

# BACE: A gretl Package for Model Averaging in Limited Dependent Variable Models

Marcin Błażejowski<sup>†</sup>, Jacek Kwiatkowski<sup>‡</sup>

<sup>†</sup>University WSB Merito in Toruń, Poland

<sup>‡</sup>Nicolaus Copernicus University, Toruń, Poland

Gretl Conference 2023

- 1 Introduction
  - The problem no. 1
  - The problem no. 2
- 2 **BACE** package in gretl
  - History
  - A new way
  - New interface
- 3 Comparison with other software
  - Logit regression model
  - Poisson regression model
  - Run times



- 1 One of the basic problems in econometric modeling is the identification of the determinants of the dependent variable.

- 1 One of the basic problems in econometric modeling is the identification of the determinants of the dependent variable.
- 2 When the number of possible explanatory variables is  $K = 30$ , the number of possible specifications is  $2^K = 1,073,741,824$ .

- 1 One of the basic problems in econometric modeling is the identification of the determinants of the dependent variable.
- 2 When the number of possible explanatory variables is  $K = 30$ , the number of possible specifications is  $2^K = 1,073,741,824$ .
- 3 Building a model with a very large number of explanatory variables can potentially lead to decision-making problems.

- 1 One of the basic problems in econometric modeling is the identification of the determinants of the dependent variable.
- 2 When the number of possible explanatory variables is  $K = 30$ , the number of possible specifications is  $2^K = 1,073,741,824$ .
- 3 Building a model with a very large number of explanatory variables can potentially lead to decision-making problems.
- 4 Moreover, different types of modeling approaches can lead to different estimates and conflicting conclusions.

# The solution

Bayesian model averaging (BMA) is a useful alternative to other variable selection procedures, because it incorporates **model uncertainty** into conclusions about estimated parameters. It is a standard Bayesian solution, where the inference on parameters is based on a weighted average over all possible models under consideration, rather than on one single regression model. These weights are Bayesian posterior probabilities of the individual models<sup>1</sup>.

---

<sup>1</sup>Błażejowski, M., & Kwiatkowski, J. (2015). Bayesian Model Averaging and Jointness Measures for gretl. *Journal of Statistical Software*, 68(i05), 1–24. <https://doi.org/10.18637/jss.v068.i05>



In some relatively simple models, such as linear regression, the marginal data density used for specification comparison can be represented by a closed-form analytic expression.

In some relatively simple models, such as linear regression, the marginal data density used for specification comparison can be represented by a closed-form analytic expression.

However, in many situations, such as limited variable models (e.g., logit and probit), it is difficult to quickly calculate the explanatory power of a very large set of competitive models.

In some relatively simple models, such as linear regression, the marginal data density used for specification comparison can be represented by a closed-form analytic expression.

However, in many situations, such as limited variable models (e.g., logit and probit), it is difficult to quickly calculate the explanatory power of a very large set of competitive models.

Numerical integration is computationally demanding and time-consuming in such cases (Newton & Raftery, 1994; Gelman et al., 2013; Pajor, 2017).

# The solutions

# The solutions

- 1 Generalized Linear Models:
- 2 Frequency approach based on Bayesian Information Criterion (BIC) – BACE

# The solutions

- 1 Generalized Linear Models:
  - **ParMA** package in `gretl`— Lucchetti and Pedini (2022) – Bayesian approach
- 2 Frequency approach based on Bayesian Information Criterion (BIC) – BACE

# The solutions

- 1 Generalized Linear Models:
  - **ParMA** package in `gretl`— Lucchetti and Pedini (2022) – Bayesian approach
  - **BMA** package in R — Raftery et al. (2022)
- 2 Frequency approach based on Bayesian Information Criterion (BIC) – BACE

# The solutions

- 1 Generalized Linear Models:
  - **ParMA** package in **gretl**— Lucchetti and Pedini (2022) – Bayesian approach
  - **BMA** package in R — Raftery et al. (2022)
- 2 Frequency approach based on Bayesian Information Criterion (BIC) – BACE
  - **BMA** package in R — Raftery et al. (2022)



# The solutions

- 1 Generalized Linear Models:
  - **ParMA** package in **gretl**— Lucchetti and Pedini (2022) – Bayesian approach
  - **BMA** package in R — Raftery et al. (2022)
- 2 Frequency approach based on Bayesian Information Criterion (BIC) – BACE
  - **BMA** package in R — Raftery et al. (2022)
  - **mlogitBMA** for **BMA** package in R — Sevcikova and Raftery (2022)

