Package 'DMRcate'

October 27, 2015

Title Illumina 450K methylation array spatial analysis methods

Version 1.6.0

Date 2015-13-07

Author Tim Peters

Maintainer Tim Peters <t.peters@garvan.org.au>

Description
De novo identification and extraction of differentially methylated regions (DMRs) in the human genome using Illumina Infinium HumanMethylation450 BeadChip array data. Provides functionality for filtering probes possibly confounded by SNPs and cross-hybridisation. Includes bedGraph generation, GRanges generation and plotting functions.

Depends R (>= 3.2.1), limma, minfi, DMRcatedata

Imports methods, graphics

biocViews DifferentialMethylation, GeneExpression, Microarray, MethylationArray, Genetics, DifferentialExpression, GenomeAnnotation, DNAMethylation, OneChannel, TwoChannel, MultipleComparison, QualityControl, TimeCourse

Suggests knitr, RUnit, BiocGenerics, IlluminaHumanMethylation450kanno.ilmn12.hg19

License file LICENSE

VignetteBuilder knitr

NeedsCompilation no

R topics documented:

DMRcate-package *Illumina 450K methylation array spatial analysis*

Description

De novo identification and extraction of differentially methylated regions (DMR) in the human genome using 450k array data. DMRcate extracts and annotates differentially methylated regions (DMRs) using an array-bias corrected smoothed estimate. Functions are provided for filtering probes possibly confounded by SNPs and cross-hybridisation. Includes bedGraph generation, GRanges generation and plotting functions.

Author(s)

Tim J. Peters <Tim.Peters@csiro.au>

References

Peters T.J., Buckley M.J., Statham, A., Pidsley R., Samaras K., Lord R.V., Clark S.J. and Molloy P.L. *De novo* identification of differentially methylated regions in the human genome. *Epigenetics & Chromatin* 2015, **8**:6, doi:10.1186/1756-8935-8-6

Examples

```
data(dmrcatedata)
myMs <- logit2(myBetas)
myMs.noSNPs <- rmSNPandCH(myMs, dist=2, mafcut=0.05)
patient <- factor(sub("-.*", "", colnames(myMs)))
type <- factor(sub(".*-", "", colnames(myMs)))
design <- model.matrix(~patient + type)
myannotation <- cpg.annotate(myMs.noSNPs, analysis.type="differential",
    design=design, coef=39)
dmrcoutput <- dmrcate(myannotation, lambda=1000)
makeBedgraphs(dmrcoutput=dmrcoutput, betas=myBetas, samps=c(1,39))
phen.col <- c(rep("orange", 38), rep("blue", 38))
DMR.plot(dmrcoutput=dmrcoutput, dmr=1, betas=myBetas, phen.col=phen.col,
    pch=16, toscale=TRUE, plotmedians=TRUE)
```

cpg.annotate *450k probe annotation*

Description

Annotates a matrix of *M*-values (logit transform of beta) with weights (depending on analysis.type) and other relevant information including gene association.

Usage

```
cpg.annotate(object, annotation=c(array="IlluminaHumanMethylation450k",
                                  annotation="ilmn12.hg19"),
         analysis.type=c("differential", "variability"),
         design, contrasts=FALSE, cont.matrix=NULL, coef, ...)
```

Arguments

object	A matrix of M-values, with unique Illumina probe IDs as rownames and unique sample IDs as column names.
annotation	A vector describing the type of annotation to affix to object. Identical context to minfi, i.e. annotation <- annotation(minfiobject) where minfiobject is a [Genomic](Methyl|Ratio)Set). Default (ilmn12.hg) is recommended.
analysis.type	"differential" for dmrcate() to return DMRs and "variability" to return VMRs.
design	Study design matrix. Identical context to differential analysis pipeline in limma. Must have an intercept if contrasts=FALSE. Applies only when analysis.type="differential".
contrasts	Logical denoting whether a limma-style contrast matrix is specified.
cont.matrix	Limma-style contrast matrix for explicit contrasting. For each call to cpg.annotate, only one contrast will be fit.
coef	The column index in design corresponding to the phenotype comparison. Corresponds to the comparison of interest in design when contrasts=FALSE, otherwise must be a column name in cont.matrix. Applies only when analysis.type="differential
...	Extra arguments passed to the limma function lmFit(). Applies only when analysis.type="differential".

