# Package 'DBModelSelect'

September 20, 2023

**Type** Package

**Title** Distribution-Based Model Selection

**Version** 0.2.0

**Date** 2023-08-22

**Description** Perform model selection using distribution and probability-based methods, including standardized AIC, BIC, and AICc. These standardized information criteria allow one to perform model selection in a way similar to the prevalent ``Rule of 2'' method, but formalize the method to rely on probability theory. A novel goodness-of-fit procedure for assessing linear regression models is also available. This test relies on theoretical properties of the estimated error variance for a normal linear regression model, and employs a bootstrap procedure to assess the null hypothesis that the fitted model shows no lack of fit. For more information, see Koeneman and Cavanaugh (2023) <arXiv:2309.10614>. Functionality to perform all subsets linear or generalized linear regression is also available.

**URL** https://github.com/shkoeneman/DBModelSelect

**License** GPL-3

**Depends** R (>= 4.1.0)

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Scott H. Koeneman [aut, cre]

**Maintainer** Scott H. Koeneman <Scott.Koeneman@jefferson.edu>

**Repository** CRAN

**Date/Publication** 2023-09-20 18:40:08 UTC

## R topics documented:

DBModelSelect-package . . . . . 2
AICc . . . . . 3
BootGOFTestLM . . . . . 4
FitGLMSubsets . . . . . 5
FitLMSubsets . . . . . 6
StandICModelSelect . . . . . 7

**Index** **9**

---

`DBModelSelect-package` *Distribution-Based Model Selection*

---

## Description

Perform model selection using distribution and probability-based methods, including standardized AIC, BIC, and AICc. These standardized information criteria allow one to perform model selection in a way similar to the prevalent "Rule of 2" method, but formalize the method to rely on probability theory. A novel goodness-of-fit procedure for assessing linear regression models is also available. This test relies on theoretical properties of the estimated error variance for a normal linear regression model, and employs a bootstrap procedure to assess the null hypothesis that the fitted model shows no lack of fit. For more information, see Koeneman and Cavanaugh (2023) <arXiv:2309.10614>. Functionality to perform all subsets linear or generalized linear regression is also available.

## Details

The DESCRIPTION file:

| | |
|---|---|
| Package: | DBModelSelect |
| Type: | Package |
| Title: | Distribution-Based Model Selection |
| Version: | 0.2.0 |
| Date: | 2023-08-22 |
| Authors@R: | person("Scott H.", "Koeneman", email = "Scott.Koeneman@jefferson.edu", role = c("aut", "cre")) |
| Description: | Perform model selection using distribution and probability-based methods, including standardized AIC, BIC |
| URL: | https://github.com/shkoeneman/DBModelSelect |
| License: | GPL-3 |
| Depends: | R (>= 4.1.0) |
| RoxygenNote: | 7.2.3 |
| Author: | Scott H. Koeneman [aut, cre] |
| Maintainer: | Scott H. Koeneman <Scott.Koeneman@jefferson.edu> |

Index of help topics:

```
AICc                    Corrected AIC for linear models
BootGOFTestLM           Bootstrap goodness-of-fit procedure for linear
                        models
DBModelSelect-package   Distribution-Based Model Selection
FitGLMSubsets           Perform all subsets regression for generalized
                        linear models
FitLMSubsets            Perform all subsets linear regression
StandICModelSelect      Model selection using standardized information
                        criteria
```

The `DBModelSelect` package provides several methods of model selection based in distributional

theory. This includes an implementation of selection using standardized information criteria in the `StandICModelSelect` function, and the implementation of an omnibus goodness-of-fit test for linear models in the `BootGOFTestLM` function.

## Author(s)

**Maintainer**: Scott H. Koeneman <Scott.Koeneman@jefferson.edu>

## See Also

Useful links:

- https://github.com/shkoeneman/DBModelSelect

---

`AICc` *Corrected AIC for linear models*

---

## Description

Calculates corrected AIC for an 'lm' linear model object.

## Usage

```
AICc(model)
```

## Arguments

| | |
|---|---|
| `model` | A fitted 'lm' object. |

## Value

The numeric value of of corrected AIC for the supplied linear model object.

## Examples

```
#generate data
set.seed(9122023)
data <- data.frame(x = rnorm(100), y = rnorm(100))
AICc(lm(y~x, data = data))
```

`BootGOFTestLM` *Bootstrap goodness-of-fit procedure for linear models*

## Description

Performs a bootstrap goodness-of-fit procedure to assess the fit of a normal linear regression model

## Usage

```
BootGOFTestLM(
  x,
  data,
  boot_iter = 1000,
  level = 0.95,
  return_dist = FALSE,
  ...
)

## S3 method for class 'BootGOFTestLM'
print(x, ...)
```

## Arguments

| | |
|---|---|
| `x` | A fitted `lm` object. |
| `data` | A dataframe used to fit the model given by `x`. |
| `boot_iter` | An integer indicating number of bootstrap iterations to perform. |
| `level` | Confidence level of the bootstrap interval used in the test. |
| `return_dist` | A logical specifying whether to optionally return the bootstrap distribution. Defaults to FALSE. |
| `...` | Additional arguments. |

## Value

A list containing the specification and results of the test. The hypothesis of adequate fit is rejected if the null value is not contained in the bootstrap interval.

