MMIX: An R package for model mixing methods

by Marie Morfin and David Makowski

Introduction

When several plausible models are available, the traditional approach is to use a selection method to find a single model. The selected model is then use for inference and prediction. Selection methods ignore model selection uncertainty and can lead, in some cases, to poor estimations and predictions. Model mixing (combining) provides an alternative to model selection. The basic idea is to combine all the available models (or at least several of these models) using a weighted average of the individual model estimations and predictions. Several model mixing methods have been proposed for linear and logistic regression: Bayesian Model Averaging (Raftery et al. (1997); Viallefont et al. (2001)), AIC-based mixing (Burnham and Anderson (2002)), Adaptive Regression by mixing with a model screening step (Yuan and Yang (2005); Yuan and Ghosh (2008)). These methods deal with model selection uncertainty and their implementation can lead to better parameter estimates and model predictions than model selection (e.g., Yuan and Yang (2005); Yuan and Ghosh (2008)).

In this article, we introduce the R package MMIX that implements several model-mixing methods for linear and logistic regression. This package can also be used to analyze the instability of stepwise selection method by bootstrapping and to assess the accuracy of model predictions by cross-validation.

Model mixing methods

In a linear regression model, the expected value of a response variable Y is related to a set of p explanatory variables $\mathbf{X} = (X_1, \ldots, X_p)$ as:

$$E(Y|\mathbf{X}) = \theta_0 + \theta_1 X_1 + \ldots + \theta_i X_i + \ldots + \theta_p X_p$$

In a logistic regression model, the relationship between Y and the p explanatory variables is expressed as:

$$logit(E(Y|\mathbf{X})) = \theta_0 + \theta_1 X_1 + \ldots + \theta_i X_i + \ldots + \theta_p X_p$$

The models defined above correspond to full models i.e the models including all the explanatory variables. Other models can be defined by relating Y to a subset of the p explanatory variables. The set of models corresponding to all the subsets of $\mathbf{X}$ is noted further Γ, $dim(\Gamma) = K = 2^p$.

ARMS

The Adaptive Regression by mixing with a Model Screening step (ARMS) was proposed by Yuan and Yang (2005) for the linear regression model and by Yuan and Ghosh (2008) for the logistic regression model. Giving a sample of size n, weights are calculated as follows:

1. A training dataset J_1 of size $\left[\frac{n}{2}\right] (= \frac{n}{2}$ if n is even) is randomly drawn from the original dataset.

2. A set of m best models, $\Gamma_s \subset \Gamma$, is selected using BIC or/and AIC.

3. Parameters of models $M_k \in \Gamma_s$ are estimated from J_1: $\hat{\theta}_{iJ_1}^{(k)}, i = 0, \ldots, p$, is set equal to the maximum likelihood parameter estimate if the ith explanatory variable is included in M_k, and is set to zero otherwise. For linear model, the residual error is also estimated at this step and is noted $\hat{\sigma}^{(k)}$.

4. Weights w_k of models $M_k \in \Gamma_s$ are calculated from the second part of the dataset, J_2, using the parameter values estimated in step 3. Two kinds of weights can be computed by MMIX:

- Likelihood-weights :

  linear model:
  $$B_k = L_k = \frac{1}{\sqrt{2\pi}\hat{\sigma}^{(k)}} exp\left\{-\frac{\sum_{Y_j \in J_2}[Y_j - (\hat{\theta}_{0J_1}^{(k)} + \sum_{i=1}^{p}\hat{\theta}_{iJ_1}^{(k)}X_{ij})]^2}{2\hat{\sigma}^{(k)2}}\right\}$$

  logistic model:
  $$B_k = L_k = \prod_{Y_j \in J_2} \hat{p}_j^{Y_j}(1 - \hat{p}_j)^{1-Y_j}$$
  where $\hat{p}_j = \frac{exp\{\hat{\theta}_{0J_1}^{(k)} + \sum_{i=1}^{p}\hat{\theta}_{iJ_1}^{(k)}X_{ij}\}}{1 + exp\{\hat{\theta}_{0J_1}^{(k)} + \sum_{i=1}^{p}\hat{\theta}_{iJ_1}^{(k)}X_{ij}\}}$

- AIC-weights :

  $B_k = L_k exp\{-[Q_k + 1]\}$ where Q_k is the number of explanatory variables included in M_k.

