

# Package ‘pomodoro’

July 23, 2025

**Type** Package

**Title** Predictive Power of Linear and Tree Modeling

**Version** 3.8.0

**Author** Seyma Kalay <seymakalay@hotmail.com>

**Maintainer** Seyma Kalay <seymakalay@hotmail.com>

**Description** Runs generalized and multinomial logistic (GLM and MLM) models, as well as random forest (RF), Bagging (BAG), and Boosting (BOOST). This package prints out to predictive outcomes easy for the selected data and data splits.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.2

**URL** <https://github.com/seymakalay/pomodoro>,  
<https://seymakalay.github.io/pomodoro/>

**BugReports** <https://github.com/seymakalay/pomodoro/issues>

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**Imports** tibble, caret, gbm, stats, randomForest, pROC, ipred

**Depends** R (>= 2.10)

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2022-03-26 12:10:02 UTC

## Contents

BAG_Model . . . . .	2
Combined_Performance . . . . .	3
Estimate_Models . . . . .	3
GBM_Model . . . . .	4

GLM_Model . . . . .	5
MLM_Model . . . . .	6
RF_Model . . . . .	7
sample_data . . . . .	8
<b>Index</b>	<b>9</b>

---

BAG_Model	<i>Bagging Model</i>
-----------	----------------------

---

**Description**

Bagging Model

**Usage**

BAG\_Model(Data, xvar, yvar)

**Arguments**

Data	The name of the Dataset.
xvar	X variables.
yvar	Y variable.

**Details**

Decision trees suffer from high variance (If we split the training data-set randomly into two parts and set a decision tree to both parts, the results might be quite different). Bagging is an ensemble procedure which reduces the variance and increases the prediction accuracy of a statistical learning method by considering many training sets ( $\hat{f}^1(x), \hat{f}^2(x), \dots, \hat{f}^B(x)$ ) from the population. Since we can not have multiple training-sets, from a single training data-set, we can generate  $B$  different bootstrapped training data-sets ( $\hat{f}^{*1}(x), \hat{f}^{*2}(x), \dots, \hat{f}^{*B}(x)$ ) by each  $B$  trees and take a majority vote. Therefore, bagging for classification problem defined as

$$\hat{f}(x) = arg \max_k \hat{f}^{*b}(x)$$

**Value**

The output from BAG\_Model.

**Examples**

```
yvar <- c("Loan.Type")
sample_data <- sample_data[c(1:750),]
xvar <- c("sex", "married", "age", "havejob", "educ", "political.af1",
"rural", "region", "fin.intermdiaries", "fin.knowldge", "income")
BchMk.BAG <- BAG_Model(sample_data, c(xvar, "networth"), yvar )
BchMk.BAG$Roc$auc
```

---

Combined_Performance	<i>Combined Performance of the Data Splits</i>
----------------------	--

---

**Description**

Combined Performance of the Data Splits

**Usage**

Combined\_Performance(Sub.Est.Mdls)

**Arguments**

Sub.Est.Mdls     is the total perfomance of exog.

**Value**

The output from [Combined\\_Performance](#).

**Examples**

```
sample_data <- sample_data[c(1:750),]
yvar <- c("Loan.Type")
xvar <- c("sex", "married", "age", "havejob", "educ", "political.af1",
"rural", "region", "fin.intermdiaries", "fin.knowldge", "income")
CCP.RF <- Estimate_Models(sample_data, yvar, xvec = xvar, exog = "political.af1",
xadd = c("networth", "networth_homequity", "liquid.assets"),
type = "RF", dnames = c("0", "1"))
Sub.CCP.RF <- list (Mdl.1 = CCP.RF$EstMdl$`D.1+networth`,
Mdl.0 = CCP.RF$EstMdl$`D.0+networth`)
CCP.NoCCP.RF <- Combined_Performance (Sub.CCP.RF)
```

---

Estimate_Models	<i>Results of the Each Data and Data Splits</i>
-----------------	---

---

**Description**

Results of the Each Data and Data Splits

**Usage**

Estimate\_Models(DataSet, yvar, exog = NULL, xvec, xadd, type, dnames)

**Arguments**

DataSet	The name of the Dataset.
yvar	Y variable.
exog	is a vector to be subtract from the calculation.
xvec	is a vector of the variables to be used.
xadd	is an additional vector to be used.
type	can be RF, GLM, MLM, BAG, and GBM.
dnames	is the unique values of exog.

