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]*Department of Biology and Plant Science.*
Pennsylvania State University, University Park, PA 16802

03 April 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.

```
**NOTE**

This is a reduced version of the cancer example with only two figures. The full
version with all the figures is available at:
https://git.psu.edu/genomath/MethylIT_Data/blob/master/cancer_example_04-03-18.pdf
```

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 MethylIT, 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*

To install MethylIT you might need to install the Bioconductor packages: 'GenomicFeatures', 'VariantAnnotation', 'ensembldb', 'GenomicRanges', 'BiocParallel', 'biovizBase', 'DESeq2', and 'genefilter'. 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")
```

Some possible troubleshooting installation on Ubuntu is given in section S1. Installation on our Windows OS machines was straightforward.

2. *Available datasets and reading*

Methylome datasets of whole-genome bisulfite sequencing (WGBS) are available at Gene Expression Omnibus (GEO DataSets). For the current example, datasets from breast tissues (normal and cancer) and embryonic stem cells will be downloaded from GEO. The data set are downloaded providing the GEO accession numbers for each data set to the function 'getGEOSuppFiles' (for details type ?getGEOSuppFiles in the R console).

```
suppressMessages(library(MethylIT))

# Embryonic stem cells datasets
esc.files = getGEOSuppFiles(GEO = c("GSM2041690", "GSM2041691", "GSM2041692"),
                            verbose = FALSE)
# Breast tissues (normal, cancer, metastasis)
cancer.files = getGEOSuppFiles(GEO = c("GSM1279517", "GSM1279514",
                                       "GSM1279513"), verbose = FALSE)
```

The file path and name of each downloaded dataset is found in the output variables 'esc.files' and 'cancer.files'.

2.1. Reading datasets

Datasets for our example can be read with function 'readCounts2GRangesList'. To specify the reading of only chromosome 13, we can specify the parameter 'chromosomes = "Chr13" '. The symbol chromosome 13, in this case "Chr13", must be consistent with the annotation provided in the given GEO dataset. Each file is wholly read with the setting 'chromosomes = "Chr13" ' and then the GRanges are built only with chromosome 13, which could be time consuming. However, users working on Linux OS can specify the reading of specific lines from each file by using regular expressions. For example, if only chromosomes 1 and 3 are required, then we can set chromosomes = NULL (default) and 'chromosome.pattern = "^Chr[1,3]" '. This will read all the lines in the downloaded files starting with the words "Chr1" or "Chr3". If we are interested in chromosomes 1 and 2, then we can set 'chromosome.pattern = "^Chr[1-2]" '. If all the chromosomes are required, then set chromosomes = NULL and chromosome.pattern = NULL (default).

```
# Embryonic stem cells datasets
ref = readCounts2GRangesList(filenames = esc.files,
                             sample.id = c("ESC1","ESC2", "ESC3"),
                             columns = c(seqnames = 1, start = 2,
                                         mC = 4, uC = 5), pattern = "^chr13",
                             remove = TRUE, verbose = FALSE)
# Breast tissues (normal, cancer, metastasis)
LR = readCounts2GRangesList(filenames = cancer.files,
                            sample.id = c("Breast_normal", "Breast_cancer",
                                          "Breast_metastasis"),
                            columns = c(seqnames = 1, start = 2,
                                        mC = 3, uC = 4),
                            remove = TRUE, pattern = "^13",
                            chromosome.names = "chr13", verbose = FALSE)
```

In the metacolumn of the last GRanges object, mC and uC stand for the methylated and unmethylated read counts, respectively. Notice that option 'remove = TRUE' remove the decompressed files (default: FALSE, see ?readCounts2GRangesList for more details about this function).

3. The reference individual

Any two objects located in a space can be compared if, and only if, there is a reference point (a coordinate system) in the space and a metric. Usually, in our daily 3D experience, our brain automatically sets up the origin of coordinates equal to zero. The differences found in the comparison depend on the reference used to perform the measurements and from the metric system. The space where the objects are located (or the set of objects) together with the metric is called metric space.

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 with respect to the same individual reference on the mentioned metric space. The reason to measure the uncertainty variation with respect to the same reference resides in that even sibling individuals follow an independent ontogenetic

development. This a consequence of the "omnipresent" 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 should be noticed that the results are sensitive to the reference used. The statistics mean, median, or sum of the read counts at each cytosine site of some control samples can be used to create a virtual reference sample. 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:

```
Ref = poolFromGRlist(ref, stat = "mean", 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]      * |         1         1
##         [2]    chr13   [19020633, 19020633]      * |         2         2
##         [3]    chr13   [19020642, 19020642]      * |         1         1
##         [4]    chr13   [19020643, 19020643]      * |         2         2
##         [5]    chr13   [19020679, 19020679]      * |         1         1
##         ...      ...                   ...    ... .       ...       ...
##   [1560633]    chr13 [115108993, 115108993]      * |         1         3
##   [1560634]    chr13 [115109022, 115109022]      * |         1         1
##   [1560635]    chr13 [115109023, 115109023]      * |         3         4
##   [1560636]    chr13 [115109523, 115109523]      * |         2         2
##   [1560637]    chr13 [115109524, 115109524]      * |         1         1
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Only direct lab experiments can reveal whether differences detected with distinct references outside the experimental conditions for control and treatment groups are real. The best reference would be estimated using a subset of individuals from control group. Such a reference will contribute to remove the intragroup variation, in control and in treatment groups, induced by environmental changes external to or not controlled by the experimental conditions.

Methylation analysis for each cystosine position is frequently performed in the bidimensional space of $(methylated, unmethylated)$ read counts. Frequently, Fisher test is applied to a single cytosine position, under the null hypothesis that the proportions $p_{ct} = methylated_{ct}/(methylated_{ct} + unmethylated_{ct})$ and $p_{tt} = methylated_{tt}/(methylated_{tt} + unmethylated_{tt})$ are the same for control and treatment, respectively. In this case, the implicit reference point for the counts at every cytosine positions is $(methylated = 0, unmethylated = 0)$, which corresponds to the point $P_i = (0, 1)$.

In our case, the Hellinger divergence (the metric used, here) of each individual in respect to the reference is the variable to test in place of $(methylated, unmethylated)$ read counts or the methylation levels $P_i = (p_i, 1 - p_i)$.

The use of references is restricted by the thermodynamics basis of the the theory. The current information-thermodynamics based approach is supported on the following postulate:

"High changes of Hellinger divergences are less frequent than low changes, provided that the divergence

is proportional to the amount of energy required to process one bit of information in methylation system".

The last postulate acknowledges the action of the second law of thermodynamics on the biomolecular methylation system. For the methylation system, it implies that the frequencies of the information divergences between methylation levels must be proportional to a Boltzmann factor (see supplementary information from reference (Sanchez and Mackenzie 2016)). In other words, the frequencies of information divergences values should follow a trend proportional to an exponential decay. If we do not observe such a behaviour, then either the reference is too far from experimental condition or we are dealing with an extreme situation where the methylation machinery in the cell is dysfunctional. The last situation is found, for example, in the silencing mutation at the gene of cytosine-DNA-methyltransferase in Arabidopsis *thaliana*. Methylation of 5-methylcytosine at CpG dinucleotides is maintained by MET1 in plants.

In our current example, the embryonic stem cells reference is far from the breast tissue samples and this could affect the nonlinear fit to a Weibull distribution (see below). To illustrate the effect of the reference on the analysis, a new reference will be built by setting:

```
Ref0 = Ref
Ref0$uC <- 0
```

The reason for the above replacement is that natural methylation changes (Ref$mC) obey the second law of thermodynamics, and we do not want to arbitrarily change the number of methylated read counts. 'mC' carries information linked to the amount of energy expended in the tissue associated with concrete methylation changes. However, 'uC' is not linked to any energy expended by the methylation machinery in the cells. In the bidimensional space $P_i = (p_i, 1 - p_i)$, reference *Ref0* corresponds to the point $P_i = (1, 0)$ at each cytosine site i, i.e., the value of methylation level at every cytosine site in reference *Ref0* is 1. The analyses with respect to both individual references, *Ref* and *Ref0*, will be performed in the downstream steps.