The weights of the models not included in Γ_s are set to zero and the normalized weights are finally computed as:

$$w_k = \frac{B_k}{\sum_{b=1}^{K} B_b}$$

5. The four previous steps are repeated N times and the weight of model M_k is defined by:

$$\overline{w}_k = \frac{1}{N} \sum_{l=1}^{N} w_k(l)$$

where $w_k(l)$ is the weight of model M_k computed from the lth sample.

Model weights can be used to estimate the parameters as a weighted sum over models:

$$\hat{\theta}_i = \sum_{M_k \in \Gamma_i} \overline{w}_k \hat{\theta}_i^{(k)}$$

where Γ_i is the set of the models including the ith explanatory variable, and $\hat{\theta}_i^{(k)}$ is the value of θ_i estimated for model M_k from the full dataset, $i = 0, \ldots, p$. The probability that a parameter θ_i is non zero can also be computed from the model weights as:

$$\hat{P}(\theta_i \neq 0) = \sum_{M_k \in \Gamma_i} \overline{w}_k$$

The function `arms` in the package MMIX implements this algorithm, with the following arguments: `data` : a data frame including the response variable and the explanatory variables. All the variables must be numeric and the response variable value must be 0 or 1 for the logistic model.
`family` : a description of the error distribution (gaussian or binomial).
`nsample` : number of generated samples (N).
`nbest` : number of models selected at the screening step (m).
`criterion` : selection criterion used at the screening step: "aic", "bic" or "both". "both" means that the `nbest` models according to AIC and the `nbest` model according to "bic" are selected.
`weight` : model weight type, "likeli" for likelihood-weights or "aic" for AIC-weights.
`maxVar` : maximum number of explanatory variables to include in the models. If `maxVar` $< p$, at most `maxVar` variables are selected before step 1 by a stepwise selection in the direction "forward" using the Akaike criterion.

`arms` output is a data frame including the estimated coefficients by ARMS (`coef`) and the probabilities that coefficients are different from zero (`pne0`). The function `plot` displays variable weights in a graph, and more results are available with the function `summary`. The output is also a list containing `coef`, `pne0`, the response variable predictions (`fitted.values`), a list of the explanatory variables included in each model (`label`), model weights (`modweights`) and a matrix of coefficient estimations for all the models in Γ (`allcoef`).

BIC-based mixing method

Bayesian Model Averaging was presented by Raftery et al. (1997), Hoeting et al. (1999), and Viallefont et al. (2001). The principle is to calculate the posterior distribution of the parameters $(\theta_i)_{i=0,\ldots,p}$, conditionally to the observed dataset D, but unconditionally to the model :

$$p(\theta_i | D) = \sum_{M_k \in \Gamma_i} p(\theta_i | D, M_k) p(M_k | D)$$

$$\overline{\theta}_i = E(\theta_i | D) = \sum_{M_k \in \Gamma_i} E(\theta_i | D, M_k) p(M_k | D)$$

Function `bmaBic` of MMIX implements this method by estimating the posterior probabilities of the models using the BIC approximation:

$$p(M_k | D) \approx \frac{exp(-BIC_k/2)}{\sum_{l=1}^{K} exp(-BIC_l/2)} = w_k^{BIC}$$

where BIC_k is the Bayesian Information Criterion of the model M_k. The posterior expected parameter values $E(\theta_i | D, M_k)$ are estimated by the maximum likelihood estimators $\hat{\theta}_i^{(k)}, M_k \in \Gamma_i, i = 0, \ldots, p$, and $\overline{\theta}_i$ is estimated by :

