Using the `LFDREmpiricalBayes` Package

Ali Karimnezhad, Johnary Kim, Anna Akpawu and David R. Bickel

September 26, 2017

Abstract

The package `LFDREmpiricalBayes` contains a series of functions aiming at analyzing the association of single nucleotide polymorphisms (SNPs) to some particular disease. The functions use estimated local false discovery rates (LFDRs) of SNPs within a sample population that we define as a "reference class". Of the proposed methods, the maximum entropy method is based on ratio of two likelihood functions generated on the basis of two alternative reference classes. The other methods are based on robust Bayes approaches and are applicable to more than two reference classes. The explanations in this report allow for a better understanding of the basic ideas of how the functions in this package work as well as detailed examples that are useful in analyzing a given data set. Although SNPs are used throughout this document, other biological data such as protein data and other gene data can be used.

Contents

1 Introduction

The package `LFDREmpiricalBayes` contains a series of functions aiming at analyzing the association of SNPs to some particular disease. These functions include `ME.log`, `caution.parameter.actions`, `SEL.caution.parameter` and `PRGM.action`. Details and the theories behind these functions have been extensively discussed in Karimnezhad and Bickel (2016a).

The function `ME.log` uses the new maximum entropy approach given two reference classes (the definition of a reference class is provided in details in subsection 3.1). The new ME approach will consider both reference classes and provides a vector of more reliable LFDR estimates copmared to the input vector of LFDRs computed based on the two alternative reference classes (Karimnezhad and Bickel, 2016a).

The function `ME.log` as well as the functions `caution.parameter.actions`, `SEL.caution.parameter` and `PRGM.action` use estimated values of local false discovery rate (LFDR) as input elements. Although in this report we use maximum likelihood (ML) LFDR estimates, all these functions except `ME.log` work with other estimates of LFDR[2].

`caution.parameter.actions` provides three robust Bayes actions (based on three different values for a caution parameter) to determine if a specific SNP is associated with some disease. As well, `SEL.caution.parameter` provides three robust Bayes estimates of LFDR (based on three different values for a caution parameter) correspoding to a specific SNP. The estimates can be used to determine if the SNP is associated with the relevent disease. `PRGM.action` provides a robust Bayes value of the correspoding LFDR estimates. These three functions are based on two or more reference classes.

This vignette was created to demonstrate practical examples for each function in `LFDREmpirical-Bayes` so that users can better understand the code output. It provides comprehensive examples of the

[2]such as binomial-based LFDR estimator (BBE) or histogram-based LFDR estimator (HBE) (Yang et al., 2013).

functions, their corresponding interpretation, and a user-friendly explanation of the theoretical aspects associated with the functions and the methodologies developed in Karimnezhad and Bickel (2016a).

2 How to Install and Use This Package

In order to install and use this package, simply input the following commands into the `R` console.

```
> source("https://bioconductor.org/biocLite.R")
> biocLite("LFDREmpiricalBayes")
```

Note that this package uses other functions in the packages `matrixStats` and `stats`. This package also uses the function `lfdr.mle` from the package `LFDR.MLE`. To install the packages and use them, input the following into the `R` console.

```
> install.packages(c("LFDR.MLE","matrixStats","stats"))
```

Ensure that the following packages are loaded before using `LFDREmpiricalBayes`:

```
> library(LFDR.MLE)
> library(matrixStats)
> library(stats)
```

For more information, consult their respective documentation.

3 Basic Definitions

Before detailing the important functions of this package, a few concepts relating to relevant methods of the functions should be explained.

3.1 Reference Classes

One of the motivating factors in estimating LFDRs, as described in Karimnezhad and Bickel (2016a), involves choosing which set of LFDR estimates gives the best estimates. The main reason for this is due to having two different sets of LFDR estimates, which is referred to as a reference class. With the two reference classes, there lies a situation in which a subset of LFDR estimates can fall into both classes. In this case, the decision on whether to favour one class over the other is highly important.

Leading to favouring one class over another, two different reference class definitions were considered as two different cases. Those two cases are described in more details in Parts 3.1.1 and 3.1.2.