# The solutions

- 1 Generalized Linear Models:
  - **ParMA** package in **gretl**— Lucchetti and Pedini (2022) – Bayesian approach
  - **BMA** package in R — Raftery et al. (2022)
- 2 Frequency approach based on Bayesian Information Criterion (BIC) – BACE
  - **BMA** package in R — Raftery et al. (2022)
  - **mlogitBMA** for **BMA** package in R — Sevcikova and Raftery (2022)
  - **BAS** package in R — Clyde (2022)

# The solutions

- 1 Generalized Linear Models:
  - **ParMA** package in `gretl`— Lucchetti and Pedini (2022) – Bayesian approach
  - **BMA** package in R — Raftery et al. (2022)
- 2 Frequency approach based on Bayesian Information Criterion (BIC) – BACE
  - **BMA** package in R — Raftery et al. (2022)
  - **mlogitBMA** for **BMA** package in R — Sevcikova and Raftery (2022)
  - **BAS** package in R — Clyde (2022)
  - **BACE** package in `gretl`— Błażejowski and Kwiatkowski (2023)

# BACE package - Main features

In terms of models with a limited dependent variable, our package can be considered as an alternative to the previously mentioned packages because:

# BACE package - Main features

In terms of models with a limited dependent variable, our package can be considered as an alternative to the previously mentioned packages because:

- 1 it offers a larger number of various models: **logit** (including ordered and multinomial), **probit** (including ordered), **Poisson** regression, **Tobit** model, **interval** regression, and **linear** regression,

# BACE package - Main features

In terms of models with a limited dependent variable, our package can be considered as an alternative to the previously mentioned packages because:

- 1 it offers a larger number of various models: **logit** (including ordered and multinomial), **probit** (including ordered), **Poisson** regression, **Tobit** model, **interval** regression, and **linear** regression,
- 2 it is fast and provides consistent results,

# BACE package - Main features

In terms of models with a limited dependent variable, our package can be considered as an alternative to the previously mentioned packages because:

- 1 it offers a larger number of various models: **logit** (including ordered and multinomial), **probit** (including ordered), **Poisson** regression, **Tobit** model, **interval** regression, and **linear** regression,
- 2 it is fast and provides consistent results,
- 3 it includes the latest proposals of BIC variants, and the latest jointness measures,

# BACE package - Main features

In terms of models with a limited dependent variable, our package can be considered as an alternative to the previously mentioned packages because:

- 1 it offers a larger number of various models: **logit** (including ordered and multinomial), **probit** (including ordered), **Poisson** regression, **Tobit** model, **interval** regression, and **linear** regression,
- 2 it is fast and provides consistent results,
- 3 it includes the latest proposals of BIC variants, and the latest jointness measures,
- 4 it is able to replicate the complex reduction strategy (Błażejowski, Kufel, & Kwiatkowski, 2020),



# BACE package - Main features

In terms of models with a limited dependent variable, our package can be considered as an alternative to the previously mentioned packages because:

- 1 it offers a larger number of various models: **logit** (including ordered and multinomial), **probit** (including ordered), **Poisson** regression, **Tobit** model, **interval** regression, and **linear** regression,
- 2 it is fast and provides consistent results,
- 3 it includes the latest proposals of BIC variants, and the latest jointness measures,
- 4 it is able to replicate the complex reduction strategy (Błażejowski, Kufel, & Kwiatkowski, 2020),
- 5 it has a GUI interface.

The **BACE** ver. 3.0 is *improved* an *extended* version of the package presented at GC2017, Athens, Greece. So, our today's speech may be treated as a continuation of that previous work. The timeline looks as follows:

The **BACE** ver. 3.0 is *improved* an *extended* version of the package presented at GC2017, Athens, Greece. So, our today's speech may be treated as a continuation of that previous work. The timeline looks as follows:

2018-07-22: initial version (1.0)

The **BACE** ver. 3.0 is *improved* an *extended* version of the package presented at GC2017, Athens, Greece. So, our today's speech may be treated as a continuation of that previous work. The timeline looks as follows:

2018-07-22: initial version (1.0)

2019-11-01: MPI based version (2.0)

The **BACE** ver. 3.0 is *improved* an *extended* version of the package presented at GC2017, Athens, Greece. So, our today's speech may be treated as a continuation of that previous work. The timeline looks as follows:

2018-07-22: initial version (1.0)

