Genome-wide methylation analysis using coMethDMR via parallel computing

Gabriel J. Odom, Lissette Gomez, Tiago Chedraoui Silva, Lanyu Zhang and Lily Wang

10/9/2019

1 Introduction

In the previous vignette "Introduction to **coMethDMR**", we discussed components of the **coMethDMR** pipeline and presented example R scripts for running an analysis with **coMethDMR** serially. However, because identifying co-methylated clusters and fitting mixed effects models on large numbers of genomic regions can be computationally expensive, we illustrate implementation of parallel computing for **coMethDMR** via the **BiocParallel** R package in this vignette.

First, we load these two packages.

```
library(coMethDMR)
library(BiocParallel)
```

In Section 2, we give a brief re-introduction to the example data. In Section 3, we present example scripts for analyzing a single type (e.g. genic regions) of genomic regions using parallel computing. In Section 4, we present example scripts for analyzing genic and intergenic regions on the Illumina arrays using parallel computing.

2 Example Dataset

For illustration, we use a subset of prefrontal cortex methylation data (GEO GSE59685) described in Lunnon et al. (2014). This example dataset includes beta values for 8552 CpGs on chromosome 22 for a random selection of 20 subjects. We assume quality control and normalization of the methylation dataset have been performed by R packages such as **minfi** or **RnBeads**.

```
data(betasChr22_df)
betasChr22_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`, `slide` (batch effect), `sex`, `Sample` and `age.brain` (age of the brain donor).

```
data(pheno_df)
head(pheno_df)
```

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

Note that only samples with both methylation data and non-missing phenotype data are analyzed by **coMethDMR**. So in this example, the sample with `subject.id = 3` which lacks AD stage information will be excluded from analysis. Please also note the phenotype file needs to have a variable called “Sample” that will be used by **coMethDMR** to link to the methylation dataset.

3 Analyzing One Type of Genomic Region via BiocParallel

As mentioned previously in “Introduction to **coMethDMR**”, there are several steps in the **coMethDMR** pipeline:

1. Obtain CpGs located closely in pre-defined genomic regions,
2. Identify co-methylated regions, and
3. Test co-methylated regions against the outcome variable (AD stage).

Suppose we are interested in analyzing genic regions on the 450k arrays. In the following, `closeByGeneAll_ls` is a list, where each item includes at least 3 CpGs located closely (max distance between 2 CpGs is 200bp by default).

```
closeByGeneAll_ls <- readRDS(
  system.file(
    "extdata",
    "450k_Gene_3_200.rds",
    package = 'coMethDMR',
    mustWork = TRUE
  )
)
```

We can inspect the first region in the list via:

```
closeByGeneAll_ls[1]
```

```
## $`chr4:91760141-91760229`
## [1] "cg01583657" "cg25325192" "cg06044899"
```

Next, for demonstration, we select regions on chromosome 22.

```
indx <- grep ("chr22", names(closeByGeneAll_ls))

closeByGene_ls <- closeByGeneAll_ls[indx]
length(closeByGene_ls)
```

```
## [1] 676
```

There are 676 genic regions to be tested for chromosome 22. The first region contains the CpGs

```
closeByGene_ls[1]
```

```
## $`chr22:30662799-30663041`
## [1] "cg19018155" "cg10467217" "cg24727122" "cg02597698" "cg25666403"
## [6] "cg02026204" "cg05539509" "cg07498879"
```

3.1 Compute residuals

This step removes uninteresting technical and biological effects using the `GetResiduals` function, so that the resulting co-methylated clusters are only driven by the biological factors we are interested in.

```
# a0 <- Sys.time()
resid_df <- GetResiduals(
  dnam = betasChr22_df,
  # converts to Mvalues for fitting linear model
  betaToM = TRUE,
  pheno_df = pheno_df,
  # Features in pheno_df used as covariates
  covariates_char = c("age.brain", "sex", "slide")
  # cores = 6
)
```

```
## Phenotype data is not in the same order as methylation data. We will use column Sample in phenotype data to put these two files in the same order.
```

```
# Sys.time() - a0
# This returns a matrix, not a data frame.
```

This step took about 37 seconds when computing in serial, and about 82 seconds when computing over 6 cores.

3.2 Finding co-methylated regions

The cluster computing with the **BiocParallel** package involves simply changing the default value of one argument: `nCores_int` . This argument enables you to perform parallel computing when you set the number of cores to an integer value greater than 1. *If you do not know how many cores your machine has, use the `detectCores()` function from the `parallel` package. Note that, if you have many applications open on your computer, you should not use all of the cores available.*

Once we know how many cores we have available, we execute the `CoMethAllRegions()` function using each worker in the cluster, to find co-methylated clusters in the genic regions. This step took about 99 seconds over 6 cores.

```
# a1 <- Sys.time()

coMeth_ls <- CoMethAllRegions(
  dnam = resid_df,
  betaToM = FALSE,
  method = "spearman",
  arrayType = "450k",
  CpGs_ls = closeByGene_ls,
  nCores_int = 6
)

# Sys.time() - a1
# 40 seconds over 6 cores on Mac
# 99 seconds over 6 cores on Windows
```

The object `coMeth_ls` is a list, with each item containing the list of CpGs within an identified co-methylated region.

3.3 Testing the co-methylated regions

Next we test these co-methylated regions against continuous phenotype `stage` , adjusting for covariates `age` and `sex` , by executing the `lmmTestAllRegions()` function.