Value

An object of class "annot", for passing to dmrcate, containing the vectors:

- ID: Illumina probe ID
- weights: *t*-statistic between phenotypes for each probe
- CHR: Chromosome which the probe maps to
- pos: hg19 position (on CHR) that the probe maps to
- gene: Matching UCSC_RefGene_Name
- group: Matching UCSC_RefGene_Group
- betafc: The beta fold change according to the given design
- indfdr: The post-kernel fitting limma fdr value

Author(s)

Tim J. Peters <Tim.Peters@csiro.au>

References

Smyth, G. K. (2005). Limma: linear models for microarray data. In: *Bioinformatics and Computational Biology Solutions using R and Bioconductor*, R. Gentleman, V. Carey, S. Dudoit, R. Irizarry, W. Huber (eds.), Springer, New York, pages 397-420.

Peters T.J., Buckley M.J., Statham, A., Pidsley R., Samaras K., Lord R.V., Clark S.J. and Molloy P.L. *De novo* identification of differentially methylated regions in the human genome. *Epigenetics & Chromatin* 2015, **8**:6, doi:10.1186/1756-8935-8-6.

Examples

```
## Not run:
data(dmrcatedata)
myMs <- logit2(myBetas)
myMs.noSNPs <- rmSNPandCH(myMs, dist=2, mafcut=0.05)
patient <- factor(sub("-.*", "", colnames(myMs)))
type <- factor(sub(".*-", "", colnames(myMs)))
design <- model.matrix(~patient + type)
myannotation <- cpg.annotate(myMs.noSNPs, analysis.type="differential",
    design=design, coef=39)
## End(Not run)
```

DMR.plot *Plotting DMRs*

Description

Plots an individual DMR as found by dmrcate.

Usage

```
DMR.plot(dmrcoutput, dmr, betas, phen.col,
         annotation=c(array="IlluminaHumanMethylation450k",
                      annotation="ilmn12.hg19"),
         samps=NULL, toscale=FALSE, plotmedians=FALSE, ...)
```

Arguments

dmrcoutput	An object of class dmrcate.output.
dmr	Row index of dmrcoutput$results. Indicates which DMR to be plotted. Will only plot regions constituted of 2 or more CpGs.
betas	Matrix of beta values for plotting, with unique Illumina probe IDs as rownames.
phen.col	Vector of colors denoting phenotypes. Should be length ncol(betas)[samps].
annotation	A vector describing the type of annotation from which plots are derived. Identical context to minfi, i.e. annotation <- annotation(minfiobject) where minfiobject is a [Genomic](Methyl|Ratio)Set). Default (ilmn12.hg) is recommended.
samps	Vector of samples to be plotted, corresponding to columns of betas. Default is all samples plotted.
toscale	TRUE denotes CpGs plotted to scale along the x-axis according to their genomic coordinates, FALSE denotes evenly spaced plotting. Default is FALSE.
plotmedians	Logical denoting whether group medians will be plotted. Groups are derived from phen.col.
...	Extra arguments passed to plot

Value

A plot to the current device. Square points along the top correspond to gene annotation; colours are as follows:

TSS1500: Light green

TSS200: Dark green

Gene Body: Red

1st Exon: Magenta

5'UTR: Dark Blue

3'UTR: Cyan

Author(s)

Tim J. Peters <Tim.Peters@csiro.au>

Examples

```
## Not run:
data(dmrcatedata)
myMs <- logit2(myBetas)
myMs.noSNPs <- rmSNPandCH(myMs, dist=2, mafcut=0.05)
patient <- factor(sub("-.*", "", colnames(myMs)))
type <- factor(sub(".*-", "", colnames(myMs)))
design <- model.matrix(~patient + type)
myannotation <- cpg.annotate(myMs.noSNPs, analysis.type="differential",
    design=design, coef=39)
dmrcoutput <- dmrcate(myannotation, lambda=1000)
phen.col <- c(rep("orange", 38), rep("blue", 38))
DMR.plot(dmrcoutput=dmrcoutput, dmr=1, betas=myBetas, phen.col=phen.col,
    pch=16, toscale=TRUE, plotmedians=TRUE)

## End(Not run)
```

`dmrcate` *DMR identification*

Description

The main function of this package. Computes a kernel estimate against a null comparison to identify significantly differentially (or variable) methylated regions in hg19.