$$\hat{\theta}_i = \sum_{M_k \in \Gamma_i} w_k^{BIC} \hat{\theta}_i^{(k)}$$

The variance of $\hat{\theta}_i$ unconditioned by the model choice is estimated in `bmaBic` by:

$$\hat{Var}(\hat{\theta}_i | D) = \sum_{k=1}^{K} w_k^{BIC} [\hat{\sigma}_i^{(k)2} + (\hat{\theta}_i^{(k)} - \hat{\theta}_i)^2]$$

where $\hat{\sigma}_i^{(k)2}$ is the estimated variance of $\hat{\theta}_i^{(k)}$.

A similar Bayesian Model Averaging method can be also applied using BMA (Raftery et al. (2005)). The function `bmaBic` of MMIX has three arguments only (`data`, `family` and `maxVar`, see section ARMS). This function is more restrictive than BMA, but can be easily applied and compared to the other methods implemented in MMIX. `bmaBic` output is a data frame including the coefficient estimates (`coef`), the probabilities that coefficients are different from zero (`pne0`), and the unconditioned standard deviation of the coefficients (`sd`). The function `plot` displays variable weights in a graph, and more results are available with the function `summary`. A `bmaBic` object is also a list including `fitted.values`, `modweights`, `allcoef`, `label` (see section ARMS), and the best three models according to the BIC criterion (`BestModels`).

AIC-based mixing method

Burnham and Anderson (2002) presented a simple model averaging technique where the model weights are computed from AIC as:

$$w_k^{AIC} = \frac{exp(-AIC_k/2)}{\sum_{l=1}^{K} exp(-AIC_l/2)}$$

where w_k^{AIC} is the weight of model M_k, and AIC_k is the value of the Akaike Information Criterion for model M_k. The parameter values θ_i, $i = 0, \ldots, p$, are estimated unconditionally to the model choice, by the sum of the maximum likelihood estimators weighted by the Akaike weights:

$$\hat{\theta}_i = \sum_{M_k \in \Gamma_i} w_k^{AIC} \hat{\theta}_i^{(k)}$$

where $\hat{\theta}_i^{(k)}$ is equal to the maximum likelihood estimate obtained with the model M_k. Variance of the estimators are computed like in `bmaBic` function using the AIC weights instead of the BIC weights. `mixAic` provides the same output as `bmaBic`. Separation problems may occur with logistic regression models.

Computation time can be long with the mixing methods implemented in MMIX when the number of explanatory variables is higher than 12-15, especially with `arms` due to the resampling procedure. Computation time can be reduced by setting `maxVar`, `nsample`, and `nbest` to low values.

`bootFreq` function

The `bootFreq` function assesses the stability of stepwise selection and mixing methods by bootstrap (Prost et al. (2008)). This function generates samples from the original dataset by sampling data with replacement (Efron and Tibshirani (1993)). Stepwise selection methods and mixing methods are then applied to each sample in order to compute the frequencies of variable selection and the standard deviation of the coefficient estimates over samples. Selection frequencies close to zero or one indicate stable results whereas selection frequencies close to 0.5 indicate instability. The function `bootFreq` implements this technique with the following arguments:
`data` : a data frame including the response variable and the explanatory variables. All the variables must be numeric and the response variable value must be 0 or 1 for the logistic model.
`family` : a description of the error distribution (gaussian or binomial).
`nboot` : number of bootstrap samples drawn from the original dataset.
`method` : the statistical method used to estimate the model parameters. *method* = 1 for `fullModel`: the model is estimated by maximum likelihood without any variable selection, *method* = 2 for `stepSel`: variables are selected by a stepwise selection method, *method* = 3 for `bmaBic`, *method* = 4 for `mixAic`, *method* = 5 for `arms`.
`file` : the path of the file where the results are stored during the run. If file = NULL no file is created.
… : the specific arguments of the called method.

