

MatSAM

Version 2Beta

Stéphane Joost
Michael Kalbermatten
EPFL-ENAC-LASIG, March 2010

Table of Contents

Introduction.....	1
Logistic regression.....	1
Probability distribution.....	1
Generalized regression model.....	2
Univariate model.....	2
Operational framework.....	3
Ratios of regression sum-of-squares.....	4
Efron's pseudo R2.....	4
McFadden's pseudo R2.....	4
McFadden's adjusted pseudo R2.....	4
Cox & Snell's pseudo	5
Nagelkerke / Cragg & Uhler's pseudo R2.....	5
AIC - information criterion.....	5
BIC - Bayesian information criterion.....	5
Remarks concerning the number of parameters.....	6
Significance of the model - Goodness of fit.....	6
Likelihood ratio chi-squared statistic (not implemented!).....	6
G test.....	7
Wald test.....	7
Remark concerning the G and Wald tests.....	7
Score test (not implemented!).....	8
Confidence interval (not implemented!).....	8
Design variables.....	8
Referenced variables.....	8
Symmetrical variables.....	9
Independent variables.....	10
Remark regarding design variables:.....	10
MatSAM.....	11
Required Data.....	11
Software components.....	11
Input matrix.....	12
Input parameters.....	12
Designing your data.....	13
Nominal predictive variables.....	14
Ordinal predictive variables.....	15
Missing values.....	16
Final verifications.....	17
Defining the parameter file.....	17
How to proceed.....	17
MatSAM results and output.....	18
Result analysis.....	18
Structure of the table.....	18
Dynamic table of analysis.....	19
Adapting significance level.....	20
Pseudo R2.....	20
Graphics ("graphics.txt").....	20
Graphics created using MatSAM.....	20
References.....	22

Introduction

This document introduces the logistic regression as it is currently implemented in MatSAM. It helps to understand the mathematical fundamentals which are linked to MatSAM, as well as the underlying tests, coefficients and statistical analysis.

Logistic regression

The logistic regression uses random binomial variables as response for the model. It tries to explain them using all kinds of predictive variables having different typologies (discrete or continuous).

Probability distribution

Let z be a random response variable, which models the success or failure of a process (or it might be the presence or absence of a phenomenon).

$$z = \begin{cases} 1 & \text{success} \\ 0 & \text{failure} \end{cases} \text{ with the probability } Pr(Z=1)=\pi, Pr(Z=0)=1-\pi$$

If we have n random variables Z_1, \dots, Z_n which are independent, then the joint probability is:

$$\prod_{i=1}^n \pi_i^{z_i} (1-\pi_i)^{1-z_i} = \exp \left[\sum_{i=1}^n \log \left(\frac{\pi_i}{1-\pi_i} \right) + \sum_{i=1}^n \log(1-\pi_i) \right]$$

n represents the number of trials.

If the π_i are equal, then $Y = \sum^n Z_i$ where Y is the number of success over n trials (Y is the realization) with $Y_i \sim B(n_i, \pi_i)$:

$$Pr(Y=y) = \binom{n}{y} \pi^y (1-\pi)^{n-y}, \quad y=0, \dots, n$$

If $Y_i \sim B(n_i, \pi_i)$, then the maximum log-likelihood is:

$$l(\pi_1, \dots, \pi_N, y_1, \dots, y_N) = \sum_{i=1}^N \left[y_i \log \left(\frac{\pi_i}{1-\pi_i} \right) + n_i \log(1-\pi_i) + \log \binom{n_i}{y_i} \right] \quad (1)$$

with N the number of sub-groups (or the number of markers taken into account).

The frequencies for N binomial distributions are given by:

	1	2	...	N
Success	Y_1	Y_2	...	Y_N
Failure	$n_1 - Y_1$	$n_2 - Y_2$...	$n_N - Y_N$
Totals	n_1	n_2	...	n_N

Generalized regression model

It is necessary to describe the success probabilities $P_i = Y_i/n_i$ of each sub-group. The expected value of the probability is $E(P_i) = \pi_i$ and these are modeled by:

$$g(\pi_i) = \mathbf{x}_i^T \boldsymbol{\beta}$$

where \mathbf{x}_i is a vector of variables, $\boldsymbol{\beta}$ is the parameter vector and g is the link function.

If we use a logistic transformation, g is the *logit* function:

$$\text{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \mathbf{x}_i^T \boldsymbol{\beta}$$

The estimation of the parameters $\boldsymbol{\beta}$ is done by using the maximum likelihood (cf. equation (1)). This one is described by:

$$l(\boldsymbol{\pi}, \mathbf{y}) = \sum_{i=1}^N \left[y_i \log(\pi_i) + (n_i - y_i) \log(1 - \pi_i) + \log\binom{n_i}{y_i} \right]$$

The deviance of the model is given by:

$$D = 2 \sum_{i=1}^N \left[y_i \log\left(\frac{y_i}{\hat{y}_i}\right) + (n_i - y_i) \log\left(\frac{n_i - y_i}{n_i - \hat{y}_i}\right) \right] = 2 \sum_{i=1}^N o \log \frac{o}{e}$$

where

- o indicate the success of y_i
- e indicates the estimated frequencies

Deviance enables to estimate the adjustment (goodness of fit) using $D \sim X^2(N - m)$, m being the number of parameters (or x_i).

Univariate model

If we use the univariate case, the model is the following:

$$\text{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 x_i + \epsilon_i \quad \text{with} \quad \pi(x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}$$

By comparison with a simple linear regression model, following differences can be illustrated:

Linear model	Logistic model
$E(Y x) = \beta_0 + \beta_1 x, \quad x \in [-\infty, \infty]$	$E(Y x) = \pi(x)$ $\pi(x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}$ $g(x) = \log\left(\frac{\pi(x)}{1 - \pi(x)}\right)$

$y = E(Y x) + \epsilon$ where ϵ follows a normal distribution, with zero mean and a constant standard deviation.	$y = \pi(x) + \epsilon$ if $y = 1$, $\epsilon = 1 - \pi(x)$ and if $y = 0$, $\epsilon = -\pi(x)$, where ϵ has zero mean and its standard deviation is $\sigma_\epsilon = \pi(x)[1 - \pi(x)]$.
---	--

Operational framework

We have n independent observations of pairs (x_i, y_i) , $i = 1, \dots, n$ where:

- y_i : dichotomous response value $y_i \in \{0, 1\}$ (absence/presence)
- x_i : value of independent variable i
- β_0, β_1 : unknown parameters

β_0 and β_1 are estimated using the maximum likelihood. It consists to maximize the probability of obtaining a data set. The first step consists of constructing the function which has to be maximized (called "maximum likelihood function), i.e. in order for the probability of $Y = 1$ to be $Pr(Y = 1|x)$:

$$l(\boldsymbol{\beta}) = \prod_{i=1}^n \pi(x_i)^{y_i} [1 - \pi(x_i)]^{1-y_i}$$

Consequently, it is necessary to maximize this function. Because of the product, it is simpler to maximize the logarithm of the function, thus the maximum log-likelihood:

$$L(\boldsymbol{\beta}) = \log[l(\boldsymbol{\beta})] = \sum_{i=1}^n [y_i \log(\pi(x_i)) + (1 - y_i) \log(1 - \pi(x_i))]$$

It is necessary to use the second derivatives (in order to take into account) to resolve the system. That is to say that a solution for the next two non-linear equations has to be found:

$$\sum_{i=1}^n [y_i - \pi(x_i)] = 0 \quad (2)$$

$$\sum_{i=1}^n x_i [y_i - \pi(x_i)] = 0$$

MatSAM uses the IRLS (Iteratively reweighted least squares) algorithm, and it is an iteration of the computed $\boldsymbol{\beta}$. The iterations stop when the difference of the $\boldsymbol{\beta}$ between two iterations are smaller than a certain threshold.