3.1.1 Considering Separate and Combined Reference Classes

In **Figure 1**, consider the two groups: ncRNA and UTR-3 groups containing 3 and 2 SNPs, respectively. Considering a series of LFDR estimates for SNPs in both the ncRNA and UTR-3 regions, a separate class considers only a small subset of the region seen in Figure 1. In this case, the region ncRNA containing the first three SNPs (i.e. SNP1, SNP2 and SNP3) is a separate reference class. The combined referece class is defined as the class that contains all SNPs in both the ncRNA and UTR-3 regions (i.e. the green rectangular). Now, to estimate LFDR for each of ncRNA SPNs, for example SNP1, there are

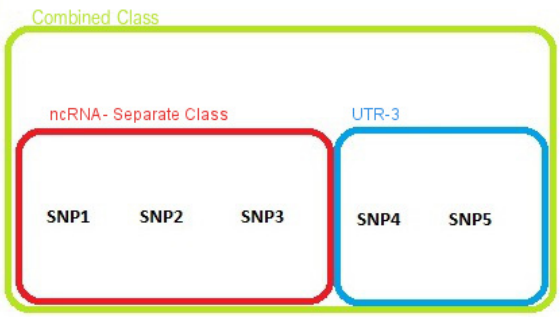

Figure 1: A visual diagram illustrating the separate and combined classes (Karimnezhad and Bickel, 2016b).

two reference classes with red and green regions in Figure 1. Determining which of the two reference classes (separate or combined) should be used to estimate the corresponding LFDR is the motivation of using the functions provided in this package.

3.1.2 Considering First and Second Reference Class Terms

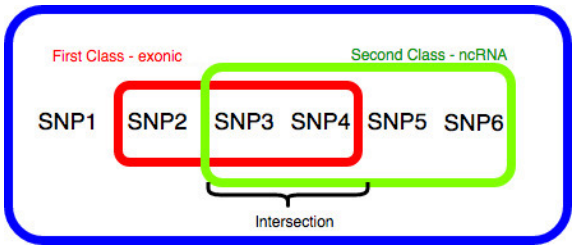

Figure 2: This figure illustrates the first and second reference classes and the intersection.

In Figure 2, we consider the first class in red to be exonic SNPs (SNP2, SNP3 and SNP4) and the second class in green to be ncRNA SNPs (containing SNP3, SNP4, SNP5 and SNP6). The intersection contains SNP3 and SNP4. The corresponding LFDR estimates based on the first class and the second class can be different. Determining an appropriate reference class to be used to reach a reliable LFDR estimate is our motivation.

3.2 Multiple Hypothesis Testing

Performing multiple hypothesis testing is important in terms of evaluating the relationship between a set of data within a population to some condition. The most general case in a biological perspective can be whether or not there is an association between an SNP and a particular disease. For example, Karimnezhad and Bickel (2016a) performed a genome-wide association analysis for the coronary heart disease (Consortium et al., 2007).

The framework of the multiple hypothesis testing problem considered here is as follows. For an ith SNP, $i = 1, 2, \ldots, N$, the null hypothesis $H_{0i} : A_i = 0$ is tested against the alternative hypothesis $H_{1i} : A_i = 1$, where A_i is an indicator that there is an association between SNP i and the disease. Under the null hypothesis, i.e. $A_i = 0$, the ith SNP is supposed not to be associated with (affected by) the underlying disease or treatment.

3.3 Local False Discovery Rates Estimation

LFDR is the posterior probability that an SNP is not associated with a particular disease (Karimnezhad and Bickel, 2016a). In a biological sense, the data are reduced to some test statistics t_1, t_2,...,t_N, and each test statistic t_i represents a value for a particular SNP, say SNP_i (i.e. t_1 represents SNP_1, t_2 represents SNP_2,..., t_N represents SNP_N). Then, LFDR_i, i.e. the probability that SNP_i is not associated with the underlying disease given the information that the value of the test statistics is t_i, is estimated.