4. Hellinger divergence estimation

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 with 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 other information theoretical divergences can be used as well. Next, the information divergence for control (Breast_normal) and treatment (Breast_cancer and Breast_metastasis) samples are estimated with 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 'estimateDivergence':

```
HD = estimateDivergence(ref = Ref, indiv = LR, Bayesian = TRUE,
                        min.coverage = 5, high.coverage = 300,
                        percentile = 0.999, num.cores = 12L, tasks = 0L,
                        verbose = FALSE)
HD0 = estimateDivergence(ref = Ref0, indiv = LR, Bayesian = TRUE,
                         min.coverage = 9, high.coverage = 300,
                         percentile = 0.999, num.cores = 12L, tasks = 0L,
                         verbose = FALSE)
HD$Breast_cancer
```

```
## GRanges object with 987895 ranges and 9 metadata columns:
##             seqnames                ranges strand |        c1        t1
##                <Rle>             <IRanges>  <Rle> | <numeric> <numeric>
##         [1]    chr13   [19020631, 19020631]      * |         1         1
##         [2]    chr13   [19020633, 19020633]      * |         2         2
##         [3]    chr13   [19020643, 19020643]      * |         2         2
##         [4]    chr13   [19020680, 19020680]      * |         0         1
##         [5]    chr13   [19020687, 19020687]      * |         1         1
##         ...      ...                   ...    ... .       ...       ...
##    [987891]    chr13 [115108788, 115108788]      * |         2         4
##    [987892]    chr13 [115108789, 115108789]      * |         2         2
##    [987893]    chr13 [115108993, 115108993]      * |         1         3
##    [987894]    chr13 [115109023, 115109023]      * |         3         4
##    [987895]    chr13 [115109524, 115109524]      * |         1         1
##                  c2        t2                p1                  p2
##             <numeric> <numeric>         <numeric>           <numeric>
##         [1]        14        24 0.413370663720767  0.375495465916207
##         [2]        14        25 0.442871729587454   0.36633665561778
##         [3]         7        38 0.442871729587454  0.170626940254775
##         [4]         1        43 0.209181014646214  0.043896699394733
##         [5]         0        46 0.413370663720767 0.0212335246596664
##         ...       ...       ...                 ...                 ...
##    [987891]         0         0  0.33036727060318  0.254902793892838
##    [987892]        27        43 0.442871729587454  0.389164474334267
##    [987893]        72         5 0.272599930600126  0.924313453079455
##    [987894]        56        36 0.405836166223695  0.606598044818736
##    [987895]        31         9 0.413370663720767  0.762392481743825
##                            TV              bay.TV                hdiv
##                     <numeric>           <numeric>           <numeric>
```

```
##         [1] -0.131578947368421 -0.0378751978045603 0.00836901839443372
##         [2] -0.141025641025641 -0.0765350739696741  0.0541295473053947
##         [3] -0.344444444444444  -0.272244789332679   0.818120566547519
##         [4] 0.0227272727272727  -0.165284315251481   0.265271064208063
##         [5]               -0.5  -0.392137139061101    1.67587572624957
##         ...               ...                 ...                 ...
##    [987891] -0.333333333333333  -0.075464476710342  0.0120745026788796
##    [987892] -0.114285714285714 -0.0537072552531863  0.0277523179966212
##    [987893]  0.685064935064935   0.651713522479329    4.95063809290373
##    [987894]  0.180124223602485    0.20076187859504   0.600010041025287
##    [987895]             0.275   0.349021818023058   0.729845744260704
##    -------
##    seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

Function 'estimateDivergence' 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 unmethylated (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 unless previous filtering by coverage had 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.

For some GEO datasets only the methylation levels for each cytosine site are provided. In this case, Hellinger divergence can be estimated as given in reference (Sanchez and Mackenzie 2016):

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

5. 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 by the function 'nonlinearFitDist'.

```r
nlms = nonlinearFitDist(HD, column = 9, num.cores = 3L, verbose = FALSE)
nlms0 = nonlinearFitDist(HD0, column = 9, num.cores = 3L, verbose = FALSE)

nlms # this returns:
```

```
## $Breast_normal
##        Estimate   Std. Error  t value Pr(>|t|)      Adj.R.Square
## shape 0.5543145 0.0002139500 2590.861         0 0.948873244359916
## scale 1.3468977 0.0005372617 2506.968         0
##                       rho       R.Cross.val              DEV               AIC
## shape 0.948873142978588 0.975345082110353 4099.05484372979 -2690639.45164462
## scale
##                       BIC     COV.shape     COV.scale COV.mu       n
## shape -2690603.97940838  4.577459e-08 -5.917927e-09     NA 1008605
## scale                   -5.917927e-09  2.886501e-07     NA 1008605
##
## $Breast_cancer
##           Estimate   Std. Error     t value    Pr(>|t|))     Adj.R.Square
## shape 5.391149e-01 1.506168e-04 3579.381429 0.000000e+00 0.96941589951915
## scale 1.134588e+00 3.739751e-04 3033.860064 0.000000e+00
## mu    7.607881e-05 1.356645e-05    5.607863 2.048944e-08
##                       rho       R.Cross.val              DEV              AIC
## shape 0.969415837601309 0.984822828646025 2409.03055126017 -3139991.5054956
## scale
## mu
##                       BIC     COV.shape     COV.scale        COV.mu      n
## shape -3139944.29216882  2.268541e-08 -5.369580e-09 -4.699170e-10 987895
## scale                   -5.369580e-09  1.398574e-07 -4.640279e-10 987895
## mu                      -4.699170e-10 -4.640279e-10  1.840486e-10 987895
##
## $Breast_metastasis
##         Estimate   Std. Error   t value Pr(>|t|))      Adj.R.Square
## shape 0.55596350 1.506398e-04 3690.6809         0 0.977972027557549
## scale 0.92855327 2.711461e-04 3424.5502         0
## mu    0.01631553 3.409143e-05  478.5817         0
##                       rho       R.Cross.val              DEV               AIC
## shape 0.977971980615213 0.989064163693439 1647.99283323508 -3291231.66890221
## scale
## mu
##                   BIC     COV.shape     COV.scale        COV.mu      n
## shape -3291184.66069  2.269235e-08 -1.522434e-10 -2.562546e-09 938514
## scale                -1.522434e-10  7.352018e-08 -2.497534e-09 938514
## mu                   -2.562546e-09 -2.497534e-09  1.162226e-09 938514
```

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).

The goodness-of-fit of Weibull to the HD0 (*Ref0*) data is better than to HD (*Ref*):

