methylDMV Package (Beta Version)
Usage Tips

Pei Fen Kuan
July 12, 2016

1 Citing methylDMV

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


2 Introduction

This is an example of using methylDMV package in R. methylDMV implements differential mean (DM) and differential variability (DV) analysis for DNA methylation using the approach described in Kuan et al. (2016). This vignette aims to demonstrate the usage of methylDMV through an example data. methylDMV is an evolving package in which new functions will be added from time to time. The current version of methylDMV implements the logistic regression for case control studies. Users can customize the functions of this package to be applied to other types of data and phenotype.

3 Description and Usage

The main function in methylDMV package is methylDMV() which implements simultaneous differential mean (DM) and differential variability (DV) analysis of DNA methylation; and candidate CpGs selection for prediction algorithm.

methylDMV(beta.mat,Y,confound.mat=NULL,logit.transform=TRUE)
4 Arguments

4.1 beta.mat

Matrix of beta values from DNA methylation of size $p \times n$, where $p$ is the number of CpGs and $n$ is the number of samples. Each column corresponds to a sample, and each row corresponds to a CpG. Please ensure that the beta values are in between 0 and 1. The beta matrix will be internally transformed using logit function if `logit.transform` is set to be TRUE. The probe IDs are denoted by row names of `beta.mat`.

```r	names(beta.mat) <- ProbeID
```

4.2 Y

Vector of length $n$, taking values 0’s and 1’s for the group membership. By convention, 0: control (e.g., normal) and 1: case (e.g., tumor).

4.3 confound.mat

Matrix of confounding variables of size $n \times K$, where $K$ is the number of variables. Each column corresponds to a confounding variable, e.g., age. If there is no confounding variable, set `confound.mat=NULL`.

4.4 logit.transform

Logical value indicating if a logit transformation should be performed on `beta.mat` or not. Default value is TRUE for methylation beta values.

Note if the user is using this package to analyze other types of genomic data, for example gene expression; then `beta.mat` is the matrix of log2 transformed gene expression values and set `logit.transform=FALSE`.

5 Value

`methylDMV` returns a list of two items. The first item is `methylDMV.res` which is the main results from `methylDMV()`. This is a data frame with nine columns. Each row corresponds to the result of a CpG. The nine columns in `methylDMV.res` are:

1. ProbeID: Probe ID which is also the row names of `beta.mat`.
2. pv_DM: P-value from Rao’s efficient score test for differential mean (DM).
3. adj.pv_DM: FDR adjusted p-value from DM analysis.
4. $p_{v_{DV}}$: P-value from Rao’s efficient score test for differential variability (DV).

5. $\text{adj.} p_{v_{DV}}$: FDR adjusted p-value from DV analysis.

6. $p_{v_{DMV}}$: P-value from Rao’s efficient score test for DM and DV. CpGs which exhibit either DM or DV, or both will have significant $p_{v_{DMV}}$.

7. $\text{adj.} p_{v_{DMV}}$: FDR adjusted p-value from DM and DV.

8. **BestModel**: Best model ranked by BIC scores. 1: (None), 2: DM only, 3: DV only, 4: Both DM and DV. In other words, if CpG $j$ takes values 4, this means that the model which considers both DM and DV has the lowest BIC score among the four models compared.

9. **WeightedMedian**: Weighted median after logit transformation. To be used for scaling the logit transformed beta values in the test set for prediction algorithm.

The second item is **BICmat** which a matrix consisting of five columns. Each row corresponds to the result of a CpG in the same order as **beta.mat** and **methylDMV.res**. Columns 1-4 correspond to the BIC scores from Model 1 to 4 of Kuan et al. (2016), respectively; where

- Model 1: only confounders are included in the model. If this model has the lowest BIC score, this indicates that CpG $j$ does not exhibit DM or DV.
- Model 2: confounders and methylation values are included in the model. If this model has the lowest BIC score, this indicates that CpG $j$ exhibits DM.
- Model 3: confounders and absolute deviations of methylation values are included in the model. If this model has the lowest BIC score, this indicates that CpG $j$ exhibits DV.
- Model 4: confounders, methylation values and absolute deviations are included in the model. If this model has the lowest BIC score, this indicates that CpG $j$ exhibits both DM and DV.

Column 5 reports the number of samples analyzed for each CpG, i.e., number of non-missing values for the CpG.

### 6 Example

You may download the package from
[http://www.ams.sunysb.edu/~pfkuan/methylDMV/methylDMV_1.0.tar.gz](http://www.ams.sunysb.edu/~pfkuan/methylDMV/methylDMV_1.0.tar.gz)

To install the package
> install.packages('methylDMV_1.0.tar.gz',type='source')

To load the package

> library(methylDMV)

An example data with 5000 CpGs is stored in exampleData. exampleData is a list consisting of 3 items. Y is an example of case control vector for 138 samples. beta.mat is an example of matrix of beta values. confound.mat is an example of matrix of two confounding variables, namely race and age.

> data(exampleData)

> names(exampleData)
[1] "Y" "beta.mat" "confound.mat"

> str(exampleData)
List of 3
$ Y : num [1:138] 0 0 0 0 0 1 0 1 1 0 ... 
$ beta.mat : num [1:5000, 1:138] 0.067 0.0915 0.1232 0.9237 0.5525 ... 
  ..- attr(*, "dimnames")=List of 2 
  .. ..$: chr [1:5000] "cg26932600" "cg00579379" "cg00036115" "cg21415998" ... 
  .. ..$: chr [1:138] "Sample1" "Sample2" "Sample3" "Sample4" ... 
$ confound.mat:'data.frame': 138 obs. of 2 variables: 
  ..$ race: chr "white" "white" "white" "white" "white" ... 
  ..$ age : num [1:138] 24 17 25 37 17 41 14 23 37 31 ...