Furthermore, we have to maximize the maximum likelihoods of $\hat{\boldsymbol{\beta}}$ and $\hat{\pi}(x_i)$ for $\boldsymbol{\beta}$ and $\pi(x_i)$.

The consequence of equation (2) is the definition of $\sum_{i=1}^n y_i = \sum_{i=1}^n \hat{\pi}(x_i)$.

As warning, we have to notice that the algorithm convergence is not assured!

Ratios of regression sum-of-squares

On the contrary of a simple linear regression, there are multiple ways to compute sum-of-squares ratios (R^2). In MatSAM, some of them were implemented (not exhaustive). If you use them to analyze the MatSAM results, please consider the following recommendations:

1. There is not any universal pseudo coefficient of determination.
2. Most of the Pseudo- R^2 have values outside of the range of $[0, 1]$. They are not completely standardized.
3. 2 or 3 Pseudo- R^2 (based on different criteria) have to be used to assess the model fit.

Most of the theoretical background and explanations hereunder can be found on the Internet¹.

Efron's pseudo R^2

This coefficient is based on the probability estimations. It is defined by:

$$EfronR^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{\pi}(x_i))^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad \text{with} \quad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$$

This coefficient has the tendency to underestimate the real R^2 .

McFadden's pseudo R^2

This coefficient is based on the maximum log-likelihood. It is defined by:

$$McfaddenR^2 = 1 - \left[\frac{\log(L)}{\log(L_2)} \right]$$

where:

- $\log(L)$: maximum log-likelihood (with variables)
- $\log(L_2)$: maximum log-likelihood (without variables)

The theoretical range of the coefficient is $0 \leq McfaddenR^2 \leq 1$

A thumb rule is that the adjustment of the model is excellent if $0.2 \leq McfaddenR^2 \leq 0.4$.

This coefficient has the tendency to underestimate the real R^2 .

McFadden's adjusted pseudo R^2

This coefficient is based on the maximum log-likelihood. It is defined by:

$$McfaddenR^2_{adjusted} = 1 - \frac{(\log(L) - m)}{\log(L_2)}$$

Compared to McFadden's usual coefficient, this one penalizes a model for including too many predictors. If the predictors in the model are effective, then the penalty will be small regarding the added information of

¹ http://www.ats.ucla.edu/stat/mult_pkg/faq/general/Psuedo_RSquareds.htm, accessed March 2010
<http://www.soziologie.uni-halle.de/langer/pdf/papers/rc33langer.pdf>, accessed March 2010

the predictors. However, if a model contains predictors that do not add sufficiently explanation to the model, then the penalty becomes noticeable and the adjusted coefficients decrease by addition of a predictor, even if the coefficient increases slightly.

Cox & Snell's pseudo

This coefficient is based on the maximum log-likelihood. It is defined by:

$$CoxR^2 = 1 - \left(\frac{\exp(\log(L2))}{\exp(\log(L))} \right)^{2/n}$$

The theoretical range of the coefficient is $0 \leq CoxR^2 \leq 1 - \exp(\log(L))^{2/n}$.

Nagelkerke / Cragg & Uhler's pseudo R²

This coefficient is based on the maximum log-likelihood. It is defined by:

$$CraggR^2 = \frac{\exp(\log(L))^{2/n} - \exp(\log(L2))^{2/n}}{1 - \exp(\log(L2))^{2/n}}$$

The theoretical range of the coefficient is $0 \leq CraggR^2 \leq 1$

AIC - information criterion

The Akaike Information Criterion (AIC) is a goodness of fit statistic. It is based on the log-likelihood function with adjustment regarding the number of estimated parameters.

$$AIC = -2 \log(L) + 2m \text{ with } m \text{ the number of parameters}$$

Akaike's Information Criterion provides a measure of model quality by simulating the situation where the model is tested on a different data set. After computing several different models, you can compare them using this criterion. According to Akaike's theory, the most accurate model has the smallest AIC. An individual AIC value is meaningless, but their differences are meaningful².

BIC - Bayesian information criterion

The Bayesian Information Criterion (BIC) is a goodness of fit statistic. It is based on the log-likelihood function with adjustment regarding the number of estimated parameters and the amount of data.

$$BIC = -2 \log(L) + 2m \log(n)$$

BIC imposes a greater penalty for additional parameters than does AIC. Thus, BIC always provides a given model with a number of parameters not greater than that chosen by AIC.

This unexplained variation in the dependent variable and the number of explanatory variables increase the value of BIC. Hence, lower BIC implies either fewer explanatory variables, better fit, or both. The BIC penalizes free parameters more strongly than does the AIC³.

2 http://en.wikipedia.org/wiki/Akaike_information_criterion , accessed March 2010

3 http://en.wikipedia.org/wiki/Bayesian_information_criterion , accessed March 2010

Remarks concerning the number of parameters

Most of the goodness of fit coefficients are dependent on the number of estimated parameters. This has a strong influence on the convergence of the model. Furthermore, it also influences the tests (see below) which are undertaken to prove the significance of a response variable.

There is no absolute answer to the limit of parameters, but one has to remember that an increase of parameters often means an increase of the parameter variance. Thus, a balance has to be found between the number of parameters to be adjusted and the expected variance.

Using least square always gives a solution for overparametrized models, but for maximum log-likelihood adjustments, the algorithm will often not converge when using too many parameters, because the variance of these will explode and likelihood tends to infinity.

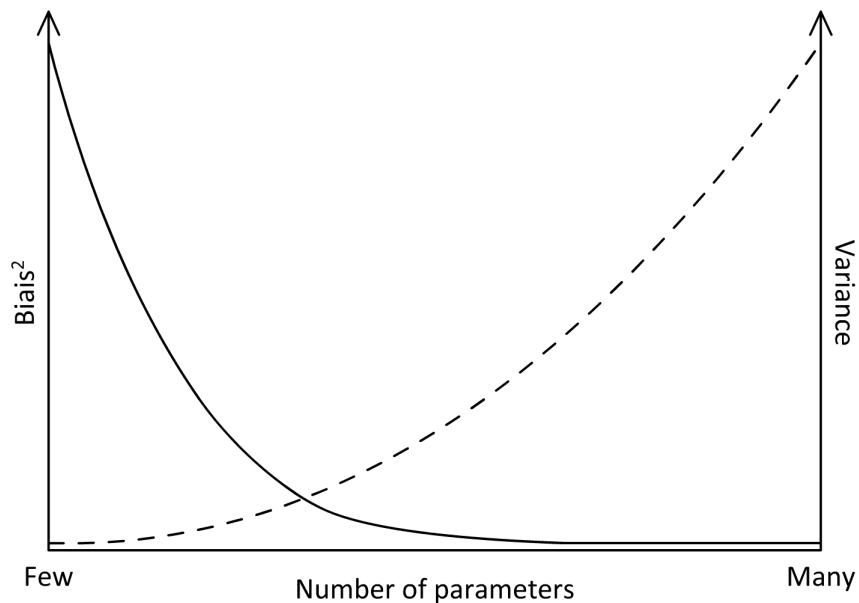


Figure: The principle of parsimony: the conceptual tradeoff between squared bias⁴ (solid line) and variance versus the number of estimable parameters in a model⁵.

