HAL
open science

Factor Analysis for Multiple Testing (FAMT): An R Package for Large-Scale Significance Testing under Dependence

David Causeur, Chloé Friguet, Magali Houee-Bigot, Maela Kloareg

To cite this version:

David Causeur, Chloé Friguet, Magali Houee-Bigot, Maela Kloareg. Factor Analysis for Multiple Testing (FAMT): An R Package for Large-Scale Significance Testing under Dependence. Journal of Statistical Software, 2011, 40 (14), pp.19. hal-00730155

HAL Id: hal-00730155

https://institut-agro-rennes-angers.hal.science/hal-00730155v1

Submitted on 5 Mar 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Journal of Statistical Software

May 2011, Volume 40, Issue 14. http://www.jstatsoft.org/

Factor Analysis for Multiple Testing (FAMT): An R Package for Large-Scale Significance Testing under Dependence

David Causeur
Agrocampus Ouest

Chloe Friguet
Agrocampus Ouest

Magalie Houee-Bigot
INRA, UMR598,
Animal Genetics

Maela Kloareg
Agrocampus Ouest

Abstract

The R package **FAMT** (factor analysis for multiple testing) provides a powerful method for large-scale significance testing under dependence. It is especially designed to select differentially expressed genes in microarray data when the correlation structure among gene expressions is strong. Indeed, this method reduces the negative impact of dependence on the multiple testing procedures by modeling the common information shared by all the variables using a factor analysis structure. New test statistics for general linear contrasts are deduced, taking advantage of the common factor structure to reduce correlation and consequently the variance of error rates. Thus, the FAMT method shows improvements with respect to most of the usual methods regarding the non discovery rate and the control of the false discovery rate (FDR).

The steps of this procedure, each of them corresponding to R functions, are illustrated in this paper by two microarray data analyses. We first present how to import the gene expression data, the covariates and gene annotations. The second step includes the choice of the optimal number of factors, the factor model fitting, and provides a list of selected genes according to a preset FDR control level. Finally, diagnostic plots are provided to help the user interpret the factors using available external information on either genes or arrays.

Keywords: factor analysis, multiple testing, dependence, false discovery rate, non discovery rate, R.

1. Introduction

Most of the existing multiple testing procedures rely on the analysis of the empirical process of p-values associated to the individual tests, under the assumption of independence. In practice, and especially in gene expression data for instance, unmodeled and/or uncontrolled

factors can interfere with the true signal and then generate dependence across the measured variables. This can be referred to as the heterogeneity of the data, as is also mentioned in Leek and Storey (2007). This data heterogeneity violates the independence assumption and induces instability in multiple testing, as shown in Friguet *et al.* (2009). Indeed, it has been demonstrated that this dependence has a negative impact on the multiple testing procedures, particularly on the variance of the number of false positive genes, thus on the control of the False Discovery Proportion (Efron 2007; Kim and Van de Wiel 2008; Friguet *et al.* 2009). The FAMT (factor analysis for multiple testing) procedure deals with this problem by modeling the common information shared by all the variables using a factor analysis structure. New test statistics for general linear contrasts are deduced, taking advantage of the common factor structure to reduce correlation and consequently the variance of error rates. The details of this method are given in Friguet *et al.* (2009).

The present paper aims at presenting the statistical handling of multiple testing dependence as proposed in the R (R Development Core Team 2011) package **FAMT** (Causeur *et al.* 2010). The crucial steps of the analysis correspond to core functions: `as.FAMTdata` to import the data and create a single R list from multi-sourced datasets, `modelFAMT` to estimate the dependence kernel and adjust the data from heterogeneity components and `defacto` to relate the heterogeneity components to external information if provided. Moreover, additional functions are proposed to summarize the results (`summaryFAMT`) and to optimize the procedure by modifying the default choices implemented in `modelFAMT`, such as the estimation procedure for the proportion of true null hypotheses (`pi0FAMT`) or the optimal number of factors (`nbfactors`).

The FAMT procedure is applied to two microarray datasets, which both describe chicken hepatic transcriptome profiles, and are provided by the Animal Genetics Laboratory (INRA-Agrocampus Ouest, Rennes, France). The first microarray data analysis studies the relationships between hepatic gene expression and abdominal fatness (Le Mignon *et al.* 2009; Blum *et al.* 2010). The normalized microarray dataset is available in the **FAMT** package, and is used here to describe the method step by step. The second microarray data analysis focuses on the feeding-to-fasting transition in chicken liver by Désert *et al.* (2008).

The use of these two examples is motivated by their relevance in illustrating the two effects of dependence. Indeed, it can be shown (Leek and Storey 2008; Friguet and Causeur 2010) that the correlation between variables impacts the distribution under the null hypotheses of the individual tests p-values in two ways. In both cases, it leads to strong departures from the uniform distribution which is expected in the independent case: under dependence, the small null p-values (resp. close-to-one p-values) can be under-represented (resp. over-represented) as in the first example (see Figure 1). Conversely, the second example illustrates the second situation induced by dependence, where small null p-values (resp. close-to-one p-values) are over-represented (resp. under-represented), see Figure 7. Actually, even if the second case is more commonplace, taking the dependence into account is recommended to improve the data analysis in both situations. Yet, we show hereafter that the FAMT method can still be used in both situations to give more insight into the multiple testing procedure and increase its overall power.

2. Data manipulation

In microarray data analysis, the selection of differentially expressed genes involves at least two datasets with different dimensions.

First, the `expression` dataset is sized $m \times n$, where m is the number of observed response variables (gene expression in a microarray experiment) and n is the sample size (number of arrays), and $n << m$. In the analysis of microarray data, a pre-processing step of normalisation is usually carried out at first. The `expression` data corresponds here to these normalised data. In the following illustrative example, this dataset concerns hepatic transcriptome profiles for $m = 9893$ genes of $n = 43$ half sib male chickens selected for their variability on abdominal fatness (denoted hereafter `Af`). The observed variable corresponds to the quantitative measure of the weight of abdominal fat. The data come from the Animal Genetics Laboratory (INRA-Agrocampus Ouest, Rennes, France), and were initially generated to map quantitative trait loci (QTL) for abdominal fatness in chickens. Animals, marker genotyping and transcriptome data acquisition and normalization are described in Le Mignon *et al.* (2009).

