Example of Methylome Analysis with MethylIT using Cancer Datasets

Robersy Sanchez[1]
rus547@psu.edu
ORCID: orcid.org/0000-0002-5246-1453

Sally Mackenzie[1]
sam795@psu.edu
ORCID: orcid.org/0000-0003-2077-5607

[1]*Departments of Biology and Plant Science.*
Pennsylvania State University, University Park, PA 16802

11 February 2018

Abstract

Methyl-IT, a novel methylome analysis procedure based on information thermodynamics and signal detection was recently released. Methylation analysis involves a signal detection problem, and the method was designed to discriminate methylation regulatory signal from background noise induced by thermal fluctuations. Methyl-IT enhances the resolution of genome methylation behavior to reveal network-associated responses, offering resolution of gene pathway influences not attainable with previous methods. Herein, an example of MethylIT application to the analysis of breast cancer methylomes is presented.

Contents

1. MethylIT **2**
1.1. *Installation of MethylIT* 2
1.2. *Available datasets* 3
1.3. *Reading dataset* 3

2. The reference individual. Creating a reference individual by pooling the methylation counts. **4**

3. Hellinger divergence estimation **5**
3.1. *Histogram and boxplots of divergences estimated in each sample* 7

4. Nonlinear fit of Weibull distribution **9**

5. Signal detection **11**
5.1. *Potential methylation signal* 11
5.2. *Histogram and boxplots of the methylation potential signal in each sample* 13
5.3. *Cutpoint estimation* 14

6. DIMPs **15**
6.1. *Histogram and boxplots of DIMPs* 15
6.2. *Venn Diagram of DIMPs* 16

7. Differentially informative methylated genomic regions (DIMRs) **18**
7.1. Differentially methylated genes (DMGs) 18

Acknowledgments **22**

Funding **22**

Supplements. **23**
S1. Datasets used in this example 23
S2. Session Information 24

References **26**

1. MethylIT

MethylIT is an R package for methylome analysis based on information thermodynamics and signal detection. The information thermodynamics-based approach is postulated to provide greater sensitivity for resolving true signal from the thermodynamic background within the methylome (Sanchez and Mackenzie 2016). Because the biological signal created within the dynamic methylome environment characteristic of plants is not free from background noise, the approach, designated Methyl-IT, includes the application of signal detection theory (Greiner, Pfeiffer, and Smith 2000; Carter et al. 2016; Harpaz et al. 2013; Kruspe et al. 2017). A basic requirement for the application of signal detection is a probability distribution of the background noise. Probability distribution, as a Weibull distribution model, can be deduced on a statistical mechanical/thermodynamics basis for DNA methylation induced by thermal fluctuations (Sanchez and Mackenzie 2016). Assuming that this background methylation variation is consistent with a Poisson process, it can be distinguished from variation associated with methylation regulatory machinery, which is non-independent for all genomic regions (Sanchez and Mackenzie 2016). An information-theoretic divergence to express the variation in methylation induced by background thermal fluctuations will follow a Weibull distribution model, provided that it is proportional to minimum energy dissipated per bit of information from methylation change.

Herein, we provide an example of MethylIT application to the analysis of breast cancer methylomes. Due to the size of human methylome the current example only covers the analysis of chromosome 13. A full description of MethylIT application of methylome analysis in plants is given in the manuscript (Sanchez et al. 2018).

1.1. *Installation of MethylIT*

Before install MethylIT, please check that both the R and bioconductor packages are up to date:

```
update.packages(ask = FALSE)
source("https://bioconductor.org/biocLite.R")
biocLite(ask = FALSE)
```

MethylIT can be installed from PSU's GitLab by typing in an R console:

```
install.packages("devtools")
devtools::install_git("https://git.psu.edu/genomath/MethylIT")
```

1.2. *Available datasets*

Methylome datasets are available at Gene Expression Omnibus (GEO DataSets). The datasets for our example and others are provided and included in the MethylIT installation. They can be accessed as follow:

```
library(MethylIT)
files = list.files(paste0(system.file(package = "MethylIT"),"/extdata"),
                   pattern = "txt.gz")
files
```

```
## [1] "GSM1279513_Breast_468LN_metastasis_chr13.txt.gz"
## [2] "GSM1279514_Breast_468PT_cancer_chr13.txt.gz"
## [3] "GSM1279517_Breast_normal_chr13.txt.gz"
## [4] "GSM2041690_WGBS_UCLA1_Primed1_chr13.txt.gz"
## [5] "GSM2041691_WGBS_UCLA1_Primed2_chr13.txt.gz"
## [6] "GSM2041692_WGBS_UCLA1_Primed3_chr13.txt.gz"
```

1.3. *Reading dataset*

Function 'readCounts2GRangesList' transforms the read counts data from each methylome into a GRanges object (from the R packages 'GenomicRanges'). The output is a list of GRanges. For example, chromosome 13 from breast tissues (cancer and normal) and embryonic stem cells can be read from the installation folder as:

```
files = paste0(system.file(package = "MethylIT"),"/extdata/", files)
LR = readCounts2GRangesList(files_names = files,
                            sample.id = c("Breast_metastasis","Breast_cancer",
                                          "Breast_normal",
                                          paste0("ESC", 1:3)),
                            columns = c( seqnames = 1, start = 2, mC = 3, uC = 4 ),
                            verbose = FALSE)
LR$Breast_cancer
```

```
## GRanges object with 803708 ranges and 2 metadata columns:
##             seqnames                ranges strand |        mC        uC
##                <Rle>             <IRanges>  <Rle> | <integer> <integer>
##         [1]    chr13   [19020631, 19020631]      * |        14        24
##         [2]    chr13   [19020633, 19020633]      * |        14        25
##         [3]    chr13   [19020643, 19020643]      * |         7        38
##         [4]    chr13   [19020680, 19020680]      * |         1        43
##         [5]    chr13   [19020687, 19020687]      * |         0        46
##         ...      ...                    ...    ... .       ...       ...
```

```
##   [803704]    chr13 [115108776, 115108776]      * |        52        20
##   [803705]    chr13 [115108789, 115108789]      * |        27        43
##   [803706]    chr13 [115108993, 115108993]      * |        72         5
##   [803707]    chr13 [115109023, 115109023]      * |        56        36
##   [803708]    chr13 [115109524, 115109524]      * |        31         9
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

In the metacolumn of the last GRanges object, mC and uC stand for the methylated and unmethylated read counts, respectively.

2. The reference individual. Creating a reference individual by pooling the methylation counts.

To evaluate the methylation differences between individuals from control and treatment we introduce a metric in the bidimensional space of methylation levels $P_i = (p_i, 1 - p_i)$. Vectors P_i provide a measurement of the uncertainty of methylation levels. However, to perform the comparison between the uncertainty of methylation levels from each group of individuals, control (c) and treatment (t), we should estimate the uncertainty variation in respect to the same individual reference on the mentioned metric space. The reason of the last statement resides in that each individual follows an independent ontogenetic development, which is a consequence of the action of the second law of thermodynamics in living organisms.

In the current example, we will create the reference individual by pooling the methylation counts from the embryonic stem cells. It is up to the user whether to apply the 'row sum', 'row mean' or 'row median' of methylated and unmethylated read counts at each cytosine site across individuals:

```r
Ref = poolFromGRlist(list(LR$ESC1, LR$ESC2, LR$ESC3), stat = "median",
                     num.cores = 12L, verbose = FALSE)

Ref
```

```
## GRanges object with 1560637 ranges and 2 metadata columns:
##             seqnames                ranges strand |        mC        uC
##                <Rle>             <IRanges>  <Rle> | <numeric> <numeric>
##         [1]    chr13   [19020631, 19020631]      * |         0         0
##         [2]    chr13   [19020633, 19020633]      * |         2         0
##         [3]    chr13   [19020642, 19020642]      * |         1         0
##         [4]    chr13   [19020643, 19020643]      * |         2         0
##         [5]    chr13   [19020679, 19020679]      * |         1         0
##         ...      ...                    ...    ... .       ...       ...
##   [1560633]    chr13 [115108993, 115108993]      * |         1         1
##   [1560634]    chr13 [115109022, 115109022]      * |         1         0
##   [1560635]    chr13 [115109023, 115109023]      * |         1         0
##   [1560636]    chr13 [115109523, 115109523]      * |         1         0
##   [1560637]    chr13 [115109524, 115109524]      * |         1         0
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Notice that a reference sample could be a virtual methylome as specified above or some specified sample.

3. Hellinger divergence estimation

Now, to perform the comparison between the uncertainty of methylation levels from each group of individuals, control (c) and treatment (t), the divergence between the methylation levels of each individual is estimated in respect to the same reference on the metric space formed by the vector set $P_i = (p_i, 1 - p_i)$ and the Hellinger divergence H. Basically, the information divergence between the methylation levels of an individual j and reference sample r is estimated according to the Hellinger divergence given by the formula:

$$H(\hat{p}_{ij}, \hat{p}_{ir}) = w_i[(\sqrt{\hat{p}_{ij}} - \sqrt{\hat{p}_{ir}})^2 + (\sqrt{1 - \hat{p}_{ij}} - \sqrt{1 - \hat{p}_{ir}})^2]$$

where $w_i = 2\frac{m_{ij}m_{ir}}{m_{ij}+m_{ir}}$, $m_{ij} = {n_i}^{mC_j} + {n_i}^{uC_j} + 1$, $m_{ir} = {n_i}^{mC_r} + {n_i}^{uC_r} + 1$ and $j \in \{c, t\}$.