## Examples

```
# generate some data
set.seed(5122023)
data <- data.frame(s = rnorm(200), t = rnorm(200))
data$y <- data$s + rnorm(200)
# determine whether candidate model shows lack of fit
model <- lm(y~s+t, data = data)
BootGOFTestLM(model, data = data, booot_iter = 100)
```

`FitGLMSubsets` *Perform all subsets regression for generalized linear models*

## Description

Fit a specified generalized linear model on all subsets of covariates supplied. May be done in parallel if a cluster is supplied. Produces an output suitable for use with the `StandICModelSelect` function.

## Usage

```
FitGLMSubsets(
  response,
  data,
  family,
  intercept = TRUE,
  force_intercept = TRUE,
  cluster = NULL,
  ...
)
```

## Arguments

| | |
|---|---|
| `response` | A character string specifying the name of the response variable. |
| `data` | A dataframe containing a column corresponding to the response variable in addition to columns for each covariate of interest. |
| `family` | A family suitable for supplying to the `glm` function specifying the error distribution and link function. |
| `intercept` | A logical indicating whether an intercept term should be considered in models. Defaults to TRUE. |
| `force_intercept` | A logical indicating whether to force an intercept term into all models if an intercept is desired. Defaults to TRUE. |
| `cluster` | A cluster created using `parallel::makeCluster`. |
| `...` | Additional arguments that may be supplied when calling `glm` to fit the models of interest. |

## Value

A list of fitted models suitable for use with the `StandICModelSelect` function.

## Examples

```
# example code
# generate some data
data <- data.frame(s = rnorm(200), t = rnorm(200))
data$y <- data$s + rnorm(200)
# perform all subsets regression
model_list <- FitGLMSubsets(response = "y", data = data, family = gaussian(),
  intercept = TRUE, force_intercept = TRUE)
# perform model selection
model_select <- StandICModelSelect(model_list, IC = "AIC")
```

---

`FitLMSubsets` *Perform all subsets linear regression*

---

## Description

Perform linear regression on all subsets of covariates supplied. May be done in parallel if a cluster is supplied. Produces an output suitable for use with the `StandICModelSelect` function.

## Usage

```
FitLMSubsets(
  response,
  data,
  intercept = TRUE,
  force_intercept = TRUE,
  cluster = NULL
)
```

## Arguments

| | |
|---|---|
| `response` | A character string specifying the name of the response variable. |
| `data` | A dataframe containing a column corresponding to the response variable in addition to columns for each covariate of interest. |
| `intercept` | A logical indicating whether an intercept term should be considered in models. Defaults to TRUE. |
| `force_intercept` | A logical indicating whether to force an intercept term into all models if an intercept is desired. Defaults to TRUE. |
| `cluster` | A cluster created using `parallel::makeCluster`. |

## Value

A list of fitted linear models suitable for use with the `StandICModelSelect` function.

## Examples

```
# example code
# generate some data
data <- data.frame(s = rnorm(200), t = rnorm(200))
data$y <- data$s + rnorm(200)
# perform all subsets regression
model_list <- FitLMSubsets(response = "y", data = data, intercept = TRUE, force_intercept = TRUE)
# perform model selection
model_select <- StandICModelSelect(model_list, IC = "AIC")
```

---

`StandICModelSelect` *Model selection using standardized information criteria*

---

## Description

Perform model selection on a list of models using standardized information criteria.

## Usage

```
StandICModelSelect(
  x,
  IC = "AIC",
  ref_model_index = NULL,
  sd_cutoff = 2,
  user_df = NULL,
  ...
)

## S3 method for class 'StandICModelSelect'
print(x, ...)

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

## Arguments

| | |
|---|---|
| `x` | A list containing the fitted model objects on which to perform model selection. Model objects must have a `logLik` method defined for them. |
| `IC` | A character string containing the base information criteria to use. Options are "AIC", "BIC", and "AICc" for linear models. Default option is 'AIC'. |
| `ref_model_index` | An integer with the index of the largest candidate model to use as the reference. If not supplied, defaults to the model with largest number of estimated coefficients in `x`. |
| `sd_cutoff` | A numeric describing how many standard deviations to use when formulating a cutoff for model viability. |

| | |
|---|---|
| user_df | An optional vector the same length as x where one can specify the degrees of freedom of each fitted model. If not supplied, the degrees of freedom for each model is calculated to be the number of estimated regression coefficients. |
| ... | Additional arguments. |

## Value

A list containing the final model selected in addition to standardized information criteria and difference in degrees of freedom for all candidate models.

## Examples

```
# example code
# generate some data
s <- rnorm(200)
t <- rnorm(200)
y <- s + rnorm(200)
# formulate and fit models
model_list <- list(lm(y~1), lm(y~s), lm(y~t), lm(y~s+t))
# perform model selection
model_select <- StandICModelSelect(model_list, IC = "AIC")
# display best model
model_select$best_model
```

# Index

AICc, 3

BootGOFTestLM, 4

DBModelSelect *(*DBModelSelect-package*)*, 2
DBModelSelect-package, 2

FitGLMSubsets, 5
FitLMSubsets, 6

plot.StandICModelSelect
        *(*StandICModelSelect*)*, 7
print.BootGOFTestLM *(*BootGOFTestLM*)*, 4
print.StandICModelSelect
        *(*StandICModelSelect*)*, 7

StandICModelSelect, 7