Significance of the model - Goodness of fit

Likelihood ratio chi-squared statistic (not implemented!)

It is the maximum likelihood ratio (with a minimal model):

$$C = 2 \sum_{i=1}^n \left[y_i \log \left(\frac{\hat{y}_i}{n \tilde{\pi}_i} \right) + (n_i - y_i) \log \left(\frac{n_i - \hat{y}_i}{n_i - n_i \tilde{\pi}_i} \right) \right]$$

where the minimal model $\tilde{\pi}_i$ is defined by:

$$\tilde{\pi} = \frac{\sum_{i=1}^n y_i}{\sum_{i=1}^n n_i}$$

⁴ The bias is the difference between the adjusted values of the parameters and their true value.

⁵ <http://www2.fmg.uva.nl/modelselection/presentations/AWMS2004-Burnham.pdf> , accessed March 2010
http://en.wikipedia.org/wiki/Occam%27s_razor#Science_and_the_scientific_method , accessed March 2010

G test

The G test is a comparison between our model using the defined parameters and a model containing only an intercept (thus computed using only β_0):

$$G = -2 \log \left[\frac{(\text{likelihood without variables})}{(\text{likelihood with variables})} \right]$$

If $n_1 = \sum_{i=1}^n y_i$ and $n_0 = \sum_{i=1}^n (1 - y_i)$ then:

$$G = -2 \log \left[\frac{\left(\frac{n_1}{n} \right)^{n_1} \left(\frac{n_0}{n} \right)^{n_0}}{\prod_{i=1}^n \hat{\pi}_i^{y_i} (1 - \hat{\pi}_i)^{(1 - y_i)}} \right]$$

Finally:

$$G = 2 \left\{ \sum_{i=1}^n [y_i \log(\hat{\pi}_i) + (1 - y_i) \log(1 - \hat{\pi}_i)] - [n_1 \log(n_1) + n_0 \log(n_0) - n \log(n)] \right\}$$

Under hypothesis that $\beta_1 = 0$, G follows a X^2 distribution with 1 degree of freedom.

Wald test

The Wald test compares the different β_i , ($i \neq 0$) to an estimation of their error.

If the regression is univariate ($\pi(x_i) = \beta_0 + \beta_1 \cdot x_i + \epsilon_i$), then the Wald test consists to compare $\hat{\beta}_1$ to its estimated error:

$$W = \frac{\hat{\beta}_1}{SE(\hat{\beta}_1)}$$

If the regression is multivariate, then the Wald test follows a chi-square distribution with $m + 1$ degrees of freedom and under the hypothesis that each of $m + 1$ parameters is equal to zero. It is computed using:

$$W = \hat{\beta}^T [Var(\hat{\beta})]^{-1} \hat{\beta}$$

Remark concerning the G and Wald tests

These tests rely on the fact that the maximum likelihood is computable. Regarding the different values of the β_i , it might tend to infinity, thus not give any solution for the tests.

Score test (not implemented!)

The score test is based, using a simple computation on the frequencies, on the theoretical distribution of the derivation of the maximum log-likelihood. For the univariate case, it is computed the following way:

$$ST = \frac{\sum_{i=1}^n x_i (y_i - \bar{y})}{\sqrt{\bar{y}(1-\bar{y}) \sum_{i=1}^n (x_i - \bar{x})^2}}$$

The multivariate case is currently not presented, because it implies direct derivation of the maximum log-likelihood.

Confidence interval (not implemented!)

For each β_i , it is possible to compute a confidence interval. It is defined by its extreme values $100(1-\alpha)$. α is the type I error. For example and for the univariate model, this means that:

- $\hat{\beta}_1 \pm z_{(1-\alpha/2)} \hat{SE}(\hat{\beta}_1)$
- $\hat{\beta}_0 \pm z_{(1-\alpha/2)} \hat{SE}(\hat{\beta}_0)$

It is also possible to compute a confidence interval for the *logit* :

$$\hat{Var}[\hat{g}(x)] = \hat{Var}(\hat{\beta}_0) + x^2 \hat{Var}(\hat{\beta}_1) + 2x \hat{Cov}(\hat{\beta}_0, \hat{\beta}_1)$$

The confidence interval based on the Wald test gives: $\hat{g}(x) \pm z_{(1-\alpha/2)} \hat{SE}(\hat{g}(x))$

Design variables

For continuous variables, the model contains parameters (β) which represent the change in the response (y) corresponding to a change of x (predictive variable). For categorical predictive variables, the parameters will represent the different levels of classes. Thus the x are chosen to exclude or include a parameter for each observation. They are called design or dummy variables.

At least $m-1$ design variables will have to be defined for m categories, groups or classes. The symmetrical case (see hereunder) is the only case which will need less design variables.

Referenced variables

A group has to be set as the reference group. For m groups, one is defined as the reference and the other groups are simply an increase of the expected value.

$$\begin{aligned} E(Y_1) &= \mu \\ E(Y_2) &= \mu + \alpha_1 \\ &\dots \\ E(Y_m) &= \mu + \dots + \alpha_{m-1} \end{aligned}$$

Design variables enable such an implementation by defining new values following ($m \times m$ matrix):

$$\begin{aligned} \text{Group 1:} & \quad [1 \ 0 \ \dots \ 0] \\ \text{Group 2:} & \quad [1 \ 1 \ \dots \ 0] \end{aligned}$$

$$\dots : \quad [1 \dots 10 \dots 0]$$

$$\text{Group } m : \quad [1 \dots 1]$$

Symmetrical variables

This time, the m groups are treated symmetrically. That is to say, it is necessary to define a central group around which the symmetry is distributed:

$$E(Y_1) = \mu$$

$$E(Y_2) = \mu + \alpha_1$$

$$\dots$$

$$E(Y_m) = \mu - \alpha_1 - \dots - \alpha_u$$

In this case, we will need $\lfloor m/2 \rfloor$ variables to express the relations between the groups ($m \times \lfloor m/2 \rfloor$ matrix):

$$\text{Group 1:} \quad [1 \quad 0 \quad \dots \quad 0]$$

$$\text{Group 2:} \quad [1 \quad 1 \quad 0 \quad \dots \quad 0]$$

$$\dots$$

$$\text{Group } m : \quad [1 \quad -1 \quad \dots \quad -1]$$

Example with 5 groups (like "very bad", "bad", "average", "good", "very good"):

In this example, the "average" group is the central group. The other groups are distributed around it.

