NHMMfdr Package (Version 1.0.0)
Usage Tips and Simulation Example

Pei Fen Kuan

January 20, 2012

1 Citing NHMMfdr

If you have used `NHMMfdr` in your work, please cite the package using the following:

Kuan, P. and Chiang, D. (2012), "Integrating Prior Knowledge in Multiple Testing Under Dependence with Applications in Detecting Differential DNA Methylation," *Biometrics*, doi: 10.1111/j.1541-0420.2011.01730.

2 Introduction

This is an example of using the `NHMMfdr` package in `R`. The `NHMMfdr` package implements False Discovery Rate (FDR) control for multiple comparisons under dependence. It allows for informative exogeneous variables to be incorporated in the model to improve detection of significant tests. The proposed model is based on Kuan and Chiang (2012). This vignette aims to demonstrate the usage of `NHMMfdr` through an example using the simulated data.

3 Description and Usage

The main function in `NHMMfdr` package is `fdr.nhmm` which computes the local index of significance (LIS). LIS can be interpreted as an analog of p-values which incorporates the dependence structure among the hypothesis tests.

```
fdr.nhmm(x, Z = NULL, dist = NULL, log.transform.dist = TRUE, alttype = "kernel",
L = 2, maxiter = 1000, nulltype = 0, modeltype = "NHMM", symmetric = FALSE)
```

4 Arguments

4.1 x

x is a vector of summary statistics. In our motivating DNA methylation dataset described in Kuan and Chiang (2012), x is z-score values of differential methylation/expression. For most genomics data where test statistics such as student's t-statistic are available, x can be obtained from normal transformation, i.e., $x_j = \Phi^{-1}(P_j)$, where P_j is the p-value for probe j (e.g., student's t-test, Mann-Whitney test, etc) and Φ is the standard Gaussian cdf.

4.2 Z

Z is a matrix containing covariates or variables, **EXCLUDING** probe spacing/inter-probe distance that could potentially improve detection of significant probes. In microarray studies, Z could be genomic annotations. For instance, Kuan and Chiang (2012) used the CpG Island, Shore and Shelf definition or GC content as Z to improve detection of differentially methylated CpG loci. Note that Z is not required, i.e., `Z = NULL` if `modeltype = "Indep"` or `modeltype = "HMM"`. Note that probe spacing/inter-probe distance will be defined by `dist` below.

4.3 dist

A vector of probe spacing/inter-probe distance, since this covariate requires a different assumption on the transition matrix, i.e., probability of self transition decreases with distance between probes.

4.4 log.transform.dist

Logical value. If `log.transform.dist = TRUE`, the argument `dist` above will be applied $\log_2(\texttt{dist} + 2)$ transformation. This is recommended for numerical stability.

4.5 alttype

Type of distribution under alternative hypothesis. Sun and Cai (2009) modeled the alternative distribution f_1 as Gaussian mixtures. This option is available as `alttype = "mixnormal"`. If this option is selected, one will have to specify the number of mixture components L. Since L is unknown, we recommend trying for L=1, 2 and 3 and choose the best L using BIC.

Alternative, one can select `alttype = "kernel"` which approximates the non-null f_1 using non-parametric Gaussian kernel density estimation with automatically selected bandwidth. This can be much faster than Gaussian mixtures which requires tuning of L.

4.6 L

Number of mixture component for `alttype = "mixnormal"` as described above.

4.7 maxiter

Maximum iterations allowed in the EM algorithm to speed up computation. One can monitor the convergence from the log likelihood from each iteration.

4.8 nulltype

Type of null hypothesis assumed in estimating f_0. This option is imported from R package `locfdr` (Efron, 2004) which implements the following options:

- 0: theoretical null, i.e. x is assumed to be $N(0, 1)$ under H_0.
- 1: empirical null with parameters estimated by maximum likelihood.
- 2: empirical null with parameters estimated by central matching. Details in Efron (2004, 2007).

NOTE: We do not recommend the use of empirical null to avoid double adjusting of the correlation structure.

4.9 modeltype

Type of dependence structure among the summary statistics x.

- Indep: This option assumes that x's are independent. This is similar to `locfdr` except that it estimates f_1 differently.
- HMM: This option assumes that the underlying dependence structure follows a homogeneous HMM. This is the model in Sun and Cai (2009).
- NHMM: This option assumes that the underlying dependence structure follows a non-homogeneous HMM (Kuan and Chiang, 2012). It allows for prior knowledge which could improve detection of significant probes to be incorporated in the model.

4.10 symmetric

Logical value. If `symmetric = TRUE`, it assumes that the non-null f_1 is a symmetric distribution. See Kuan and Chiang (2012) for motivation on why assuming a symmetric f_1 could be advantageous. Currently, this option is only available for `alttype = "kernel"`.

5 Value

5.1 LIS

Local index of significance. This is an analog of p-values. It incorporates the dependence structure if `modeltype = "HMM"` or `modeltype = "NHMM"`. This quantity will be used to declare statistically significant probes/tests.