`bootFreq` returns an object of class "classMMIX". A data frame with the main results is printed and a graph with the weights of the explanatory variables is available with the function `plot`. A `bootFreq` object is also a list including the frequency of selection of each variable across the bootstrap samples (`frequency`), the estimated parameter values for all the bootstrap samples (`coef`), the mean of the estimated parameter values across the bootstrap samples (`mean`), the standard deviation of the estimated

parameter values across the bootstrap samples (`sd`) and `pne0`, the mean values of the variable weights computed using a model mixing method, or frequency if `method` = 1 or 2.

Performance criteria

MMIX can be used to compare the performances of stepwise selection methods and model mixing methods. Two criteria can be computed: PMSE (Predictive Mean Square Error) for linear models and AUC (Area Under Roc Curve) for logistic models. The function `pmseCV` estimates PMSE by a "leave-np-out" cross-validation, which means that np data are removed for parameter estimation but used to estimate the mean square error. This is done for all the sets of np observations (only if `np=1`), or for a number of random drawing specified in `npermu`:

$$PMSE = \frac{1}{npermu} \sum_{l=1}^{npermu} PMSE_l$$
$$= \frac{1}{npermu} \sum_{l=1}^{npermu} \frac{1}{np} \sum_{Y_j \in J_l} (Y_j - Y_j^*(l))^2$$

J_l is the sample left out and $Y_j^*(l)$ the prediction of Y_j at the lth iteration of the cross-validation procedure.

For logistic models, the function `aucCV` estimates AUC by a "leave-np-pair-out" cross-validation, which means that data are removed by pairs of individuals (one from each of the two modalities). At each iteration j the AUC is estimated with the Wilcoxon statistic by AUC_j and the final AUC is calculated as:

$$AUC = \frac{1}{npermu} \sum_{j=1}^{npermu} AUC_j$$

For these two functions, arguments are:
`data` : a data frame including the response variable (first column) and the explanatory variables. All the variables must be numeric and the response variable value must be 0 or 1 for the logistic model.
`method` : the statistical method used to estimate the model parameters. `method` = 1 for `fullModel`, `method` = 2 for `stepSel`, `method` = 3 for `bmaBic`, `method` = 4 for `mixAic`, `method` = 5 for `arms`.
`np` : number of observations (`pmseCV`) or pairs of observations (`aucCV`) left out for computing the PMSE or AUC.
`random` : observations are selected at random if TRUE. `random` can be FALSE only if `np` = 1. In this case all the possible sets are selected.
`npermu` : number of random samples of np observations if `random` = TRUE.
`file` : the path of the file where the results are stored during the run. If `file` is not NULL the `np` predictions (first column) and the `np` corresponding obser-

vations (second column) are saved at each iteration.
... : the specific arguments of the called method.

```
>read.table("tabPC.txt",header=TRUE)
>tabPC

PC   NNI    WD       PQG    NG    NV    BAF
0    0.59   105.09   1.34   141   30    0
1    0.66   77       1.07   132   0     0
0    0.27   24       1.26   33    0     1
1    0.62   13.33    0.92   84    -10   1
1    0.69   91       1.16   144   20    1
0    0.46   36       1.26   50    20    0
1    0.57   117      1.1    127   30    0
0    0.67   5        1.08   110   10    1
0    0.49   66       1.15   217   0     0
1    0.34   493.5    1.07   264   0     0
0    0.46   42.67    1.26   50    20    0
0    0.48   178.25   1.13   176   20    0
0    0.49   62.5     1.17   132   0     0
1    0.49   150      1.07   127   30    1
1    0.48   113      0.93   110   0     1
1    0.52   48       1.17   0     30    1
```