2019-11-01: MPI based version (2.0)

2023-03-15: refactored version with completely new internal strategy for managing (storing and updating) posterior results obtained for already visited specifications + several models added + many more (3.0)

The **BACE** ver. 3.0 is *improved* an *extended* version of the package presented at GC2017, Athens, Greece. So, our today's speech may be treated as a continuation of that previous work. The timeline looks as follows:

2018-07-22: initial version (1.0)

2019-11-01: MPI based version (2.0)

2023-03-15: refactored version with completely new internal strategy for managing (storing and updating) posterior results obtained for already visited specifications + several models added + many more (3.0)

**BACE** for gretl project main page:

<https://sourceforge.net/projects/bace-for-gretl>

The architecture of **BACE** 1.0 – even used in parallel via MPI interface – turned out to be a dead end due to the following factors:

The architecture of **BACE** 1.0 – even used in parallel via MPI interface – turned out to be a dead end due to the following factors:

- estimation based on "manual" algebra operations via `mols` was effective only for linear regression



The architecture of **BACE** 1.0 – even used in parallel via MPI interface – turned out to be a dead end due to the following factors:

- estimation based on "manual" algebra operations via `mols` was effective only for linear regression
- bundle-based posterior results managing with hexadecimal keys with multiple `hex`  $\leftrightarrow$  `bin` and `bin`  $\rightarrow$  `int` operations was massively CPU time consuming

The architecture of **BACE** 1.0 – even used in parallel via MPI interface – turned out to be a dead end due to the following factors:

- estimation based on "manual" algebra operations via `mols` was effective only for linear regression
- bundle-based posterior results managing with hexadecimal keys with multiple `hex`  $\leftrightarrow$  `bin` and `bin`  $\rightarrow$  `int` operations was massively CPU time consuming
- furthermore, many updates (write back operations) lead to memory fragmentation slowing the package down dramatically

# Surprisingly effective trick

## Surprisingly effective trick

- We new that we must reduce any write back operations to minimum, i.e. updating only(!) models counter and leaving the rest of posterior results untouched.

## Surprisingly effective trick

- We new that we must reduce any write back operations to minimum, i.e. updating only(!) models counter and leaving the rest of posterior results untouched.
  - ✓ ... and we managed to do it quite easily.

## Surprisingly effective trick

- We new that we must reduce any write back operations to minimum, i.e. updating only(!) models counter and leaving the rest of posterior results untouched.
  - ✓ ... and we managed to do it quite easily.
- But there was still an inefficient method of mapping `hex`  $\leftrightarrow$  `bin`  $\rightarrow$  `int`, that was problematic.

## Surprisingly effective trick

- We new that we must reduce any write back operations to minimum, i.e. updating only(!) models counter and leaving the rest of posterior results untouched.
  - ✓ ... and we managed to do it quite easily.
- But there was still an inefficient method of mapping `hex`  $\leftrightarrow$  `bin`  $\rightarrow$  `int`, that was problematic.
  - ✓ ... and we were thinking of how to speed up the package and – after the number of different trials – **we decided to test something very strange:**

## Surprisingly effective trick

- We new that we must reduce any write back operations to minimum, i.e. updating only(!) models counter and leaving the rest of posterior results untouched.
  - ✓ ... and we managed to do it quite easily.
- But there was still an inefficient method of mapping `hex`  $\leftrightarrow$  `bin`  $\rightarrow$  `int`, that was problematic.
  - ✓ ... and we were thinking of how to speed up the package and – after the number of different trials – **we decided to test something very strange:**

```
|| model_id = sprintf("%d", M.mod_struct_old)
```

where `M.mod_struct_old` is a binary vector:

$$\begin{bmatrix} 0 & 1 & 1 & 1 & \dots & 0 & 1 & 0 \end{bmatrix}_{1 \times K}$$



# BACE core function

Core function that runs and controls the main loop:

```
function bundle BACE (list Y "Dependent variable",  
  list X_list "List of independent variables (X)",  
  bundle Options [null])
```

# BACE core function

Core function that runs and controls the main loop:

```
function bundle BACE (list Y "Dependent variable",
  list X_list "List of independent variables (X)",
  bundle Options [null])
```

The first two options are mandatory, while the last one is an optional set of additional arguments<sup>2</sup>.