Thus:

$$E(Y_{\text{very bad}}) = \mu - \alpha_1 - \alpha_2$$

$$E(Y_{\text{bad}}) = \mu - \alpha_1$$

$$E(Y_{\text{average}}) = \mu$$

$$E(Y_{\text{good}}) = \mu + \alpha_1$$

$$E(Y_{\text{very good}}) = \mu + \alpha_1 + \alpha_2$$

And the matrix is:

$$\text{Group "very bad":} \quad [1 \quad -1 \quad -1]$$

$$\text{Group "bad":} \quad [1 \quad -1 \quad 0]$$

$$\text{Group "average":} \quad [1 \quad 0 \quad 0]$$

$$\text{Group "good":} \quad [1 \quad 1 \quad 0]$$

$$\text{Group "very good":} \quad [1 \quad 1 \quad 1]$$

Conceptually, μ represents the overall average effect and α_i the group differences. The sum of expected values is and has to be null:

$$[E(Y_{\text{very bad}}) - \mu] + [E(Y_{\text{bad}}) - \mu] + [E(Y_{\text{average}}) - \mu] + [E(Y_{\text{good}}) - \mu] + [E(Y_{\text{very good}}) - \mu] =$$

$$-\alpha_2 - \alpha_1 - \alpha_1 + \alpha_1 + \alpha_1 + \alpha_2 = 0$$

Independent variables

Each group is independent of the other m groups. Furthermore, it means that there is no intercept in the regression model:

$$\begin{aligned} E(Y_1) &= \alpha_1 \\ E(Y_2) &= \alpha_2 \\ &\dots \\ E(Y_m) &= \alpha_m \end{aligned}$$

Design variables enable such an implementation by defining new values following ($m \times m$ matrix), thus an identity matrix:

$$\begin{array}{l} \text{Group 1:} \\ \text{Group 2:} \\ \dots : \\ \text{Group } m : \end{array} \begin{array}{l} [1 \ 0 \ \dots \ 0] \\ [0 \ 1 \ \dots \ 0] \\ [0 \ \dots \ 0 \ 1 \ 0 \ \dots \ 0] \\ [0 \ \dots \ 1] \end{array}$$

Remark regarding design variables:

The model is not anymore univariate. It can only be multivariate.

MatSAM

This short notice is intended to MatSAM v2Beta users. It gives an introduction on how to configure your data, use MatSAM v2Beta and it gives hints on how to interpret results.

The detection of adaptive loci in the genome is important as it supports the understanding of what proportion of the genome is shaped by natural selection. It also gives the possibility to identify regions of the genome involved in adaptation processes. Several methods were developed to detect loci under selection (see a review in Joost et al. 2007), but the uncovering of environmental parameters responsible for selection was so far difficult to realize. This is precisely a task MatSAM is able to fulfill.

All the examples presented in the current document use the example data file. This one can be downloaded in the download area.

Required Data

The method the SAM software implements is described in details in Joost et al. (2007). Geographic coordinates of the place where the animal/ plant is sampled are necessary. They permit to retrieve environmental information to characterize the sampling location. Required data are:

1. At least one environmental variable describing the sampling location;
2. A matrix with the presence (1) or the absence (0) of a given molecular marker at the sampling location.

The molecular data sets used for analysis are in the form of matrices; each row of the matrix corresponds to a sampled individual, while the columns are organized according to the sampled individual's geographic coordinates and contain binary information (1 or 0), relating to the status of the genetic marker. For AFLP markers, the numbers 1 or 0 respectively indicate the phenotypes « presence of band » and « absence of band ». For microsatellite markers, the numbers 1 and 0 respectively, indicate the presence or absence of a given allele at the locus in question. The method was also recently successfully applied to SNPs (Pariset, Joost, Ajmone and Valentini 2009). For microsatellites and SNPs, an encoding phase is necessary, while AFLP data are ideal for logistic regression because they provide binomial information.

Software components

To use MatSAM, you will need two main components:

- The Matlab Component Runtime v. 7.9 (261Mb);
- The MatSAM v2Beta zip file (0.7Mb).

Furthermore, you might want to have an example of input files:

- Example zip file.

The MatSAM v2Beta zip file contains a Windows executable file (tested only on XP) containing the main procedure, processing of many simultaneous logistic regression models, based on the GLMfit function, see MacCullagh & Nelder (1989).

The main component "matsam_v2beta.exe" was developed with Matlab ©1994-2010 The MathWorks Inc. The advantage of Matlab is that it is really fast and efficient to simultaneously process an important number of models. The present version of matSAM is compiled and can be run without having to purchase Matlab.

But it still requires a few Matlab components (mainly libraries), which are made available by the Matlab Component Runtime. The drawback is that this component is heavy (261Mb). But it can be freely distributed in a non profit perspective, for academic use.

Input matrix

During the preparation of the input matrix, it is important to sort out both environmental variables and genetic markers according to any criteria, and then to number the environmental variables and the markers according to this criteria. Typically a good idea is to sort out and number markers according to their frequency among sampled animals or plants, and to produce a matrix of genetic markers with low frequencies on the left and high frequencies on the right. For environmental variables, it can be the thematic order or the alphabetical order.

The input matrix has to be a text file (.txt, .csv or .dat), delimited with spaces. The initial row (title line with the name of the environmental variables and the name of the loci or alleles) **MUST BE LEFT** in the file! The name of the column of this first row must **NOT** contain any space or special character (replace spaces with an underscore).

VERIFY after table export from Excel that the header row does not contain any quotes (like " or ').

The initial column (name of animals or samples) has to be removed to be processed by the "matsam_v2beta.exe" program.

The section "Designing your data" gives an example of data design in the input matrix.

Input parameters

You will need to define at least three input parameters. If you have special (nominal or cardinal) predictors, you will need to define 5 parameters. Each line of the parameter file contains a certain parameter, which is:

1. Number of environmental variables;
2. Number of genetic markers;
3. Create graphics (0 = do not create / 1 = create). Creating graphics will extend the processing time. Moreover, it will create a sub-directory "graphics" containing a graphic for each environmental variable versus each marker.

This is only valid for non-nominal and non-ordinal environmental variables. Otherwise and as the regression is multivariate, a graphic does not make sense.

If using 1 (create graphics), please be aware not to touch anything on your screen. MatSAM is sensitive to the position of your windows on the screen. Thus, once you started MatSAM, do not touch or move any window on your screen, but wait until the end of the computation.

4. Column number of the nominal or cardinal predictors (separated using a space);
5. Type of recoding that has to be undertaken for the nominal or cardinal predictors. At the present, MatSAM implements three different methods: reference recoding (code: r), symmetrical recoding (code: s) and independent recoding (code: i). Thus, to specify which recoding method that you want to apply, use the letters in brackets for the optimal recoding (r, s or i) separated by a space.

Look at the "Designing your data" section for an example of data design using different predictors.

Designing your data

Different data types (nominal, ordinal or cardinal) might be used. For nominal and ordinal data, the input values have to be designed specifically regarding the used method (see hereunder).

As example, we will use the given example file. The non-formatted version of this file is composed of the following columns:

- farmid: an identification code which is unique to each farm in which an animal was sampled;
- animal: a unique identification code for each animal;
- location: a nominal variable which describe the type of habitat where the animal lives. This variable has 7 distinct classes, which are nominal;
- tmp_class: a temperature class which describes the habitat of each animal. This variable has 5 distinct classes, which are ordinal, but there is no indication of the class interval;
- Longitude: the longitude of the sampled animal;
- latitude: the latitude of the sampled animal;
- altitude: the altitude at which the animal lives;
- ph_fao: pH soil classes. This variable is made of 6 classes, which are ordinal. Moreover, the class interval is given for each class;
- sunyear: percent of maximum possible sunshine (percent of day length), annual average;
- pryear: precipitations in mm, annual average;
- wndyear: windspeed in m/s 10 meters above the ground, annual average;
- E32_T38_3a to E45_T32_39: genetic markers (1=presence / 0=absence).