This equation for Hellinger divergence is given in reference (Basu, Mandal, and Pardo 2010), but others information theoretical divergences can be used as well.

Next, the information divergence for control (Breast_normal) and treatments (Breast_cancer and Breast_metastasis) samples are estimated in respect to the reference virtual individual. A Bayesian correction of counts can be selected or not. In a Bayesian framework, methylated read counts are modeled by a beta-binomial distribution, which accounts for both, the biological and sampling variations (Hebestreit, Dugas, and Klein 2013; Robinson et al. 2014; Dolzhenko and Smith 2014). In our case we adopted the Bayesian approach suggested in reference (Baldi and Brunak 2001) (Chapter 3). In a Bayesian framework with uniform priors, the methylation level can be defined as: $p = (mC + 1)/(mC + uC + 2)$. However, the most natural statistical model for replicated BS-seq DNA methylation measurements is beta-binomial (the beta distribution is a prior conjugate of binomial distribution). We consider the parameter p (methylation level) in the binomial distribution as randomly drawn from a beta distribution. The hyper-parameters α and β from the beta-binomial distribution are interpreted as pseudo-counts. The information divergence is estimated here using the function 'infDivergence':

```
Indiv = list(LR$Breast_normal, LR$Breast_cancer, LR$Breast_metastasis)
names(Indiv) <- c("Breast_normal", "Breast_cancer", "Breast_metastasis")

HD = infDivergence(ref = Ref, indiv = Indiv, Bayesian = TRUE,  min.coverage = 4,
                   high.coverage = 300, percentile = 0.999, num.cores = 12L,
                   tasks = 0L, verbose = FALSE )
HD$Breast_cancer
```

```
## GRanges object with 791245 ranges and 9 metadata columns:
##            seqnames                ranges strand |        c1        t1
##               <Rle>             <IRanges>  <Rle> | <numeric> <numeric>
##        [1]    chr13   [19020631, 19020631]      * |         0         0
##        [2]    chr13   [19020633, 19020633]      * |         2         0
##        [3]    chr13   [19020643, 19020643]      * |         2         0
##        [4]    chr13   [19020680, 19020680]      * |         0         0
```

```
##        [5]    chr13   [19020687, 19020687]      * |         1         0
##        ...      ...                     ...    ... .       ...       ...
##   [791241]    chr13 [115108776, 115108776]      * |         1         0
##   [791242]    chr13 [115108789, 115108789]      * |         3         0
##   [791243]    chr13 [115108993, 115108993]      * |         1         1
##   [791244]    chr13 [115109023, 115109023]      * |         1         0
##   [791245]    chr13 [115109524, 115109524]      * |         1         0
##                   c2        t2                p1                 p2
##            <numeric> <numeric>         <numeric>          <numeric>
##        [1]        14        24 0.264954576121836   0.37780204012486
##        [2]        14        25 0.766218632300514  0.368583236193762
##        [3]         7        38 0.766218632300514  0.172517448216519
##        [4]         1        43 0.264954576121836 0.0457825616393211
##        [5]         0        46 0.688028610158752  0.023032937123599
##        ...       ...       ...               ...                ...
##   [791241]        52        20 0.688028610158752  0.717815024916889
##   [791242]        27        43 0.813069422141182  0.390448796364428
##   [791243]        72         5 0.515586604211175  0.925596444726253
##   [791244]        56        36 0.688028610158752  0.607620134938738
##   [791245]        31         9 0.688028610158752  0.764742183694093
##                            TV              bay.TV               hdiv
##                     <numeric>           <numeric>          <numeric>
##        [1]  0.368421052631579   0.112847464003024  0.0286322422065044
##        [2] -0.641025641025641  -0.397635396106752   0.941775219813708
##        [3] -0.844444444444444  -0.593701184083995    2.21471911969555
##        [4] 0.0227272727272727  -0.219172014482515   0.204926777646219
##        [5]                 -1  -0.664995673035153     2.4711625302714
##        ...                ...                 ...                 ...
##   [791241] -0.277777777777778  0.0297864147581361 0.00413739288457813
##   [791242] -0.614285714285714  -0.422620625776753    1.49962289680636
##   [791243]  0.435064935064935   0.410009840515078    1.37901291763423
##   [791244] -0.391304347826087 -0.0804084752200139  0.0278109185157832
##   [791245]             -0.225  0.0767135735353407  0.0283399173099231
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Function 'infDivergence' returns a list of GRanges objects with the four columns of counts, the information divergence, and additional columns:

1. The original matrix of methylated (c_i) and unmathylated (t_i) read counts from control ($i = 1$) and treatment ($i = 2$) samples.
2. "p1" and "p2": methylation levels for control and treatment, respectively.
3. "bay.TV": total variation TV = p2 - p1.
4. "TV": total variation based on simple counts: $TV = c1/(c1 + t1) - c2/(c2 + t2)$.
5. "hdiv": Hellinger divergence.

If Bayesian = TRUE, results are based on the posterior estimations of methylation levels $p1$ and $p2$. Filtering by coverage is provided at this step, which would be used if not previous filtering by coverage have been applied. This is a pairwise filtering. Cytosine sites with 'coverage' >

'min.coverage' and 'coverage' < 'percentile' (e.g., 99.9 coverage percentile) in at least one of the samples are preserved. The coverage percentile used is the maximum estimated from both samples: reference and individual.

3.1. *Histogram and boxplots of divergences estimated in each sample*

First, the data of interest (Hellinger divergences, "hdiv") are selected from the GRanges objects:

```
normal = HD$Breast_normal[, "hdiv"]
normal = normal[ normal$hdiv > 0 ]
metastasis = HD$Breast_metastasis[, "hdiv"]
metastasis = metastasis[ metastasis$hdiv > 0 ]
cancer = HD$Breast_cancer[, "hdiv"]
cancer = cancer[ cancer$hdiv > 0 ]
```

Next, a single GRanges object is built from the above set GRanges objects using the function 'uniqueGRanges'. Notice that the number of cores to use for parallel computation can be specified.

```
hd = uniqueGRanges(list(normal, cancer, metastasis), missing = NA,
                   verbose = FALSE, num.cores = 12L)
hd
```

```
## GRanges object with 821240 ranges and 3 metadata columns:
##            seqnames                ranges strand |                hdiv
##               <Rle>             <IRanges>  <Rle> |           <numeric>
##        [1]    chr13   [19020631, 19020631]      * |    0.29900661793179
##        [2]    chr13   [19020633, 19020633]      * | 0.00037994395648263
##        [3]    chr13   [19020643, 19020643]      * |  0.0422470312205984
##        [4]    chr13   [19020680, 19020680]      * |  0.0861466701480782
##        [5]    chr13   [19020687, 19020687]      * |   0.382111181938756
##        ...      ...                     ...    ... .                 ...
##   [821236]    chr13 [115108776, 115108776]      * |  0.0206791333698288
##   [821237]    chr13 [115108789, 115108789]      * |   0.184070741986262
##   [821238]    chr13 [115108993, 115108993]      * |     1.2952881688155
##   [821239]    chr13 [115109023, 115109023]      * |   0.222873631170147
##   [821240]    chr13 [115109524, 115109524]      * |   0.117062809736541
##                         hdiv.1             hdiv.2
##                       <numeric>          <numeric>
##        [1]  0.0286322422065044   1.13992463993004
##        [2]   0.941775219813708  0.374550157781891
##        [3]    2.21471911969555  0.671031008209812
##        [4]   0.204926777646219 0.0187956994198048
##        [5]     2.4711625302714   1.61131188657488
##        ...                 ...                ...
##   [821236] 0.00413739288457813   1.56893759546956
##   [821237]    1.49962289680636   5.08307076368914
##   [821238]    1.37901291763423   1.54118355448437
##   [821239]  0.0278109185157832  0.002825180468834
##   [821240]  0.0283399173099231   1.07835351600496
```

```
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Now, the Hellinger divergences estimated for each sample are in a single matrix on the metacolumn of the GRanges object and we can proceed to build the histogram and boxplot graphics for these data.

```r
library(ggplot2) # graphic
library(reshape2) # To reshape the data frame
library(grid) # For multiple plots
library(gridExtra) # For multiple plots
data <- data.frame(normal = hd$hdiv, cancer = hd$hdiv.1, metastasis = hd$hdiv.2)
data = melt(data)
```