```
nlms0
```

```
## $Breast_normal
##        Estimate   Std. Error   t value Pr(>|t|))      Adj.R.Square
## shape 0.8294116 0.0001082846  7659.556         0 0.995937143067973
## scale 0.3103328 0.0000296877 10453.243         0
##                      rho       R.Cross.val              DEV               AIC
## shape 0.995937132223697 0.998305231882957 253.748299340483 -3860960.92341557
## scale
##                      BIC     COV.shape     COV.scale COV.mu      n
## shape -3860926.34268738  1.172554e-08 -7.197877e-10     NA 749311
## scale                   -7.197877e-10  8.813597e-10     NA 749311
##
## $Breast_cancer
##         Estimate   Std. Error  t value Pr(>|t|))      Adj.R.Square
## shape 0.65808523 1.592997e-04 4131.114         0 0.990007423933119
## scale 0.71350113 1.551868e-04 4597.693         0
## mu    0.01231425 3.325003e-05  370.353         0
##                      rho       R.Cross.val              DEV               AIC
## shape 0.990007395583978 0.995153381084488 587.148032395742 -2998041.98350013
## scale
## mu
##                      BIC     COV.shape     COV.scale        COV.mu      n
## shape -2997996.11987504  2.537640e-08 -3.720161e-09 -2.704483e-09 704967
## scale                   -3.720161e-09  2.408294e-08 -1.475318e-09 704967
## mu                      -2.704483e-09 -1.475318e-09  1.105565e-09 704967
##
## $Breast_metastasis
##          Estimate   Std. Error   t value Pr(>|t|))     Adj.R.Square
## shape 0.590229316 1.198288e-04 4925.6057         0 0.99130928757275
## scale 1.103193068 2.464984e-04 4475.4572         0
## mu    0.006755526 2.520266e-05  268.0482         0
##                      rho       R.Cross.val              DEV               AIC
## shape 0.991309259423371 0.995659479338505 447.581035974421 -2711720.97325685
## scale
## mu
##                      BIC     COV.shape     COV.scale        COV.mu      n
## shape -2711675.63969437  1.435894e-08 -4.500865e-09 -1.129274e-09 617473
## scale                   -4.500865e-09  6.076146e-08 -1.140726e-09 617473
## mu                      -1.129274e-09 -1.140726e-09  6.351740e-10 617473
```

The goodness-of-fit indicators suggest that the fit to Weibull distribution model for *Ref0* is better than for *Ref*.

6. 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.

6.1. *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 'getPotentialDIMP':

```
PS = getPotentialDIMP(LR = HD, nlms = nlms, div.col = 9, alpha = 0.05,
                      tv.col = 7, tv.cut = 0.2)
PS0 = getPotentialDIMP(LR = HD0, nlms = nlms0, div.col = 9, alpha = 0.05,
                       tv.col = 7, tv.cut = 0.2)

PS$Breast_cancer
```

```
## GRanges object with 959 ranges and 10 metadata columns:
##         seqnames                ranges strand |        c1        t1        c2
##            <Rle>             <IRanges>  <Rle> | <numeric> <numeric> <numeric>
##     [1]    chr13   [20137885, 20137885]      * |         7         8        31
##     [2]    chr13   [20267416, 20267416]      * |         6         6        57
##     [3]    chr13   [20279401, 20279401]      * |         8         8        33
##     [4]    chr13   [20285268, 20285268]      * |         0         5        30
##     [5]    chr13   [20680750, 20680750]      * |         5         6        53
##     ...      ...                   ...    ... .       ...       ...       ...
##   [955]    chr13 [114995714, 114995714]      * |         2         6       104
##   [956]    chr13 [114995719, 114995719]      * |         1         6        98
##   [957]    chr13 [115003506, 115003506]      * |         0         4        89
##   [958]    chr13 [115049352, 115049352]      * |         5         6        45
##   [959]    chr13 [115090019, 115090019]      * |         3         6        77
##               t2                p1                p2                TV
```

```
##         <numeric>          <numeric>         <numeric>         <numeric>
##     [1]         0   0.45048218742099 0.970328544095088 0.533333333333333
##     [2]         0  0.475815388309824 0.983404482207982               0.5
##     [3]         0  0.481227946148085 0.972024139204215               0.5
##     [4]         1 0.0874396287267155  0.94002393396164 0.967741935483871
##     [5]         0  0.435095578441169  0.98219749770619 0.545454545454545
##     ...       ...                ...               ...               ...
##   [955]         3  0.263443639719927 0.963493855019361  0.72196261682243
##   [956]         7  0.180432322273311 0.925427707571774 0.790476190476191
##   [957]         2  0.102328131612797 0.967965998825198 0.978021978021978
##   [958]         0  0.435095578441169 0.979167173630546 0.545454545454545
##   [959]         1  0.331185648084296  0.97526063050039 0.653846153846154
##                    bay.TV             hdiv              wprob
##                 <numeric>        <numeric>          <numeric>
##     [1] 0.519846356674098 9.00956640068852 0.0470836932334934
##     [2] 0.507589093898158 9.45940418698377  0.043405888839761
##     [3] 0.490796193056131  8.8670225410283 0.0483316945609365
##     [4] 0.852584305234925 9.68800576922036  0.041676901917537
##     [5] 0.547101919265021 9.66096829306787 0.0418767769318518
##     ...               ...              ...               ...
##   [955] 0.700050215299434  11.039595608022  0.033053793767963
##   [956] 0.744995385298462 10.2400645328768 0.0378476601046051
##   [957] 0.865637867212401 9.78238187227201 0.0409886409957129
##   [958] 0.544071595189377 9.09110832004754 0.0463882979784358
##   [959] 0.644074982416095  10.759779809129 0.0346398125223427
##   -------
##   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 quantitative 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 $H(\hat{p}_{ij}, \hat{p}_{ir})$.

7. Cutpoint estimation

Laws of statistical physics can account for background methylation, a response to thermal fluctuations that presumably functions 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 = estimateCutPoint(PS, control.names = "Breast_normal",
                             treatment.names = c("Breast_cancer",
                                                 "Breast_metastasis"),
                             div.col = 9, verbose = FALSE)
cutpoints
```

```
## $cutpoint
##                    Breast_normal
## Breast_cancer          9.539561
## Breast_metastasis      6.848653
##
## $auc
##                    Breast_normal
## Breast_cancer        0.25034090
## Breast_metastasis    0.08249673
##
## $accuracy
##                    Breast_normal
## Breast_cancer         0.4820937
## Breast_metastasis     0.8723602
```

```
cutpoints0 = estimateCutPoint(PS0, control.names = "Breast_normal",
                              treatment.names = c("Breast_cancer",
                                                  "Breast_metastasis"),
                              div.col = 9, verbose = FALSE)
cutpoints0
```

```
## $cutpoint
##                    Breast_normal
## Breast_cancer          3.514418
## Breast_metastasis      2.418451
##
## $auc
##                    Breast_normal
## Breast_cancer         0.9762920
## Breast_metastasis     0.9985477
##
## $accuracy
##                    Breast_normal
## Breast_cancer         0.9737442
## Breast_metastasis     0.8517964
```

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 $H_{cutpoint}$.

8. 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 with respect to a given control. Notice that the definition of DIMP is not deterministic in an ordinary sense, but stochastic-deterministic in physico-mathematical terms.

DIMPs are selected with the function:

```
DIMPs = selectDIMP(PS, div.col = 9, cutpoint = 6.848653 )
```

8.1. *Venn Diagram of DIMPs*

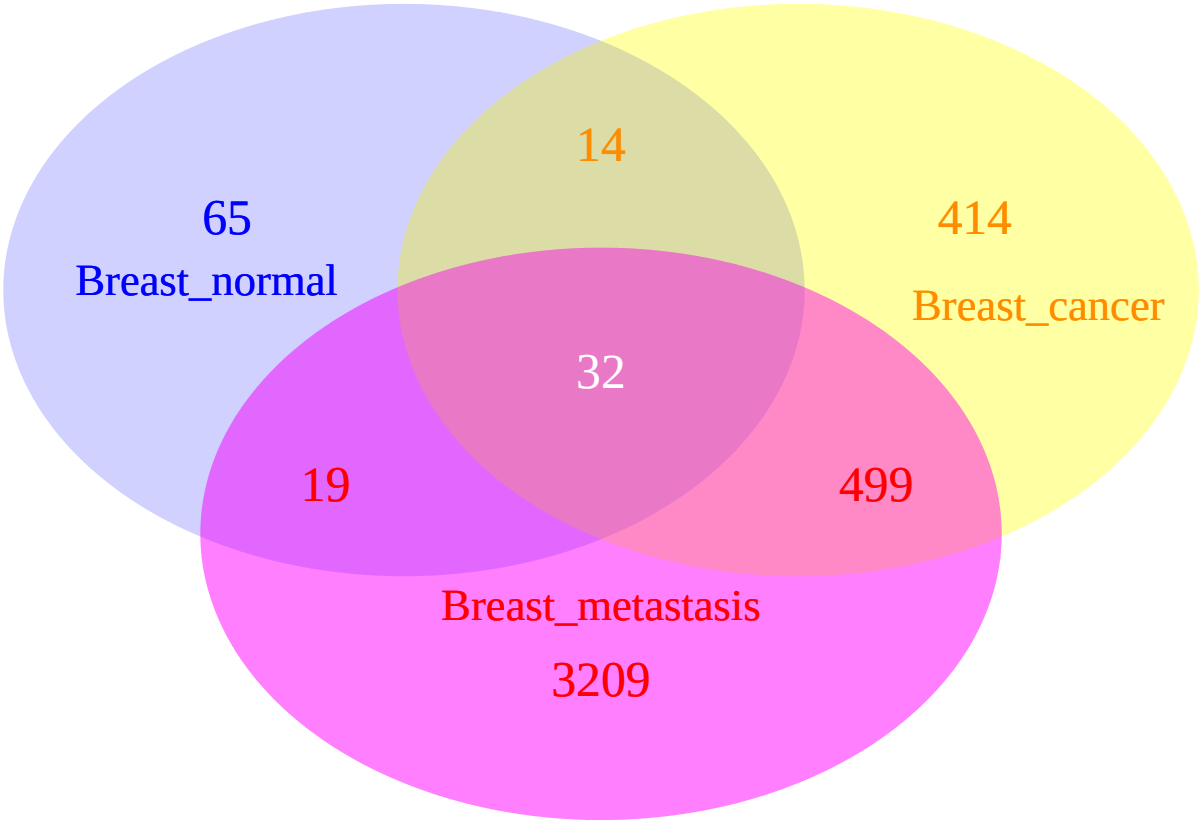
The Venn diagram of DIMPs reveals that the number cytosine site carrying methylation signal with a divergence level comparable to that observed in breast tissues with cancer and metastasis is relatively small (2797 DIMPs). The number of DIMPs decreased in the breast tissue with metastasis, but, as shown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



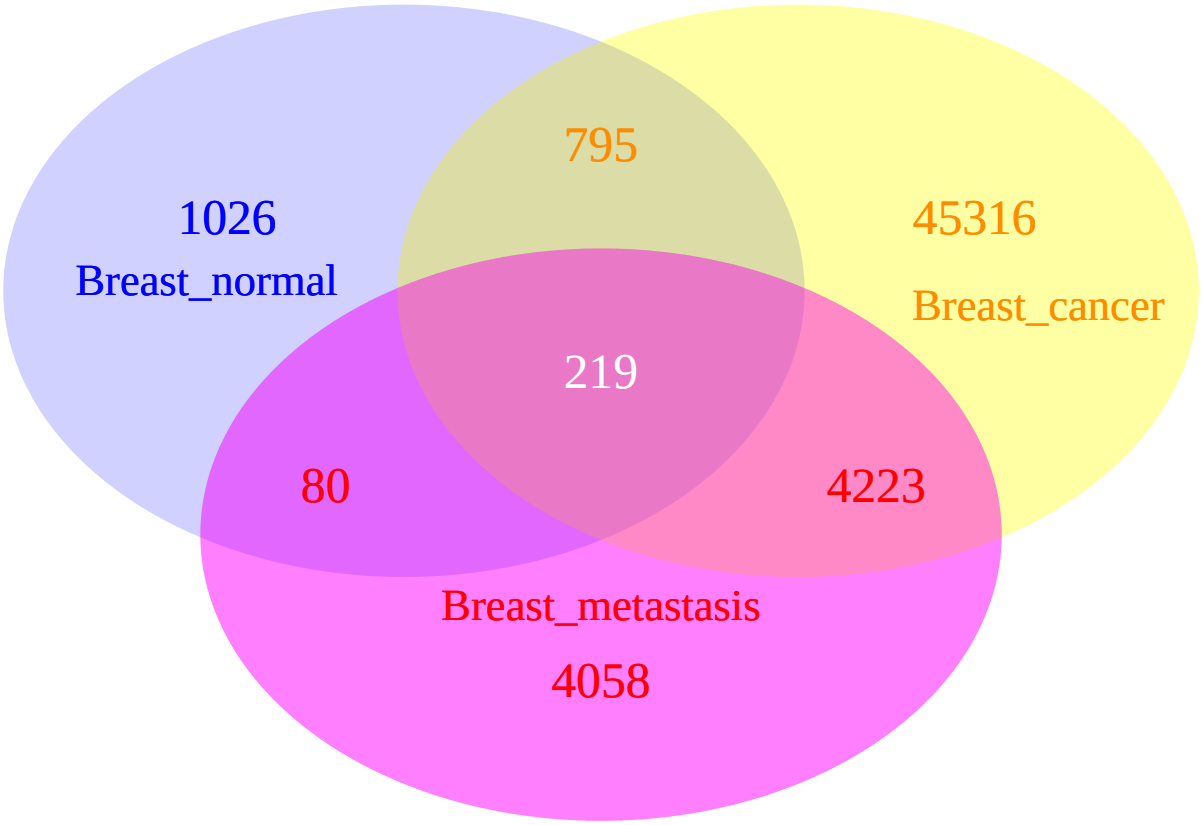
Notice that natural methylation regulatory signals (not induced by the treatment) are present in both groups, control and treatment. The signal detection step permits us to discriminate the "ordinary" signals observed in the control from those induced by the treatment (a disease in the current case). In addition, this diagram reflects a classification of DIMPs only based on the cytosine positions. That is, this Venn diagram cannot tell us whether DIMPs at the same position can be distinguishable or not. For example, DIMPs at the same positions in control and treatment can happened with different probabilities estimated from their corresponding fitted Weibull distributions (see below).

8.2. *Venn Diagram of DIMPs for reference Ref0*

```
DIMPs0 = selectDIMP(PS0, div.col = 9, cutpoint = 3.514418)
```

```
n12 = length(GenomicRanges::intersect(DIMPs0$Breast_normal,
                                      DIMPs0$Breast_cancer))
n13 = length(GenomicRanges::intersect(DIMPs0$Breast_normal,
                                      DIMPs0$Breast_metastasis))
n23 = length(GenomicRanges::intersect(DIMPs0$Breast_cancer,
                                      DIMPs0$Breast_metastasis))
n123 = length(Reduce(GenomicRanges::intersect,
                     list(DIMPs0$Breast_normal, DIMPs0$Breast_cancer,
                          DIMPs0$Breast_metastasis)))

grid.newpage()
v = draw.triple.venn(area1 = length(DIMPs0$Breast_normal),
                     area2 = length(DIMPs0$Breast_cancer),
                     area3 = length(DIMPs0$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)
```



9. 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 our GLM test looks for 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 (GLM) 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.

9.1. Differentially methylated genes (DMGs)

We shall call DMGs those DIMRs restricted to gene-body regions. DMGs are detected using function 'countTest'. 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(con = paste0("ftp://ftp.ensembl.org/pub/release-91/gff3/",
                         "homo_sapiens/Homo_sapiens.GRCh38.91.gff3.gz"))
genes = AG[ AG$type == "gene", c( "gene_id", "biotype" ) ]
genes = genes[ genes$biotype == "protein_coding", "gene_id" ]
seqlevels(genes, "coarse") <- "13" # To keep a consistent chromosome annotation
seqlevels(genes) <- "chr13"
```