The file looks like the above figure:

	A	B	C	D	E	F	G	H	I	J	K	L	DF	DG	DH
1	farmid	animal	location	tmp_class	longitude	latitude	altitude	ph_fao	sunyear	pryear	wndyear	E32_T38_3a	E45_T32_37	E45_T32_38	E45_T32_39
2	AL-0027	CHALMUZ23	Mediterranean Basin	Temperateness	20.21	40.12	660	3	53.91	107.28	1.97	0	1	1	0
3	AL-0027	CHALMUZ24	Mediterranean Basin	Temperateness	20.21	40.12	660	3	53.91	107.28	1.97	0	1	1	0
4	AL-0027	CHALMUZ25	Mediterranean Basin	Temperateness	20.21	40.12	660	3	53.91	107.28	1.97	0	1	1	1
5	AL-0028	CHALMUZ26	Mediterranean Basin	Temperateness	20.21	40.11	684	3	53.91	107.28	1.97	0	1	0	0
6	AL-0028	CHALMUZ27	Mediterranean Basin	Temperateness	20.21	40.11	684	3	53.91	107.28	1.97	0	1	0	0
7	AL-0028	CHALMUZ28	Mediterranean Basin	Temperateness	20.21	40.11	684	3	53.91	107.28	1.97	0	1	0	0
8	AL-0029	CHALMUZ31	Mediterranean Basin	Temperateness	20.21	40.1	396	3	53.91	107.28	1.97	0	1	0	0
9	AL-0030	CHALMUZ32	Mediterranean Basin	Temperateness	20.22	40.09	427	3	53.91	107.28	1.97	0	1	1	0
10	AL-0030	CHALMUZ33	Mediterranean Basin	Temperateness	20.22	40.09	427	3	53.91	107.28	1.97	0	1	1	1
11	AL-0031	CHALMUZ36	Mediterranean Basin	Temperateness	20.19	40.07	197	3	53.91	107.28	1.97	0	1	1	0
12	AL-0031	CHALMUZ37	Mediterranean Basin	Temperateness	20.19	40.07	197	3	53.91	107.28	1.97	0	1	1	0
13	AL-0031	CHALMUZ38	Mediterranean Basin	Temperateness	20.19	40.07	197	3	53.91	107.28	1.97	0	1	1	0
14	AL-0032	CHALCAP1	Mediterranean Basin	Temperateness	20.73	40.9	705	4	49.78	66.08	2.07	0	1	1	1
15	AL-0032	CHALCAP2	Mediterranean Basin	Temperateness	20.73	40.9	705	4	49.78	66.08	2.07	0	1	1	0
16	AL-0032	CHALCAP3	Mediterranean Basin	Temperateness	20.73	40.9	705	4	49.78	66.08	2.07	0	1	1	1

The first step is to delete the columns MatSAM will not be able to use. This concerns columns "farmid" and "animal".

	A	B	C	D	E	F	G	H	I	J	K	L	DF	DG	DH
1	location	tmp_class	longitude	latitude	altitude	ph_fao	sunyear	pryear	wndyear	E32_T38_3a	E32_T38_4	E32_T38_4	E45_T32_39		
2	Mediterranean Basin	Temperateness	20.21	40.12	660	3	53.91	107.28	1.97	0	0	0	0		
3	Mediterranean Basin	Temperateness	20.21	40.12	660	3	53.91	107.28	1.97	0	0	0	0		
4	Mediterranean Basin	Temperateness	20.21	40.12	660	3	53.91	107.28	1.97	0	0	0	1		
5	Mediterranean Basin	Temperateness	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0		
6	Mediterranean Basin	Temperateness	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0		
7	Mediterranean Basin	Temperateness	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0		
8	Mediterranean Basin	Temperateness	20.21	40.1	396	3	53.91	107.28	1.97	0	0	0	0		
9	Mediterranean Basin	Temperateness	20.22	40.09	427	3	53.91	107.28	1.97	0	0	0	0		
10	Mediterranean Basin	Temperateness	20.22	40.09	427	3	53.91	107.28	1.97	0	0	0	1		
11	Mediterranean Basin	Temperateness	20.19	40.07	197	3	53.91	107.28	1.97	0	0	0	0		
12	Mediterranean Basin	Temperateness	20.19	40.07	197	3	53.91	107.28	1.97	0	0	0	0		
13	Mediterranean Basin	Temperateness	20.19	40.07	197	3	53.91	107.28	1.97	0	0	0	0		
14	Mediterranean Basin	Temperateness	20.73	40.9	705	4	49.78	66.08	2.07	0	0	0	1		
15	Mediterranean Basin	Temperateness	20.73	40.9	705	4	49.78	66.08	2.07	0	0	0	0		
16	Mediterranean Basin	Temperateness	20.73	40.9	705	4	49.78	66.08	2.07	0	0	0	1		

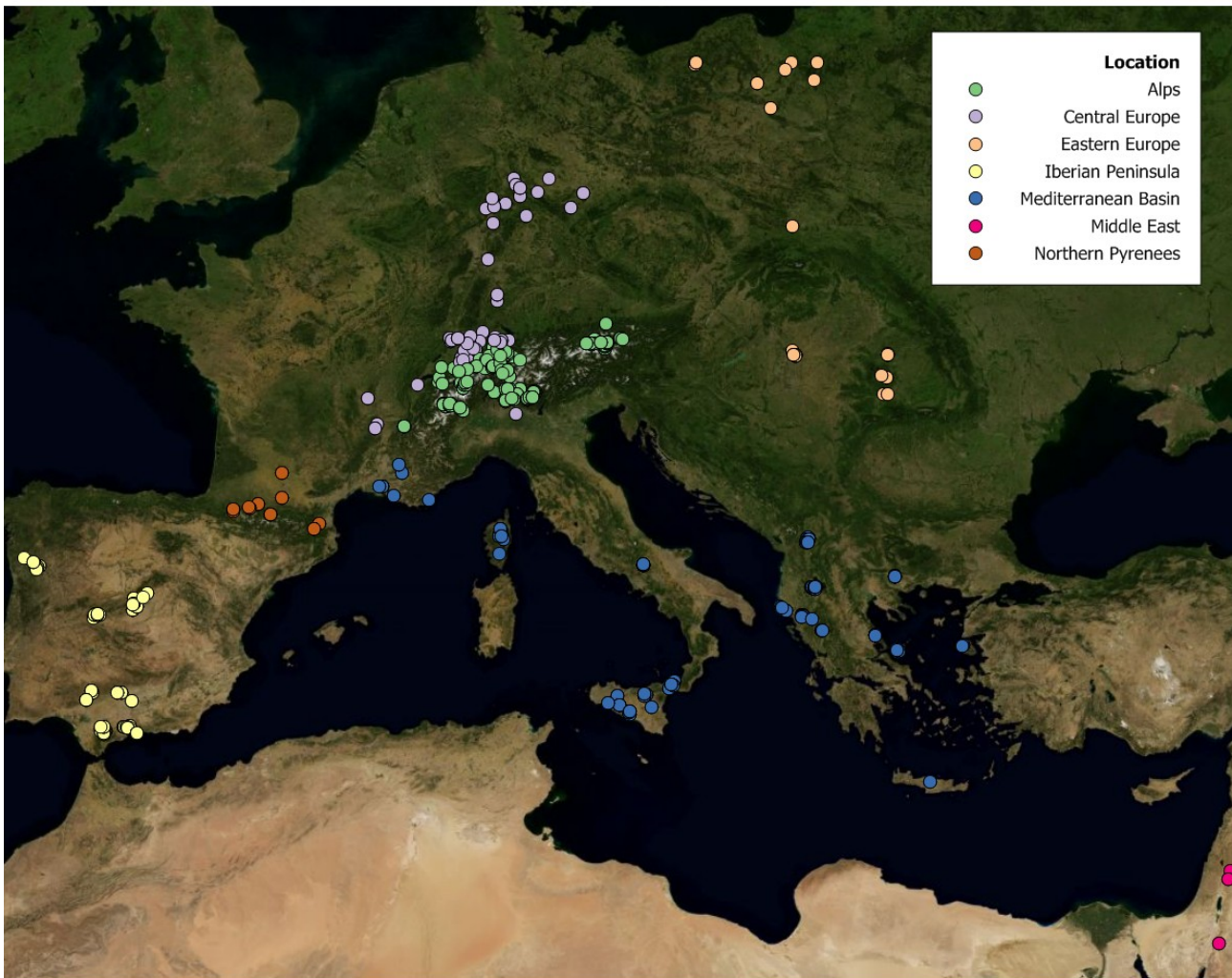
The next step is to recode the nominal / ordinal variables out of which we are not able to re-create a quantitative value, thus variables "location" and tmp_class".