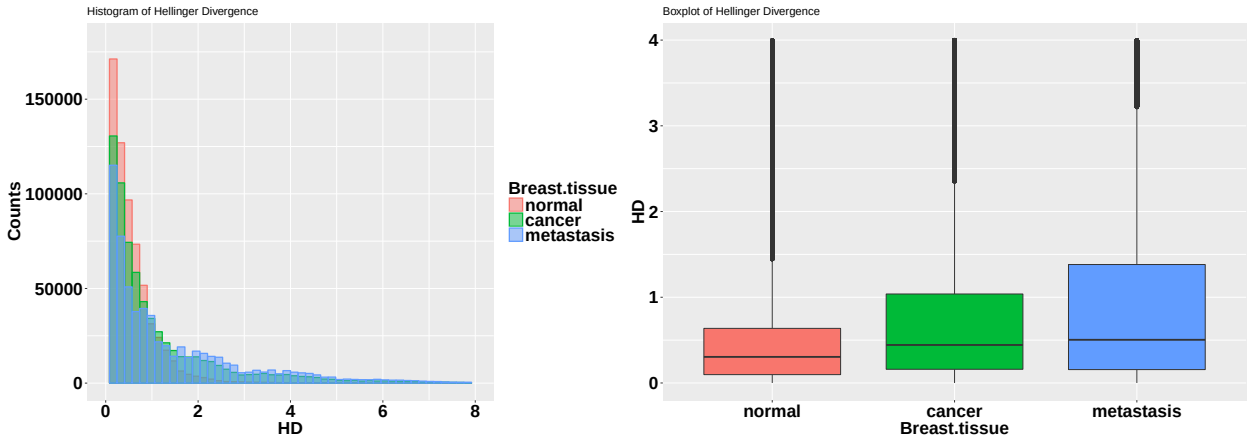
```
## No id variables; using all as measure variables
```

```r
colnames(data) <- c("Breast.tissue", "HD")
head(data)
```

```
##   Breast.tissue          HD
## 1        normal 0.299006618
## 2        normal 0.000379944
## 3        normal 0.042247031
## 4        normal 0.086146670
## 5        normal 0.382111182
## 6        normal 0.760190695
```

```r
# For visualization purposes HD is limited to the interval 0 to 8
p1 = ggplot(data, aes(x = HD, fill = Breast.tissue, colour = Breast.tissue)) +
  geom_histogram(alpha = 0.5, bins = 50, position = "identity", na.rm = TRUE,
                 size = 0.7) +
  theme(axis.title.x = element_text(face = "bold", size = 20),
        axis.text.x = element_text(face = "bold", size = 20, color = "black",
                                   hjust = 0.5, vjust = 0.75),
        axis.text.y = element_text(face = "bold", size = 20, color = "black"),
        axis.title.y = element_text(face = "bold", size = 20,color = "black"),
        legend.text = element_text(size = 20, face = "bold"),
        legend.title = element_text(size = 20, face = "bold")
        ) +
  xlim(0, 8) + ylab( "Counts" ) +
  ggtitle("Histogram of Hellinger Divergence")

# For visualization purposes HD is limited to the interval 0 to 4
dt = data[ which(data$HD < 4), ]
p2 = ggplot(dt,aes(x = Breast.tissue, y = HD , fill = Breast.tissue)) +
  geom_boxplot(na.rm = TRUE) +
  theme(axis.title.x = element_text(face = "bold", size = 20),
        axis.text.x = element_text(face = "bold", size = 20, color = "black",
                                   hjust = 0.5, vjust = 0.75),
        axis.text.y = element_text(face = "bold", size = 20, color = "black"),
```

```
        axis.title.y = element_text(face = "bold", size = 20,color = "black"),
        legend.position = "none"
        ) +
  ggtitle("Boxplot of Hellinger Divergence")
grid.arrange(p1, p2, ncol = 2)
```



Except for the tail, most of the methylation changes occurred under the area covered by the density curve corresponding to the normal breast tissue. This is theoretically expected. This area is explainable in statistical physical terms and, theoretically, it should fit a Weibull distribution. The tails regions cover the methylation changes that, with high probability, are not induced by thermal fluctuation and are not addressed to stabilize the DNA molecule. These changes are methylation signal. Professor David J. Miller (Department of Electrical Engineering, Penn State) proposed modeling of distribution as a mixed Weibull distribution to simultaneously describe the background methylation noise and the methylation signal (personal communication, January, 2018). This model approach seems to be supported by the above histogram, but it must be studied before being incorporated in a future version of Methyl-IT.

4. Nonlinear fit of Weibull distribution

A basic requirement for the application of signal detection is the knowledge of the probability distribution of the background noise. Probability distribution, as a Weibull distribution model, can be deduced on a statistical mechanical/thermodynamics basis for DNA methylation induced by thermal fluctuations (Sanchez and Mackenzie 2016). Assuming that this background methylation variation is consistent with a Poisson process, it can be distinguished from variation associated with methylation regulatory machinery, which is non-independent for all genomic regions (Sanchez and Mackenzie 2016). An information-theoretic divergence to express the variation in methylation induced by background thermal fluctuations will follow a Weibull distribution model, provided that it is proportional to the minimum energy dissipated per bit of information associated with the methylation change.

The nonlinear fit to a Weibull distribution model is performed through the function 'nonlinearFitWD', which is a wrapper of 'Weibull3Ps' function to operate on a list of GRanges.

```r
nlms = nonlinearFitWD(HD, column = 9, num.cores = 3L, verbose = FALSE)

nlms # this returns:
```

```
## $Breast_normal
##        Estimate   Std. Error   t value Pr(>|t|))      Adj.R.Square
## shape 0.8545840 1.101518e-04  7758.243         0 0.995572205269834
## scale 0.4437931 4.096469e-05 10833.552         0
##                     rho       R.Cross.val              DEV
## shape 0.995572194376627 0.998127914492171 300.015811227137
## scale
##                     AIC               BIC     COV.shape     COV.scale
## shape -4118965.90909961 -4118931.08383233  1.213341e-08 -9.030226e-10
## scale                                     -9.030226e-10  1.678105e-09
##       COV.mu      n
## shape     NA 812948
## scale     NA 812948
##
## $Breast_cancer
##        Estimate   Std. Error  t value Pr(>|t|))      Adj.R.Square
## shape 0.7650625 5.547232e-05 13791.79         0 0.998620470750984
## scale 0.7995806 4.898943e-05 16321.49         0
##                     rho      R.Cross.val              DEV
## shape 0.998620467263991 0.99934922817425 90.9620738850743
## scale
##                     AIC               BIC     COV.shape     COV.scale
## shape -4931858.28082519 -4931823.53673639  3.077178e-09 -8.481762e-10
## scale                                     -8.481762e-10  2.399964e-09
##       COV.mu      n
## shape     NA 791245
## scale     NA 791245
##
## $Breast_metastasis
##        Estimate   Std. Error  t value Pr(>|t|))      Adj.R.Square
## shape 0.6829475 3.476534e-05 19644.49         0 0.999305142795581
## scale 1.0593938 5.214489e-05 20316.35         0
##                     rho       R.Cross.val              DEV
## shape 0.999305140907252 0.999658607906166 42.6294379287731
## scale
##                    AIC               BIC     COV.shape    COV.scale COV.mu
## shape -5091668.5701483 -5091634.04339184  1.208629e-09 -4.54658e-10     NA
## scale                                    -4.546580e-10  2.71909e-09     NA
##            n
## shape 735951
## scale 735951
```

Cross-validations for the nonlinear regressions (R.Cross.val) were performed as described in reference (Stevens 2009). In addition, Stein's formula for adjusted R squared (ρ) was used as an estimator of

the average cross-validation predictive power (Stevens 2009).

5. Signal detection

The information thermodynamics-based approach is postulated to provide greater sensitivity for resolving true signal from the thermodynamic background within the methylome (Sanchez and Mackenzie 2016). Because the biological signal created within the dynamic methylome environment characteristic of plants is not free from background noise, the approach, designated Methyl-IT, includes the application of signal detection theory (Greiner, Pfeiffer, and Smith 2000; Carter et al. 2016; Harpaz et al. 2013; Kruspe et al. 2017). Signal detection is a critical step to increase sensitivity and resolution of methylation signal by reducing the signal-to-noise ratio and objectively controlling the false positive rate and prediction accuracy/risk