Function 'getDIMPatGenes' is used to count the number of DIMPs at gene-body. The operation of this function is based on the 'findOverlaps' function from the 'GenomicRanges' Bioconductor R package. The 'findOverlaps' function has several critical parameters like, for example, 'maxgap', 'minoverlap', and 'ignore.strand'. In our function 'getDIMPatGenes', except for setting ignore.strand = TRUE and type = "within", we preserve the rest of default 'findOverlaps' parameters. In this case, these are important parameter settings 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 = getDIMPatGenes(GR = DIMPs$Breast_normal, GENES = genes)
DIMPsBC = getDIMPatGenes(GR = DIMPs$Breast_cancer, GENES = genes)
DIMPsBM = getDIMPatGenes(GR = DIMPs$Breast_metastasis, GENES = genes)
```

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")
```

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 'countTest'

```
DMGs = countTest(DIMR, num.cores = 3L, minCountPerIndv = 4, countFilter = TRUE,
                 Minlog2FC = 1, pvalCutOff = 0.05,
                 MVrate = .95, verbose = FALSE)
```

```
## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates
```

```
DMGs
```

```
## GRanges object with 10 ranges and 11 metadata columns:
##                    seqnames                ranges strand | Breast_normal
##                       <Rle>             <IRanges>  <Rle> |     <integer>
##   ENSG00000132932    chr13 [ 25372071,  26025851]      * |             1
##   ENSG00000132938    chr13 [ 28820348,  29505947]      * |             1
##   ENSG00000102763    chr13 [ 41566837,  41961120]      * |             1
##   ENSG00000183098    chr13 [ 93226842,  94407401]      * |             1
##   ENSG00000175198    chr13 [100089015, 100530437]      * |             4
##   ENSG00000125247    chr13 [100603927, 100675093]      * |             1
##   ENSG00000102452    chr13 [101053776, 101416492]      * |             1
##   ENSG00000204442    chr13 [107163510, 107866735]      * |             1
##   ENSG00000185974    chr13 [113667155, 113737735]      * |             1
##   ENSG00000185989    chr13 [113977783, 114132611]      * |             1
##                   Breast_normal1 Breast_cancer Breast_metastasis    log2FC
##                        <integer>     <integer>         <integer> <numeric>
##   ENSG00000132932              1             2                51  3.725693
##   ENSG00000132938              1            17               125  4.649187
##   ENSG00000102763              1             2                14  2.397895
##   ENSG00000183098              1            12                14  2.525729
##   ENSG00000175198              4            28                29  4.127134
##   ENSG00000125247              1           110               295  3.654978
##   ENSG00000102452              1             8                24  3.020425
##   ENSG00000204442              1             1                43  2.639057
##   ENSG00000185974              1             3                 7  2.302585
##   ENSG00000185989              1             5                23  2.944439
##                         pvalue          model     adj.pval CT.SignalDensity
##                      <numeric>    <character>    <numeric>        <numeric>
```

```
##   ENSG00000132932 1.923633e-07   Neg.Binomial 9.618167e-07     0.0015295642
##   ENSG00000132938 2.059568e-03   Neg.Binomial 4.119135e-03     0.0014585764
##   ENSG00000102763 1.167245e-03   Neg.Binomial 2.918113e-03     0.0025362429
##   ENSG00000183098 1.994458e-02   Neg.Binomial 3.095665e-02     0.0008470556
##   ENSG00000175198 1.752839e-07 Neg.Binomial.W 9.618167e-07     0.0090616030
##   ENSG00000125247 1.709922e-05 Neg.Binomial.W 5.699740e-05     0.0140514564
##   ENSG00000102452 2.166965e-02   Neg.Binomial 3.095665e-02     0.0027569703
##   ENSG00000204442 3.927490e-02 Neg.Binomial.W 4.087004e-02     0.0014220180
##   ENSG00000185974 3.997781e-02 Neg.Binomial.W 4.087004e-02     0.0141681189
##   ENSG00000185989 4.087004e-02   Neg.Binomial 4.087004e-02     0.0064587384
##                   TT.SignalDensity SignalDensityVariation
##                          <numeric>              <numeric>
##   ENSG00000132932       0.04053345             0.03900389
##   ENSG00000132938       0.10355893             0.10210035
##   ENSG00000102763       0.02028994             0.01775370
##   ENSG00000183098       0.01101172             0.01016467
##   ENSG00000175198       0.06456392             0.05550232
##   ENSG00000125247       2.84541993             2.83136847
##   ENSG00000102452       0.04411152             0.04135455
##   ENSG00000204442       0.03128440             0.02986238
##   ENSG00000185974       0.07084059             0.05667248
##   ENSG00000185989       0.09042234             0.08396360
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

9.2. DMGs for reference Ref0

```
DIMPs0BN = getDIMPatGenes(GR = DIMPs0$Breast_normal, GENES = genes)
DIMPs0BC = getDIMPatGenes(GR = DIMPs0$Breast_cancer, GENES = genes)
DIMPs0BM = getDIMPatGenes(GR = DIMPs0$Breast_metastasis, GENES = genes)

Genes.DIMPs0 = uniqueGRanges( list(DIMPs0BN[, 2], DIMPs0BN[, 2],
                                   DIMPs0BC[, 2], DIMPs0BM[, 2]),
                              type = "equal", verbose = FALSE,
                              ignore.strand = TRUE )
colnames( mcols(Genes.DIMPs0)) <- c("Breast_normal", "Breast_normal1",
                                    "Breast_cancer", "Breast_metastasis")

GeneID = subsetByOverlaps(genes, Genes.DIMPs0, type = "equal",
                          ignore.strand = FALSE)
dmps = data.frame( mcols( Genes.DIMPs0 ) )
dmps = apply( dmps, 2, as.numeric )
rownames( dmps ) <- GeneID$gene_id

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")

DIMR0 <- DESeqDataSetFromMatrix( countData = dmps,
                                 colData = condition,
                                 design = formula( ~ condition ),
                                 rowRanges = Genes.DIMPs0)
```

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

```
DMGs0 = countTest(DIMR0, num.cores = 3L, minCountPerIndv = 9,
                  countFilter = TRUE, Minlog2FC = 1, pvalCutOff = 0.05,
                  MVrate = .95, verbose = FALSE)
```

```
## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates
```

```
DMGs0
```

```
## GRanges object with 89 ranges and 11 metadata columns:
##                    seqnames                ranges strand | Breast_normal
##                       <Rle>             <IRanges>  <Rle> |     <integer>
##   ENSG00000132958    chr13   [19422877, 19536762]      * |             2
##   ENSG00000121390    chr13   [19674752, 19783019]      * |             2
##   ENSG00000121741    chr13   [19958670, 20091829]      * |             1
##   ENSG00000172458    chr13   [20702127, 20723098]      * |            10
##   ENSG00000132953    chr13   [20777329, 20903048]      * |             2
##               ...      ...                    ...    ... .           ...
##   ENSG00000150403    chr13 [113490995, 113554590]      * |            10
##   ENSG00000198176    chr13 [113584721, 113641470]      * |            11
##   ENSG00000185974    chr13 [113667155, 113737735]      * |             4
##   ENSG00000184497    chr13 [113759240, 113816995]      * |            10
##   ENSG00000185989    chr13 [113977783, 114132611]      * |            12
##                    Breast_normal1 Breast_cancer Breast_metastasis    log2FC
##                         <integer>     <integer>         <integer> <numeric>
##   ENSG00000132958               2           184                37  3.591424
##   ENSG00000121390               2           174                15  4.326023
##   ENSG00000121741               1           110                12  3.701302
##   ENSG00000172458              10             1                 0 -3.761200
##   ENSG00000132953               2            33                 6  2.683953
##               ...             ...           ...               ...       ...
##   ENSG00000150403              10            35                79  1.957427
##   ENSG00000198176              11            15                57  1.349155
##   ENSG00000185974               4            17                80  2.320078
##   ENSG00000184497              10            45                34  1.301737
##   ENSG00000185989              12            91                62  1.813947
##                          pvalue          model     adj.pval CT.SignalDensity
```