The `covariates` dataset has n rows and the variables describe the experimental conditions: the identifier of each row (arrays), which correspond to the column names of `expression`, is provided together with the value of the main explanatory variable in the testing issue (`Af` in the present example) and possibly some other covariates if provided in the study. This dataset is optional: if not provided, the procedure aims at testing the significance of the mean expressions.

Finally, the `annotations` dataset, with m rows, provides additional information about the response variables that can be further used as an interpretation tool: in the example, the functional characterization of each gene extracted from the Gene Ontology is useful to directly connect the list of differentially expressed genes to biological processes. In the present example, some other additional variables characterizing the location of the spots on the microarray (block, row, column) are also considered. One column of this dataset must be named `ID` and gives the variable (gene) identifier that will be used in the final output of the procedure. This dataset is optional: if not provided, a basic `annotations` dataset is created with row indices as variable identifiers.

The first step of the FAMT method uses the `as.FAMTdata` function to create a single R list containing the multi-sourced datasets. To avoid violations of the correspondence between the columns of the `expression` dataset and the rows of the `covariates` dataset, this function also checks that one column in `covariates`, the index of which is given by the argument `idcovar`, gives the individual (array) identifier, matching the column names of `expression`. Some simple tests for the compatibility of the datasets' dimensions are also performed: the number of columns of `expression` must correspond to the number of rows of `covariates` and furthermore, `expression` and `annotations` must have the same number of rows. The subclass of the output is named `FAMTdata` and the belonging to this class is required by the other functions of the package.

In our example, three datasets are provided:

- `expression`: It contains 9893 gene expressions, and 43 individuals.
- `covariates`: It contains 6 variables: `AfClass` (abdominal fatness class, with 3 levels: `F` = fat, `L` = lean, `NC` = intermediate), `ArrayName` (identifying the arrays), `Mere` (dam of the offsprings, a factor with 8 levels), `Lot` (hatch, a factor with 4 levels), `Pds9s` (body weight, a numeric vector), and `Af` (abdominal fatness, a numeric vector). `Af` is the experimental condition of main interest in this example.

- `annotations`: It contains 6 variables: `ID` (gene identifier), `Name` (gene functional category), `Block`, `Column` and `Row` (location on the microarray), `Length` (oligonucleotide size).

The following code provides the `FAMTdata` object called `chicken`.

```
R> chicken <- as.FAMTdata(expression, covariates, annotations, idcovar = 2)

$`Rows with missing values'
integer(0)

$`Columns with missing values'
integer(0)
```

By default, `idcovar = 1`. Here, we use `idcovar = 2` because the array identification is given in the second column of `covariates`.

Besides, this step checks for missing data, since the FAMT method cannot be applied to incomplete observations. The `as.FAMTdata` function gives the indices of the rows and columns of `expression` with missing values. If needed, using `na.action = TRUE`, the missing values are imputed by the nearest neighbor averaging (function `impute.knn` of the package **impute**, Hastie *et al.* 2011). Here, the `expression` component of `chicken` has no missing data.

Some classical componentwise summaries can be obtained from a `FAMTdata` object thanks to the `summaryFAMT` function which provides:

- for `expression`: The number of tests (which is the number of rows) and the sample size (which is the number of columns)
- for `covariates` and `annotations`: Classical summaries as returned by the generic function `summary` of package **base**.

The code to perform the summary of a `FAMTdata` object is:

```
R> summaryFAMT(chicken)

$expression
$expression$`Number of tests'
[1] 9893

$expression$`Sample size'
[1] 43

$covariates
 AfClass  ArrayName        Mere    Lot         Pds9s            Af
 F :18   F10     : 1   GMB05555:10   L2:16   Min.   :1994   Min.   :-25.5397
 L :19   F11     : 1   GMB05625: 7   L3:11   1st Qu.:2284   1st Qu.: -8.0042
 NC: 6   F12     : 1   GMB05562: 5   L4: 8   Median :2371   Median :  2.7166
         F13     : 1   GMB05599: 5   L5: 8   Mean   :2370   Mean   :  0.2365
```

```
         F14     : 1   GMB05554: 4           3rd Qu.:2474   3rd Qu.:  8.6037
         F15     : 1   GMB05589: 4           Max.   :2618   Max.   : 18.1024
         (Other):37   (Other) : 8

$annotations
         ID           Name               Block          Column
 RIGG00001:   1   Length:9893        23     : 237   9      : 502
 RIGG00002:   1   Class :character   25     : 237   6      : 501
 RIGG00003:   1   Mode  :character   9      : 232   12     : 499
 RIGG00005:   1                      17     : 231   18     : 499
 RIGG00006:   1                      29     : 230   14     : 494
 RIGG00007:   1                      43     : 229   16     : 491
 (Other)  :9887                      (Other):8497   (Other):6907
      Row           Length
 20     : 500   Min.   :60.00
 15     : 498   1st Qu.:70.00
 16     : 493   Median :70.00
 8      : 483   Mean   :69.57
 17     : 481   3rd Qu.:70.00
 9      : 468   Max.   :75.00
 (Other):6970
```

This step is especially useful to check the class of variables in `covariates` and `annotations`. The dataset called `annotation` is composed of variables that will be further used to characterize the latent factors after model fitting, except the `ID` column which is only used to match the rows of the `annotation` dataset with the ones of the `expression` dataset. Attention should be drawn to the fact that the variable associated to the gene names must be considered as a `character variable` and not as a `factor`, with a number of levels that would equal the number of genes. The `summaryFAMT` function can help to check it before going further in the data analysis.

3. Multiple testing

This section is dedicated to the use of the FAMT method as a classical multiple testing procedure controlling the false discovery rate (FDR), without any modeling for the dependence structure across the variables.

3.1. Multiple F-tests for general linear hypotheses

The scope of models for the relationship between the responses and the explanatory variable(s) of interest is restricted to linear models. Let $\mathbf{Y} = (Y^{(1)}, \ldots, Y^{(m)})^\top$ be the m-vector of response variables and $\mathbf{x} = (x^{(1)}, \ldots, x^{(p)})^\top$ the $p-$vector of explanatory variables. It is assumed that:

$$Y^{(k)} \quad = \quad \beta_0^{(k)} + \mathbf{x}^\top \beta^{(k)} + \varepsilon^{(k)}, \tag{1}$$

where $\varepsilon = (\varepsilon^{(1)}, \ldots, \varepsilon^{(m)})^\top$ is a normally distributed $m-$vector with mean $\mathbf{0}$ and variance-covariance $\mathbf{\Sigma}$.

