of predictions.

Examples

```
x <- matrix(rnorm(80), ncol = 2)
colnames(x) <- c("a", "b")
model <- fit_CAP_model(x, rowSums(x), num_knots = 2)
predict(model, x[1:3, , drop = FALSE])
```

Index

CAP_LIME, [2](#)

CAP_stability, [3](#)

fit_CAP_model, [4](#)

plot_CAP_LIME, [5](#)

predict_CAP_model (predict_CAP_model), [5](#)

predict_CAP_model, [5](#)