

# Package ‘supervisedPRIM’

October 1, 2016

**Title** Supervised Classification Learning and Prediction using Patient Rule Induction Method (PRIM)

**Version** 2.0.0

**Date** 2016-10-01

**Description** The Patient Rule Induction Method (PRIM) is typically used for “bump hunting” data mining to identify regions with abnormally high concentrations of data with large or small values. This package extends this methodology so that it can be applied to binary classification problems and used for prediction.

**Depends** R (>= 3.1.1), stats, prim (>= 1.0.16)

**Suggests** kernlab, testthat

**License** GPL-3

**URL** <https://github.com/dashaub/supervisedPRIM>

**BugReports** <https://github.com/dashaub/supervisedPRIM/issues>

**LazyData** true

**RoxygenNote** 5.0.1

**ByteCompile** true

**NeedsCompilation** no

**Author** David Shaub [aut, cre]

**Maintainer** David Shaub <davidshaub@gmx.com>

**Repository** CRAN

**Date/Publication** 2016-10-01 14:39:25

## R topics documented:

predict.supervisedPRIM	2
supervisedPRIM	2
<b>Index</b>	<b>4</b>

---

```
predict.supervisedPRIM
```

*Model Predictions*

---

### Description

Perform prediction on a trained supervisedPRIM model. Output to either predicted class or positive class probability is supported.

### Usage

```
## S3 method for class 'supervisedPRIM'
predict(object, newdata, classProb = FALSE, ...)
```

### Arguments

object	A trained model of class supervisedPRIM returned by <a href="#">supervisedPRIM</a>
newdata	The new data on which to create predictions
classProb	Should the function return the estimated class
...	additional arguments (ignored) probabilities instead of the predicted class?

### Author(s)

David Shaub

### Examples

```
# Train a model to determine if a flower is setosa
data(iris)
yData <- factor(ifequal(iris$Species == "setosa", "setosa", "other"), levels = c("setosa", "other"))
xData <- iris
xData$Species <- NULL
primModel <- supervisedPRIM(x = xData, y = yData)
# Predict on the original dataset
predictions <- predict(primModel, newdata = xData)
```

---

```
supervisedPRIM
```

*Fit PRIM model to a labeled dataset*

---

### Description

perform supervised classification using Patient Rule Induction Method (PRIM)

### Usage

```
supervisedPRIM(x, y, peel.alpha = 0.05, paste.alpha = 0.01,
  mass.min = 0.05, threshold.type = 1, ...)
```

**Arguments**

<code>x</code>	matrix of data values
<code>y</code>	binary vector of 0/1 response values
<code>peel.alpha</code>	peeling quantile tuning parameter
<code>paste.alpha</code>	pasting quantile tuning parameter
<code>mass.min</code>	minimum mass tuning parameter
<code>threshold.type</code>	threshold direction indicator: 1 = " $\geq$ threshold", -1 = " $\leq$ threshold"
<code>...</code>	additional arguments to pass to <a href="#">prim.box</a>

**Details**

Fit

**Value**

an object of class `supervisedPRIM`. See additional details in [prim.box](#)

**Author(s)**

David Shaub

**Examples**

```
# Train a model to determine if a flower is setosa
data(iris)
yData <- factor(ifelse(iris$Species == "setosa", "setosa", "other"), levels = c("setosa", "other"))
xData <- iris
xData$Species <- NULL
primModel <- supervisedPRIM(x = xData, y = yData)
```

# Index

`predict.supervisedPRIM`, 2

`prim.box`, 3

`supervisedPRIM`, 2, 2