The individual tests are the usual Fisher tests for the marginal effect of one or more explanatory variables of interest among x, considering the other ones as covariates. In most of the cases, only one explanatory variable x is included in the model and the aim is then to test the significance of the relationship between each variable and x. However, more complex situations also occur, where the effect of this explanatory variable shall be examined after adjustment from other effects, which have been accounted for in the experimental design. Note also that, if no `covariates` dataset is provided, then model (1) is the null model and the significance of the mean of each variable is tested.

The thresholding procedure applied on the p-values of the F-tests to control the FDR at a given level α is the Benjamini-Hochberg (BH) procedure (Benjamini and Hochberg 1995). The cut-off on the p-values under which the hypotheses are rejected is derived from the increasingly ordered p-values $p_{(k)}$ as follows: $t_\alpha = p_{(k^*)}$ with $k^* = \arg\max_k\{m\pi_0 p_{(k)}/k \leq \alpha\}$, provided the proportion of true null hypotheses π_0 is known. Many multiple testing procedures assume that the fraction of non-null hypotheses among the tests is negligible regarding the large number of tests ($\pi_0 \approx 1$). For example, with the Benjamini-Hochberg procedure, approximating π_0 by 1 leads to a FDR control at level $\pi_0\alpha$ instead of α. Generally, plugging-in an estimate of π_0 into the expression of the FDR corrects this for control level and results in a less conservative procedure (see Benjamini and Hochberg 1995; Black 2004; Storey 2002, for details).

Two methods are proposed in the **FAMT** package to estimate π_0: the first one is based on a non-parametric estimate of the density function of the p-values by a convex curve using the approach of Langaas *et al.* (2005) and the other one uses the smoothing splines approach proposed by Storey and Tibshirani (2003).

In the following, the use of the `modelFAMT` function is illustrated using the `chicken` dataset introduced in section 2.

3.2. Results

In the `chicken` example, the aim is to test the significance of the relationship between each gene expression and the abdominal fatness variable (6th column of `covariates`), taking into account the effect of the dam (3rd column of `covariates`). The Fisher test statistics and the corresponding p-values are obtained using the `modelFAMT` function with arguments `x = c(3, 6)` to give the column numbers of the explanatory variables in the `covariates` component of `chicken` and `test = 6` to give the column number of the explanatory variable of interest. Model (1) is fitted here, thus the following code also uses argument `nbf = 0`. The interest of this argument is specified in the following section.

```
R> chicken.raw <- modelFAMT(chicken, x = c(3, 6), test = 6, nbf = 0)
R> hist(chicken.raw$pval, main = "Histogram of p-values", xlab = "p-values")
```

Figure 1 displays the histogram of the raw p-values, as produced by the command lines above. The shape of the histogram clearly shows an abnormal under-representation of the p-values in the neighborhood of 0. Indeed, if all the gene expressions were truly under the null hypothesis, the p-values should be uniformly distributed on $[0, 1]$ and the proportion of observed p-values under 0.05 should be close to 0.05, provided the gene expressions are independent. This marked departure of the empirical distribution of p-values from the density function of a uniform distribution has been recently considered by some authors as the impact

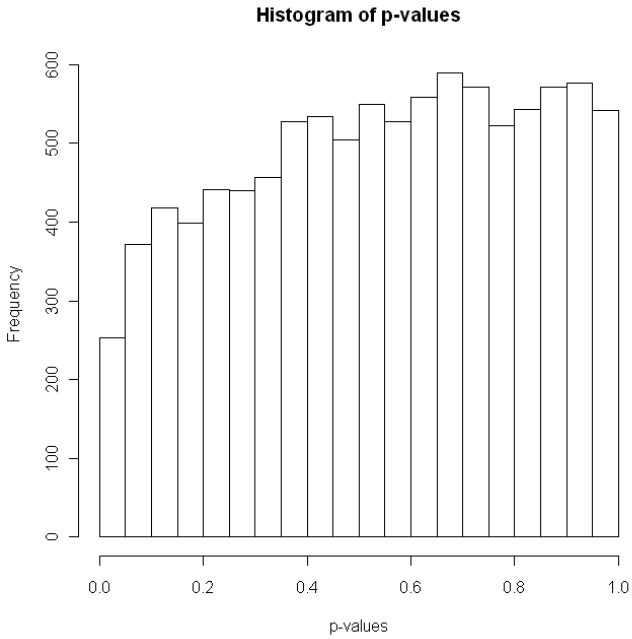

Figure 1: Histogram of the raw p-values for the `chicken` dataset.

of a high amount of dependence among tests (see Efron 2007; Leek and Storey 2007; Friguet and Causeur 2010).

The `modelFAMT` function creates a R list with subclass `FAMTmodel`. This subclass is required for the main input of the other functions in the package. Thus, the `summaryFAMT` function can be applied to a `FAMTmodel` object to get the list of positive tests for a control of the FDR at a preset level α (the default level is `alpha = 0.15`). Moreover, some useful information about the positive responses is provided, using the argument `info` which gives the names of columns in the `annotations` component of `chicken`. Here, the columns named `ID` and `Name` give the gene identifier and the functional annotation of the significant genes.

```
R> summaryFAMT(chicken.raw, alpha = 0.05, info = c("ID", "Name"))

$nbreject
  alpha Raw analysis FA analysis
1  0.05            0           0

$DE
integer(0)

$pi0
[1] 1
```

The `nbreject` component in the output of `summaryFAMT` is a table providing the number of positive tests using the raw multiple testing procedure and the factor analytic approach, for possibly different values of the FDR control level α. In this special use of `summaryFAMT`

with `nbf = 0`, the columns named `Raw analysis` and `FA analysis` gives the same result since they are equivalent. The result shows no positive genes for a FDR control at level 0.05. The `DE` component of the output also provides the additional information on the responses specified in the argument `info`.