5.1. *Potential methylation signal*

The first estimation in our signal detection step is the identification of the cytosine sites carrying potential methylation signal PS. The methylation regulatory signal does not hold Weibull distribution and, consequently, for a given level of significance α (Type I error probability, e.g. $\alpha = 0.05$), cytosine positions k with information divergence $H_k >= H_{\alpha=0.05}$ can be selected as sites carrying potential signals PS. The value of α can be specified. For example, potential signals with $H_k > H_{\alpha=0.01}$ can be selected. For each sample, cytosine sites are selected based on the corresponding fitted Weibull distribution model estimated in the previous step. Additionally, since cytosine with $|TV_k| < 0.1$ are the most abundant sites, depending on the sample (experiment), cytosine positions k with $H_k >= H_{\alpha=0.05}$ and $|TV_k| < 0.1$ can be observed. To prevent the last situation we can select the PS with the additional constraint $|TV_k| > TV_0$, where TV_0 ('tv.cut') is a user specified value. The PS is detected with the function 'potentialSignal':

```
PS = potentialSignal(LR = HD, nlms = nlms, div.col = 9, alpha = 0.05,
                     tv.col = 7, tv.cut = 0.2)

PS$Breast_cancer
```

```
## GRanges object with 55068 ranges and 10 metadata columns:
##           seqnames                ranges strand |        c1        t1
##              <Rle>             <IRanges>  <Rle> | <numeric> <numeric>
##       [1]    chr13   [19020862, 19020862]      * |         2         0
##       [2]    chr13   [19026482, 19026482]      * |         2         0
##       [3]    chr13   [19028595, 19028595]      * |         3         0
##       [4]    chr13   [19029464, 19029464]      * |         3         1
##       [5]    chr13   [19029877, 19029877]      * |         2         1
##       ...      ...                    ...    ... .       ...       ...
##   [55064]    chr13 [115079248, 115079248]      * |         4         1
##   [55065]    chr13 [115093831, 115093831]      * |         2         3
##   [55066]    chr13 [115105364, 115105364]      * |         3         0
##   [55067]    chr13 [115105564, 115105564]      * |         2         0
##   [55068]    chr13 [115106665, 115106665]      * |         1         2
```

```
##                  c2        t2                p1                 p2
##           <numeric> <numeric>         <numeric>          <numeric>
##       [1]         1        64 0.766218632300514 0.0314288779884335
##       [2]         0        24 0.766218632300514 0.0425360903302844
##       [3]         1        80 0.813069422141182 0.0253689612788728
##       [4]         0        31 0.677329651772352 0.0335082705163899
##       [5]         1        52 0.612665128583912 0.0382883730073182
##       ...       ...       ...               ...                ...
##   [55064]         3        59 0.723491935315773 0.0641614781574425
##   [55065]        64         2 0.437365343287846  0.957686869754536
##   [55066]         2        62 0.813069422141182 0.0470609281757438
##   [55067]         0        21 0.766218632300514 0.0480886996522946
##   [55068]        79         2 0.412260835026643  0.965335509559106
##                           TV             bay.TV             hdiv
##                     <numeric>          <numeric>        <numeric>
##       [1] -0.984615384615385 -0.734789754312081 4.23510527103979
##       [2]                 -1  -0.72368254197023  3.7109273658828
##       [3] -0.987654320987654 -0.787700460862309 6.55305338136463
##       [4]              -0.75 -0.643821381255962 5.03186897183347
##       [5] -0.647798742138365 -0.574376755576594 3.52316389201568
##       ...                ...                ...               ...
##   [55064] -0.751612903225807  -0.65933045715833 6.04482054891634
##   [55065]   0.56969696969697   0.52032152646669 4.37270623885233
##   [55066]           -0.96875 -0.766008493965438 5.76265595806898
##   [55067]                 -1  -0.71812993264822 3.55138249599445
##   [55068]  0.641975308641975  0.553074674532463 3.45414436972374
##                       wprob
##                    <numeric>
##       [1]  0.0278702553113445
##       [2]  0.0393215749249383
##       [3] 0.00673974174640233
##       [4]  0.0168245409656175
##       [5]  0.0446029226284479
##       ...                 ...
##   [55064]  0.0090927478496443
##   [55065]  0.0255056798009794
##   [55066]  0.0107647053872193
##   [55067]  0.0437617562038316
##   [55068]  0.0467361856694653
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Notice that the total variation distance $|TV|$ is an information divergence as well and it can be used in place of Hellinger divergence (Sanchez and Mackenzie 2016). The set of vectors $P_i = (p_i, 1 - p_i)$ and distance function $|TV|$ integrate a metric space. In particular:

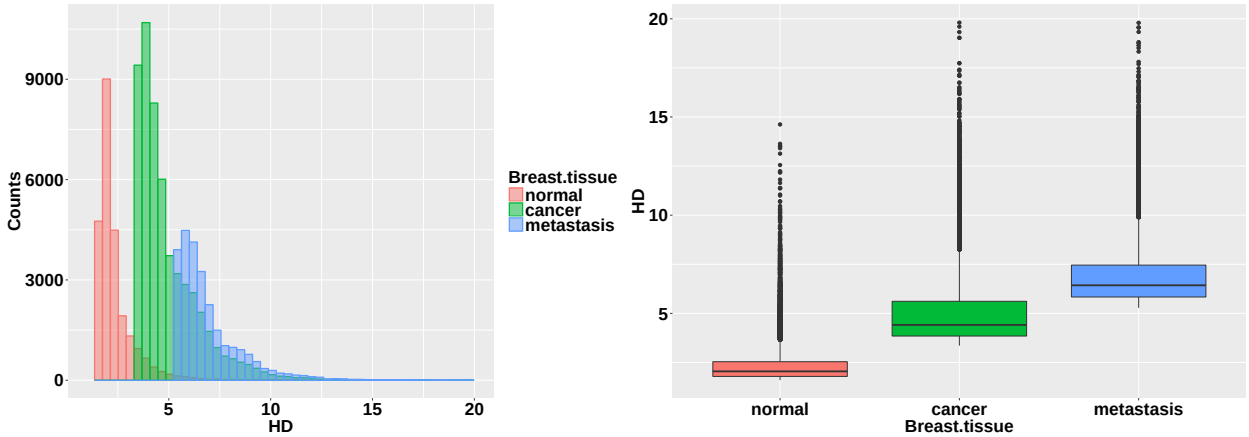
$$|TV| = \frac{1}{2}(|\hat{p}_{ij} - \hat{p}_{ir}| + |(1 - \hat{p}_{ij}) - (1 - \hat{p}_{ir})|) = |\hat{p}_{ij} - \hat{p}_{ir}|$$

That is, the quantitive effect of the vector components $1 - \hat{p}_{ij}$ and $1 - \hat{p}_{ir}$ (in our case, the effect of unmethylated read counts) is not present in TV as in $H(\hat{p}_{ij}, \hat{p}_{ir})$.

5.2. *Histogram and boxplots of the methylation potential signal in each sample*

As before, a single GRanges object is built from the above set GRanges objects using the function 'uniqueGRanges', and the Hellinger divergences of the cytosine sites carrying PS (for each sample) are located in a single matrix on the metacolumn of the GRanges object.

```
ps = uniqueGRanges(PS, missing = NA, verbose = FALSE, num.cores = 12L)
data <- data.frame(normal = ps$hdiv, cancer = ps$hdiv.1, metastasis = ps$hdiv.2)
data = suppressMessages(melt(data))
colnames(data) <- c("Breast.tissue", "HD")

# For visualization purposes HD is limited to the interval 0 to 20
dt = data[ which(data$HD < 20), ]

p1 = ggplot(data, aes(x = HD, fill = Breast.tissue, colour = Breast.tissue)) +
  geom_histogram(alpha = 0.5, bins = 50, position = "identity", na.rm = TRUE,
                 size = 0.7) + xlim(1, 20) + ylab( "Counts" ) +
  theme(axis.title.x = element_text(face = "bold", size = 20),
        axis.text.x = element_text(face = "bold", size = 20, color = "black",
                                   hjust = 0.5, vjust = 0.75),
        axis.text.y = element_text(face = "bold", size = 20, color = "black"),
        axis.title.y = element_text(face = "bold", size = 20,color = "black"),
        legend.text = element_text(size = 20, face = "bold"),
        legend.title = element_text(size = 20, face = "bold")
        )

p2 = ggplot(dt,aes(x = Breast.tissue, y = HD , fill = Breast.tissue)) +
  geom_boxplot(na.rm = TRUE) +
  theme(axis.title.x = element_text(face = "bold", size = 20),
        axis.text.x = element_text(face = "bold", size = 20, color = "black",
                                   hjust = 0.5, vjust = 0.75),
        axis.text.y = element_text(face = "bold", size = 20, color = "black"),
        axis.title.y = element_text(face = "bold", size = 20,color = "black"),
        legend.position = "none"
        )
grid.arrange(p1, p2, ncol = 2)
```



5.3. *Cutpoint estimation*

Laws of statistical physics can account for background methylation, a response to thermal fluctuations that presumably function in DNA stability (Sanchez and Mackenzie 2016). True signal is detected based on the optimal cutpoint (López-Ratón et al. 2014), which can be estimated from the area under the curve (AUC) of a receiver operating characteristic (ROC) curve built from a logistic regression performed with the potential signals from controls and treatments. The ROC AUC is equivalent to the probability that a randomly-chosen positive instance is ranked more highly than a randomly-chosen negative instance (Fawcett 2005). In the current context, the AUC is equivalent to the probability to distinguish a randomly-chosen methylation regulatory signal induced by the treatment from a randomly-chosen signal in the control.