Nominal predictive variables

Variable "location"

These values are purely nominal and they cannot be recoded into quantitative values or intervals. Furthermore, we want to use the "Central Europe" value as reference value. This consideration is purely conceptual, but it forces us to recode the variable in a specific way. MatSAM always uses the smallest value as the reference value when using the "reference" design type.

Their distribution over Europe and Asia is:



In the matrix, the variable "location" can have the following values and these will be recoded as:

"Location" value	Recoded value
Alps	2
Central Europe	1
Eastern Europe	3
Iberian Peninsula	4
Mediterranean Basin	5

Middle East	6
Northern Pyrenees	7

In the table, we will recode "location" into a new variable called "location_nb" and then delete the column "location" (as we do not want to use it for the regression):

	A	B	C	D	E	F	G	H	I	J	K	L	M	DF	DG	DH
1	location_nb	tmp_class	longitude	latitude	altitude	ph_fao	suryear	pryear	wndyear	E32_T38_3a	E32_T38_4	E32_T38_4a	E32_T38_4b	E45_T32_39		
2		5	20.21	40.12	660	3	53.91	107.28	1.97	0	0	1	0	0		
3		5	20.21	40.12	660	3	53.91	107.28	1.97	0	0	1	0	0		
4		5	20.21	40.12	660	3	53.91	107.28	1.97	0	0	1	1	0		
5		5	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0	0		
6		5	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0	0		
7		5	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0	0		
8		5	20.21	40.1	396	3	53.91	107.28	1.97	0	0	1	0	0		
9		5	20.22	40.09	427	3	53.91	107.28	1.97	0	0	1	0	0		
10		5	20.22	40.09	427	3	53.91	107.28	1.97	0	0	1	1	0		
11		5	20.19	40.07	197	3	53.91	107.28	1.97	0	0	1	0	0		
12		5	20.19	40.07	197	3	53.91	107.28	1.97	0	0	1	0	0		
13		5	20.19	40.07	197	3	53.91	107.28	1.97	0	0	1	0	0		
14		5	20.73	40.9	705	4	49.78	66.08	2.07	0	0	0	1	0		
15		5	20.73	40.9	705	4	49.78	66.08	2.07	0	0	1	0	0		
16		5	20.73	40.9	705	4	49.78	66.08	2.07	0	0	1	1	0		

Ordinal predictive variables

Variable "tmp_class"

The temperature class variable has 5 different values and we do not know how these classes were created, thus we do not know which the specific temperature interval for each class is:

"tmp_class" value	Recoded value
Very cold	-2
Cold	-1
Temperateness	0
Warm	1
Very warm	2

As we want to use a symmetrical design, we have to define around which value the symmetry has to be designed. In this case, it is quite obvious that the center value is "Temperateness" and that the other values are distributed around it. Thus, the simplest way to recode this kind of design is to assign the value 0 to the center value and design the other values has an increase / decrease of it.

Again, in the table, we will recode the "tmp_class" into a new variable called "tmp_class2" and then delete the column "tmp_class" (as we do not want to use it for the regression):

	A	B	C	D	E	F	G	H	I	J	K	L	M	DF	DG	DH
1	location_nb	tmp_class2	longitude	latitude	altitude	ph_fao	suryear	pryear	wndyear	E32_T38_3a	E32_T38_4	E32_T38_4a	E32_T38_4b	E45_T32_39		
2		0	20.21	40.12	660	3	53.91	107.28	1.97	0	0	1	0	0		
3		0	20.21	40.12	660	3	53.91	107.28	1.97	0	0	1	0	0		
4		0	20.21	40.12	660	3	53.91	107.28	1.97	0	0	1	1	0		
5		0	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0	0		
6		0	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0	0		
7		0	20.21	40.11	684	3	53.91	107.28	1.97	0	0	0	0	0		
8		0	20.21	40.1	396	3	53.91	107.28	1.97	0	0	1	0	0		
9		0	20.22	40.09	427	3	53.91	107.28	1.97	0	0	1	0	0		
10		0	20.22	40.09	427	3	53.91	107.28	1.97	0	0	1	1	0		
11		0	20.19	40.07	197	3	53.91	107.28	1.97	0	0	1	0	0		
12		0	20.19	40.07	197	3	53.91	107.28	1.97	0	0	1	0	0		
13		0	20.19	40.07	197	3	53.91	107.28	1.97	0	0	1	0	0		
14		0	20.73	40.9	705	4	49.78	66.08	2.07	0	0	0	1	0		
15		0	20.73	40.9	705	4	49.78	66.08	2.07	0	0	1	0	0		
16		0	20.73	40.9	705	4	49.78	66.08	2.07	0	0	1	1	0		

Variable "ph_fao"

For this variable, we have a transition table from the FAO⁶:

- category 0 : water
- category 1 : pH<4.5
- category 2 : pH>= 4.5-5.5
- category 3 : pH> 5.5-7.2
- category 4 : pH> 7.2-8.5
- category 5 : pH>= 8.5

If it is possible, we will ALWAYS try to recode an ordinal / nominal variable into a quantitative variable. For the first two special variables we saw that this was not possible. Here we can clearly do it.

For surface water systems, the pH is usually between 6.5-8.5, thus we will consider an average value of 7.5. For category 1, we will consider a minimum value of 0, thus an average value of 2.25. For category 5, the maximum value will be set to 10, which is already a high pH value in natural systems, thus an average value of 9.25. For all other value, we will consider the average value of the interval:

"ph_fao" value	Recoded value
category 0	7.5
category 1	2.25
category 2	5
category 3	6.35
category 4	7.85
category 5	9.25

In the table, we will recode the "ph_fao" into a new variable called "ph_fao_avg" and then delete the column "ph_fao" (as we do not want to use it for the regression):