---

<sup>2</sup>Note, that in case of `intreg` estimator the Y list must contain two series with *minvar* and *maxvar* (left- and right- bounds respectively)

# BACE core function

Core function that runs and controls the main loop:

```
function bundle BACE (list Y "Dependent variable",
  list X_list "List of independent variables (X)",
  bundle Options [null])
```

The first two options are mandatory, while the last one is an optional set of additional arguments<sup>2</sup>.

Note, that some of the above-mentioned optional parameters are **over-loadable**, which means that their value may be passed either as an integer or as a string (case-insensitive). Those parameters are: `bic_type`, `model_prior`, `with_const`, `do_joint`, `data_trans`, and `model_type`.

---

<sup>2</sup>Note, that in case of `intreg` estimator the Y list must contain two series with *minvar* and *maxvar* (left- and right- bounds respectively)

# BACE optional parameters

**string** Options["model\_type"] = "logit": the type of model that should be applied to the data; possible options are: 1—linear, 2—logit, 3—probit, 4—poisson, 5—tobit, 6—intreg, 7—mlogit

**scalar** Options["bic\_type"] = 2: the version of Bayesian Information Criterion that should be used; possible values are: 1—bic, 2—hbic, 3—ibic, 4—ebic

**scalar** Options["model\_prior"] = 1: the prior model size distribution to use; possible values are: 0—none, 1—Binomial, 2—Beta-Binomial

# BACE optional parameters

**scalar** Options["with\_const"] = 1: the inclusion of a constant term in each estimated model; possible values are: 0—Never (without constant), 1—Always (constant is always included), 2—Can be dropped (constant may be dropped)

**scalar** Options["avg\_model\_size"] = avg\_model\_size\_tmp  
the expected prior model size is calculated as:

$$\text{avg\_model\_size\_tmp} = \begin{cases} K / 2 & \text{if } \text{with\_const} \neq 2, \\ (K + 1) / 2 & \text{elsewhere,} \end{cases}$$

and  $K$  stands for the number of independent variables

**scalar** Options["data\_trans"] = 0: the transformation of explanatory variables; possible options are: 0—none, 1—centering (i.e., subtracting the mean from each value of a variable), 2—standarizing (centering a variable and additional dividing by its standard deviation)

**scalar** Options["do\_joint"] = 0: the type of jointness analysis; possible values are: 0—none, 1—DW1 (Doppelhofer-Weeks Measure 1), 2—DW2 (Doppelhofer-Weeks Measure 2), 3—LS1 (Ley-Steel Measure 1), 4—LS2 (Ley-Steel Measure 2), 5—SM (Strachan Measure), 6—YQ (Yule's Q Measure), 7—YQM (Modified Yule's Q Measure)

**scalar** Options["alpha"] = alpha\_tmp: the significance level for the initial model, where **scalar** alpha\_tmp takes the following values:

$$\text{alpha\_tmp} = \begin{cases} 1.0 & \text{if } \text{model\_type} = \text{"intreg"}, \\ 0.6 & \text{elsewhere;} \end{cases}$$

**scalar** Options["Nrank"] = 5: the number of top-ranked models to be printed

**scalar** Options["Npredict"] = 0: the number of out-of-sample forecasts

**scalar** Options["y\_lag"] = 0: defines the lag order of the dependent

**scalar** Options["hyperparameter"] = 0.5: the hyperparameter for prior-based Bayesian information criterion; for now, it is interpreted as  $\gamma$  for the extended Bayesian information criterion (EBIC)

**scalar** Options["binary\_threshold"] = 0.5: this hyper-parameter is applicable only in models for binary-dependent variables; it indicates the lowest probability for which  $\hat{y}_i$  should be considered as 1

**scalar** Options["low"] = 0: this option is applicable only for the **tobit** model; it controls the `--llimit` flag of the tobit model, which sets the left bound in censoring

**scalar** Options["high"] = NA: applicable only for **tobit** model; this option controls tobit's `--rlimit` flag for setting right bound in truncation

**scalar** Options["verbosity"] = 2: verbosity of the output: 1—basic results, 2—full results; this option controls what is printed as output by BACE: if set to 2, all possible results are printed, and if set to 1, printing of the initial model and models in ranking is suppressed

**scalar** Options["\_gui"] = 0: this parameter is for internal use only and must not be modified by the user under any circumstances

**scalar** Options["Nrep"] = Nrep: the total number of Monte Carlo iterations (model draws) in MC<sup>3</sup> algorithm, where **scalar** Nrep takes the following values:

$$\text{Nrep} = \begin{cases} 10^4 & \text{if } K \leq 15, \\ 10^5 & \text{if } K \in [16; 19], \\ 10^6 & \text{if } K \geq 20; \end{cases}$$

