

# coMethDMR's User Guide

Lisette Gomez, Gabriel Odom, Lily Wang

June 1, 2019

coMethDMR is an R package that identifies genomic regions that are both co-methylated and differentially methylated in Illumina array datasets. Instead of testing all CpGs within a genomic region, coMethDMR carries out an additional step that selects co-methylated sub-regions first without using any outcome information. Next, coMethDMR tests association between methylation within the sub-region and continuous phenotype using a random coefficient mixed effects model, which models both variations between CpG sites within the region and differential methylation simultaneously. coMethDMR is available from GitHub, and will be submitted to Bioconductor soon.

## 1. Quick start

### 1.1 Installation

The latest version can be installed by

```
library(devtools)
install_github("lisettegomez/coMethDMR")
```

After installation, the coMethDMR package can be loaded into R using:

```
library(coMethDMR)
```

### 1.2 Datasets

The input of coMethDMR are methylation beta values. We assume quality control and normalization of the methylation dataset have been performed, by R packages such as minfi or RnBeads. For illustration, we use a subset of prefrontal cortex methylation data (GEO GSE59685) from a recent Alzheimer's disease epigenome-wide association study which was described in Lunnon et al. (2014). This example dataset contains beta values for 8552 CpGs on chromosome 22 for a random selection of 20 subjects.

```
data(betaMatrixChr22_df)
betaMatrixChr22_df [1:5, 1:5]
```

##	GSM1443279	GSM1443663	GSM1443434	GSM1443547	GSM1443577
## cg00004192	0.9249942	0.8463296	0.8700718	0.9058205	0.9090382
## cg00004775	0.6523025	0.6247554	0.7573476	0.6590817	0.6726261
## cg00012194	0.8676339	0.8679048	0.8484754	0.8754985	0.8484458
## cg00013618	0.9466056	0.9475467	0.9566493	0.9588431	0.9419563
## cg00014104	0.3932388	0.5525716	0.4075900	0.3997278	0.3216956

The corresponding phenotype dataset included variables `stage` (Braak AD stage), `subject.id`, `Mplate` (batch effect), `Sex`, `Sample` and `age.brain` (age of the brain donor).

```
data(pheno_df)
```

```
head(pheno_df)
```

##	stage	subject.id	Mplate	sex	Sample	age.brain
## 3	0	1	6042316048	Sex: FEMALE	GSM1443251	82
## 8	2	2	6042316066	Sex: FEMALE	GSM1443256	82
## 10	NA	3	6042316066	Sex: MALE	GSM1443258	89
## 15	1	4	7786923107	Sex: FEMALE	GSM1443263	81
## 21	2	5	6042316121	Sex: FEMALE	GSM1443269	92
## 22	1	6	6042316099	Sex: MALE	GSM1443270	78

### 1.3 Vanilla analysis

We are interested in identifying co-methylated genomic regions associated with AD stages (`stage` treated as a linear variable). Here we illustrate analysis of genomic regions mapped to CpG islands, however the workflow can be similarly conducted for other types of genomic region as well. See section 2.1 below.

There are several steps: (1) obtain CpGs located closely (see details in Section 2.1 below) in genomic regions mapped to CpG islands, (2) identify co-methylated regions, and (3) test co-methylated regions against the outcome variable AD stage.

For the first step, we use the following commands:

```
CpGisland_ls <- readRDS(  
  system.file (  
    "extdata",  
    "CpGislandsChr22_ex.RDS",  
    package = 'coMethDMR',  
    mustWork = TRUE  
  )  
)
```

Here, `CpGisland_ls` is a list of 20 items, with each item of the list including a group of CpG probe IDs located closely within a particular CpG island region. Section 2.1 discusses how to import additional types of genomic regions.

Next, we identify co-methylated regions:

```
coMeth_ls <- CoMethAllRegions (  
  betaMatrix = betaMatrixChr22_df,  
  betaToM = TRUE,  
  CpGs_ls = CpGisland_ls,  
  arrayType = "450k",  
  returnAllCpGs = FALSE,  
  output = "CpGs"  
)
```

```

coMeth_ls

## $`chr22:18268062-18268249`
## [1] "cg12460175" "cg14086922" "cg21463605"
##
## $`chr22:18324579-18324769`
## [1] "cg19606103" "cg14031491" "cg03816851"
##
## $`chr22:18531243-18531447`
## [1] "cg25257671" "cg06961233" "cg08819022"

```

coMeth\_ls is list with that contains groups of CpG probeIDs corresponding to co-methylated regions. Three comethylated regions were identified in this example.