	A	B	C	D	E	F	G	H	I	J	K	L		DF	DG	DH
1	location_nb	tmp_class2	longitude	latitude	altitude	ph_fao_avg	suryear	pryear	wndyear	E32_T38_3a	E32_T38_4	E32_T38_4a	E32	E45_T32_39		
2	5	0	20.21	40.12	660	6.35	53.91	107.28	1.97	0	0	1	0	0		
3	5	0	20.21	40.12	660	6.35	53.91	107.28	1.97	0	0	1	0	0		
4	5	0	20.21	40.12	660	6.35	53.91	107.28	1.97	0	0	1	1	0		
5	5	0	20.21	40.11	684	6.35	53.91	107.28	1.97	0	0	0	0	0		
6	5	0	20.21	40.11	684	6.35	53.91	107.28	1.97	0	0	0	0	0		
7	5	0	20.21	40.11	684	6.35	53.91	107.28	1.97	0	0	0	0	0		
8	5	0	20.21	40.1	396	6.35	53.91	107.28	1.97	0	0	1	0	0		
9	5	0	20.22	40.09	427	6.35	53.91	107.28	1.97	0	0	1	0	0		
10	5	0	20.22	40.09	427	6.35	53.91	107.28	1.97	0	0	1	1	0		
11	5	0	20.19	40.07	197	6.35	53.91	107.28	1.97	0	0	1	0	0		
12	5	0	20.19	40.07	197	6.35	53.91	107.28	1.97	0	0	1	0	0		
13	5	0	20.19	40.07	197	6.35	53.91	107.28	1.97	0	0	1	0	0		
14	5	0	20.73	40.9	705	7.85	49.78	66.08	2.07	0	0	0	1	0		
15	5	0	20.73	40.9	705	7.85	49.78	66.08	2.07	0	0	1	0	0		
16	5	0	20.73	40.9	705	7.85	49.78	66.08	2.07	0	0	1	1	0		

Missing values

The "NaN" command (N=upper case a= lower-case N=upper case) can be placed where you have a missing value (an environmental variable value or the presence/absence of a marker) in the input matrix (environmental variables or presence or absence of a given marker). In Matlab syntax, NaN means "Not-a-Number". The main impact is that the G test cannot be computed when the presence/absence of a marker show a missing value, and "NaN" will appear in the matrix of results in the corresponding column. But this

⁶ Food and Agriculture Organization of the United Nations

does not affect the Wald test. In this case, you will have to assess your results on the basis of the Wald test only, and this makes the "Cumulated test" mentioned here above unusable. This is due to the elements used by both tests to produce a statistical test. G; the distribution of this statistic is a chi-square with a number of degrees of freedom equal to the number of investigated parameters.

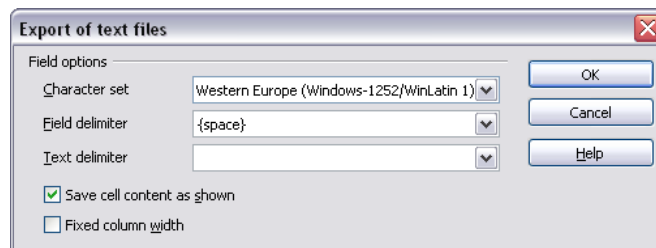
Final verifications

The following points should absolutely be verified in order to be sure to have a clean and usable input matrix:

- The first line has to contain the column names. These must not contain any space character (replace them with an underscore character).
- Delete all the columns which are not necessary for the analysis.
- The matrix has to be formatted in order to have first the environmental predictors and then the genetic markers (response variable).

If the input matrix is ready, save it in a text format using a space character for the column separator. Do not use any quotes for the column names.

In Openoffice, you can choose the separator and the quotes when exporting in the csv format:



In Excel (2007), use the "CSV (Comma delimited)*.csv" export format. Once this is done, open the saved file in a text editor (Notepad, Scite, ...) and use the replace command to replace the semi-colon character with the space character. Finally save your file.

Defining the parameter file

As we have seen, this file contains information about the input matrix. In the example developed above, this would mean:

First line:	number of environmental variables:	9	
Second Line:	number of response variables:	101	
Third line	create PNG graphical files	1	(or 0 for not creating them)
Fourth line:	column number of ordinal/cardinal variables	1 2	
Fifth line:	type of design variables to be used	r s	

The fourth line shows to the system that you have two special columns (the pH is used as an usual cardinal variable). The first one of these is using the "reference" design (location of the animals) and the second one is using the "symmetrical" design (temperature classes).

For the third type of design, thus the "independent", you should use the letter i.

How to proceed

1. Install MCRIInstaller.exe. The installer will propose to create the following folders \MATLAB\MATLAB

- Component Runtime, accept and wait the installation process to finish. This can take several minutes. UNINSTALL all PRIOR VERSION of the Matlab Runtime;
2. Extract the content of the "matsam_v2beta.zip" in a folder (all elements have to be in the same folder);
 3. Open the MS DOS command console ("Start" in the Windows toolbar, then "Run...", "cmd" in the text area, and then "OK".);
 4. "matsam_v2beta.exe" has to be run from the MS DOS console (no double click on the "matsam_v2beta.exe" file! It won't work!). Before running "matsam_v2beta.exe", all existing "output.txt", "graphics.txt" and "pseudoR2.txt" files in the current folder will be overwritten. Beware to remove all such existing files in the current folder.
 5. Write the following command (square brackets are shown here to highlight elements separated by spaces, and are not part of the command). This is a 2 argument command :

```
C:\path...\matsam_v2beta.exe [InputMatrix.txt] [InputParameters.txt]
```

- Element 1 is the name of the input matrix (space separated text file with HEADER);
- Element 2 is the name of the parameter files (minimum two lines, maximum five lines);

6. To run "matsam_v2beta.exe" on the example file provided:

```
C:\path...\matsam_v2beta.exe input_matrix.csv example_parameter.txt
```

7. Do not mind the message about the system locale settings. "matsam_v2beta.exe" will run (model processing) until the prompt with the path is displayed again, and 3 files appear in the current folder: "output.txt", "pseudoR2.txt" and "graphics.txt".
8. If you have chosen to create the graphical results of your regressions (third line of the parameter file containing a 1), **DO NOT TOUCH YOUR COMPUTER** until the end of the computation. Matlab uses a screenshot-like process to create the images. Modifying the screen layout during the process would result in bad image containing the things you were touching.

MatSAM results and output

MatSAM produce three output files: "output.txt", "pseudoR2.txt" and "graphics.txt".

Result analysis

To analyze the results, simply open the "output.txt" file in a notepad like software, select everything ([ctrl-a]), and paste everything in an Openoffice or Excel sheet.

Structure of the table

The table containing the results is made of 15 different groups of statistical data. Each group is constituted of n rows, where n represents the number of environmental variables. Columns correspond to genetic markers. These groups contain the following information:

1. Log Likelyhood2
2. Log Likelyhood1
3. Degrees of freedom
4. G value
5. P value for G
6. Null hypothesis rejected for G (default confidence level = 99%)
7. Wald for Beta 0
8. Wald for Beta 1

9. P value for Wald Beta 0
10. P value for Wald Beta 1
11. Null hypothesis rejected for Wald Beta 0 (default confidence level = 99%)
12. Null hypothesis rejected for Wald Beta 1 (default confidence level = 99%)

The 3 next groups constitute the dynamic section of the rejection table.

13. Dynamic null hypothesis analysis for G and Wald Beta 1 : Null hypothesis for G
14. Dynamic null hypothesis analysis for G and Wald Beta 1 : Null hypothesis for Wald Beta 1
15. Dynamic null hypothesis analysis for G and Wald Beta 1 : Cumulated test

We will mainly focus on those 3 groups to carry out the analysis. The other above groups contain the basic statistics and are made available in case that it is necessary to refine the analysis. The next groups are parameters to verify that everything went fine during the computation.

16. Marker frequency
17. Warning - Log likelihood2
18. Warning - Log likelihood1
19. Warning - Wald computation

Three kinds of warning might appear: a, b and c.