3.4 Bayes Action on LFDRs Applied to Hypothesis Testing

A Bayes action is used to test a particular null hypothesis A_i. Once an estimated LFDR for SNP i is available, the Bayes action (decision) correspoding to that SNP is considered as below

$$\text{Decision} = \begin{cases} \texttt{0} \text{ if estimated } \text{LFDR}_i > \text{threshold}, \\ \texttt{1} \text{ if estimated } \text{LFDR}_i \leq \text{threshold}. \end{cases} \tag{1}$$

The interpretation of such a decision is as follows. If an estimated LFDR_i corresponding to SNP_i is less than or equal to the pre-determined threshold, then the null hypothesis is rejected and as a result, the SNP_i is considered to be associated with the disease. Otherwise, the null hypothesis would fail to be rejected and consequently, SNP_i is inferred not be be associated with the disease (Karimnezhad and Bickel, 2016a). The threshold considered above is determined by considering some loss values which are the subject of the next part.

3.4.1 Zero-One Loss Function

The threshold considered in the previous subsection is determind by considering some loss values as important components in the decision theory. The loss values considered are based on type-I (false-positive) and type-II (false-negative) errors. In using loss values, a value or a weight is assigned to either one of a type-I error or a type-II error. The weight is based on an incurred penalty associated in obtaining one of those errors. Introducing the fact that a type-I error is a less desirable outcome than a type-II error, the weighting of a loss value associated with a type-I would be higher than that of a type-II error.

Referring to the type-I and type-II errors by the loss values l_I and l_{II}, Karimnezhad and Bickel (2016a) use the following loss function

$$L_{ZO} = \begin{cases} \texttt{0} \text{ if } \delta_i = A_i \in \{0,1\}, \\ l_I \text{ if } \delta_i = \texttt{1},\ A_i = \texttt{0}, \\ l_{II} \text{ if } \delta_i = 0,\ A_i = \texttt{1}, \end{cases}$$

where $\delta_i = \delta(t_i)$ is the decision rule (1).

The decision rule (1) is called the Bayes rule if the threshold is taken to be

$$\text{threshold} = \frac{l_{II}}{l_I + l_{II}}. \tag{2}$$

Recalling that type-I error should be avoided as much as possible, it is recommended that l_I be greater than l_{II}. Following this recommendation, a threshold of 0.2 was chosen (Efron, 2005), assigning a value of `4` to l_I and a value of `1` to l_{II} .

In addition to the Bayes action, `caution.parameter.actions` uses a decision-theoretic approach in forming a large scale hypothesis problem and draws conclusions based on the experimental data set.

`caution.parameter.actions` applies some additional information in terms of reference classes (two or more) to test the null hypothesis based on some avaialbe estimated LFDR values. The output from this function gives two values 0 and 1 for each caution-type decision.

3.4.2 Squared Error Loss Function

In estimating the hypothesis indicator A_i, Karimnezhad and Bickel (2016a) use the following squared error loss function

$$L_{SEL} = (\delta_i - A_i),$$

where $\delta_i = \delta(t_i)$ is an arbitrary decision rule with support $(-\infty, +\infty)$.

4 The Maximum Entropy Method

The maximum entropy method, where `ME.log` is the corresponding function, provides a vector of more reliable LFDR estimates based on comparing two likelihood functions constructed based on two alternative reference classes. It overcomes the lack of knowledge that specifies whether a separate reference class or a combined reference class should be used to get more reliable estimates of LFDRs for SNPs. This approach leads to a "selected reference class" and giving credit to the selected reference class, an estimate of LFDR is computed for each SNP.

4.1 Inputs of `ME.log`

The inputs of `ME.log` are the following:

- `stat`: test statistic values corresponding to SNPs in the separate and combined reference classes
- `lfdr.C`: estimated LFDRs for SNPs in the combined reference class
- `p0.C`: proportion of unassociated SNPs belonging to the combined class
- `p0.S`: proportion of unassociated SNPs belonging to the separate class
- `ncp.C`: the non-centrality parameter of a chi-square distribution for associated SNPs belonging to the combined reference class
- `ncp.S`: the non-centrality parameter of a chi-square distribution for associated SNPs belonging to the separate reference class

Within the `ME.log` function, there are a series of parameters. Different positive values can be chosen for parameter a indicating grades of evidence against the separate reference class and in favor of its alternative. The defalut value for this parameter is set to 3 (Bickel, 2015)