Example : Wheat grain number and grain protein content

We present here the results of an application of the package MMIX to two real datasets. Experiments were carried out in 16 organic winter wheat plots in France (Casagrande et al. (In press)) where grain number per m^2 (GN) and a binary variable indicating high (PC=1) or low (PC=0) grain protein content were measured. Nine additional variables were measured in each plot: Water Balance during the Vegetation period (WBV), Photothermal Quotient during the Vegetation period (PQV), and after flowering (PQG), Weed Density (WD), Nitrogen Nutrition Index (NNI), cultivar type (BAF), Soil Compaction (SC), Soil nitrogen before flowering (NV) and after flowering (NG). The explanatory variables considered for GN are: WBV, PQV, WD, NNI, BAF and SC. The explanatory variables considered for PC are: NNI, WD, PQG, NG, NV and BAF. Two data frames including the response variables and the explanatory variables were created; tabPC for the response variable PC and tabGN for the response variable GN.

```
>tabGN<-read.table("tabGN.txt",header=TRUE)
>tabGN

GN          WBV      PQV    WD       NNI    BAF   SC
11439.12    -8.55    1.56   105.09   0.59   0     0
17036.09    -28.94   1.38   77.00    0.66   0     0
3770.00     -2.33    0.98   24.00    0.27   1     0
9065.44     -4.81    1.43   13.33    0.62   1     0
14002.14    -21.69   1.17   91.00    0.69   1     0
15803.02    -2.98    1.11   36.00    0.46   0     1
13567.92    -16.29   1.23   117.00   0.57   0     0
6831.00     -39.95   0.80   5.00     0.67   1     1
11444.00    -24.31   0.80   66.00    0.49   0     0
7662.27     -21.44   1.25   493.50   0.34   0     0
11637.31    -3.49    1.11   42.67    0.46   0     1
12972.65    -12.76   1.20   178.25   0.48   0     0
15843.75    -22.96   1.17   62.50    0.49   0     0
6834.85     -12.25   1.25   150.00   0.49   1     0
3874.00     -6.46    0.60   113.00   0.48   1     1
7168.00     -24.16   0.80   48.00    0.52   1     0
```

Linear regression models were fitted to tabGN to relate the response variable Grain Number to the explanatory variables. Model parameters were estimated by using several methods: the full linear model fitted by maximum likelihood (*full*), stepwise selection in direction "both" using the criteria AIC and BIC (*stepA* and *stepB*), bayesian model averaging (*bma*), AIC-based mixing (*mixA*), and ARMS using both criterion AIC and BIC at the screening step, with likelihood or AIC weights (*armsL, armsA*). Estimation results are summarized in Figure 1 and Table 1. Only two variables (NNI and BAF) were selected by the stepwise selection methods. For these variables, the weights computed by the model mixing methods ranged from 0.5 to 1. A weight higher than 0.5 was also computed for WD by `mixAic`.
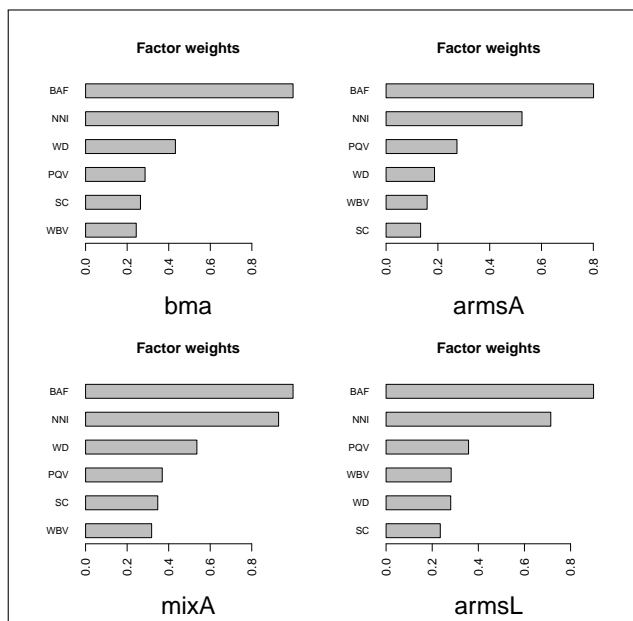
Figure 1: Weights of the six explanatory variables of the "Grain Number" dataset, estimated by four different mixing methods (`bmaBic`, `mixAic`, `arms` with likelihood and AIC weights) with linear models.