Note that, if not explicitly provided as an input of the function `summaryFAMT` using argument `pi0`, the proportion π_0 of true null hypotheses is estimated from the histogram of p-values, by the method proposed by Storey and Tibshirani (2003), and returned in the `pi0` component of the output. Here, $\hat{\pi}_0 = 1$ is a direct consequence of the abnormal shape of the histogram of p-values as displayed in Figure 1. Indeed, the dependence across genes induces a bias in the estimation of π_0. Analysis of this dependence among gene expressions is addressed in section 5 but a first possible biological explanation is that all the chickens in this experiment are half sib males, which is to say genetically very similar.

This example is a typical situation where the dependence among genes must be taken into account to have a chance to reveal significant relationships between the hepatic transcriptome profile and the quantity of abdominal fatness.

4. Multiple testing dependence using FAMT

4.1. Method

The details of the method are described in Friguet *et al.* (2009). The main innovation with respect to most classical methods consists in capturing the components of dependence between variables into latent factors and integrating this latent structure in the calculation of the test statistics. It is indeed assumed that the conditional covariance matrix Σ of the responses, given the explanatory variables is represented by a factor analysis model:

$$\mathbf{\Sigma} \quad = \quad \mathbf{\Psi} + \mathbf{B}\mathbf{B}^\top, \tag{2}$$

where $\mathbf{\Psi}$ is a diagonal $m \times m$ matrix of uniquenesses ψ_k^2 and $\mathbf{B}$ is a $m \times q$ matrix of factor loadings. In the above decomposition, the diagonal elements ψ_k^2 in $\mathbf{\Psi}$ are also referred to as the specific variances of the responses. Therefore, $\mathbf{B}^\top\mathbf{B}$ appears as the shared variance in the common factor structure.

The factor analysis representation of the covariance is equivalent to the following mixed effects regression modeling of the data: for $k = 1, \ldots, m$,

$$Y^{(k)} \quad = \quad \beta_0^{(k)} + \mathbf{x}^\top \beta^{(k)} + \mathbf{b}_k^\top Z + \varepsilon^{(k)}, \tag{3}$$

where $\mathbf{b}_k$ is the kth row of B, $\mathbf{Z} = (Z^{(1)}, \ldots, Z^{(q)})$ are latent factors supposed to concentrate in a small dimension space the common information in the m responses, $\mathbf{Z}$ is normally distributed with expectation $\mathbf{0}$ and variance $\mathbf{I}_q$ and $\varepsilon = (\varepsilon^{(1)}, \ldots, \varepsilon^{(m)})^\top$ is a normally distributed $m-$vector, independent of $\mathbf{Z}$, with mean $\mathbf{0}$ and variance-covariance $\mathbf{\Psi}$. Therefore, factor analysis can be viewed as simultaneous mixed effects regression models sharing common covariance components.

An expectation-maximization (EM) algorithm inspired from Rubin and Thayer (1982) is used to estimate $\mathbf{\Psi}$, B and $\mathbf{Z}$ (see Friguet *et al.* 2009 for details). Since this algorithm

only implies inversions of $q \times q$ matrices, fitting the factor analysis (FA) model on high-dimensional datasets is computationally much less cumbersome than more usual algorithms such as Principal Factoring used in psychometrics. As recommended in Friguet *et al.* (2009), the number of factors is chosen according to an *ad-hoc* procedure which consists in minimizing the variance of the number of false discoveries. Once the factor model is estimated, the so called factor-adjusted test statistics are derived as F-tests calculated on the adjusted response variables $Y^{(k)} - \mathbf{b}_k^\top Z$ obtained by subtracting the dependence kernel from the data. Friguet *et al.* (2009) show that the resulting test statistics are asymptotically uncorrelated, which improves the overall power of the multiple testing procedure.

4.2. Results of the FAMT analysis

Optimal number of factors

The `modelFAMT` function implements the whole `FAMT` procedure with default options for the estimation of π_0 and the number of factors. As mentioned in the previous section, the method proposed by Storey and Tibshirani (2003) is implemented to estimate π_0.

Concerning the number of factors, the dependence in the residual correlation matrix resulting from the k-factor analysis model fitting induces an inflation of the variance of the number of false positives. This variance has a negative impact on the actual control of the false discovery proportion. Hence, as explained by Friguet *et al.* (2009), the number of factors considered in the model is chosen to reduce this variance. In order to avoid the overestimation of the number of factors, the function is implemented in such a way that the optimal number of factors corresponds to the largest number of factors for which the decrease of the variance inflation criterion is lower than 5 % of the previous value (see the Catell scree test criterion, Cattell 1966). Nevertheless, the optimal number of factors can also be specified by the `nbf` argument in the `modelFAMT` function (see the second illustrative example of this paper). Once the optimal number of factors is chosen, the model parameters are estimated using an EM algorithm. Factor-adjusted test statistics are derived, as well as the corresponding p-values.

The testing issue is the same as in the previous section.

```
R> modelfinal <- modelFAMT(chicken, x = c(3, 6), test = 6)
R> modelfinal$nbf
```

```
[1] 3
```

A side effect of the `modelFAMT` function is to produce a diagnostic plot, displaying the values of the variance inflation criterion along with the number of factors. Figure 2 shows that the optimal number of factors obtained by the `modelFAMT` function, which is `modelfinal$nbf = 3`, also corresponds in this case to the minimum value of the variance inflation criterion. The model parameters are estimated with this choice of a 3-factor structure and π_0 is estimated using the method by Storey and Tibshirani (2003) applied on the factor-adjusted p-values.

The number of positive tests is provided for each level of FDR control chosen by the user (in our example below, the levels are defined by the argument `alpha = seq(0, 0.3, 0.05)`). The list of positive genes (`DE` component) is given for the highest `alpha`.

```
R> summaryFAMT(modelfinal, alpha = seq(0, 0.3, 0.05))
```