The parameter `p0` indicates the proportion of SNPs that are not associated with the disease. The parameters `lower.p0` and `upper.p0` represent pre-assumed lower and upper bounds of `p0`, respectively. By default, those values are set to 0 and 1, respectively. `lower.ncp` and `upper.ncp` are used to denote the lower and upper limits of the non-centrality parameter `ncp`, respectively. By default, those values are set to 0.1 and 50, respectively. The methodolgy behind the ME method requires that the interval [`lower.p0`,`upper.p0`] be divided to `length.p0` partitions. As well, the interval [`lower.ncp`,`upper.ncp`] is be divided to `length.ncp` partitions. By default, `length.p0` and `length.ncp` values are set to 200.

The function `ME.log` depends on the function `lfdr.mle` for obtaining LFDR estimates. The LFDR estimates are then used in `lfdr.C` for the input of `ME.log`.

5 Using `ME.log`

The following subsections demonstrates the steps needed to use the `ME.log` function.

5.1 Obtaining Test Statistics and Associated Parameters

The code below is an example of a simulation study. Artificial SNPs are created, and test statistics are obtained. It is assumed that the separate reference class 20 artificial SNPs that are not associated with a specific disease. Also, it is assumed that the separate reference class is a subset of a combined reference class which contains 20 nonassociated and 20 associated SNPs. Given that the separate reference class is a subset of the combined reference class, the 20 SNPs in the separate reference class, each has two possible LFDR estimates.

```
> #import function "lfdr.mle" from package "LFDR.MLE"
> library(LFDR.MLE)
> ##From the simulation study, create artificial SNPs and obtain test statistics.
> sdORS<-sdORC<-sqrt(.02) #some parameters required for simulation.
> real.OR1.S<-1.25
> real.OR1.C<-1.25
> nS1<-0  ##Number of artificial SNPs associated with a
> ## disease in a separate reference class.
> nS2<-20 ## Number of artificial SNPs not associated with
> ## a specific disease in a separate reference class.
> nC1<-20 ##Number of artificial SNPs associated with a specific disease
> ## outside the separate reference class but inside a combined reference class.
> nC2<-0 ##Number of artificial SNPs not associated with a specific disease
> ## outside the separate reference class but inside a combined reference class.
>
> ##zS1 generates test statistics for artificial SNPs associated with a
> ##specific disease in the separate reference class.
> zS1<-rnorm(nS1,mean=log(real.OR1.S),sd=sdORS)
> ##zS2 generates test statistics for artificial SNPs not associated with a
> ##specific disease in the separate reference class.
> zS2<-rnorm(nS2,mean=log(1),sd=sdORS)##
> zSmall<-c(zS1,zS2) ## test statistics from the 20 artificial SNPs
> ##zC1 generates test statistics for artificial SNPs associated with a specific
> ##disease in the combined reference class.
> zC1<-rnorm(nC1,mean=log(real.OR1.C),sd=sdORC)
> ##zC2 generates test statistics for artificial SNPs not associated with a
> ##specific disease in the combined reference class.
> zC2<-rnorm(nC2,mean=log(1),sd=sdORC)
> zBig<-c(zSmall,zC1,zC2) ## test statistics from the 40 artificial SNPs
>
```