```
##                     <numeric>       <factor>    <numeric>        <numeric>
##   ENSG00000132958 0.0139844607 Neg.Binomial.W 0.0170495479      0.017561421
##   ENSG00000121390 0.0001587262   Neg.Binomial 0.0003924064      0.018472679
##   ENSG00000121741 0.0360133028 Neg.Binomial.W 0.0377080464      0.007509763
##   ENSG00000172458 0.0005714643 Neg.Binomial.W 0.0011056593      0.476826245
##   ENSG00000132953 0.0082692229   Neg.Binomial 0.0111509218      0.015908368
##               ...          ...           ...          ...              ...
##   ENSG00000150403 1.581619e-02   Neg.Binomial 1.902218e-02       0.15724259
##   ENSG00000198176 4.434089e-02   Neg.Binomial 4.484477e-02       0.19383260
##   ENSG00000185974 3.441486e-07   Neg.Binomial 2.187802e-06       0.05667248
##   ENSG00000184497 2.321227e-02   Neg.Binomial 2.648579e-02       0.17314218
##   ENSG00000185989 1.308002e-02 Neg.Binomial.W 1.616836e-02       0.07750486
##                   TT.SignalDensity SignalDensityVariation
##                          <numeric>             <numeric>
##   ENSG00000132958       0.97026851             0.9527071
##   ENSG00000121390       0.87283408             0.8543614
##   ENSG00000121741       0.45809552             0.4505858
##   ENSG00000172458       0.02384131            -0.4529849
##   ENSG00000132953       0.15510659             0.1391982
##               ...              ...                   ...
##   ENSG00000150403        0.8962828             0.7390402
##   ENSG00000198176        0.6343612             0.4405286
##   ENSG00000185974        0.6871538             0.6304813
##   ENSG00000184497        0.6839116             0.5107694
##   ENSG00000185989        0.4940935             0.4165886
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

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

```
# DMGs0
DMGs0[ grep( "ENSG00000139618", names(DMGs0) ) ]
```

```
## GRanges object with 1 range and 11 metadata columns:
##                   seqnames              ranges strand | Breast_normal
##                      <Rle>           <IRanges>  <Rle> |     <integer>
##   ENSG00000139618    chr13 [32315474, 32400266]      * |             3
##                   Breast_normal1 Breast_cancer Breast_metastasis    log2FC
##                        <integer>     <integer>         <integer> <numeric>
##   ENSG00000139618              3           125                31  3.516508
##                         pvalue        model     adj.pval CT.SignalDensity
##                      <numeric>     <factor>    <numeric>        <numeric>
##   ENSG00000139618 3.447489e-06 Neg.Binomial 1.804862e-05       0.03538028
##                   TT.SignalDensity SignalDensityVariation
##                          <numeric>             <numeric>
##   ENSG00000139618        0.9198873              0.884507
##   -------
##   seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

10. Classification of DIMPs into two classes

The regulatory methylation signal is an output from a natural process that continuously takes place across the ontogenetic development of the organism. Therefore, we expect to see methylation signal in natural, ordinary conditions. Function 'evaluateDIMPclass' can be used to perform a classification of DIMPs into two classes: DIMPS from control and DIMPs from treatment samples, as well as an evaluation of the classification performance (for more details see ?evaluateDIMPclass). In the setting below, a logistic regression: group versus divergence (at DIMPs), will be executed after randomly splitting the original DIMP dataset into two subsets: training (60%) and testing (40%).

The performance of the logistic classifier using reference 'Ref' is:

```
conf.mat <- evaluateDIMPclass(DIMPs,
                              column = c(hdiv = TRUE, TV = TRUE,
                                         wprob = TRUE, pos = TRUE),
                              control.names = "Breast_normal",
                              treatment.names = c("Breast_cancer",
                                                  "Breast_metastasis"),
                              output = "conf.mat", prop = 0.6)
```

```
## Model: treat ~ hdiv + TV + logP + pos
```

```
conf.mat$conf.mat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   CT   TT
##         CT   41    0
##         TT   11 1888
##
##                Accuracy : 0.9943
##                  95% CI : (0.9899, 0.9972)
##     No Information Rate : 0.9732
##     P-Value [Acc > NIR] : 3.972e-12
##
##                   Kappa : 0.8789
##  Mcnemar's Test P-Value : 0.002569
##
##             Sensitivity : 1.0000
##             Specificity : 0.7885
##          Pos Pred Value : 0.9942
##          Neg Pred Value : 1.0000
##              Prevalence : 0.9732
##          Detection Rate : 0.9732
##    Detection Prevalence : 0.9789
##       Balanced Accuracy : 0.8942
##
```

```
##        'Positive' Class : TT
##
```

The best fitted logistic model using reference 'Ref' is:

```
summary(conf.mat$model)
```

```
##
## Call:
## glm(formula = formula, family = binomial(link = "logit"), data = dt)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8978   0.0004   0.0025   0.0083   8.4904
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.020e+01  6.162e+00  -9.769  < 2e-16 ***
## hdiv        -3.803e+00  3.653e-01 -10.411  < 2e-16 ***
## TV          -2.509e-01  6.790e-01  -0.370  0.71172
## logP        -7.227e+01  7.151e+00 -10.107  < 2e-16 ***
## pos          1.461e-08  5.045e-09   2.897  0.00377 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 718.38  on 2907  degrees of freedom
## Residual deviance: 225.67  on 2903  degrees of freedom
## AIC: 235.67
##
## Number of Fisher Scoring iterations: 11
```

In this case, the only variable not included in the model is total variation TV and all the rest are significant. The generalized linear regression can be performed by removing the variables TV. There are three other classifiers available: "pca.logistic", "pca.lda", and "pca.qda" (type ?evaluateDIMPclass in R console for more details). Principal component analysis (PCA) is used to convert a set of observations of possibly correlated predictor variables into a set of values of linearly uncorrelated variables (principal components, PCs). Then, the PCs are used as new uncorrelated predictor variables for LDA, QDA, and logistic classifiers. In the current case, the best classification result is obtained with the combination PCA + Quadratic Discriminant Analysis (PCA + QDA, "pca.qda").