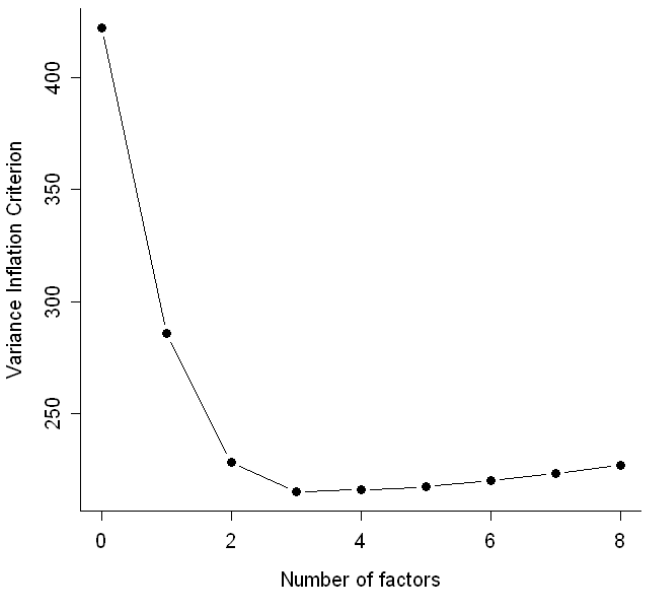

Figure 2: Variance inflation criterion for the determination of the optimal number of factors.

```
$nbreject
  alpha Raw analysis FA analysis
1  0.00            0           0
2  0.05            0           0
3  0.10            0           2
4  0.15            0           6
5  0.20            0           6
6  0.25            0           8
7  0.30            0          11

$DE
            ID
6722 RIGG05436
3885 RIGG04393
1119 RIGG15056
3484 RIGG05478
463  RIGG09893
124  RIGG12578
9859 RIGG03755
9840 RIGG04507
3925 RIGG10355
4968 RIGG05365
3855 RIGG13434
```

```
6722                                                    Same gene X54200
3885      Weakly similar to CAE03429 (CAE03429) OSJNBa0032F06.12 protein
1119                                                ENSGALT00000015290.1
3484 ...ilar to Q8IUG4 (Q8IUG4) Rho GTPase activating protein (Fragment)
463                                                 ENSGALT00000000452.1
124                                                 ENSGALT00000008042.1
9859                                                 Contig Hit 348847.1
9840 ...r to Q8AWZ8 (Q8AWZ8) Voltage-gated potassium channel subunit MiR
3925                 Transforming protein p54/c-ets-1. [Source:SWISSPROT
4968                                               Genome Hit Contig7.437
3855         Troponin T fast skeletal muscle isoforms. [Source:SWISSPROT


$pi0
[1] 0.9738531
```

With a FDR control at level 0.15, there is no differentially expressed gene with the raw analysis, whereas 6 genes are differentially expressed with the `FAMT` analysis based on factor-adjusted tests statistics. In order to figure out the differences between both analyses, Figure 3 compares the empirical distributions of the raw and the factor-adjusted p-values.

```
R> par(mfrow = c(1, 2))
R> hist(modelfinal$pval, main = "Histogram of p-values",
+    xlab = "Unadjusted p-values")
R> hist(modelfinal$adjpval, main = "Histogram of adjusted p-values",
+    xlab = "Adjusted p-values")
```

Factor-adjustment restores independence between test statistics, which results in a correction of the distribution of the p-values from the concave shape observed on the left panel plot of Figure 3. Indeed, it seems that a large amount of p-values are uniformly distributed and a few small p-values shall correspond to significant genes.

Note that the user can choose to focus on two aspects of the multiple testing procedure, which are the choice of the optimal number of factors with the `nbfactors` function and the estimation of π_0 with the `piOFAMT` function.

```
R> nbfactors(chicken, x = c(3, 6), test = 6, diagnostic.plot = TRUE)
```

This function gives the optimal number of factors as obtained from the `modelFAMT` function and produces the same plot as shown in Figure 2.

The `piOFAMT` function provides 2 algorithms to estimate π_0. The `density` method is based on Langaas *et al.* (2005)'s approach where the density function f of the p-value distribution is estimated assuming f is a convex function: the estimation of π_0 is then $f(p = 1)$. The `smoother` method uses the smoothing spline approach proposed by Storey and Tibshirani (2003). In most situations, these two methods give similar results but the `smoother` method is numerically less time-consuming.

The following code uses the `density` method to estimate π_0 and produces a histogram of the p-values (Figure 4), on which the convex estimation of f is represented.

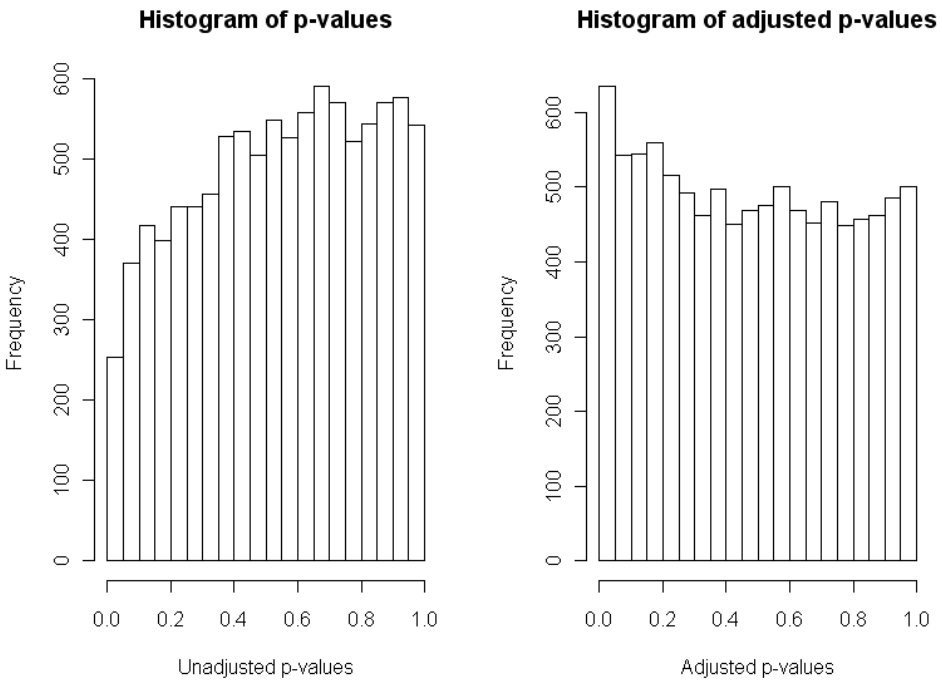

Figure 3: Histograms of raw p-values (left) and factor-adjusted p-values (right).

```
R> pi0FAMT(modelfinal, method = "density", diagnostic.plot = TRUE)
```

The estimated value of π_0 is 0.95, which is slightly less than with the `smoother` method ($\hat{\pi}_0 = 0.97$).