```
# a2 <- Sys.time()

res_df <- lmmTestAllRegions(
  betas = betasChr22_df,
  region_ls = coMeth_ls,
  pheno_df = pheno_df,
  contPheno_char = "stage",
  covariates_char = c("age.brain", "sex"),
  modelType = "randCoef",
  arrayType = "450k",
  nCores_int = 6,
  outLogFile = "res_lmm_log.txt"
)

# Sys.time() - a2
# 15 seconds over 6 cores on Mac
# 37 seconds over 6 cores on Windows
```

Model fit messages and diagnostics for each region will be saved to the log file specified with the `outLogFile` argument. For a single region, this will return a one row of model fit statistics similar to the following:

chrom	start	end	nCpGs	Estimate	StdErr	Stat	pValue
chr22	24823455	24823519	4	-0.0702	0.0290	-2.4184	0.0155

4 coMethDMR Analysis Pipeline for 450k Methylation Arrays Datasets via BiocParallel

In this section, we provide example scripts for testing genic and intergenic regions using **coMethDMR** and **BiocParallel** R packages.

First, we read in clusters of CpGs located closely on the array, in genic and intergenic regions, then combine them into a single list.

```
closeByGene_ls <- readRDS(
  system.file(
    "extdata",
    "450k_Gene_3_200.RDS",
    package = 'coMethDMR',
    mustWork = TRUE
  )
)

closeByInterGene_ls <- readRDS(
  system.file(
    "extdata",
    "450k_InterGene_3_200.RDS",
    package = 'coMethDMR',
    mustWork = TRUE
  )
)

# put them together in one list
closeBy_ls <- c(closeByGene_ls, closeByInterGene_ls)
```

For demonstration purposes, we will select regions on Chromosome 22 only.

```
# select regions on chrom 22 for demonstration
indx <- grep ("chr22", names(closeBy_ls))

closeByChr22_ls <- closeBy_ls[indx]
length(closeByChr22_ls)
```

```
## [1] 752
```

Next we remove uninteresting biological and technical effects, execute the `CoMethAllRegions()` function over each worker on the cluster (this completes in roughly 2.6 minutes using 20 cores and 64Gb of RAM), then pass these results to `lmmTestAllRegions()`:

```
# aAll <- Sys.time()

residChr22_df <- GetResiduals(
  dnam = betasChr22_df,
  betaToM = TRUE, #converts to Mvalues for fitting linear model
  pheno_df = pheno_df,
  covariates_char = c("age.brain", "sex", "slide"),
  nCores_int = 6
)

coMethChr22_ls <- CoMethAllRegions(
  dnam = residChr22_df,
  betaToM = FALSE, # these are residuals, not beta values
  method = "spearman",
  arrayType = "450k",
  CpGs_ls = closeByChr22_ls,
  nCores_int = 6
)

resChr22_df <- lmmTestAllRegions(
  betas = betasChr22_df,
  region_ls = coMethChr22_ls,
  pheno_df = pheno_df,
  contPheno_char = "stage",
  covariates_char = c("age.brain", "sex", "slide"),
  modelType = "randCoef",
  arrayType = "450k",
  nCores_int = 6,
  outLogFile = paste0("lmm_log.txt")
)

# Sys.time() - aAll
# 3 minutes over 6 cores on Mac
# 4.4 minutes over 6 cores on Windows
```

4.1 Annotate results

Finally, we can annotate the results using the `AnnotateResults()` function.

```
lmmResAnnotatedChr22_df <- AnnotateResults(
  lmmRes_df = resChr22_df,
  arrayType = "450k"
)
head(lmmResAnnotatedChr22_df)
```

5 Implementing Parallel Computing for coMethDMR Analysis of EPIC Methylation Arrays Datasets

The analysis for EPIC methylation arrays would be the same as those for 450k arrays, except by testing genomic regions in files “EPIC_Gene_3_200.RDS” and “EPIC_InterGene_3_200.RDS”, instead of “450k_Gene_3_200.RDS” and “450k_InterGene_3_200,RDS”.

So the following scripts replace those in Section 4.

```
closeByGene_ls <- readRDS(
  system.file(
    "extdata",
    "EPIC_Gene_3_200.RDS",
    package = 'coMethDMR',
    mustWork = TRUE
  )
)

closeByInterGene_ls <- readRDS(
  system.file(
    "extdata",
    "EPIC_InterGene_3_200.RDS",
    package = 'coMethDMR',
    mustWork = TRUE
  )
)
```

6 Additional Comments on Computational Time and Resources

In this vignette, we have analyzed a small subset of a real EWAS dataset (i.e. only chromosome 22 data on 20 subjects). To give users a more realistic estimate of time for analyzing real EWAS datasets, we also measured time used for analyzing the entire Lunnon et al. (2014) dataset with 110 samples on all chromosomes. These computation times measured on a Dell Precision 5810 with 64Gb of RAM, an Intel Xeon E5-2640 CPU at 2.40Ghz, and using up to 20 cores. More specifically, in Section 4, the entire **coMethDMR** workflow for took 103 minutes with 6 cores and used a maximum of 24Gb of RAM (for the `CoMethAllRegions()` function). We're currently working improving the speed and reducing the size of **coMethDMR**, so please check back soon for updates.