If we want to look at co-methylation within the first co-methylated region:

```

WriteCorrPlot <- function (beta_mat){

  require (corrplot)
  require (coMethDMR)

  CpGs_char <- row.names (beta_mat)

  CpGsOrd_df <- OrderCpGsByLocation(
    CpGs_char, arrayType=c("450k"), output = "dataframe"
  )

  betaOrdered_mat <- t(beta_mat [CpGsOrd_df$cpg ,])

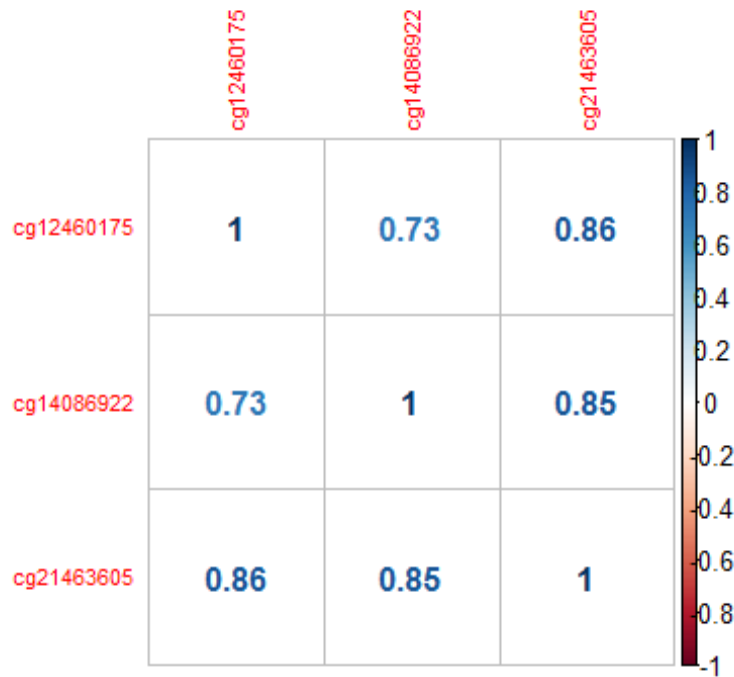
  corr <- cor (
    betaOrdered_mat, method = "spearman", use = "pairwise.complete.obs"
  )

  corrplot(corr, method="number", number.cex = 1, tl.cex = 0.7)
}

# subsetting beta values to include only co-methylated probes
betas_df <- subset(
  betaMatrixChr22_df,
  row.names(betaMatrixChr22_df) %in% coMeth_ls [[1]]
)

WriteCorrPlot (betas_df)

```



Next, we test these co-methylated regions against stage using a random coefficient model (more details in section 2.3 below), adjusting for age.brain.

```
out_df <- lmmTestAllRegions(
  beta_df = betaMatrixChr22_df,
  region_ls = coMeth_ls,
  pheno_df,
  contPheno_char = "stage",
  covariates_char = "age.brain",
  modelType = "randCoef",
  arrayType = "450k"
)
```

out\_df

```
##   chrom   start     end nCpGs   Estimate   StdErr   Stat
## 1 chr22 18268062 18268249     3 -0.07320788 0.03943577 -1.856383
## 2 chr22 18324579 18324769     3  0.03061423 0.02953080  1.036688
## 3 chr22 18531243 18531447     3 -0.07170046 0.04631595 -1.548073
##      pValue      FDR
## 1 0.06339902 0.1824072
## 2 0.29988117 0.2998812
## 3 0.12160480 0.1824072
```

Here out\_df is a data frame of genomic regions, with corresponding p-values and false discovery rate (FDRs) from the random coefficient mixed model.

To further examine the significant regions, we can also extract individual CpG p-values within these significant regions. For example, for the most significant region chr22:18268062-18268249,

```
outCpGs_df <- CpGsInfoOneRegion(
  regionName_char = "chr22:18268062-18268249",
  betas_df = betaMatrixChr22_df,
  pheno_df, contPheno_char = "stage",
  covariates_char = "age.brain",
  arrayType = "450k"
)

outCpGs_df
```

##		Region	cpg	chr	pos	slopeEstimate
## 1		chr22:18268062-18268249	cg12460175	chr22	18268062	-0.0387
## 2		chr22:18268062-18268249	cg14086922	chr22	18268239	-0.0795
## 3		chr22:18268062-18268249	cg21463605	chr22	18268249	-0.1015
##	slopePval	UCSC_RefGene_Name	UCSC_RefGene_Accession	UCSC_RefGene_Group		
## 1	0.3152					
## 2	0.0504					
## 3	0.0214					