**scalar** Options["show\_progress"] = do\_show\_progress: this controls whether the overall progress is shown, where **scalar** do\_show\_progress takes the following values:

$$\text{do\_show\_progress} = \begin{cases} 1 & \text{if } \text{\$sysinfo["gui\_mode"]}, \\ 0 & \text{elsewhere;} \end{cases}$$

**scalar** Options["do\_plot"] = do\_plot\_tmp: this option controls whether to produce graphs, where **scalar** do\_plot\_tmp takes the following values:

$$\text{do\_plot\_tmp} = \begin{cases} 1 & \text{if } \text{\$sysinfo["gui\_mode"]}, \\ 0 & \text{elsewhere;} \end{cases}$$



# BACE main window

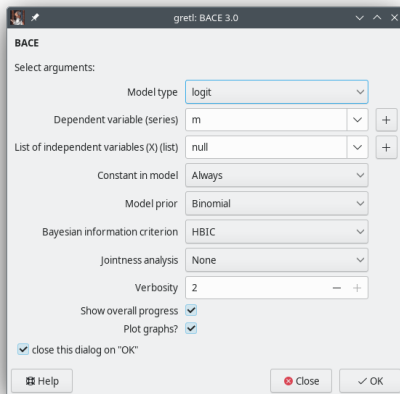


Figure: The BACE GUI window interface.

Variable	PIP			
	BACE	ParMA	BMA	BAS
const	1.000	1.000	1.000	1.000
ht	0.842	0.746	0.675	0.388
lwt	0.809	0.789	0.694	0.411
smoke	0.633	0.721	0.369	0.289
Drace_2	0.574	0.569	0.245	0.238
ptl	0.546	0.555	0.423	0.316
ui	0.485	0.496	0.296	0.238
Drace_3	0.443	0.580	0.167	0.197
age	0.232	0.242	0.089	0.150
ftv	0.151	0.153	0.014	0.087

**Table:** Comparison of Posterior Inclusion Probabilities (PIPs) among BACE, ParMA, BMA, and BAS packages for the **logit regression model**. Dataset *bithwt* presented in Hosmer and Lemeshow (2000), which consists of 189 births with eight risk factors associated with a dummy variable for low infant birth weight and determinants: *age* – mother’s age, *lwt* – mother’s weight, *race* – mother’s race, *smoke* – smoking status, *ptl* – number of previous premature labours, *ht* – history of hypertension, *ui* – presence of uterine irritability, *ftv* – number of physician visits

Variable	PIP			
	BACE	ParMA	BMA	BAS
const	1.000	1.000	1.000	1.000
ILLNESS	1.000	1.000	1.000	0.999
ACTDAYS	1.000	1.000	1.000	1.000
SEX	0.931	0.912	1.000	0.875
HSCORE	0.818	0.763	0.723	0.070
AGE	0.632	0.688	0.642	0.681
FREEPOOR	0.614	0.616	0.350	0.031
AGESQ	0.355	0.316	0.358	0.283
INCOME	0.302	0.338	0.020	0.015
LEVYPLUS	0.126	0.078	0.000	0.001
CHCOND2	0.105	0.042	0.000	0.001
CHCOND1	0.075	0.054	0.000	<0.001
FREEREPA	0.064	0.030	0.000	0.004

**Table:** Comparison of Posterior Inclusion Probabilities (PIPs) among BACE, ParMA, BMA, and BAS packages for the **Poisson regression model**. Cameron and Trivedi (1986).  
Dependent variable: the number of doctor visits. Determinants: *SEX* (1 if female, 0 otherwise), *AGE* (in years divided by 100), *AGESQ* (*AGE* squared), *INCOME* (annual, in tens of thousands of dollars), *LEVYPLUS* (1 if private insurance, 0 otherwise), *FREEPOOR* (1 if free government insurance due to low income, 0 otherwise), *FREEREPA* (1 if free government insurance due to old-age, etc., 0 otherwise), *ILLNESS* (number of illnesses), *ACTDAYS* (number of days of reduced activity), *HSCORE* (health questionnaire score, high score means bad health), *CHCOND1* (1 if chronic condition(s) but not limited in activity, 0 otherwise), *CHCOND2* (1 if chronic condition(s) and limited in activity, 0 otherwise).