```
cutpoints = cutPointEstimation(PS, control.names = "Breast_normal",
                               treatment.names = c("Breast_cancer",
                                                   "Breast_metastasis"),
                               div.col = 9, verbose = FALSE)
cutpoints
```

```
## $cutpoint
##                   Breast_normal
## Breast_cancer          3.355682
## Breast_metastasis      5.279089
##
## $auc
##                   Breast_normal
## Breast_cancer         0.9542813
## Breast_metastasis     0.9905928
##
## $accuracy
##                   Breast_normal
## Breast_cancer         0.9648128
## Breast_metastasis     0.9897372
```

In practice, potential signals are classified as "control"" (CT) and "treatment"" (TT) signals (prior classification) and the logistic regression (LG): signal (with levels CT (0) and TT (1)) versus

H_k is performed. LG output yields a posterior classification for the signal. Prior and posterior classifications are used to build the ROC curve and then to estimate AUC and cutpoint $H_{cutpoint}$.

6. DIMPs

Cytosine sites carrying a methylation signal are designated *differentially informative methylated positions* (DIMPs). The probability that a DIMP is not induced by the treatment is given by the probability of false alarm (P_{FA}, false positive). That is, the biological signal is naturally present in the control as well as in the treatment. Each DIMP is a cytosine position carrying a significant methylation signal, which may or may not be represented within a differentially methylated position (DMP) according to Fisher's exact test (or other current tests). A DIMP is a DNA cytosine position with high probability to be differentially methylated or unmethylated in the treatment in respect to a given control. Notice that the definition of DIMP is not deterministic in an ordinary sense, but a stochastic-deterministic definition in physico-mathematical terms.

DIMPs are selected with the function:

```
DIMPs = selectDIMP(PS, div.col = 9, cutpoint = 3.355682)
```

6.1. *Histogram and boxplots of DIMPs*

The cutpoint detected with the signal detection step is very close (in this case) to the Hellinger divergence value $H_{\alpha=0.05}$ estimated for cancer tissue. The natural methylation regulatory signal is still present in a patient with cancer and reduced during the metastasis step. This signal is detected here as false alarm (P_{FA}, false positive)

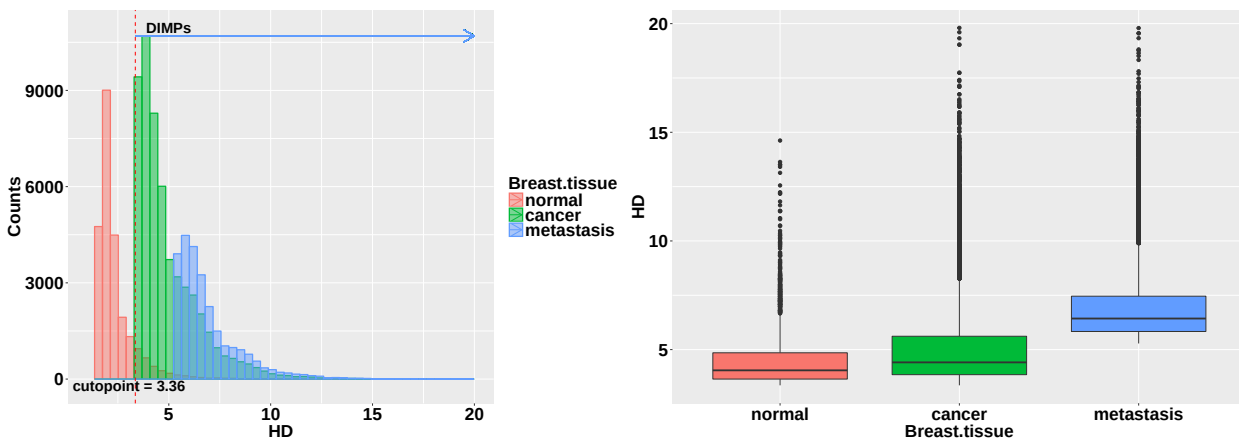
The list GRanges with DIMPs are integrated into a single GRanges object with the matrix of 'hdiv' values on its metacolumn:

```
dimp = uniqueGRanges(DIMPs, missing = NA, verbose = FALSE, num.cores = 12L)
dat <- data.frame(normal = dimp$hdiv, cancer = dimp$hdiv.1,
                  metastasis = dimp$hdiv.2)
dat = suppressMessages(melt(dat))
colnames(dat) <- c("Breast.tissue", "HD")

# For visualization purposes HD is limited to the interval 0 to 20
dt = dat[ which(dat$HD < 20), ]
```

The multiplot with the histogram and the boxplot can now built:

```
p1 = ggplot(data, aes(x = HD, fill = Breast.tissue, colour = Breast.tissue)) +
  geom_histogram(alpha = 0.5, bins = 50, position = "identity", na.rm = TRUE,
                 size = 0.7) + xlim(1, 20) + ylab( "Counts" ) +
  geom_vline(xintercept = 3.355682, color = "red", linetype = "dashed") +
  annotate(geom = "text", x = 3.05, y = -200, fontface = 2, size = 6,
           label = paste0("cutopoint = ", 3.36)) +
  annotate(geom = "text", x = 5, y = 10950, label = "DIMPs",
           fontface = 2, size = 6) +
```

```
  geom_segment(aes(x = 3.36, xend = 20, y = 10700, yend = 10700),
               arrow = arrow(length = unit(0.5, "cm"))) +
  theme(axis.title.x = element_text(face = "bold", size = 20),
        axis.text.x = element_text(face = "bold", size = 20, color = "black",
                                   hjust = 0.5, vjust = 0.75),
        axis.text.y = element_text(face = "bold", size = 20, color = "black"),
        axis.title.y = element_text(face = "bold", size = 20,color = "black"),
        legend.text = element_text(size = 20, face = "bold"),
        legend.title = element_text(size = 20, face = "bold")
        )

p2 = ggplot(dt,aes(x = Breast.tissue, y = HD , fill = Breast.tissue)) +
  geom_boxplot(na.rm = TRUE) +
  theme(axis.title.x = element_text(face = "bold", size = 20),
        axis.text.x = element_text(face = "bold", size = 20, color = "black",
                                   hjust = 0.5, vjust = 0.75),
        axis.text.y = element_text(face = "bold", size = 20, color = "black"),
        axis.title.y = element_text(face = "bold", size = 20,color = "black"),
        legend.position = "none"
        )
grid.arrange(p1, p2, ncol = 2)
```

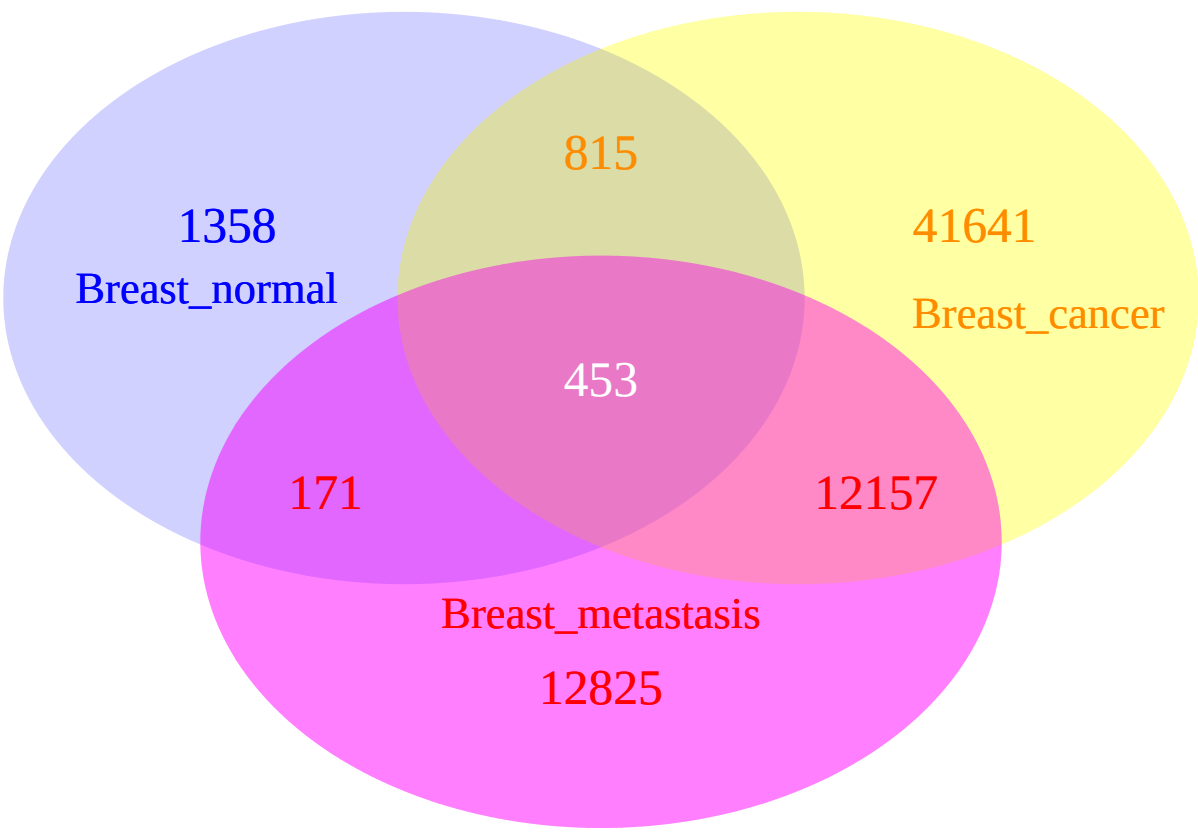


6.2. *Venn Diagram of DIMPs*

The Venn diagram of DIMPs reveals that the number cytosine site carrying methylation signal with a divergence level comparable with that one observed in breast tissues with cancer and metastasis is relatively small (2797 DIMPs). The number of DIMPs decreased in the breast tissue with metastasis, but as shown in the last boxplot the intensity of the signal increased.