Usage

```
dmrcate(object,
        lambda = 1000,
        C=2,
        p.adjust.method = "BH",
        pcutoff = "limma",
        consec = FALSE,
        conseclambda = 10,
        betacutoff = NULL
        )
```

Arguments

object	A class of type "annot", created from cpg.annotate.
lambda	Gaussian kernel bandwidth for smoothed-function estimation. Also informs DMR bookend definition; gaps >= lambda between significant probes will be in separate DMRs. Support is truncated at 5*lambda. Default is 1000 nucleotides. See details for further info.
C	Scaling factor for bandwidth. Gaussian kernel is calculated where lambda/C = sigma. Empirical testing shows that when lambda=1000, near-optimal prediction of sequencing-derived DMRs is obtained when C is approximately 2, i.e. 1 standard deviation of Gaussian kernel = 500 base pairs. Cannot be < 0.2.
p.adjust.method	Method for *p*-value adjustment from the significance test. Default is "BH" (Benjamini-Hochberg).
pcutoff	*p*-value cutoff to determine DMRs. Default is automatically determined by the number of significant probes returned by limma for that contrast, but can be set manually with a numeric value.
consec	Use DMRcate in consecutive probe mode. Treats CpG sites as equally spaced.
conseclambda	Bandwidth in *probes* (rather than nucleotides) to use when consec=TRUE. When specified the variable lambda simply becomes the minumum distance separating DMRs.
betacutoff	Optional filter; removes any region from the results that does not have at least one CpG site with a beta fold change exceeding this value.

Details

The values of lambda and C should be chosen with care. We recommend that half a kilobase represent 1 standard deviation of support (lambda=1000 and C=2). If lambda is too small or C too large then the kernel estimator will not have enough support to significantly differentiate the weighted estimate from the null distribution. If lambda is too large then dmrcate will report very long DMRs spanning multiple gene loci, and the large amount of support will likely give Type I errors. If you are concerned about Type I errors we recommend using the default value of pcutoff, although this will return no DMRs if no DM probes are returned by limma either.

Many gene loci have lengths reaching into the hundreds of thousands of base pairs, so it is quite possible that multiple signficant regions will have identical values in results$gene_assoc. This is fine; these regions are distinct in that they are at the very least lambda nucleotides apart, and is preferable to attempting collapse into a super-DMR by increasing lambda.

Value

A list containing 2 data frames (input and results) and a numeric value (cutoff). input contains the contents of the annot object, plus calculated *p*-values:

- ID: As per annotation object input
- weights: As per annotation object input
- CHR: As per annotation object input
- pos: As per annotation object input
- gene: As per annotation object input
- group: As per annotation object input
- betafc: As per annotation object input

- raw: Raw *p*-values from the significance test
- fdr: Adjusted *p*-values from the significance test

results contains an annotated data.frame of significant regions, ranked by minpval:

- gene_assoc: Complete list of gene loci overlapping the region, comma-separated
- group: Complete list of gene annotations (e.g. TSS1500, 5'UTR etc.) overlapping the region, comma-separated
- hg19coords: Coordinates of the significant region in hg19. IGV-friendly.
- no.probes: Number of probes constituting the significant region. Tie-breaker when sorting probes by minpval. A few regions may report no.probes=1, which may seem counter-intuitive, but this is only because the adjacent probes are either just below the significance threshold, or it is a highly DM probe in a sparse region. Unless pcutoff is highly conservative, it is unlikely that these regions will report at the head of the sorted list.
- minpval: Minimum adjusted *p*-value from the probes constituting the significant region.
- meanpval: Mean adjusted *p*-value from the probes constituting the significant region.
- maxbetafc: Maximum absolute beta fold change within the region

cutoff is the signficance *p*-value cutoff provided in the call to dmrcate.

Author(s)

Tim J. Peters <Tim.Peters@csiro.au>, Mike J. Buckley <Mike.Buckley@csiro.au>, Tim Triche Jr. <tim.triche@usc.edu>

References

Peters T.J., Buckley M.J., Statham, A., Pidsley R., Samaras K., Lord R.V., Clark S.J. and Molloy P.L. *De novo* identification of differentially methylated regions in the human genome. *Epigenetics & Chromatin* 2015, **8**:6, doi:10.1186/1756-8935-8-6

Wand, M.P. & Jones, M.C. (1995) *Kernel Smoothing*. Chapman & Hall.