5. Interpretation of the common factors

The `defacto` function helps the user to give more insight on the common factors using some available external information on either response variables or individuals (see Blum *et al.* 2010). The external information is variables of `covariates`, which are not used in the model, and the categorical variables of `annotations`. As in principal component analysis (PCA), a transformation of the dataset allows to represent the individuals and variables graph through the matrix of factors loadings and the matrix of estimated factors. An analysis of variance test assesses the significance of the relationship between the factor and the external information.

The use of this function requires a `FAMTmodel` as returned from the function `modelFAMT` and one or more explanatory variables in `covariates`. As the factors are designed to be independent from the explanatory variables (the abdominal fatness and the dam in our example), they shall be described according to the other covariates. In our example, the argument `select.covar = 4:5` gives the 2 column numbers in `covariates` picking two external variables: `Lot` and `Pds9s`, which are respectively the hatch and the body weight of the chickens. Similarly, the argument `select.annot = 3:6` picks 4 external variables in `annotations`: `Block`, `Column`, `Row` and `Length`.

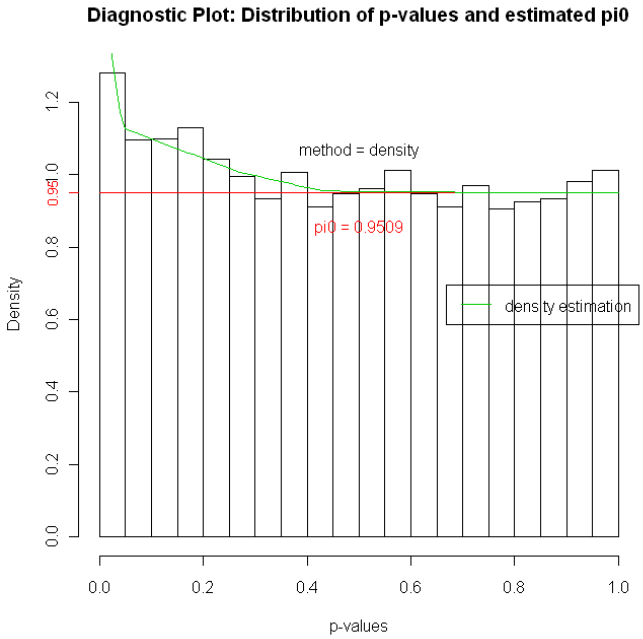

Figure 4: Estimation of the proportion of true null hypotheses using a non-parametric estimate of the density function of the p-values proposed by Langaas *et al.* (2005).

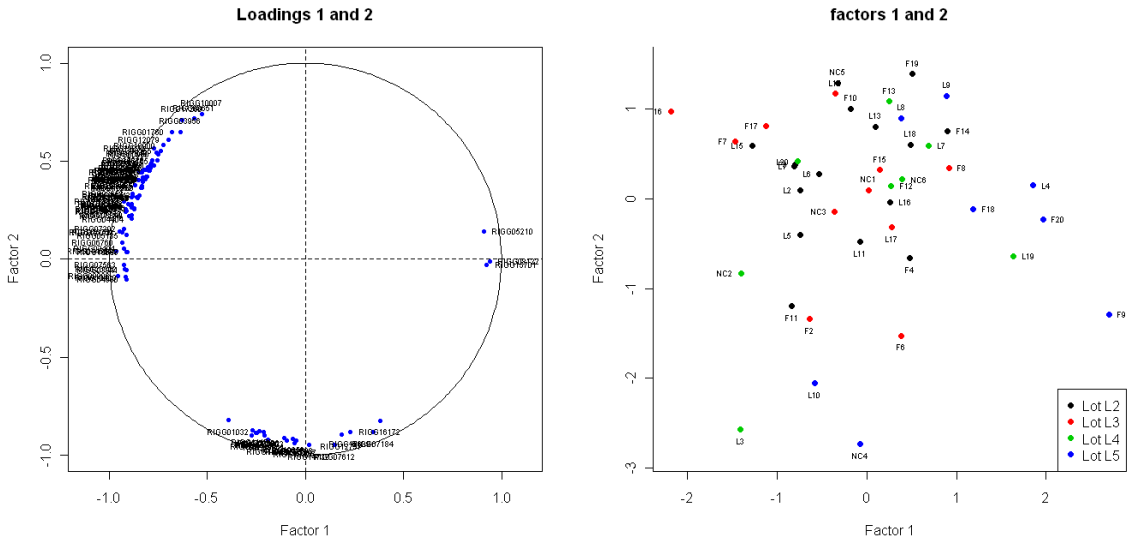

Figure 5: Loadings circle plot of the genes in the `chicken` example (left). Score plot of the microarrays in the `chicken` example (right).

As for many implementations of PCA-like methods, two plots are provided to summarize the relationships between the latent factors extracted from the data and the external variables. First, if there are at least 2 common factors in the FA model, the `defacto` function provides a

loadings circle plot displaying the largest loadings along with two factors, the index of which are given by the argument `axes` (the default option is `axes = c(1, 2)`). Points are automatically labelled by their identifier as given in `annotations` (see Figure 5, left). Moreover, the score plot displays the coordinates of the individuals along the two factors, with different colors according to the levels of the factors selected in `covariates` (their hatch in the `chicken` example, see Figure 5, right).

```
R> chicken.defacto <- defacto(modelfinal, axes = 1:2, select.covar = 4:5,
+    select.annot = 3:6, cex = 0.6)
```

In addition to these plots, F-tests are provided for the significance of the linear relationship between each component of the external information and each factor. The corresponding p-values are given in the `covariates` and `annotations` components of the `defacto` function.

```
R> chicken.defacto$covariates

                 Lot      Pds9s
Factor 1 0.006437319 0.27847793
Factor 2 0.258859549 0.00124608
Factor 3 0.271648846 0.96813322

R> chicken.defacto$annotations

                  Block    Column        Row
Loadings 1 8.148075e-25 0.2368072 0.21767892
Loadings 2 3.328477e-19 0.9323152 0.01889079
Loadings 3 0.000000e+00 0.1030426 0.68201616
```

The p-values inferior to a 5% treshold show a significant relation between the external information and the factor. Here, `Factor 1` is clearly affected by a hatch effect, and `Factor 2` by a body weight effect. Thus, part of the expression heterogeneity is probably due to these marked biological effects, which are independent of the abdominal fatness, the explanatory variable of the main interest in this study.