**Value**

The output from [Estimate\\_Models](#).

**Examples**

```
sample_data <- sample_data[c(1:750),]
m2.xvar0 <- c("sex", "married", "age", "havejob", "educ", "rural", "region", "income")
CCP.RF <- Estimate_Models(sample_data, yvar = c("Loan.Type"),
  exog = "political.af1", xvec = m2.xvar0,
  xadd = "networth", type = "RF", dnames = c("0", "1"))
```

GBM\_Model

*Gradient Boosting Model***Description**

Gradient Boosting Model

**Usage**

```
GBM_Model(Data, xvar, yvar)
```

**Arguments**

Data	The name of the Dataset.
xvar	X variables.
yvar	Y variable.

**Details**

Unlike bagging trees, boosting does not use bootstrap sampling, rather each tree is fit using information from previous trees. An event probability of stochastic gradient boosting model is given by

$$\hat{\pi}_i = \frac{1}{1 + \exp[-f(x)]'}$$

where  $f(x)$  is in the range of  $[-\infty, \infty]$  and its initial estimate of the model is  $f_i^{(0)} = \log(\frac{\pi_i}{1-\pi_i})$ , where  $\hat{\pi}$  is the estimated sample proportion of a single class from the training set.

Value

The output from [GBM\\_Model](#).

Examples

```
yvar <- c("Loan.Type")
sample_data <- sample_data[c(1:120),]
xvar <- c("sex", "married", "age", "havejob", "educ", "political.af1",
"rural", "region", "fin.intermdiaries", "fin.knowldge", "income")
BchMk.GBM <- GBM_Model(sample_data, c(xvar, "networth"), yvar )
BchMk.GBM$finalModel
BchMk.GBM$Roc$auc
```

---

GLM_Model	<i>Generalized Linear Model</i>
-----------	---------------------------------

---

Description

Generalized Linear Model

Usage

```
GLM_Model(Data, xvar, yvar)
```

Arguments

Data	The name of the Dataset.
xvar	X variables.
yvar	Y variable.

Details

Let **y** be a vector of response variable of accessing credit for each applicant *n*, such that  $y_i = 1$  if the applicant-*i* has access to credit, and zero otherwise. Furthermore, let  $\mathbf{x} = x_{ij}$ , where  $i = 1, \dots, n$  and  $j = 1, \dots, p$  characteristics of the applicants. The log-odds can be define as:

$$\log(\frac{\pi_i}{1 - \pi_i}) = \beta_0 + \mathbf{x}_i\beta = \beta_0 + \sum_{i=1}^p \beta_i \mathbf{x}_i$$

$\beta_0$  is the intercept,  $\beta = (\beta_1, \dots, \beta_p)$  is a  $p \times 1$  vector of coefficients and  $\mathbf{x}_i$  is the  $i_{th}$  row of **x**.

Value

The output from [GLM\\_Model](#).

## Examples

```
yvar <- c("multi.level")
sample_data <- sample_data[c(1:750),]
xvar <- c("sex", "married", "age", "havejob", "educ", "political.af1",
"rural", "region", "fin.intermdiarries", "fin.knowldge", "income")
BchMk.GLM <- GLM_Model(sample_data, c(xvar, "networth"), yvar )
BchMk.GLM$finalModel
BchMk.GLM$Roc$auc
```

---

MLM\_Model

*Multinomial Logistic Model*


---

## Description

Multinomial Logistic Model

## Usage

```
MLM_Model(Data, xvar, yvar)
```

## Arguments

Data	The name of the Dataset.
xvar	X variables.
yvar	Y variable.