These CpGs mapped to intergenic regions, so there are no gene names associated with the probes. For genic regions such as chr22:19709548-19709755, we would have results such as the following:

```
CpGsInfoOneRegion(
  regionName_char = "chr22:19709548-19709755",
  betas_df = betaMatrixChr22_df,
  pheno_df, contPheno_char = "stage",
  covariates_char = "age.brain",
  arrayType = "450k"
)

##
```

##		Region	cpg	chr	pos	slopeEstimate
## 1		chr22:19709548-19709755	cg04533276	chr22	19709548	-0.0731
## 2		chr22:19709548-19709755	cg20193802	chr22	19709696	-0.0554
## 3		chr22:19709548-19709755	cg05726109	chr22	19709755	-0.0111
##	slopePval	UCSC_RefGene_Name	UCSC_RefGene_Accession	UCSC_RefGene_Group		
## 1	0.1229	SEPT5		NM_002688		Body
## 2	0.2097	SEPT5;GP1BB		NM_002688;NM_000407		Body;TSS1500
## 3	0.7714	SEPT5;GP1BB		NM_002688;NM_000407		Body;TSS1500

## 2. Details of the coMethDMR workflow

### 2.1 Genomic regions

Genomic regions on the Illumina arrays can be defined based on their relations to genes or CpG Islands. Genomic regions related to genes include TSS1500, TSS200, UTR5, EXON1, GENEBOY and UTR3. Genomic regions related to CGIs are NSHORE, NSHELF, ISLAND, SSHORE and SSHELF.

In coMethDMR package, the probe IDs for CpGs within each genomic region, as defined by Illumina 450K array annotation, were obtained from <https://rforge.net/IMA/>, under "Annotation file". From this list of pre-defined genomic regions, using the function `WriteCloseByAllRegions`, we identified clusters of CpGs located closely (i.e. the maximum separation between any two consecutive probes is 200bp; `maxGap = 200`), and we required each cluster to have at least 3 CpGs (`minCpGs = 3`).

These pre-computed genomic regions can be accessed from our GitHub repository `coMethDMRdata`. For each genomic region, a file with these clusters was created and saved with the suffix '3\_200'. For example, to download genomic regions with cluster of CpGs mapped to CpG islands, we use the following commands:

```
gitHubPath_char <- "https://raw.githubusercontent.com/lisettegomez/coMethDMRdata/master/"

CpGisland_ls <- readRDS (
  url(paste0(gitHubPath_char, "ISLAND_3_200.rds"))
)
```

Here `CpGisland_ls` is a list, with each item containing a character vector of CpGs IDs for a particular CpG island.

Similarly, CpGs in other types of genomic regions such as TSS200 can be extracted using the following commands:

```
TSS200_ls <- readRDS (
  url(paste0(gitHubPath_char, "TSS200_3_200.rds"))
)
```

In addition, customized genomic regions can be specified in a .gmt file ([http://software.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data\\_formats#GMT:\\_Gene\\_Matrix\\_Transposed\\_file\\_format\\_.28.2A.gmt.29](http://software.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data_formats#GMT:_Gene_Matrix_Transposed_file_format_.28.2A.gmt.29)) and read into coMethDMR using the `read_gmt` function.

```
# library(devtools)
# install_github ("gabrielodom/pathwayPCA")
library(pathwayPCA)

TSS200_ls <- read_gmt (
  url(paste0(gitHubPath_char, "TSS200_3_200.gmt")),
```

```

    description = FALSE,
    setType = "regions"
)

names(TSS200_ls$regions) <- TSS200_ls$TERMS

```

To extract clusters of close-by CpGs from pre-defined genomic regions with different values of maxGap and minCpGs, the WriteCloseByAllRegions function can be used.

## 2.2 Identify co-methylated regions

Within each genomic region, we next identify contiguous and co-methylated CpGs sub-regions without using any outcome information. To select these co-methylated sub-regions, we use the rdrop statistic, which is the correlation between each CpG with the sum of methylation levels in all other CpGs. The default is rDropThresh\_num = 0.4. We recommend this setting based on our simulation study. Note that higher rDropThresh\_num values lead to fewer co-methylated regions.

For example, if we are interested in identifying co-methylated sub-region within the first genomic region in Cgi\_ls:

```

Cgi_ls <- readRDS(
  system.file (
    "extdata",
    "CpGislandsChr22_ex.RDS",
    package = 'coMethDMR',
    mustWork = TRUE
  )
)

coMeth_ls <- CoMethAllRegions (
  betaMatrix = betaMatrixChr22_df,
  betaToM = TRUE,
  CpGs_ls = Cgi_ls[1],
  arrayType = "450k",
  returnAllCpGs = FALSE,
  output = "CpGs"
)

coMeth_ls
## NULL

```

The results indicate there is no co-methylated sub-region within the first genomic region.