```
suppressMessages(library(VennDiagram))

n12 = length(GenomicRanges::intersect(DIMPs$Breast_normal, DIMPs$Breast_cancer))
n13 = length(GenomicRanges::intersect(DIMPs$Breast_normal, DIMPs$Breast_metastasis))
n23 = length(GenomicRanges::intersect(DIMPs$Breast_cancer, DIMPs$Breast_metastasis))
```

```r
n123 = length(Reduce(GenomicRanges::intersect,
                     list(DIMPs$Breast_normal, DIMPs$Breast_cancer,
                          DIMPs$Breast_metastasis)))

grid.newpage()
v = draw.triple.venn(area1 = length(DIMPs$Breast_normal),
                     area2 = length(DIMPs$Breast_cancer),
                     area3 = length(DIMPs$Breast_metastasis),
                     n12 = n12, n23 = n23, n13 = n13, n123 = n123,
                     category = c("Breast_normal", "Breast_cancer",
                                  "Breast_metastasis"),
                     lty = rep("blank", 3), fill = c("blue", "yellow", "magenta"),
                     alpha = c(0.1, 0.2, 0.3),
                     cat.pos = c(-80, 90, 0),
                     cat.col = c("blue", "darkorange", "red"),
                     cat.dist = c( -0.1, -0.08, -0.26),
                     cex = rep(1.7, 7),
                     cat.cex = c( 1.5, 1.5, 1.5),
                     label.col = c( "blue", "darkorange", "darkorange", "red",
                                    "white", "red", "red"),
                     scaled = TRUE)
grid.draw(v)
```



Notice that natural methylation regulatory signals (not induced by the treatment) are present in both groups, control and treatment. The signal detection step permits us to discriminate the "ordinary" signals observed in the control from those induced by the treatment (a disease in the current case).

7. Differentially informative methylated genomic regions (DIMRs)

Our degree of confidence in whether DIMP counts in both groups of samples, control and treatment, represent true biological signal was determined in the signal detection step. To estimate DIMRs, we followed similar steps to those proposed in Bioconductor R package DESeq2 (Love, Huber, and Anders 2014), but the test looks for the statistical difference between the groups based on gene-body DIMP counts overlapping a given genomic region rather than read counts. The regression analysis of the generalized linear model (GLMs) with logarithmic link was applied to test the difference between group counts. The fitting algorithmic approaches provided by 'glm' and 'glm.nb' functions from the R packages stat and MASS, respectively, were used for Poisson (PR), Quasi-Poisson (QPR) and Negative Binomial (NBR) linear regression analyses, respectively.

7.1. Differentially methylated genes (DMGs)

We shall call DMGs those DIMRs restricted to gene-body regions. DMGs are detected using function 'COUNT.TEST'. We used computational steps from DESeq2 packages. In the current case we follow the steps:

```
suppressMessages(library(DESeq2))
suppressMessages(library(rtracklayer))
# To load human gene annotation
AG = import(paste0(system.file(package = "MethylIT"),
                   "/extdata/Homo_sapiens.GRCh38.91.chromosome.13.gff3.gz")
            )
genes = AG[ AG$type == "gene", c( "gene_id", "biotype" ) ]
genes = genes[ genes$biotype == "protein_coding", "gene_id" ]
seqlevels( genes ) <- "chr13" # To keep a consistent chromosome annotation
```

Function'dimpAtGenes' is used to count the number of DIMPs at gene-body. The operation of this function is based on 'findOverlaps' function from 'GenomicRanges' Bioconductor R package. 'findOverlaps' function has several critical parameters like, for example, 'maxgap', 'minoverlap', and 'ignore.strand'. In our function 'dimpAtGenes', except for setting ignore.strand = TRUE and type = "within", we preserve the rest of default 'findOverlaps' parameters. In this case, these are important parameter setting because the local mechanical effect of methylation changes on a DNA region where a gene is located is independent of the strand where the gene is encoded. That is, methylation changes located in any of the two DNA strands inside the gene-body region will affect the flexibility of the DNA molecule (Choy et al. 2010; Severin et al. 2011).

```
DIMPsBN = dimpAtGenes(GR = DIMPs$Breast_normal, GENES = genes)
DIMPsBC = dimpAtGenes(GR = DIMPs$Breast_cancer, GENES = genes)
DIMPsBM = dimpAtGenes(GR = DIMPs$Breast_metastasis, GENES = genes)
```

```
DIMPsBN
```

```
## GRanges object with 216 ranges and 2 metadata columns:
##         seqnames                ranges strand |          GeneID     DIMPs
##            <Rle>             <IRanges>  <Rle> |        <factor> <integer>
##     [1]    chr13   [19422877, 19536762]      - | ENSG00000132958         4
##     [2]    chr13   [19674752, 19783019]      - | ENSG00000121390         3
##     [3]    chr13   [19823482, 19863636]      - | ENSG00000132950         2
##     [4]    chr13   [20138255, 20161049]      - | ENSG00000121743         4
##     [5]    chr13   [20403667, 20525857]      - | ENSG00000165475         5
##     ...      ...                     ...    ... .             ...       ...
##   [212]    chr13 [113977783, 114132611]      - | ENSG00000185989        15
##   [213]    chr13 [114179238, 114223084]      + | ENSG00000283361         2
##   [214]    chr13 [114234887, 114272723]      + | ENSG00000130177         3
##   [215]    chr13 [114281584, 114305817]      + | ENSG00000169062         1
##   [216]    chr13 [114314513, 114327328]      + | ENSG00000198824         1
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

The number of DIMPs on the strand where a gene is encoded is obtained by setting ignore.strand = FALSE. However, for the current example results will be the same since the datasets downloaded from GEO do not have strand information.

Next, the above GRanges objects carrying the DIMP counts from each sample are grouped into a single GRanges object. Since we have only one control, to perform group comparison and to move forward with this example, we duplicated 'Breast_normal' sample. Obviously, the confidence on the results increases with the number of sample replications per group (in this case, it is only an illustrative example on how to perform the analysis, since a fair comparison requires for more than one replicate in the control group).

```
Genes.DIMPs = uniqueGRanges( list(DIMPsBN[, 2], DIMPsBN[, 2],
                                  DIMPsBC[, 2], DIMPsBM[, 2]),
                             type = "equal", verbose = FALSE,
                             ignore.strand = TRUE )
colnames( mcols(Genes.DIMPs)) <- c("Breast_normal", "Breast_normal1",
                                   "Breast_cancer", "Breast_metastasis")

Genes.DIMPs
```

```
## GRanges object with 303 ranges and 4 metadata columns:
##         seqnames                ranges strand | Breast_normal
##            <Rle>             <IRanges>  <Rle> |     <numeric>
##     [1]    chr13   [19173770, 19181852]      - |             0
##     [2]    chr13   [19422877, 19536762]      - |             4
##     [3]    chr13   [19633681, 19673459]      + |             0
##     [4]    chr13   [19674752, 19783019]      - |             3
##     [5]    chr13   [19823482, 19863636]      - |             2
##     ...      ...                     ...    ... .           ...
##   [299]    chr13 [113977783, 114132611]      - |            15
```

```
##   [300]    chr13 [114179238, 114223084]      + |             2
##   [301]    chr13 [114234887, 114272723]      + |             3
##   [302]    chr13 [114281584, 114305817]      + |             1
##   [303]    chr13 [114314513, 114327328]      + |             1
##         Breast_normal1 Breast_cancer Breast_metastasis
##              <numeric>     <numeric>         <numeric>
##     [1]              0             1                 0
##     [2]              4           186                71
##     [3]              0            98                19
##     [4]              3           172                45
##     [5]              2            32                10
##     ...            ...           ...               ...
##   [299]             15            98               136
##   [300]              2             8                13
##   [301]              3             5                 4
##   [302]              1             8                 0
##   [303]              1            10                 9
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Next, the set of mapped genes are annotated

```
GeneID = subsetByOverlaps(genes, Genes.DIMPs, type = "equal",
                          ignore.strand = FALSE)
dmps = data.frame( mcols( Genes.DIMPs ) )
dmps = apply( dmps, 2, as.numeric )
rownames( dmps ) <- GeneID$gene_id
```

Now, we build a 'DESeqDataSet' object using functions DESeq2 package.