## Details

Multi-nominal model is the generalized form of generalized logistic model and can be define as

$$\pi_i^h = P(y_i^h = 1 | \mathbf{x}_i^h)$$

where  $h$  presents the class labels ("1-of-h") on the basis of an input vector  $x_j$ , in our case  $x_j$  is loan types ("Formal Loan", "Informal Loan", "Both Loan", and "No Loan"). Furthermore,

$y_i^h = 1$  if the weight  $\mathbf{w}$  of  $x_j$  corresponds to belong a class and  $y_i^h = 0$  otherwise. For  $i \in 1, \dots, h$  and the weight vectors  $\mathbf{w}^i$  corresponds to class  $i$ .

We set  $w^h = 0$  and the parameters to be learned are the weight vectors  $\mathbf{w}^i$  for  $i \in 1, \dots, h - 1$ . And the class probabilities must satisfy

$$\sum_{i=1}^h P(y_i^h = 1 | \mathbf{x}_i^h, \mathbf{w}) = 1.$$

## Value

The output from [MLM\\_Model](#).

**Examples**

```

yvar <- c("Loan.Type")
sample_data <- sample_data[c(1:750),]
xvar <- c("sex", "married", "age", "havejob", "educ", "political.af1",
"rural", "region", "fin.intermdiaries", "fin.knowldge", "income")
BchMk.MLM <- MLM_Model(sample_data, c(xvar, "networth"), yvar )
BchMk.MLM$finalModel
BchMk.MLM$Roc$auc

```

RF\_Model

*Random Forest***Description**

Random Forest

**Usage**

```
RF_Model(Data, xvar, yvar)
```

**Arguments**

Data	The name of the Dataset.
xvar	X variables.
yvar	Y variable.

**Details**

Rather than considering the random sample of  $m$  predictors from the total of  $p$  predictors in each split, random forest does not consider a majority of the  $p$  predictors, and considers in each split a fresh sample of  $m_{try}$  which we usually set to  $m_{try} \approx \sqrt{p}$  Random forests which de-correlate the trees by considering  $m_{try} \approx \sqrt{p}$  show an improvement over bagged trees  $m = p$ .

**Value**

The output from [RF\\_Model](#).

**Examples**

```

sample_data <- sample_data[c(1:750),]
yvar <- c("Loan.Type")
xvar <- c("sex", "married", "age", "havejob", "educ", "political.af1",
"rural", "region", "fin.intermdiaries", "fin.knowldge", "income")
BchMk.RF <- RF_Model(sample_data, c(xvar, "networth"), yvar )
BchMk.RF

```

---

sample_data	<i>Sample data for analysis. A dataset containing information of access to credit.</i>
-------------	--

---

**Description**

Sample data for analysis.

A dataset containing information of access to credit.

**Usage**

sample\_data

**Format**

A data\_frame with 53940 rows and 10 variables:

**x1** hhid, household id number

**x2** swgt, survey weight

**x3** region, 3 factor level, west, east, and center

**x4** No.Loan, if the household has no loan

**x5** Formal, if the household has formal loan

**x6** Both, if the household has both loan

**x7** Informal, if the household has informal loan

**x8** sex, if the household has male

**y1** Loan.Type, 4 factor level type of the loan

**y2** multi.level, 2 factor level if the household has access to loan or not ...



# Index

## \* **datasets**

sample\_data, [8](#)

BAG\_Model, [2](#), [2](#)

Combined\_Performance, [3](#), [3](#)

Estimate\_Models, [3](#), [4](#)

GBM\_Model, [4](#), [5](#)

GLM\_Model, [5](#), [5](#)

MLM\_Model, [6](#), [6](#)

RF\_Model, [7](#), [7](#)

sample\_data, [8](#)