5.2 BIC

Bayesian Information Criterion scores for the fitted model. This can be used to rank and compare competing models and to select the number of mixture components for `alttype = "mixnormal"`.

5.3 Other returned values

- pii: Initial probabilities.
- A: Transition probability matrix.
- f0: Null distribution f_0.
- f1: Alternative distribution f_1.
- logL: Log likelihood of the final iteration.
- trans.par2: Transition parameters for State 1.
- logL.iter: Log likelihood from each iteration. Can be used to monitor convergence.

6 Simulation Example

To load the package

```
> library(NHMMfdr)
Loading required package: MASS
Loading required package: locfdr
Loading required package: splines
```

Let us simulate a NHMM model with 2000 observations, i.e., the length of the Markov chain is 2000. We first simulate a covariate Z (other than inter-probe distance) from a Gaussian distribution. The simulation which involves inter-probe distance is presented later.

```
> ### Set the random seed to reproduce the results presented here
> set.seed(1234)
> NUM1 <- 2000
> Z <- rnorm(NUM1)
> Z <- matrix(Z,ncol=1)
> Z <- apply(Z,2,scale)
```

`NHMMfdr` assumes that the non-stationary hidden state transition follows a logistic regression:

$$\begin{aligned} \pi_s(\boldsymbol{x}) &= P(\theta_1 = s|\boldsymbol{X}_1 = \boldsymbol{x}) = \frac{\exp(\lambda_s + \boldsymbol{\rho}_s^t \boldsymbol{x})}{\sum_{s=0}^{1} \exp(\lambda_s + \boldsymbol{\rho}_s^t \boldsymbol{x})}, \\ a_{rs}(\boldsymbol{x}) &= P(\theta_j = s|\theta_{j-1} = r, \boldsymbol{X}_j = \boldsymbol{x}) = \frac{\exp(\sigma_{rs} + \boldsymbol{\rho}_s^t \boldsymbol{x})}{\sum_{s=0}^{1} \exp(\sigma_{rs} + \boldsymbol{\rho}_s^t \boldsymbol{x})}, \text{ for } j \geq 2, \end{aligned}$$

where λ_s, $\sigma_{rs} \in \mathbb{R}$ and $\boldsymbol{\rho}_s \in \mathbb{R}^D$ are the parameters in the transition probabilities and θ_j denote the hidden state $r, s \in \{0, 1\}$. Here $\boldsymbol{X}_j$ denotes a matrix of D columns with candidate covariates including probe spacing, assay type and genomic annotations. For identifiability, we set $\lambda_0 = \sigma_{00} = \sigma_{10} = \boldsymbol{\rho}_0 = 0$.

NOTE: When probe spacing/inter-probe distance is included as a covariate, the formulation in the transition probability matrix is modified to ensure that probability of self transition decreases with distance. See Kuan and Chiang (2012) for further details.

Now, in our simulation, let us set $(\lambda_1, \sigma_{01}, \sigma_{11}, \rho_1)$= c(0.3, -2, -1, 1) and compute the transition probabilities given the transition parameters as follows:

```
> trans.par1.true <- c(0,0,0,0)
> trans.par2.true <- c(0.3,-2,-1,1)

> ########################################
> # compute the transition probabilities
> # given transition parameters
> ########################################

> A.true <- compute.A.nhmm(Z, trans.par1.true, trans.par2.true,
+ dist.included = FALSE)$A
> pii.true <- compute.A.nhmm(Z, trans.par1.true, trans.par2.true,
+ dist.included = FALSE)$pii
```

Now assume that under the null hypothesis, $x \sim f_0 = N(0, 1)$ and under the alternative hypothesis $x \sim f_1 = N(3, 1)$. We can now simulate the NHMM data with function `simdata.nhmm` as follows.

```
> ### the null distribution
```

```
> f0 <- c(0, 1)

> ### the alternative distribution
> f1 <- c(3, 1)

> ### simulate NHMM data

> simdat <- simdata.nhmm(NUM1, pii.true, A.true, f0, 1, f1)

> ### simulated observed values
> x <- simdat$o

> ### simulated unobserved true states
> theta1 <- simdat$s

> table(theta1)
theta1
   0    1
1640  360
```

Note that we simulated 360 tests to be true positives. We can now apply our `fdr.nhmm` function to this simulated data.

```
> #################
> # Model fitting
> #################
>
> fit.nhmm <- fdr.nhmm(x, Z, dist = NULL, log.transform.dist = FALSE,
+ alttype ='mixnormal', L=1, maxiter = 100, nulltype = 0, modeltype = 'NHMM')
NOTE: symmetric option is only available for alttype kernel
Running with alltype  mixnormal , nulltype 0 , modeltype  NHMM ...
Scaling covariates
DONE!

> str(fit.nhmm)
List of 11
 $ pii       : num [1:2] 1.0 3.6e-08
 $ A         : num [1:2, 1:2, 1:1999] 0.848 0.751 0.152 0.249 0.712 ...
 $ f0        : num [1:2] 0 1
 $ f1        : num [1:2] 3.069 0.903
 $ LIS       : num [1:2000] 1 0.997 0.995 0.968 0.747 ...
 $ logL      : num -3409
```

```
 $ BIC       : num -3432
 $ ni        : num 9
 $ trans.par2: num [1:4] -15.931 -2.001 -1.388 0.995
 $ converged : logi TRUE
 $ logL.iter : num [1:9] -4231 -3419 -3410 -3409 -3409 ...


> fit.nhmm$f1
[1] 3.0688593 0.9034724

> fit.nhmm$trans.par2
[1] -15.931395  -2.001475  -1.388179   0.994551
```

The estimated parameters are close to the true parameters. Note that the estimated parameter for initial distribution $\hat{\lambda}_0$ is off. This is usually the case because we only have one Markov chain, and this has negligible effect on the inference.