5.2 Obtaining LFDR Estimates Using `lfdr.mle`

Prior to using the function `ME.log`, chi-square test statistics need to be obtained. The test statistics from the separate and combined reference class, are transformed in order to obtain a chi-square distribution. The function `ME.log` then uses the function `lfdr.mle` from the package `LFDR.MLE` to compute LFDR estimates (Padilla et al., 2012).

```
> ##Then obtain chi-square test statistics
> ## Separate reference class
```

```
> xS1<-(zS1/sdORS)^2
> xS2<-(zS2/sdORS)^2
> xSmall<-c(xS1,xS2) ##chi-square test statistics from 20 SNPs
> ## Combined reference class
> xC1<-(zC1/sdORC)^2
> xC2<-(zC2/sdORC)^2
> xBig<-c(xSmall,xC1,xC2) ##chi-square test-statistics from 40 SNPs
> #Using lfdr.mle, a series of arguments are used.
> dFUN=dchisq;df=1;
> lower.ncp = .1;upper.ncp = 5
> lower.p0 = 0;upper.p0 = 1;
> #Estimate the corresponding LFDRs using lfdr.mle
> ## Separate reference class
> opt.S<-lfdr.mle(x =xSmall, dFUN = dFUN, lower.ncp = lower.ncp, upper.ncp = upper.ncp,
+                 lower.p0 = lower.p0, upper.p0 = upper.p0,df=df)
> lfdr.S <- opt.S$LFDR.hat ## Estimate the corresponding LFDRs
> p0.S<-opt.S$p0.hat
> ncp.S<-opt.S$ncp.hat
> ## Combined reference class
> opt.C<-lfdr.mle(x =xBig, dFUN = dFUN, lower.ncp = lower.ncp, upper.ncp = upper.ncp,
+                 lower.p0 = lower.p0, upper.p0 = upper.p0,df=df)
> lfdr.C <- opt.C$LFDR.hat
> p0.C<-opt.C$p0.hat
> ncp.C<-opt.C$ncp.hat
>
```

5.3 Illustrating `ME.log`

Now the parameters from the previous subsections can be used into the `ME.log` function.

```
> library(stats)
> library(matrixStats)
> # Using lfdr.mle, a series of arguments are used.
> ## if ommitted they will have the default values.
> LFDR.ME<-ME.log(xSmall,lfdr.C,p0.C,ncp.C,p0.S,ncp.S,a=3,lower.p0=0,upper.p0=1,lower.ncp=0.1,
+                 upper.ncp=50,length.p0=200,length.ncp=200)
> LFDR.ME


$p0.hat
 [1] 0.6281289 0.6281289 0.6281289 0.6281289 0.6281289 0.6281289 0.6281289
 [8] 0.6281289 0.6281289 0.6281289 0.6281289 0.6281289 0.6281289 0.6281289
[15] 0.6281289 0.6281289 0.6281289 0.6281289 0.6281289 0.6281289

$ncp.hat
 [1] 3.550765 3.550765 3.550765 3.550765 3.550765 3.550765 3.550765 3.550765
 [9] 3.550765 3.550765 3.550765 3.550765 3.550765 3.550765 3.550765 3.550765
[17] 3.550765 3.550765 3.550765 3.550765

$LFDR.hat
 [1] 0.8767839 0.8444716 0.6904314 0.8336484 0.9031935 0.9084548 0.5537349
 [8] 0.9043444 0.8555761 0.9087873 0.8550451 0.7369274 0.9025667 0.8285536
[15] 0.5080896 0.2707493 0.8800038 0.8952452 0.8361598 0.8998372


>
```

LFDR estimates were obtained from the output as `$LFDR.hat`. The LFDRs are of same length as the test statistic vector (`stat`) of the input. In fact, the new LFDR estimates have the same length as the SNPs falling into the intersection of both reference classes. Each `$LFDR.hat` is based on the corresponding `p0` and `ncp` given by `$p0.hat` and `$ncp.hat`. These `$p0.hat` and `$ncp.hat` are determined according to the likelihood set constructed based on two likelihood functions generated on the basis of the separate and combined reference classes. For more details refer to Karimnezhad and Bickel (2016a).

6 `caution.parameter.actions`

This function provides three actions based on three different values for a caution parameter defined in Karimnezhad and Bickel (2016a).

6.1 Input

The function `caution.parameter.actions` allows for a user to input two vectors of LFDR estimates of any size. The two inputs vectors `x1` and `x2` should be of the same size. These vectors refer to LFDR estimates for SNPs falling into the intersection of the two reference classes defined in subsection 3.1.

6.2 Output

The output of `caution.parameter.actions` is a list containing three vectors of actions which have been extensively discussed in Karimnezhad and Bickel (2016a).