```
condition = data.frame(condition = factor(c("BN", "BN", "BC", "BC"),
                                          levels = c("BN", "BC")))
rownames(condition) <- c("Breast_normal", "Breast_normal1",
                         "Breast_cancer", "Breast_metastasis")

DIMR <- DESeqDataSetFromMatrix( countData = dmps,
                                colData = condition,
                                design = formula( ~ condition ),
                                rowRanges = Genes.DIMPs)
```

```
## converting counts to integer mode
```

DMG analysis is performed with the function 'COUNT.TEST'

```
DMGs = COUNT.TEST( DIMR, num.cores = 3L, minCountPerIndv = 9, countFilter = TRUE,
                   Minlog2FC = 1, pvalCutOff = 0.05,
                   MVrate = .95 )
```

```
## *** Number of genes after filtering counts 181
```

```
## *** Estimating dispersion...
```

```
## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## *** GLM...
```

```
DMGs
```

```
## GRanges object with 129 ranges and 11 metadata columns:
##                   seqnames                ranges strand | Breast_normal
##                      <Rle>             <IRanges>  <Rle> |     <integer>
##   ENSG00000132958    chr13   [19422877, 19536762]      - |             4
##   ENSG00000121390    chr13   [19674752, 19783019]      - |             3
##   ENSG00000132950    chr13   [19823482, 19863636]      - |             2
##   ENSG00000150456    chr13   [20728731, 20773958]      - |             3
##   ENSG00000132953    chr13   [20777329, 20903048]      - |            10
##               ...      ...                     ...    ... .           ...
##   ENSG00000185974    chr13 [113667155, 113737735]      + |             4
##   ENSG00000184497    chr13 [113759240, 113816995]      + |             9
##   ENSG00000185989    chr13 [113977783, 114132611]      - |            15
##   ENSG00000283361    chr13 [114179238, 114223084]      + |             2
##   ENSG00000198824    chr13 [114314513, 114327328]      + |             1
##                   Breast_normal1 Breast_cancer Breast_metastasis    log2FC
##                        <integer>     <integer>         <integer> <numeric>
##   ENSG00000132958              4           186                71  4.604337
##   ENSG00000121390              3           172                45  2.713369
##   ENSG00000132950              2            32                10  2.466215
##   ENSG00000150456              3            32                69  3.303217
##   ENSG00000132953             10            78                49  2.817520
##               ...            ...           ...               ...       ...
##   ENSG00000185974              4            32               177  4.173104
##   ENSG00000184497              9            74               119  2.026853
##   ENSG00000185989             15            98               136  1.154323
##   ENSG00000283361              2             8                13  2.172223
##   ENSG00000198824              1            10                 9  3.713572
##                         pvalue          model     adj.pval
##                      <numeric>       <factor>    <numeric>
##   ENSG00000132958 1.072726e-15 Neg.Binomial.W 3.459542e-14
##   ENSG00000121390 2.753143e-07   Neg.Binomial 1.145663e-06
##   ENSG00000132950 1.147777e-02 Neg.Binomial.W 1.346029e-02
##   ENSG00000150456 3.081212e-06   Neg.Binomial 9.463722e-06
##   ENSG00000132953 9.088263e-03 Neg.Binomial.W 1.122584e-02
##               ...          ...            ...          ...
##   ENSG00000185974 3.183554e-11 Neg.Binomial.W 4.428195e-10
##   ENSG00000184497 3.701518e-09   Neg.Binomial 2.652755e-08
##   ENSG00000185989 7.234106e-11   Neg.Binomial 8.483633e-10
##   ENSG00000283361 1.422959e-03   Neg.Binomial 2.323565e-03
##   ENSG00000198824 3.086748e-02 Neg.Binomial.W 3.336551e-02
```

```
##                   CT.SignalDensity TT.SignalDensity SignalDensityVariation
##                          <numeric>        <numeric>              <numeric>
##   ENSG00000132958       0.03512284        1.1283213              1.0931985
##   ENSG00000121390       0.02770902        1.0021428              0.9744338
##   ENSG00000132950       0.04980700        0.5229735              0.4731665
##   ENSG00000150456       0.06633059        1.1165650              1.0502344
##   ENSG00000132953       0.07954184        0.5050907              0.4255488
##               ...              ...              ...                    ...
##   ENSG00000185974       0.05667248        1.4805684              1.4238959
##   ENSG00000184497       0.15582797        1.6708221              1.5149941
##   ENSG00000185989       0.09688108        0.7556724              0.6587913
##   ENSG00000283361       0.04561315        0.2394691              0.1938559
##   ENSG00000198824       0.07802747        0.7412609              0.6632335
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

BRCA2, a breast cancer associated risk gene, is found between the DMGs

```r
DMGs[ grep( "ENSG00000139618", names(DMGs) ) ]
```

```
## GRanges object with 1 range and 11 metadata columns:
##                   seqnames               ranges strand | Breast_normal
##                      <Rle>            <IRanges>  <Rle> |     <integer>
##   ENSG00000139618    chr13 [32315474, 32400266]      + |             1
##                   Breast_normal1 Breast_cancer Breast_metastasis    log2FC
##                        <integer>     <integer>         <integer> <numeric>
##   ENSG00000139618              1           122                73  4.518159
##                        pvalue          model   adj.pval CT.SignalDensity
##                     <numeric>       <factor>  <numeric>        <numeric>
##   ENSG00000139618 0.009137309 Neg.Binomial.W 0.01122584       0.01179343
##                   TT.SignalDensity SignalDensityVariation
##                          <numeric>              <numeric>
##   ENSG00000139618         1.149859               1.138066
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Acknowledgments

We thank Professor David J Miller for valuable conversations and suggestions on our mathematical modeling.

Funding

The work was supported by funding from NSF-SBIR (2015-33610-23428-UNL) and the Bill and Melinda Gates Foundation (OPP1088661).

Supplements.

S1. Datasets used in this example

The dataset used in this example are chromosome 13's methylomes from human breast tissues. BRCA2 gene, a breast cancer associated risk gene, is located in chromosome 13. BS-Seq experiments can be downloaded from GEO DataSet and then be read by the MethylIT function 'readCounts2GRangesList'. For the sake of brevity and to reduce file sizes, we already did it. For example, for a dataset of embryonic stem cells we used the script:

```
setwd("/data/HumanMethy/StemCells/GSE76970")
files = list.files(path = "/data/HumanMethy/StemCells/GSE76970",
                   pattern = "CGmethratio.tab.gz" )

# If not chromosome is specified, all are included.
LR = readCounts2GRangesList(files_names = files, sample.id = paste0("Primed", 1:3),
                            columns = c( seqnames = 1, start = 2, strand = 3,
                                         mC = 4, coverage = 5 ),
                            chromosomes = "chr13")

# Only to build the example dataset and save space.
files = c("GSM2041690_WGBS_UCLA1_Primed1_chr13.txt",
          "GSM2041691_WGBS_UCLA1_Primed2_chr13.txt",
          "GSM2041692_WGBS_UCLA1_Primed3_chr13.txt")
for (k in 1:3) {
  x = as.data.frame(LR[[k]])
  x = x[, c("seqnames", "start", "mC", "uC")]
  write.table(x, file = files[k], sep = "\t", row.names = FALSE,
              col.names = FALSE)
  system(paste0("gzip -9 ", files[k]))
}
```

Notice that we have specified the column table where the data of interest are found (see 'readCounts2GRangesList'). We opted not to define a new type of object specific for our package, but to use the useful 'GRanges' objects from Biocoductor R package 'GenomicRanges'. Chromosomes are located in the GRanges objects in the "seqnames" column. It is important to be consistent with chromosome notation for all the samples. For example, if for one dataset chromosomes are named as "chr1", "chr2", ..., etc, then this notation must be preserved. Let's suppose that in the GRanges object GR chromosomes are named "1", "2", and "3", and we need to specify then as "Chr1", "Chr2", "Chr3", then we can do it as:

```
GR = as.data.frame(GR)
GR$seqnames <- paste0("Chr", GR$seqnames)
# and recover the GR object by using:
GR = makeGRangesFromDataFrame(GR, keep.extra.columns = TRUE)

# or alternatively
# Chromosome order must be preserved!
```

```r
seqlevels(GR) <- c("Chr1", "Chr2", "Chr3")
```

S2. Session Information

```
## R version 3.4.3 (2017-11-30)
## Platform: x86_64-redhat-linux-gnu (64-bit)
## Running under: CentOS Linux 7 (Core)
##
## Matrix products: default
## BLAS/LAPACK: /usr/lib64/R/lib/libRblas.so
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=C
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
##  [1] parallel  stats4    grid      stats     graphics  grDevices utils
##  [8] datasets  methods   base
##
## other attached packages:
##  [1] rtracklayer_1.38.3         DESeq2_1.18.1
##  [3] SummarizedExperiment_1.8.1 DelayedArray_0.4.1
##  [5] matrixStats_0.53.0         Biobase_2.38.0
##  [7] GenomicRanges_1.30.1       GenomeInfoDb_1.14.0
##  [9] IRanges_2.12.0             S4Vectors_0.16.0
## [11] BiocGenerics_0.24.0        VennDiagram_1.6.18
## [13] futile.logger_1.4.3        gridExtra_2.3
## [15] reshape2_1.4.3             ggplot2_2.2.1
## [17] knitr_1.19                 MethylIT_0.1.0
##
## loaded via a namespace (and not attached):
##   [1] backports_1.1.2               Hmisc_4.1-1
##   [3] AnnotationHub_2.10.1          plyr_1.8.4
##   [5] lazyeval_0.2.1                splines_3.4.3
##   [7] BiocParallel_1.12.0           digest_0.6.15
##   [9] foreach_1.4.4                 BiocInstaller_1.28.0
##  [11] ensembldb_2.2.1               htmltools_0.3.6
##  [13] magrittr_1.5                  checkmate_1.8.5
##  [15] memoise_1.1.0                 BSgenome_1.46.0
##  [17] cluster_2.0.6                 sfsmisc_1.1-1
##  [19] etm_0.6-2                     annotate_1.56.1
##  [21] recipes_0.1.2                 Biostrings_2.46.0
```