> dim(exampleData$beta.mat)
[1] 5000 138

> head(rownames(exampleData$beta.mat))

> length(exampleData$Y)
[1] 138

> str(exampleData$confound.mat)
'data.frame': 138 obs. of 2 variables: 
$ race: chr "white" "white" "white" "white" "white" ... 
$ age : num 24 17 25 37 17 41 14 23 37 31 ...

> dim(exampleData$confound.mat)
[1] 138 2
To run methylDMV:

```r
> fit.methylDMV <- methylDMV(beta.mat=exampleData$beta.mat,Y=exampleData$Y,
confound.mat=exampleData$confound.mat,logit.transform=TRUE)
Analyzing 5000 CpGs...
0 CpGs omitted due to excessive missing values...
Looping over 5000 CpGs...
Finished 1000 CpGs
Finished 2000 CpGs
Finished 3000 CpGs
Finished 4000 CpGs
Finished 5000 CpGs
DONE!
```

```r
> str(fit.methylDMV)
List of 2
  $ methylDMV.res:'data.frame': 5000 obs. of 9 variables:
    ..$ ProbeID : Factor w/ 5000 levels "cg00001534","cg00004979",...
    ..$ pv_DM : num [1:5000] 0.1726 0.0327 0.678 0.714 0.0713 ...
    ..$ adj.pv_DM : num [1:5000] 0.293 0.078 0.783 0.808 0.147 ...
    ..$ pv_DV : num [1:5000] 0.1375 0.2378 0.7933 0.0476 0.8276 ...
    ..$ adj.pv_DV : num [1:5000] 0.381 0.526 0.931 0.19 0.944 ...
    ..$ pv_DMV : num [1:5000] 0.113 0.1 0.851 0.112 0.195 ...
    ..$ adj.pv_DMV : num [1:5000] 0.207 0.189 0.903 0.206 0.318 ...
    ..$ BestModel : num [1:5000] 5 5 5 5 5 5 5 2 5 2 ...
    ..$ WeightedMedian: num [1:5000] -2.984 -2.676 -2.043 2.604 0.571 ...
  $ BICmat : num [1:5000, 1:5] 176 176 172 175 175 ...
```

To identify the CpGs which are significant at FDR = 0.05 for differential mean (DM) and differential variability (DV), respectively:

```r
### DM ###
> id.DM <- which(fit.methylDMV$methylDMV.res$adj.pv_DM<=0.05)
> length(id.DM)
[1] 1912

### DV ###
> id.DV <- which(fit.methylDMV$methylDMV.res$adj.pv_DV<=0.05)
```
To identify the CpGs which are significant at FDR = 0.05 for either differential mean (DM) or differential variability (DV) or both:

```r
> id.DMV <- which(fit.methylDMV$methylDMV.res$adj.pv_DMV<=0.05)
> length(id.DMV)
[1] 1909
```

NOTE: `methylDMV()` will automatically filter CpGs which have too many missing values. In particular, if a CpG has non-missing values in fewer than 3 samples in each group, it will be omitted from analysis. If the user prefers to only analyze CpGs with at least `p`% (e.g., 90%) of non-missing values, the user can post-process the output `fit.methylDMV` as follows:

```r
> id.include <- which(fit.methylDMV$BICmat[,5]>round(0.9*dim(exampleData$beta.mat)[2]))
> length(id.include)
[1] 4919
```

```r
> ### adjust the p-values via FDR for this subset of CpGs ###
> sub.methylDMV.res <- fit.methylDMV$methylDMV.res[id.include,]
> sub.methylDMV.res$adj.pv_DM <- p.adjust(sub.methylDMV.res$pv_DM,method='BH')
> sub.methylDMV.res$adj.pv_DV <- p.adjust(sub.methylDMV.res$pv_DV,method='BH')
> sub.methylDMV.res$adj.pv_DMV <- p.adjust(sub.methylDMV.res$pv_DMV,method='BH')
> dim(sub.methylDMV.res)
[1] 4919 9
```

```r
> sub.BICmat <- fit.methylDMV$BICmat[id.include,]
> dim(sub.BICmat)
[1] 4919 5
```

If the user wants to select CpGs which rank Model 4 as the best model as candidate feature set in the prediction algorithm,

```r
> selectCpG <- fit.methylDMV$methylDMV.res$ProbeID[which(fit.methylDMV$methylDMV.res$BestModel==4)]
> length(selectCpG)
[1] 278
```

```r
> head(selectCpG)
[1] cg22514112 cg07908868 cg08259168 cg17070108 cg18305271 cg22547559
```
The users can then proceed to include the selected CpGs in their favorite prediction algorithms (e.g., \texttt{glmnet} or \texttt{randomForest}). If the deviation measure is going to be included as candidate feature set in the prediction algorithm, it is important to calibrate the deviation measure of the test set using the weighted median computed from the training set. For example, assuming that \texttt{exampleData} is the training data, the weighted median is stored in \texttt{fit.methylDMV$methylDMV.res$WeightedMedian}. Please refer to Kuan et al. (2016) for details.