- CGM1: refers to conditional-Gamma minimax action with caution parameter 1
- CGM0: refer to conditional-Gamma minimin action with caution parameter 0
- CGM0.5: refers to a caution-type action with caution parameter 0.5 which is a balance between CGM0 and CGM1

`CGM1`, the caution-type estimator describes the maximum amount of caution taken towards ambiguity while `CGM0` describes the minimal caution towards ambiguity (Bickel, 2015). Given an individual's prior belief of their subjective probability estimates, once the outcome is given, the individual's belief is updated. Note that the prior beliefs cannot be assessed with certainty. In this case, an individual must perform a best guess based on the prior information given to them. This is defined as ambiguity (Dobbs, 1991).

The theory behind these actions involves the loss fucntion L_{ZO} introduced earlier Part 3.4.1. The outputs from the three caution-type decisions is either `0` or `1`.

6.2.1 Interpreting the results from `caution.parameter.actions`

When interpreting the results of the function, the three caution-type decisions should be considered independently. In order to simplify matters, a vector of size 1, which corresponds to SNP1, is used to help illustrate a hypothetical case.

Example 1: Consider the following cases:

```
$CGM1
[1] 0
$CGM0
[1] 0
```

```
$CGM0.5
[1] 1
```

By **Example 1**, the outputs of `CGM1` and `CGM0` contain the same value. Interpreting the values of `CGM1` and `CGM0` of SNP1, the two show a value of `0` reflecting that there is no association between SNP1 and the disease. In this case, the null hypothesis is failed to be rejected. On the other hand, the result of `CGM0.5` is a value of `1`. Thus, the null hypothesis is rejected and it is interpreted that there is an associaiton between SNP1 and the disease. Depending on the caution-type estimator chosen, the decision can be different depending on which of `CGM1`, `CGM0` or `CGM0.5` the user chooses.

A more realistic example on how to interpret the results can be illustrated more effectively. From the R documentation `LFDREmpiricalBayes`, an example was provided where two vectors of size 4 are used. Noting that each index corresponds to a particular SNP value (e.g. index 1 corresponds to SNP1, index 2 corresponds to SNP2, etc.) a more detailed explanation is presented below.

Example 2: A basic interpretation of the outputs.

```
> LFDR.Separate <- c(.14,.8,.16,.30)
> LFDR.Combined <- c(.21,.61,.12,.10)
> output <- caution.parameter.actions(LFDR.Separate, LFDR.Combined)
> output


$CGM1
[1] 1 0 1 0

$CGM0
[1] 1 0 1 1

$CGM0.5
[1] 1 0 1 0
```

In this example, there are 4 SNPs. Here, SNP1, SNP2 and SNP3 have the same output for the three caution-type estimators. SNP4, on the other hand, has the same result for caution-type decisions `CGM1` and `CGM0.5` but a different result for `CGM0`. Thus, `CGM1` and `CGM0.5` reflects that there is no association between SNP4 and the disease, whereas on the basis of `CGM0` there is an association between SNP4 and the disease (the null hypothesis is rejected). Ultimately, the user would have to choose amongst one of the three caution-type estimators for their analysis. For example, in the simulation studies performed with the coronary artery disease in Karimnezhad and Bickel (2016a), `CGM1` and `CGM0.5` were shown to perform better than `CGM0`.

7 SEL.caution.parameter

This function provides three actions based on three different values for a caution parameter defined in Karimnezhad and Bickel (2016a). Unlike the function `caution.parameter.actions` which is based on the L_{ZO} loss function, this function is based on the L_{SE} loss function introduced earlier in Part 3.4.2.

7.1 Input

Much like `caution.parameter.actions`, `SEL.caution.parameter` accepts two vectors `x1` and `x2` which correspond to the two vectors of the separate and combined reference classes that were used in the former function.

7.2 Output

This output of this function is a list containing three vectors. The name of the three vectors are the same as the ones in `caution.parameter.actions`, i.e. `CGM1`, `CGM0`, and `CGM0.5`. Unlike the function `caution.parameter.actions`, which returns a `0` or a `1` value, this function returns any value between `0` to `1`. The closer the value it is to zero, the higher the accuracy is interpreted in estimating the hypothesis indicator A_i (Karimnezhad and Bickel, 2016a).

7.3 Examples

For a better clarificiation of performance of the function `SEL.caution.parameter`, the same example as seen in `caution.parameter.actions` is used.