Moreover, some second-order technological biases turn out to have an impact on the correlation structure of the gene expressions, since the location of the spots on the microarray (`Block`, `Column`, and `Row` here) appears as significantly related to the loadings. According to Qiu *et al.* (2005), such kinds of effects on the correlation between gene expressions may be induced by the normalization procedure itself. The effect of `Block` is captured by all the factors, and the effect of `Row` by factor 2. The p-values of the effect of `Column` are all higher than the 5% treshold, so none of the factors are characterized by this effect.

6. Second illustrative example

The Animal Genetics Laboratory (INRA-Agrocampus Ouest, Rennes, France) studies the transcriptome profiling of the feeding-to-fasting transition in chicken liver. Désert *et al.* (2008) show that numerous genes are altered by starvation in chickens, and the study suggests

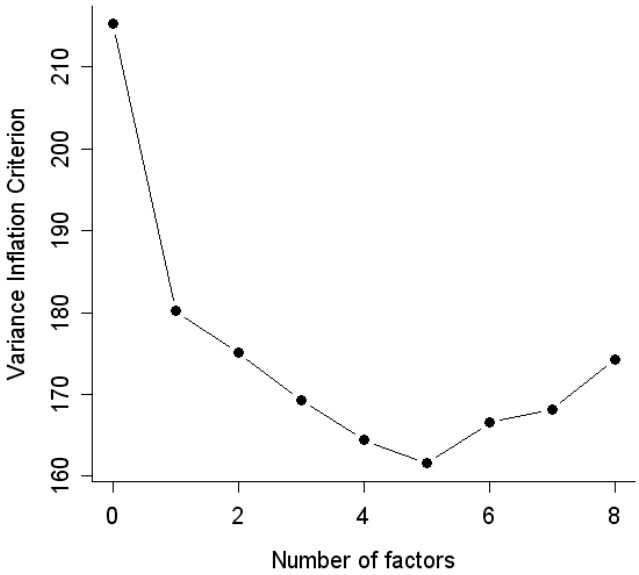

Figure 6: Variance inflation criterion along with the number of factors included in the model for Désert *et al.* (2008) dataset.

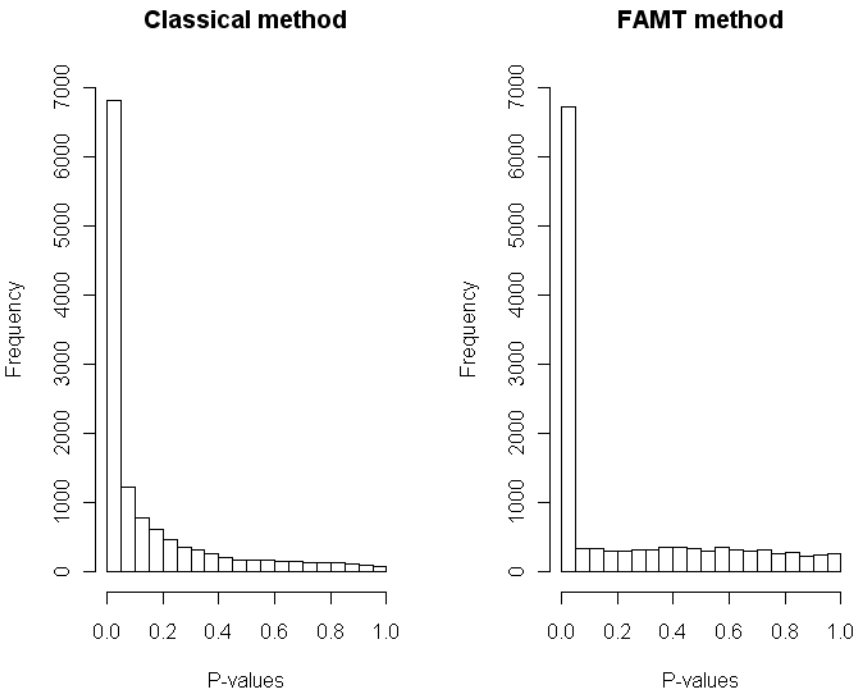

Figure 7: Histograms of raw p-values and factor-adjusted p-values for Désert *et al.* (2008) dataset.

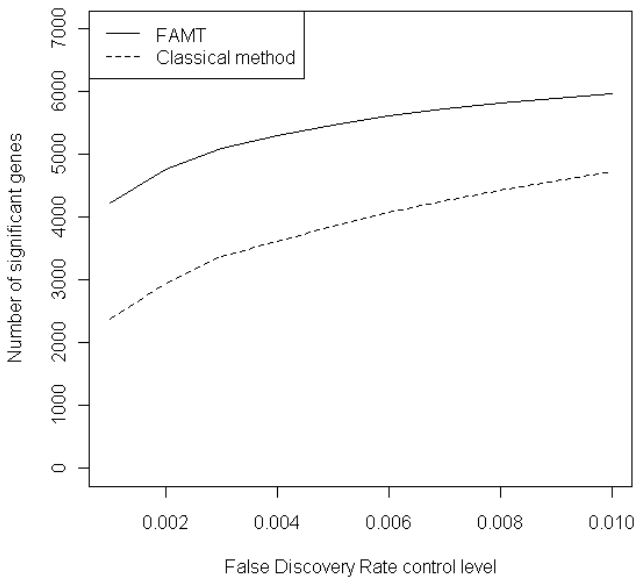

Figure 8: Number of significant genes with the raw method and FAMT for Désert *et al.* (2008) dataset.

a global repression of cellular activity in response to this stressor. In this section, the related gene expression data are used to illustrate that the FAMT method is still useful to increase the power of the multiple testing procedure in a case where a large proportion of genes are significant.

From the 20460 oligos present in the microarray data, 13057 aligning with a unique coding region of the 2.1 Washington University assembly of the chicken sequence genome, were chosen for statistical analyses. The dataset was finally restricted to 7419 genes (out of 13057) presenting a human ortholog with a human gene ontology (HUGO) symbol allowing for the recovery of functional annotations from these databases. 18 microarrays were analyzed: 6 corresponding to fed chickens, 5 to 16-hour fasted animals and 7 to 48-hour fasted animals. We calculate the p-values of the classical Fisher tests. The left panel plot of Figure 7 shows that a large number of genes have small p-values, which means that many genes are involved in the fasting process.