Next we look at a region (13th region in Cgi\_ls) where there is a co-methylated sub-region:

```

coMeth_ls <- CoMethAllRegions (
  betaMatrix = betaMatrixChr22_df,

```

```

    betaToM = TRUE,
    CpGs_ls = Cgi_ls[13],
    arrayType = "450k",
    returnAllCpGs = FALSE,
    output = "CpGs"
)

```

```
coMeth_ls
```

```
## $`chr22:18268062-18268249`
## [1] "cg12460175" "cg14086922" "cg21463605"
```

coMeth\_ls is a list, where each item is a list of CpG probe IDs for a co-methylated sub-region.

If we want to see the detailed output of the coMethDMR algorithm, that is, how the co-methylated region was obtained, we can specify output = "dataframe":

```

coMethData_ls <- CoMethAllRegions (
  betaMatrix = betaMatrixChr22_df,
  betaToM = TRUE,
  CpGs_ls = Cgi_ls[13],
  arrayType = "450k",
  returnAllCpGs = FALSE,
  output = "dataframe"
)

```

```
coMethData_ls
```

```
## $`chr22:18267969-18268249`
##           Region      CpG   Chr  MAPINFO    r_drop keep
## 1 chr22:18267969-18268249 cg18370151 chr22 18267969 0.3939650    0
## 2 chr22:18267969-18268249 cg12460175 chr22 18268062 0.8341630    1
## 3 chr22:18267969-18268249 cg14086922 chr22 18268239 0.7918945    1
## 4 chr22:18267969-18268249 cg21463605 chr22 18268249 0.8604731    1
##   keep_contiguous
## 1                0
## 2                1
## 3                1
## 4                1
```

coMethData\_ls provides the details on how the co-methylated region was obtained: Here keep = 1 if rDropThresh\_num > 0.4 (i.e. a co-methylated CpG), and keep\_contiguous indicates if the probe is in a contiguous co-methylated region.

## 2.3 Testing genomic regions against a continuous phenotype

To test association between a continuous phenotype and methylation values in a contiguous co-methylated region, two mixed models have been implemented in the



function `lmmTestAllRegions`: a random coefficient mixed model (`modelType = "randCoef"`) and a simple linear mixed model (`modelType = "simple"`).

The random coefficient mixed model includes both a systematic component that models the mean for each group of CpGs, and a random component that models how each CpG varies with respect to the group mean (random probe effects). It also includes random sample effects that model correlations between multiple probes within the same sample.

More specifically, the random coefficient model is `methylation M value ~ contPheno_char + covariates_char + (1|Sample) + (contPheno_char|CpG)`. The last two terms are random intercepts and slopes for each CpG.

The simple linear mixed model includes all the terms in the random coefficient model except random probe effects.

The simple linear mixed model is

`methylation M value ~ contPheno_char + covariates_char + (1|Sample)`

To test one genomic region against the continuous phenotype stage, adjusting for `age.brain`:

```
lmmTestAllRegions(  
  beta_df = betaMatrixChr22_df,  
  region_ls = coMeth_ls[1],  
  pheno_df,  
  contPheno_char = "stage",  
  covariates_char = "age.brain",  
  modelType = "randCoef",  
  arrayType = "450k"  
)  
  
##   chrom   start     end nCpGs   Estimate   StdErr   Stat  
## 1 chr22 18268062 18268249     3 -0.07320788 0.03943577 -1.856383  
##           pValue      FDR  
## 1 0.06339902 0.06339902
```

If we don't want to adjust for any covariate effect, we can set `covariates_char` to `NULL`:

```
lmmTestAllRegions(  
  beta_df = betaMatrixChr22_df,  
  region_ls = coMeth_ls[1],  
  pheno_df,  
  contPheno_char = "stage",  
  covariates_char = NULL,  
  modelType = "randCoef",  
  arrayType = "450k"  
)  
  
##   chrom   start     end nCpGs   Estimate   StdErr   Stat   pValue  
## 1 chr22 18268062 18268249     3 -0.06678558 0.03883719 -1.71963 0.08549977
```

```
##          FDR
## 1 0.08549977
```

### 3. Reference

Lunnon K, Smith R, Hannon E, De Jager PL, Srivastava G, Volta M, Troakes C, Al-Sarraj S, Burrage J, Macdonald R, et al (2014) Methylomic profiling implicates cortical deregulation of ANK1 in Alzheimer's disease. *Nat Neurosci* 17:1164-1170.