```
##  [23] gower_0.1.2                   dimRed_0.1.0
##  [25] ArgumentCheck_0.10.2          prettyunits_1.0.2
##  [27] colorspace_1.3-2              blob_1.1.0
##  [29] dplyr_0.7.4                   crayon_1.3.4
##  [31] RCurl_1.95-4.10               roxygen2_6.0.1
##  [33] genefilter_1.60.0             bindr_0.1
##  [35] zoo_1.8-1                     survival_2.41-3
##  [37] VariantAnnotation_1.24.5      iterators_1.0.9
##  [39] glue_1.2.0                    DRR_0.0.3
##  [41] gtable_0.2.0                  ipred_0.9-6
##  [43] zlibbioc_1.24.0               XVector_0.18.0
##  [45] kernlab_0.9-25                ddalpha_1.3.1.1
##  [47] DEoptimR_1.0-8                scales_0.5.0
##  [49] futile.options_1.0.0          DBI_0.7
##  [51] Rcpp_0.12.15                  cmprsk_2.2-7
##  [53] xtable_1.8-2                  progress_1.1.2
##  [55] htmlTable_1.11.2              FAdist_2.2
##  [57] foreign_0.8-69                bit_1.1-12
##  [59] Formula_1.2-2                 lava_1.6
##  [61] prodlim_1.6.1                 htmlwidgets_1.0
##  [63] httr_1.3.1                    RColorBrewer_1.1-2
##  [65] acepack_1.4.1                 pkgconfig_2.0.1
##  [67] XML_3.98-1.9                  nnet_7.3-12
##  [69] locfit_1.5-9.1                caret_6.0-78
##  [71] labeling_0.3                  tidyselect_0.2.3
##  [73] rlang_0.1.6                   AnnotationDbi_1.40.0
##  [75] munsell_0.4.3                 tools_3.4.3
##  [77] RSQLite_2.0                   devtools_1.13.4
##  [79] broom_0.4.3                   evaluate_0.10.1
##  [81] stringr_1.2.0                 yaml_2.1.16
##  [83] ModelMetrics_1.1.0            bit64_0.9-7
##  [85] robustbase_0.92-8             purrr_0.2.4
##  [87] AnnotationFilter_1.2.0        bindrcpp_0.2
##  [89] nlme_3.1-131                  mime_0.5
##  [91] RcppRoll_0.2.2                xml2_1.2.0
##  [93] biomaRt_2.34.2                compiler_3.4.3
##  [95] rstudioapi_0.7                curl_3.1
##  [97] interactiveDisplayBase_1.16.0 testthat_1.0.2
##  [99] e1071_1.6-8                   geneplotter_1.56.0
## [101] tibble_1.4.2                  stringi_1.1.6
## [103] desc_1.1.1                    Epi_2.24
## [105] GenomicFeatures_1.30.3        lattice_0.20-35
## [107] ProtGenerics_1.10.0           Matrix_1.2-12
## [109] commonmark_1.4                psych_1.7.8
## [111] pillar_1.1.0                  data.table_1.10.4-3
## [113] bitops_1.0-6                  httpuv_1.3.5
## [115] R6_2.2.2                      latticeExtra_0.6-28
## [117] RMySQL_0.10.13                codetools_0.2-15
```

```
## [119] lambda.r_1.2                  dichromat_2.0-0
## [121] MASS_7.3-48                   assertthat_0.2.0
## [123] CVST_0.2-1                    rprojroot_1.3-2
## [125] minpack.lm_1.2-1              withr_2.1.1
## [127] GenomicAlignments_1.14.1      Rsamtools_1.30.0
## [129] mnormt_1.5-5                  GenomeInfoDbData_1.0.0
## [131] rpart_4.1-12                  timeDate_3042.101
## [133] tidyr_0.8.0                   class_7.3-14
## [135] rmarkdown_1.8                 nls2_0.2
## [137] biovizBase_1.26.0             numDeriv_2016.8-1
## [139] shiny_1.0.5                   lubridate_1.7.1
## [141] base64enc_0.1-3
```

References

Baldi, Pierre, and Soren Brunak. 2001. *Bioinformatics: the machine learning approach*. Second. Cambridge: MIT Press.

Basu, A., A. Mandal, and L. Pardo. 2010. "Hypothesis testing for two discrete populations based on the Hellinger distance." *Statistics & Probability Letters* 80 (3-4). Elsevier B.V.: 206–14. doi:10.1016/j.spl.2009.10.008.

Carter, Jane V., Jianmin Pan, Shesh N. Rai, and Susan Galandiuk. 2016. "ROC-ing along: Evaluation and interpretation of receiver operating characteristic curves." *Surgery* 159 (6). Mosby: 1638–45. doi:10.1016/j.surg.2015.12.029.

Choy, John S., Sijie Wei, Ju Yeon Lee, Song Tan, Steven Chu, and Tae Hee Lee. 2010. "DNA methylation increases nucleosome compaction and rigidity." *Journal of the American Chemical Society* 132 (6). American Chemical Society: 1782–3. doi:10.1021/ja910264z.

Dolzhenko, Egor, and Andrew D Smith. 2014. "Using beta-binomial regression for high-precision differential methylation analysis in multifactor whole-genome bisulfite sequencing experiments." *BMC Bioinformatics* 15 (1). BioMed Central: 215. doi:10.1186/1471-2105-15-215.

Fawcett, Tom. 2005. "An introduction to ROC analysis." doi:10.1016/j.patrec.2005.10.010.

Greiner, M, D Pfeiffer, and R D Smith. 2000. "Principles and practical application of the receiver-operating characteristic analysis for diagnostic tests." *Preventive Veterinary Medicine* 45 (1-2): 23–41. doi:10.1016/S0167-5877(00)00115-X.

Harpaz, Rave, William DuMouchel, Paea LePendu, Anna Bauer-Mehren, Patrick Ryan, and Nigam H Shah. 2013. "Performance of Pharmacovigilance Signal Detection Algorithms for the FDA Adverse Event Reporting System." *Clin Pharmacol Ther* 93 (6): 1–19. doi:10.1038/clpt.2013.24.Performance.

Hebestreit, Katja, Martin Dugas, and Hans-Ulrich Klein. 2013. "Detection of significantly differentially methylated regions in targeted bisulfite sequencing data." *Bioinformatics (Oxford, England)* 29 (13): 1647–53. doi:10.1093/bioinformatics/btt263.

Kruspe, Sven, David D. Dickey, Kevin T. Urak, Giselle N. Blanco, Matthew J. Miller, Karen C. Clark, Elliot Burghardt, et al. 2017. "Rapid and Sensitive Detection of Breast Cancer Cells in

Patient Blood with Nuclease-Activated Probe Technology." *Molecular Therapy - Nucleic Acids* 8 (September). Cell Press: 542–57. doi:10.1016/j.omtn.2017.08.004.

Love, M I, W Huber, and S Anders. 2014. "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2." *Genome Biology* 15 (12): 1–34. doi:Artn 550\rDoi 10.1186/S13059-014-0550-8.

López-Ratón, Mónica, Mar\'\ia Xosé Rodríguez-Álvarez, Carmen Cadarso-Suárez, Francisco Gude-Sampedro, and Others. 2014. "OptimalCutpoints: an R package for selecting optimal cutpoints in diagnostic tests." *Journal of Statistical Software* 61 (8). Foundation for Open Access Statistics: 1–36. https://www.jstatsoft.org/article/view/v061i08.

Robinson, Mark D., Abdullah Kahraman, Charity W. Law, Helen Lindsay, Malgorzata Nowicka, Lukas M. Weber, and Xiaobei Zhou. 2014. "Statistical methods for detecting differentially methylated loci and regions." *Frontiers in Genetics* 5 (SEP). Frontiers: 324. doi:10.3389/fgene.2014.00324.

Sanchez, Robersy, and Sally A. Mackenzie. 2016. "Information Thermodynamics of Cytosine DNA Methylation." Edited by Barbara Bardoni. *PLOS ONE* 11 (3). Public Library of Science: e0150427. doi:10.1371/journal.pone.0150427.

Sanchez, Robersy, Xiaodong Yang, Hardik Kundariya, Jose Raul Barreras, Yashitola Wamboldt, and Sally Mackenzie. 2018. "Enhancing resolution of natural methylome reprogramming behavior in plants." *bioRxiv*, January. Cold Spring Harbor Laboratory, 252106. doi:doi.org/10.1101/252106.

Severin, Philip M D, Xueqing Zou, Hermann E Gaub, and Klaus Schulten. 2011. "Cytosine methylation alters DNA mechanical properties." *Nucleic Acids Research* 39 (20): 8740–51. doi:10.1093/nar/gkr578.

Stevens, James P. 2009. *Applied Multivariate Statistics for the Social Sciences.* Fifth Edit. Routledge Academic.