Suppose we want to identify which observations are coming from the alternative hypothesis at FDR level 0.1. This can be carried out using the function `LIS.adjust`. Note that the adjusted LIS in `LIS.adjust$aLIS` is analogous to adjusted p-values and can be used to identify statistically significant probes at other FDR level, by selecting the observations with adjusted LIS $\leq$ FDR level. In our simulation example, we declare 338 tests to be significant at FDR level 0.1. Among these 338 tests, 29 of them are false positives which is equivalent to empirical FDR of 0.086.

```
> ################
> # Adjust LIS
> ################
>
> LIS.adjust <- LIS.adjust(fit.nhmm$LIS, fdr = 0.1, adjust = TRUE)
>
> str(LIS.adjust)
List of 2
 $ States: num [1:2000] 0 0 0 0 0 1 0 0 0 0 ...
 $ aLIS  : num [1:2000] 0.831 0.598 0.575 0.386 0.162 ...

> ### tests which are statistically significant
> sig.test <- which(LIS.adjust$States == 1)
> length(sig.test)
[1] 338
>
> ### Number of true positive
> TP <- length(which(LIS.adjust$States == 1 & theta1 == 1))
> TP
```

```
[1] 309
>
> ### Number of false positive
>
> FP <- length(which(LIS.adjust$States == 1 & theta1 == 0))
> FP
[1] 29
>
> ### Empirical FDR
>
> emp.fdr <- FP/length(sig.test)
> emp.fdr
[1] 0.08579882
```

Now, let us simulate inter-probe distance from a Uniform distribution. Set $(\lambda_1, \sigma_{01}, \sigma_{11}, \rho_1)$= c(0.3,-3,-1,2,-1). Note that the coefficient for inter-probe distance (=2 in this simulation) has to be positive to ensure that probability of self transition decreases with inter-probe distance.

```
> set.seed(1234)
> NUM1 <- 2000
> Z <- rnorm(NUM1)
> Z <- matrix(Z,ncol=1)
> Z <- apply(Z,2,scale)
> Dist <- runif(NUM1,min=0,max=1)
> log2Dist <- log2(Dist+2)
> covariate <- cbind(log2Dist,Z)
> trans.par1.true <- c(0,0,0,0,0)
> trans.par2.true <- c(0.3,-3,-1,2,-1)

> ########################################
> # compute the transition probabilities
> # given transition parameters
> ########################################

> A.true <- compute.A.nhmm(covariate, trans.par1.true, trans.par2.true,
+ dist.included=TRUE)$A
> pii.true <- compute.A.nhmm(covariate, trans.par1.true, trans.par2.true,
+ dist.included=TRUE)$pii

> ### simulate NHMM data
```

```
> simdat <- simdata.nhmm(NUM1, pii.true, A.true, f0, 1, f1)

> ### simulated observed values
> x <- simdat$o

> ### simulated unobserved true states
> theta1 <- simdat$s

> table(theta1)
theta1
   0    1
1398  602


> #################
> # Model fitting
> #################

> fit.nhmm <- fdr.nhmm(x, Z, dist = Dist, log.transform.dist = TRUE,
+ alttype ='mixnormal', L=1, maxiter = 100, nulltype = 0, modeltype = 'NHMM')
NOTE: symmetric option is only available for alttype kernel
Transforming distance
Running with alltype  mixnormal , nulltype 0 , modeltype  NHMM ...
Scaling covariates
DONE!

> fit.nhmm$f1
[1] 3.038126 1.019147

> fit.nhmm$trans.par2
[1]  0.2144416 -3.0082774 -1.4076304  1.9348853 -1.0574905
```

References

Efron, B. (2004), "Large-scale simultaneous hypothesis testing: the choice of a null hypothesis," *Journal of the American Stat. Assoc*, 99, 96–104.

— (2007), "Size, power and false discovery rates," *Annals of Statistics*, 35, 1351–1377.

Kuan, P. and Chiang, D. (2012), "Integrating Prior Knowledge in Multiple Testing Under Dependence with Applications in Detecting Differential DNA Methylation," *To appear in Biometrics, Early view online version doi: 10.1111/j.1541-0420.2011.01730.*

Sun, W. and Cai, T. (2009), "Large-scale multiple testing under dependence," *J. R. Stat. Soc B*, 71, 393–424.