A bootstrap analysis applied to the stepwise selection methods showed that selection frequencies were close to 0.5 for several variables (Table 2). Only the variable BAF was always selected, whereas the five other variable selection frequencies varied from 0.5 to 0.7. This result shows that there is significant uncertainty about variable selection. Method performances were compared by computing PMSE with the `pmseCV` function. The most accurate predictions were obtained with `bmaBic`. Performances of `mixAic` and `arms` with likelihood weights were quite similar. The full model and the stepwise selection methods led to less accurate predictions for this dataset.

	RPMSE	$\frac{RPMSE - min(RPMSE)}{min(RPMSE)}$
full	4070.1	37%
stepA	3735.9	26%
stepB	3735.9	26%
bma	2960.4	0%
mixA	3112.1	5%
armsA	3554.9	20%
armsL	3245.1	10%

The same type of analysis was performed with the "Protein Content" dataset using logistic regression models to relate the response variable to six explanatory variables. We only present below the AUC values estimated by leave-one-pair-out cross-validation, obtained with `aucCV`, np=1 and random=FALSE. According to the AUC values, the best results were obtained with the two `arms` proce-

dures, with `bmaBic` and `mixaic`.

	AUC	$\frac{max(AUC) - AUC}{max(AUC)}$
full	0.69	13%
stepA	0.69	13%
stepB	0.69	13%
bma	0.76	4%
mixA	0.76	4%
armsA	0.77	2%
armsL	0.80	0%

Conclusion

Package MMIX provides functions for estimating the parameters of linear and logistic regression models using stepwise selection techniques and model mixing methods. It also includes functions for assessing the stability and the performance of these methods. MMIX can thus be used to compare the performances of selection and model mixing methods for different types of dataset. This package is available on CRAN.

Acknowledgments

Bibliography

K. Burnham and D. Anderson. *Model selection and multimodel inference : A Practical Information-Theoric Approach.* Springer, 2nd edition, 2002.

M. Casagrande, C. David, M. Valentin-Morison, D. Makowski, and M.-H. Jeuffroy. Factors limiting the grain protein content of organic winter wheat in south-eastern france: a mixed-model approach. *Agronomy for Sustainable Development*, In press.

B. Efron and R. Tibshirani. *An introduction to the bootstrap.* Chapman & Hall/CRC, 1993.

J. Hoeting, D. Madigan, A. Raftery, and C. Volinsky. Bayesian model averaging: a tutorial. *Statistical Science*, 14:382–417, 1999.

L. Prost, D. Makowski, and M.-H. Jeuffroy. Comparison of stepwise selection and bayesian model averaging for yield gap analysis. *Ecological Modelling*, 219:66–76, 2008.

A. Raftery, D. Madigan, and J. Hoeting. Bayesian model averaging for linear regression models. *Journal of the American Statistical Assocation*, 92:179–191, 1997.

	full	*stepA*	*stepB*	*bma*			*mixA*			*armsA*		*armsL*	
	coef	coef	coef	coef	pne0	sd	coef	pne0	sd	coef	pne0	coef	pne0
Int.	4943	3626	3626	4534	1.0	4009	4597	1.0	4071	6384	1.0	4939	1.0
WBV	-3.1	0.0	0	2.6	0.2	41.6	2.4	0.3	48	-6	0.2	-4	0.3
PQV	1882	0.0	0	727	0.3	2142.5	931	0.4	2435	1395	0.3	1479	0.4
WD	-9	0.0	0	-3.7	0.4	6.1	-4.6	0.5	6.5	-2	0.2	-3	0.3
NNI	14226	18673	18673	16283	0.9	7605	15950	0.9	7766	9369	0.5	12802	0.7
BAF	-6283	-6239	-6239	-6244	1.0	1359	-6255	1	1370	-4807	0.8	-5482	0.9
SC	-929	0	0.0	-273	0.3	932	-364	0.3	1074	-113	0.1	-189	0.2