Run time	Dataset	$N$	$K$	BACE	ParMA	BMA	BAS
Logit	<i>bithwt</i>	189	9	0.17	2.04	0.42	0.24
Ordered logit	<i>nels</i>	6649	13	5.13	—	—	—
Probit	<i>bithwt</i>	189	9	0.22	2.47	2.56	—
Ordered probit	<i>nels</i>	6649	13	5.59	—	—	—
Multinomial logit	<i>hsbdemo</i>	200	10	0.35	—	9.24	—
Poisson	<i>rac3d</i>	5190	12	4.14	8.88	4.30	224.0
Tobit (Type I)	<i>tobit</i>	200	4	0.07	—	—	—
Interval regression	<i>wtp</i>	312	5	0.12	—	—	—
Linear regression	<i>FLS</i>	72	41	45.38	14041.1	11.18	84.85

**Table:** Run time in seconds of BACE, ParMA, BMA, and BAS packages for selected datasets,  $N$  – the number of observations,  $K$  – the total number of explanatory variables in estimated model.

# Empirical example no. 1: **logit** model

```
set verbose off
set seed 1000000

# We load packages
include BACE.gfn

# We load birthwt dataset
open birthwt.gdt --frompkg=BACE --quiet

# We specify list of explanatory variables
list X = age lwt Drace_2 Drace_3 smoke ptl ht ui ftv

set stopwatch
RES = BACE(low, X, _(model_type="logit", model_prior="beta-binomial", \
bic_type="hbic", do_joint="myq", binary_threshold=0.6))
printf "Total execution time: %.2f sec.\n", $stopwatch
```

Run

## Empirical example no. 2: **tobit** model

```

set seed 1000000

# We load packages
include BACE.gfn








# We load tobit dataset [https://stats.oarc.ucla.edu/r/dae/tobit-models/]
open tobit.gdt --frompkg=BACE --quiet

# We specify list of explanatory variables
list X = read math proggeneral progvocational

#####
#           BACE                                     #
#####
set stopwatch
BACE(apt, X, _(model_type="tobit", high=800))
printf "Total execution time: %.2f sec.\n", $stopwatch

```

Run

-  Newton, M. A., & Raftery, A. E. (1994). Approximate Bayesian Inference with the Weighted Likelihood Bootstrap. *Journal of the Royal Statistical Society. Series B (Methodological)*, 56(1), 3–48. Retrieved January 14, 2023, from <http://www.jstor.org/stable/2346025>
-  Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A., & Rubin, D. (2013). *Bayesian Data Analysis, Third Edition*. Taylor & Francis.
-  Błażejowski, M., & Kwiatkowski, J. (2015). Bayesian Model Averaging and Jointness Measures for gretl. *Journal of Statistical Software*, 68(i05), 1–24. <https://doi.org/10.18637/jss.v068.i05>
-  Pajor, A. (2017). Estimating the Marginal Likelihood Using the Arithmetic Mean Identity. *Bayesian Analysis*, 12(1), 261–287. <https://doi.org/10.1214/16-BA1001>
-  Błażejowski, M., Kufel, P., & Kwiatkowski, J. (2020). Model simplification and variable selection: A replication of the UK inflation model by Hendry (2001). *Journal of Applied Econometrics*, 35(5), 645–652. <https://doi.org/https://doi.org/10.1002/jae.2760>
-  Clyde, M. (2022). *BAS: Bayesian Model Averaging using Bayesian Adaptive Sampling*. <https://github.com/merlisclyde/BAS>
-  Lucchetti, R., & Pedini, L. (2022). ParMA: Parallelized Bayesian Model Averaging for Generalized Linear Models. *Journal of Statistical Software*, 104(1), 1–39. <https://doi.org/10.18637/jss.v104.i02>



Raftery, A., Hoeting, J., Volinsky, C., Painter, I., & Yeung, K. (2022). *Bayesian Model Averaging – Package BMA*.

<http://CRAN.R-project.org/package=BMA>



Sevcikova, H., & Raftery, A. (2022). *mlogitBMA: Bayesian Model Averaging for Multinomial Logit Models*.

<https://cran.r-project.org/package=mlogitBMA>



Błażejowski, M., & Kwiatkowski, J. (2023). *BACE: A gretl Package for Model Averaging in Limited Dependent Variable Models* (gretl working papers No. 9). Università Politecnica delle Marche (I), Dipartimento di Scienze Economiche e Sociali. <https://ideas.repec.org/p/anc/wgretl/9.html>



Thank You for Your attention!