Figure 6, resulting from the `modelFAMT` function, shows that the variance inflation criterion is minimum for 5 factors. Yet, the `modelFAMT` function proposes `nbf = 1` as optimal number of factors, using the Catell scree test criterion (see the previous section). In this case, the plot appears as a useful tool to modify, if necessary, the default number of factors resulting from `modelFAMT`. We finally chose to fit the factor analysis model with 5 factors.

The following code fits the FA model with 5 factors and extracts the numbers of rejected genes for the given FDR control levels.

```
R> model <- modelFAMT(Poulets, x = 2, nbf = 5)
R> rejections <- summaryFAMT(model, alpha = seq(0.001, 0.01, 0.001))$nbreject
```

```
R> plot(rejections[, 1], rejections[, 3], type = "l", lty = 1,
+    ylab = "Number of significant genes",
+    xlab = "False Discovery Rate control level",
+    ylim = c(0, 7000))
R> lines(rejections[, 1], rejections[, 2], type = "l", lty = 2)
R> legend("topleft", c("FAMT", "Classical method"), lty = c(1, 2))
```

The number of significant genes for various FDR control levels are plotted in Figure 8. For a same level of FDR control, more genes are considered as differentially expressed with the FAMT method than using the raw p-values. This illustrates that the FAMT procedure improves the power of the multiple testing procedure since, for a same FDR control level, more genes are significant.

7. Conclusion

The R package **FAMT** provides a powerful method for large-scale significance testing under dependence. It is essentially based on a factor modeling of the conditional covariance structure of the response variables. As these factors capture the dependence, they can be used to restore independence among tests, which results in a gain in terms of control of the false discovery proportion and on the overall power of the multiple testing procedure.

The main functions of the package are described in the paper, and are illustrated using a gene expression dataset available in the package. The package also offers tools to help the user describe and interpret the factors using some external information on either genes or arrays. The functions of the package, their arguments and values, are detailed in the help files. The website http://famt.free.fr/ sums up the **FAMT** package and gives news about eventual updates.

Forthcoming versions of the package should include currently studied procedures aiming at inferring on the gene regulatory network using a Gaussian graphical model. **Excel** add-ins should also be included in the next update in order to help non-R users to analyse microarray data using FAMT.

References

Benjamini Y, Hochberg Y (1995). "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing." *Journal of the Royal Statistical Society B*, **57**, 289–300.

Black MA (2004). "A Note on the Adaptative Control of False Discovery Rates." *Journal of the Royal Statistical Society B*, **66**, 297–304.

Blum Y, Le Mignon G, Lagarrigue S, Causeur D (2010). "A Factor Model to Analyze Heterogeneity in Gene Expression." *BMC Bioinformatics*, **11**(368).

Cattell RB (1966). "The Scree Test for the Number of Factors." *Multivariate Behavioral Research*, **1**, 245–276.

Causeur D, Friguet C, Houee-Bigot M, Kloareg M (2010). *Factor Analysis for Multiple Testing (**FAMT**): Simultaneous Tests under Dependence in High-Dimensional Data.* R package version 2.20.0, URL http://CRAN.R-project.org/package=FAMT.

Désert C, Duclos MJ, Blavy P, Lecerf F, Moreews F, Klopp C, Aubry M, Herault F, Le Roy P, Berri C, Douaire M, Diot C, Lagarrigue S (2008). "Transcriptome Profiling of the Feeding-to-Fasting Transition in Chicken Liver." *BMC Genomics*, **9**(611).

Efron B (2007). "Correlation and Large-Scale Simultaneous Testing." *Journal of the American Statistical Association*, **102**, 93–103.

Friguet C, Causeur D (2010). "Estimation of the Proportion of True Null Hypotheses in High-Dimensional Data under Dependence."

Friguet C, Kloareg M, Causeur D (2009). "A Factor Model Approach to Multiple Testing under Dependence." *Journal of the American Statistical Association*, **104**, 1406–1415.

Hastie T, Tibshirani R, Narasimhan B, Chu G (2011). ***impute**: Imputation for Microarray Data.* R package version 1.26.0, URL http://CRAN.R-project.org/package=impute.

Kim KI, Van de Wiel M (2008). "Effects of Dependence in High-Dimensional Multiple Testing Problems." *BMC Bioinformatics*, **9**(114).

Langaas M, Lindqvist BH, Ferkingstad E (2005). "Estimating the Proportion of True Null Hypotheses, with Application to DNA Microarray Data." *Journal of the Royal Statistical Society B*, **67**, 555–572.

Le Mignon G, Désert C, Pitel F, Leroux S, Demeure O, Guernec G, Abasht B, Douaire M, Le Roy P, Lagarrigue S (2009). "Using Transriptome Profiling To Characterize QTL Regions On Chicken Chromosome 5." *BMC Genomics*, **10**(575).

Leek JT, Storey J (2007). "Capturing Heterogeneity in Gene Expression Studies by Surrogate Variable Analysis." *PLoS Genetics*, **3**, e161.

Leek JT, Storey J (2008). "A General Framework for Multiple Testing Dependence." *Proceedings of the National Academy of Sciences*, **105**, 18718–18723.

Qiu X, Brooks A, Klebanov L, Yakovlev A (2005). "The Effects of Normalization on the Correlation Structure of Microarray Data." *BMC Bioinformatics*, **6**(120).

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Rubin DB, Thayer DT (1982). "EM Algorithms for ML Factor Analysis." *Psychometrika*, **47**, 69–76.

Storey JD (2002). "A Direct Approach to False Discovery Rates." *Journal of the Royal Statistical Society B*, **64**, 479–498.

Storey JD, Tibshirani R (2003). "Statistical Significance for Genomewide Studies." *Proceedings of the National Academy of Sciences*, **100**, 9440–9445.

Affiliation:

David Causeur
Agrocampus Ouest
Applied Mathematics Department
65 rue de St-Brieuc
35000 Rennes, France
E-mail: David.Causeur@agrocampus-ouest.fr
URL: http://www.agrocampus-ouest.fr/math/causeur/

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 40, Issue 14 *Submitted:* 2010-07-09
May 2011 *Accepted:* 2011-04-01