```
conf.mat <- evaluateDIMPclass(DIMPs,
                              column = c(hdiv = TRUE, TV = TRUE,
                                         wprob = TRUE, pos = TRUE),
                              classifier = "pca.qda", n.pc = 4,
                              center = TRUE, scale = TRUE,
                              control.names = "Breast_normal",
                              treatment.names = c("Breast_cancer",
```

```
                                                    "Breast_metastasis"),
                                output = "conf.mat", prop = 0.6)
```

```
## Model: treat ~ hdiv + TV + logP + pos
```

```
conf.mat$conf.mat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   CT   TT
##         CT   47    0
##         TT    5 1888
##
##                Accuracy : 0.9974
##                  95% CI : (0.994, 0.9992)
##     No Information Rate : 0.9732
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.9482
##  Mcnemar's Test P-Value : 0.07364
##
##             Sensitivity : 1.0000
##             Specificity : 0.9038
##          Pos Pred Value : 0.9974
##          Neg Pred Value : 1.0000
##              Prevalence : 0.9732
##          Detection Rate : 0.9732
##    Detection Prevalence : 0.9758
##       Balanced Accuracy : 0.9519
##
##        'Positive' Class : TT
##
```

```
summary(conf.mat$model)
```

```
##     Length Class  Mode
## qda 8      qda    list
## pca 5      prcomp list
```

Monte Carlo (bootstrap) validation with 500 resamplings is performed by using the option 'output = "mc.val" ':

```
conf.mat <- evaluateDIMPclass(DIMPs,
                              column = c(hdiv = TRUE, TV = TRUE,
                                         wprob = TRUE, pos = TRUE),
                              classifier = "pca.qda", n.pc = 4,
                              center = TRUE, scale = TRUE,
                              control.names = "Breast_normal",
                              treatment.names = c("Breast_cancer",
```

```
                                                    "Breast_metastasis"),
                                output = "mc.val", prop = 0.6,
                                mc.cores = 12L, num.boot = 500)
```

```
## Model: treat ~ hdiv + TV + logP + pos
```

```
conf.mat
```

```
##     Accuracy          Kappa        AccuracyLower    AccuracyUpper
##  Min.   :0.9943   Min.   :0.8813   Min.   :0.9899   Min.   :0.9972
##  1st Qu.:0.9974   1st Qu.:0.9482   1st Qu.:0.9940   1st Qu.:0.9992
##  Median :0.9979   Median :0.9597   Median :0.9947   Median :0.9994
##  Mean   :0.9979   Mean   :0.9580   Mean   :0.9947   Mean   :0.9993
##  3rd Qu.:0.9985   3rd Qu.:0.9706   3rd Qu.:0.9955   3rd Qu.:0.9997
##  Max.   :1.0000   Max.   :1.0000   Max.   :0.9981   Max.   :1.0000
##
##   AccuracyNull    AccuracyPValue      McnemarPValue       Sensitivity
##  Min.   :0.9732   Min.   :0.000e+00   Min.   :0.007661   Min.   :0.9984
##  1st Qu.:0.9732   1st Qu.:0.000e+00   1st Qu.:0.145816   1st Qu.:0.9995
##  Median :0.9732   Median :5.000e-18   Median :0.479500   Median :0.9995
##  Mean   :0.9732   Mean   :1.437e-14   Mean   :0.524451   Mean   :0.9996
##  3rd Qu.:0.9732   3rd Qu.:5.100e-17   3rd Qu.:1.000000   3rd Qu.:1.0000
##  Max.   :0.9732   Max.   :3.972e-12   Max.   :1.000000   Max.   :1.0000
##                                       NA's   :2
##   Specificity     Pos Pred Value   Neg Pred Value     Precision
##  Min.   :0.8077   Min.   :0.9947   Min.   :0.9400   Min.   :0.9947
##  1st Qu.:0.9183   1st Qu.:0.9978   1st Qu.:0.9787   1st Qu.:0.9978
##  Median :0.9423   Median :0.9984   Median :0.9804   Median :0.9984
##  Mean   :0.9366   Mean   :0.9983   Mean   :0.9837   Mean   :0.9983
##  3rd Qu.:0.9615   3rd Qu.:0.9989   3rd Qu.:1.0000   3rd Qu.:0.9989
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##
##      Recall             F1           Prevalence     Detection Rate
##  Min.   :0.9984   Min.   :0.9971   Min.   :0.9732   Min.   :0.9716
##  1st Qu.:0.9995   1st Qu.:0.9987   1st Qu.:0.9732   1st Qu.:0.9727
##  Median :0.9995   Median :0.9989   Median :0.9732   Median :0.9727
##  Mean   :0.9996   Mean   :0.9989   Mean   :0.9732   Mean   :0.9728
##  3rd Qu.:1.0000   3rd Qu.:0.9992   3rd Qu.:0.9732   3rd Qu.:0.9732
##  Max.   :1.0000   Max.   :1.0000   Max.   :0.9732   Max.   :0.9732
##
##  Detection Prevalence Balanced Accuracy
##  Min.   :0.9716       Min.   :0.9036
##  1st Qu.:0.9737       1st Qu.:0.9585
##  Median :0.9742       Median :0.9709
##  Mean   :0.9745       Mean   :0.9681
##  3rd Qu.:0.9753       3rd Qu.:0.9808
##  Max.   :0.9778       Max.   :1.0000
##
```

The performance of the PCA+QDA classifier using reference 'Ref0' is:

```
conf.mat0 <- evaluateDIMPclass(DIMPs0,
                               column = c(hdiv = TRUE, TV = TRUE,
                                          wprob = TRUE, pos = TRUE),
                               classifier = "pca.qda", n.pc = 4,
                               center = TRUE, scale = TRUE,
                               control.names = "Breast_normal",
                               treatment.names = c("Breast_cancer",
                                                   "Breast_metastasis"),
                               output = "conf.mat", prop = 0.6)
```

```
## Model: treat ~ hdiv + TV + logP + pos
```

```
conf.mat0$conf.mat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    CT    TT
##         CT   848     0
##         TT     0 23654
##
##                Accuracy : 1
##                  95% CI : (0.9998, 1)
##     No Information Rate : 0.9654
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##  Mcnemar's Test P-Value : NA
##
##             Sensitivity : 1.0000
##             Specificity : 1.0000
##          Pos Pred Value : 1.0000
##          Neg Pred Value : 1.0000
##              Prevalence : 0.9654
##          Detection Rate : 0.9654
##    Detection Prevalence : 0.9654
##       Balanced Accuracy : 1.0000
##
##        'Positive' Class : TT
##
```

Monte Carlo (bootstrap) validation with 500 resamplings using reference 'Ref0' can be now performed:

```
conf.mat01 <- evaluateDIMPclass(DIMPs0,
                                column = c(hdiv = TRUE, TV = TRUE,
                                           wprob = TRUE, pos = TRUE),
                                classifier = "pca.qda", n.pc = 4,
                                center = TRUE, scale = TRUE,
```

```
                                control.names = "Breast_normal",
                                treatment.names = c("Breast_cancer",
                                                    "Breast_metastasis"),
                                output = "mc.val", prop = 0.6,
                                mc.cores = 12L, num.boot = 500)
```

```
## Model: treat ~ hdiv + TV + logP + pos
```

```
conf.mat01
```

```
##     Accuracy     Kappa   AccuracyLower    AccuracyUpper  AccuracyNull
##  Min.   :1   Min.   :1   Min.   :0.9998   Min.   :1     Min.   :0.9654
##  1st Qu.:1   1st Qu.:1   1st Qu.:0.9998   1st Qu.:1     1st Qu.:0.9654
##  Median :1   Median :1   Median :0.9998   Median :1     Median :0.9654
##  Mean   :1   Mean   :1   Mean   :0.9998   Mean   :1     Mean   :0.9654
##  3rd Qu.:1   3rd Qu.:1   3rd Qu.:0.9998   3rd Qu.:1     3rd Qu.:0.9654
##  Max.   :1   Max.   :1   Max.   :0.9998   Max.   :1     Max.   :0.9654
##
##  AccuracyPValue McnemarPValue  Sensitivity  Specificity Pos Pred Value
##  Min.   :0      Min.   : NA   Min.   :1    Min.   :1    Min.   :1
##  1st Qu.:0      1st Qu.: NA   1st Qu.:1    1st Qu.:1    1st Qu.:1
##  Median :0      Median : NA   Median :1    Median :1    Median :1
##  Mean   :0      Mean   :NaN   Mean   :1    Mean   :1    Mean   :1
##  3rd Qu.:0      3rd Qu.: NA   3rd Qu.:1    3rd Qu.:1    3rd Qu.:1
##  Max.   :0      Max.   : NA   Max.   :1    Max.   :1    Max.   :1
##                 NA's   :500
##  Neg Pred Value   Precision     Recall        F1      Prevalence
##  Min.   :1      Min.   :1   Min.   :1   Min.   :1   Min.   :0.9654
##  1st Qu.:1      1st Qu.:1   1st Qu.:1   1st Qu.:1   1st Qu.:0.9654
##  Median :1      Median :1   Median :1   Median :1   Median :0.9654
##  Mean   :1      Mean   :1   Mean   :1   Mean   :1   Mean   :0.9654
##  3rd Qu.:1      3rd Qu.:1   3rd Qu.:1   3rd Qu.:1   3rd Qu.:0.9654
##  Max.   :1      Max.   :1   Max.   :1   Max.   :1   Max.   :0.9654
##
##  Detection Rate   Detection Prevalence Balanced Accuracy
##  Min.   :0.9654   Min.   :0.9654       Min.   :1
##  1st Qu.:0.9654   1st Qu.:0.9654       1st Qu.:1
##  Median :0.9654   Median :0.9654       Median :1
##  Mean   :0.9654   Mean   :0.9654       Mean   :1
##  3rd Qu.:0.9654   3rd Qu.:0.9654       3rd Qu.:1
##  Max.   :0.9654   Max.   :0.9654       Max.   :1
##
```

That is, with high accuracy level, DIMPs from control group can be discriminated from DIMPs found in cancer tissues.

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. Troubleshooting installation on Ubuntu

Herein, a possible path to prevent potential issues originated during MethylIT installation on Ubuntu is given:

1. To update R:

   i. To added an R CRAN repository typing in the terminal:
   sudo echo “deb /bin/linux/ubuntu xenial/” | sudo tee -a /etc/apt/sources.list

   For example:

```
sudo echo "deb https://cran.mtu.edu/bin/linux/ubuntu xenial/" |
  sudo tee -a /etc/apt/sources.list
```

   ii. sudo apt update

   iii. sudo apt upgrade

2. Install Bioconductor:
   source(“https://bioconductor.org/biocLite.R”)
   biocLite()

3. Install Bioconductor packages: ‘GenomicFeatures’, ‘VariantAnnotation’, ‘ensembldb’, ‘GenomicRanges’, ‘BiocParallel’, ‘biovizBase’, ‘DESeq2’, and ‘genefilter’. Package ‘GenomicFeatures’ depends on the R package ‘RMySQL’, which is not in ’Bioconductor. To install “RMySQL” from CRAN you might require the ’installation of the library “libmysqlclient-dev”. If this is the case, ’then you can solve it by typing in the Ubuntu Teminal:

```
sudo apt install libmysqlclient-dev
```

   Next, in the R console:

```
install.packages("RMySQL")
```

4. install.packages(“devtools”)

5. devtools::install_git(“https://git.psu.edu/genomath/MethylIT”)

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=en_US.UTF-8
##  [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] grid      parallel  stats4    stats     graphics  grDevices utils
##  [8] datasets  methods   base
##
## other attached packages:
##  [1] VennDiagram_1.6.19         futile.logger_1.4.3
##  [3] MethylIT_0.3.1             rtracklayer_1.38.3
##  [5] DESeq2_1.18.1              SummarizedExperiment_1.8.1
##  [7] DelayedArray_0.4.1         matrixStats_0.53.1
##  [9] Biobase_2.38.0             GenomicRanges_1.30.3
## [11] GenomeInfoDb_1.14.0        IRanges_2.12.0
## [13] S4Vectors_0.16.0           BiocGenerics_0.24.0
## [15] knitr_1.20
##
## 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           ggplot2_2.2.1
##   [9] digest_0.6.15                 foreach_1.4.4
##  [11] BiocInstaller_1.28.0          ensembldb_2.2.2
##  [13] htmltools_0.3.6               magrittr_1.5
##  [15] checkmate_1.8.5               memoise_1.1.0
##  [17] BSgenome_1.46.0               cluster_2.0.6
##  [19] sfsmisc_1.1-2                 etm_0.6-2
##  [21] recipes_0.1.2                 Biostrings_2.46.0
##  [23] annotate_1.56.2               gower_0.1.2
##  [25] dimRed_0.1.0                  ArgumentCheck_0.10.2
##  [27] prettyunits_1.0.2             colorspace_1.3-2
##  [29] blob_1.1.1                    dplyr_0.7.4
##  [31] RCurl_1.95-4.10               genefilter_1.60.0
##  [33] bindr_0.1.1                   survival_2.41-3
##  [35] VariantAnnotation_1.24.5      zoo_1.8-1
##  [37] iterators_1.0.9               glue_1.2.0
##  [39] DRR_0.0.3                     gtable_0.2.0
```

```
##  [41] ipred_0.9-6                   zlibbioc_1.24.0
##  [43] XVector_0.18.0                kernlab_0.9-25
##  [45] ddalpha_1.3.1.1               DEoptimR_1.0-8
##  [47] scales_0.5.0                  futile.options_1.0.0
##  [49] DBI_0.8                       Rcpp_0.12.16
##  [51] xtable_1.8-2                  progress_1.1.2
##  [53] cmprsk_2.2-7                  htmlTable_1.11.2
##  [55] foreign_0.8-69                bit_1.1-12
##  [57] Formula_1.2-2                 lava_1.6
##  [59] prodlim_1.6.1                 htmlwidgets_1.0
##  [61] httr_1.3.1                    RColorBrewer_1.1-2
##  [63] acepack_1.4.1                 pkgconfig_2.0.1
##  [65] XML_3.98-1.10                 nnet_7.3-12
##  [67] locfit_1.5-9.1                caret_6.0-78
##  [69] tidyselect_0.2.4              rlang_0.2.0
##  [71] reshape2_1.4.3                AnnotationDbi_1.40.0
##  [73] munsell_0.4.3                 tools_3.4.3
##  [75] RSQLite_2.0                   broom_0.4.3
##  [77] evaluate_0.10.1               stringr_1.3.0
##  [79] yaml_2.1.18                   ModelMetrics_1.1.0
##  [81] bit64_0.9-7                   robustbase_0.92-8
##  [83] purrr_0.2.4                   AnnotationFilter_1.2.0
##  [85] bindrcpp_0.2                  nlme_3.1-131.1
##  [87] mime_0.5                      RcppRoll_0.2.2
##  [89] biomaRt_2.34.2                compiler_3.4.3
##  [91] rstudioapi_0.7                curl_3.1
##  [93] interactiveDisplayBase_1.16.0 e1071_1.6-8
##  [95] tibble_1.4.2                  geneplotter_1.56.0
##  [97] stringi_1.1.7                 GenomicFeatures_1.30.3
##  [99] Epi_2.26                      lattice_0.20-35
## [101] ProtGenerics_1.10.0           Matrix_1.2-12
## [103] psych_1.7.8                   pillar_1.2.1
## [105] data.table_1.10.4-3           bitops_1.0-6
## [107] httpuv_1.3.6.2                R6_2.2.2
## [109] latticeExtra_0.6-28           RMySQL_0.10.14
## [111] gridExtra_2.3                 codetools_0.2-15
## [113] lambda.r_1.2                  dichromat_2.0-0
## [115] MASS_7.3-49                   assertthat_0.2.0
## [117] CVST_0.2-1                    rprojroot_1.3-2
## [119] minpack.lm_1.2-1              withr_2.1.2
## [121] GenomicAlignments_1.14.1      Rsamtools_1.30.0
## [123] mnormt_1.5-5                  GenomeInfoDbData_1.0.0
## [125] rpart_4.1-13                  timeDate_3043.102
## [127] tidyr_0.8.0                   class_7.3-14
## [129] nls2_0.2                      rmarkdown_1.9
## [131] biovizBase_1.26.0             lubridate_1.7.3
## [133] numDeriv_2016.8-1             shiny_1.0.5
## [135] 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.