- a: iteration limit reached: this warning occurs when the algorithm does not converge (no optimal solution for the regression parameters). This happens mostly when the response variable frequency is near of 0 or 1. It is a complete separation of the binomial response in the regression. If the response frequency is true/false (0/1), the maximum likelihood may be infinite, thus the algorithm does not converge.
Another issue is that you may not have enough observations for the number of parameters you are trying to estimate. This is a statistical issue, not a software issue. Simplify your model, or collect more data. This might occur when using design variables and having too many classes. Empirically, the number of classes should not exceed 5-10. But 10 is already a lot!
- b: nearly singular matrix: this warning occurs mostly when using design variables. It happens as the system tries to invert the covariance matrix in order to compute the Wald test (and is often due to warning a). If the system was not able to converge, then some of the variances and covariances are high, others very low, thus an almost singular matrix.
- c: singular matrix: same as warning b, but this time the covariance matrix is clearly singular. It results in a "Nan" value for the Wald test.

If you get warnings, the first thing to do is to verify:

- Your marker frequencies (near 0 or 1)
- The number of observations in your system. Is the redundancy big enough?
- If using a design variable, how many classes have you got? If too many try to aggregate the classes into fewer.

Dynamic table of analysis

In the dynamic groups (see above), formulas make it possible to set up a dynamic rejection table, whose results will evolve according to the confidence level you chose. In these 3 groups (13 to 15), cells display a "1" when the null hypothesis is rejected for the chosen confidence level, and a "0" when the null hypothesis is not rejected (the investigated variable does not significantly contribute in explaining more variance than a model with a constant only). In the last group (15), cells show a "1" only when both tests (G and Wald) failed to reject the null hypothesis (the reason is statistical robustness, see Joost et al. 2007).

On the 3 matrices (13, 14, 15) apply the Excel conditional formatting with a given color when cells contain a

1, and no color when cells contain a 0. This way, significant models are dynamically highlighted when you change the significance threshold.

Adapting significance level

The confidence level can be adjusted in order to take into account the multiple hypotheses testing context. Here we simply apply the Bonferroni correction: the confidence level you choose is divided by the number of models and the result stored in the Bonferroni correction cell. This last one is used to reject or not the null hypothesis, what makes the approach very conservative (see arguments in Joost et al. 2007).

Pseudo R2

The "pseudoR2.txt" file contains information about each computed model. Like for the "output.txt", this file can be copy/paste from a notepad to an Excel or Openoffice sheet.

Several Pseudo R2, the AIC and BIC are computed for each combination of environmental variable and marker. The comparison between the results may give some hints about the goodness of fit of the different models.

It is important to compare the R2 to another and not to just look at one value. It is only a basis for comparison. Moreover a good value of R2, without comparison to another R2, does not mean that the fit of the model was good. The same is true for the AIC and BIC indicators.

Graphics ("graphics.txt")

The "graphics.txt" file contains 5 lines for each marker regarding each environmental variable which is not nominal or ordinal:

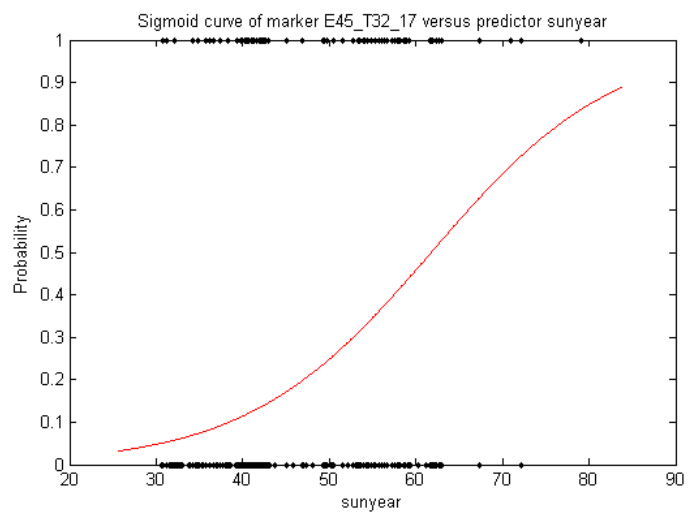
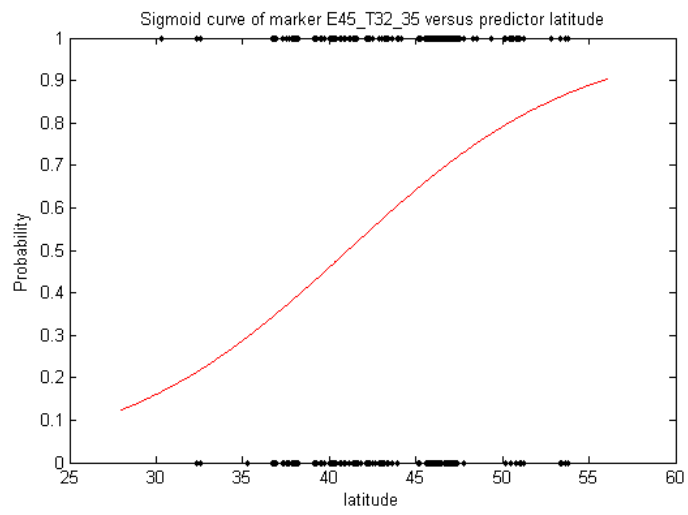
- Line 1: Name of environmental variable and marker name;
- Line 2: values of the environmental variable;
- Line 3: presence or absence of the genetic marker;
- Line 4: subdivision of the X axis (scale given by the statistic distribution of the environmental variable investigated);
- Line 5: probability that the genetic marker is present for the corresponding environmental variable.

To build graphics, you have to use lines 2 to 5. First use lines 4 (x-axis) and 5 (y-axis) to draw the sigmoid curve, than use lines 2 (x-axis value) and 3 (y-axis value) to draw the markers (0: absence, 1:presence). You can use line 1 to create the title and labels of the graphic.

Graphics created using MatSAM

If you choose to produce graphics using MatSAM, it will result in a graphic for each environmental variable for each marker.

MatSAM uses the PNG image format to create them. The two above images show result for the example files:



References

- Dobson, A. J. and Barnett, A. G., 2008, *An Introduction to Generalized Linear Models, Third Edition*, CRC Press, 307 p.
- Hosmer, D. W. and Lemeshow, S., 2000, *Applied Logistic Regression, Second Edition*, Wiley series in probability and statistics, 375 p.
- Joost, S., Bonin, A., Bruford, M. W., Després, L., Conord, C., Erhardt, G. and Taberlet, P., 2007, *A Spatial Analysis Method (SAM) to detect candidate loci for selection: towards a landscape genomics approach to adaptation*, *Molecular Ecology*, Vol.16, No 18, pp. 3955–3969.
- Joost, S., Kalbermatten, M. and Bonin, A., 2008, *Spatial Analysis Method (SAM): a software tool combining molecular and environmental data to identify candidate loci for selection*, *Molecular Ecology Resources*, 8:957–960.
- McCullagh, P. and Nelder, J. A., 1989, *Generalized Linear Models, Second Edition*, CRC Press – Chapman & Hall, 511 p.
- Pariset, L., Joost, S., Ajmone Marsan, P. and Valentini, A., 2009, *Landscape genomics and biased FST approaches reveal Single Nucleotide Polymorphisms under selection in goat breeds of North-East Mediterranean*, *BMC Genetics*, 10:7.

MK, 24.03.10