Table 1: Results of the estimation methods (full model, stepwise using AIC, stepwise using BIC, BMA, AIC based mixing model, ARMS using AIC weights, ARMS using likelihood weights), applied to the "Grain Number" dataset with linear models.

	stepA			*stepB*		
	mean	sd	frequency	mean	sd	frequency
Int.	-859.8	118539.7	1.0	6227.2	28931.7	1.0
WBV	-25.2	679.2	0.6	-47.7	304.8	0.5
PQV	4914.5	69988.2	0.6	1223.8	22305.2	0.6
WD	-6.3	29.7	0.7	-9.4	32.9	0.6
NNI	17697.2	104793.9	0.7	11910.0	25173.3	0.7
BAF	-4656.0	23613.1	1.0	-5971.5	9201.8	1.0
SC	-366.5	6854.2	0.6	-484.3	10393.6	0.5

Table 2: Results obtained from 500 bootstrap samples for two stepwise selection methods (using AIC and BIC), applied to the "Grain Number" dataset with linear models.

```
M1 <- fullModel(data=tabGN, family=gaussian("identity"))
M2 <- stepSel(data=tabGN, family=gaussian("identity"), direction="both", criterion="aic")
M3 <- stepSel(data=tabGN, family=gaussian("identity"), direction="both", criterion="bic")
M4 <- bmaBic(data=tabGN, family=gaussian("identity"))
M5 <- mixAic(data=tabGN, family=gaussian("identity"))
M6 <- arms(data=tabGN, family=gaussian("identity"), nbest=40, criterion="both", weight="aic")
M7 <- arms(data=tabGN, family=gaussian("identity"), nbest=40, criterion="both", weight="likeli")

B2 <- bootFreq(data=tabGN, family=gaussian('identity'), nboot=500, method=2, criterion='aic', trace=0)
B3 <- bootFreq(data=tabGN, family=gaussian('identity'), nboot=500, method=2, criterion='bic',trace=0)

par(mfcol=c(2,2))
plot(M4)
title(sub="bma")
plot(M5)
title(sub="mixaic")
plot(M6)
title(sub="arms aic")
plot(M7)
title(sub="arms likeli")

P1 <- pmseCV(data=tabGN, method=1, np=1, random=F)
P2 < -pmseCV(data=tabGN, method=2,np=1,criterion="aic", random=F, trace=0)
P3 < -pmseCV(data=tabGN, method=2,np=1, criterion="bic", random=F)
P4 <- pmseCV (data=tabGN, method=3, random=F,np=1)
P5 <- pmseCV (data=tabGN, method=4, random=F,np=1)
P6 <- pmseCV (data=tabGN, method=5, weight="aic", nbest=40, nsample=20, random=F, np=1)
P7 <- pmseCV (data=tabGN, method=5, weight="likeli", nbest=40, nsample=20, random=F, np=1)
rpmseCv <-sqrt(rbind(P1, P2, P3, P4, P5, P6, P7)[,1])
```

Figure 2: Commands used for the "Grain Number" case study: implementation of the five estimation methods, plot of the variable weights, bootstrap analysis for stepwise selection methods, and PMSE estimations.

A. Raftery, I. Painter, and C. Volinsky. Bma : An r package for bayesian model averaging. *R News*, 5: 2–8, 2005.

V. Viallefont, A. Raftery, and S. Richardson. Variable selection and bayesian model averaging in case-control studies. *Statistics in medicine*, 20:3215–3230, 2001.

Z. Yuan and D. Ghosh. Combining multiple biomarker models in logistic regression. *Biometrics*, 64:431–439, 2008.

Z. Yuan and Y. Yang. Combing linear regression models: When and how? *Journal of the American Statistical Association*, 100:1202–1214, 2005.

Marie Morfin, David Makowski
UMR 211 Agronomie INRA/AgroParisTech
Thiverval-Grignon, France
UMR 518 Mathématiques et Informatiques Appliquées
INRA/AgroParisTech
Paris, France
`makowski@grignon.inra.fr`