Example 1: Consider the same Example 1 in `caution.parameter.actions`.

```
> LFDR.Separate <- c(.14,.8,.16,.3)
> LFDR.Combined <- c(.21,.61,.12,.1)
> output <- SEL.caution.parameter(LFDR.Separate, LFDR.Combined)
> output


$CGM1
[1] 0.86 0.20 0.88 0.90

$CGM0
[1] 0.825 0.295 0.860 0.800

$CGM0.5
[1] 0.79 0.39 0.84 0.70
```

Notice that the vectors have the same length in both the outputs of both `caution.parameter.actions` and `SEL.caution.parameter`. In this case, each corresponding element gives an estimate of the hypothesis indicator A_i from the three caution-type actions seen in `caution.parameter.actions`.

8 PRGM.action

`PRGM.action` is a function that computes the posterior regret Gamma minimax estimate of the hypothesis indicator A_i based on the two reference classes discussed earlier in Subsection 3.1. The thoery behind this function is on the basis the L_{SE} loss function introduced in Part 3.4.2.

8.1 Input

Similar to the functions `caution.parameter.actions` and `SEL.caution.parameter`, the inputs are two vectors of LFDR estimates `x1` and `x2` coresponding to the reference classes.

8.2 Output

The output of `PRGM.action` is equivalent to the output `CGM0` from the function `SEL.caution.parameter`. It returns a list which is one minus the average of `x1` and x2.

8.3 Examples

The following example will provide a small demonstratation of the relationship between the output of `PRGM` and the `CGM0` output vector of `SEL.caution.parameter`.

Example 1: Demonstrating the equivalence of `PRGM` and `CGM0`.

```
> LFDR.Separate <- c(.14,.8,.16,.3)
> LFDR.Combined <- c(.21,.61,.12,.1)
> output <- PRGM.action(x1 = LFDR.Separate, x2 = LFDR.Combined)
> output


$PRGM
[1] 0.825 0.295 0.860 0.800
```

Comparing the outputs of `PRGM` and that of `CGM0` of `SEL.caution.paramter`, as seen in Subsection 7.3, the outputs are the same.

Acknowledgements

We thank Abbas Rahal and Justin Chitpin for their contributions to this package.

References

Bickel, D. R. (2015). Inference after checking multiple bayesian models for data conflict and applications to mitigating the influence of rejected priors. *International Journal of Approximate Reasoning*, 66:53–72.

Consortium, W. T. C. C. et al. (2007). Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*, 447(7145):661.

Dobbs, I. M. (1991). A bayesian approach to decision-making under ambiguity. *Economica*, pages 417–440.

Efron, B. (2005). Local false discovery rates.

Karimnezhad, A. and Bickel, D. R. (2016a). Incorporating prior knowledge about genetic variants into the analysis of genetic association data: An empirical bayes approach. *Working Paper. University of Ottawa, deposited in uO Research at http://hdl.handle.net/10393/34889.*

Karimnezhad, A. and Bickel, D. R. (2016b). Information-theoretic and robust bayes multiple hypothesis testing with application to genetic association data. *44th Annual Meeting of the Statistical Society of Canada, Brock University, St. Catharines, Ontario, Canada.*

Padilla, M., Bickel, D. R., et al. (2012). Estimators of the local false discovery rate designed for small numbers of tests. *Statistical applications in genetics and molecular biology*, 11(5):1–42.

Yang, Z., Li, Z., and Bickel, D. R. (2013). Empirical bayes estimation of posterior probabilities of enrichment: A comparative study of five estimators of the local false discovery rate. *BMC bioinformatics*, 14(1):87.

Affiliation

David R. Bickel,
Ottawa Institute of Systems Biology,
Biochemistry, Microbiology and Immunology Department,
Mathematics and Statistics Department.
University of Ottawa
451 Smyth Road,
Ottawa, Ontario.
E-mail: dbickel@uottawa.ca

Ali Karimnezhad,
Ottawa Hospital Research Institute,
Biochemistry, Microbiology and Immunology Department,
University of Ottawa
451 Smyth Road,
Ottawa, Ontario.
E-mail: a.karimnezhad@uottawa.ca