Duong T. (2013) Local significant differences from nonparametric two-sample tests. *Journal of Nonparametric Statistics*. 2013 **25**(3), 635-645.

Examples

```
## Not run:
data(dmrcatedata)
myMs <- logit2(myBetas)
myMs.noSNPs <- rmSNPandCH(myMs, dist=2, mafcut=0.05)
patient <- factor(sub("-.*", "", colnames(myMs)))
type <- factor(sub(".*-", "", colnames(myMs)))
design <- model.matrix(~patient + type)
myannotation <- cpg.annotate(myMs.noSNPs, analysis.type="differential",
    design=design, coef=39)
dmrcoutput <- dmrcate(myannotation, lambda=1000)

## End(Not run)
```

extractRanges *Create GRanges object from dmrcate output.*

Description

Takes a dmrcate.output object and produces the corresponding GRanges object.

Usage

```
extractRanges(dmrcoutput)
```

Arguments

dmrcoutput	An object of class dmrcate.output.

Value

A GRanges object.

Author(s)

Tim Triche Jr. <tim.triche@usc.edu>, Tim Peters <Tim.Peters@csiro.au>

Examples

```
## Not run:
data(dmrcatedata)
myMs <- logit2(myBetas)
myMs.noSNPs <- rmSNPandCH(myMs, dist=2, mafcut=0.05)
patient <- factor(sub("-.*", "", colnames(myMs)))
type <- factor(sub(".*-", "", colnames(myMs)))
design <- model.matrix(~patient + type)
myannotation <- cpg.annotate(myMs.noSNPs, analysis.type="differential",
    design=design, coef=39)
dmrcoutput <- dmrcate(myannotation, lambda=1000)
myRanges <- extractRanges(dmrcoutput)

## End(Not run)
```

makeBedgraphs *Outputs bedGraphs*

Description

Makes bedGraphs, 1 per sample, each containing all significant regions found via dmrcate. Bedgraphs are written are written to the working directory.

Usage

```
makeBedgraphs(dmrcoutput, betas,
    annotation=c(array="IlluminaHumanMethylation450k",
        annotation="ilmn12.hg19"),
    samps=NULL)
```

Arguments

dmrcoutput	An object of class dmrcate.output.
betas	Matrix of beta values to be converted to bedGraph rows, with unique Illumina probe IDs as rownames.
annotation	A vector describing the type of annotation from which to derive bedgraph output. Identical context to minfi, i.e. annotation <- annotation(minfiobject) where minfiobject is a [Genomic](Methyl|Ratio)Set). Default (ilmn12.hg) is recommended.
samps	Vector of samples to be converted to bedGraph files, corresponding to columns of betas. Default is all samples plotted.

Value

Writes zero or more bedGraph files to the working directory.

Author(s)

Tim J. Peters <Tim.Peters@csiro.au>

Examples

```
## Not run:
data(dmrcatedata)
myMs <- logit2(myBetas)
myMs.noSNPs <- rmSNPandCH(myMs, dist=2, mafcut=0.05)
patient <- factor(sub("-.*", "", colnames(myMs)))
type <- factor(sub(".*-", "", colnames(myMs)))
design <- model.matrix(~patient + type)
myannotation <- cpg.annotate(myMs.noSNPs, analysis.type="differential",
    design=design, coef=39)
dmrcoutput <- dmrcate(myannotation, lambda=1000)
makeBedgraphs(dmrcoutput=dmrcoutput, betas=myBetas, samps=c(1,39))

## End(Not run)
```

---

rmSNPandCH *Filter probes*

Description

Filters a matrix of M-values (or beta values) by distance to SNP. Also (optionally) removes cross-hybridising probes and sex-chromosome probes.

Usage

```
rmSNPandCH(object, dist = 2, mafcut = 0.05, and = TRUE, rmcrosshyb = TRUE, rmXY=FALSE)
```

Arguments

object	A matrix of M-values or beta values, with unique Illumina probe IDs as row-names.
dist	Maximum distance (from CpG to SNP) of probes to be filtered out. See details for when Illumina occasionally lists a CpG-to-SNP distance as being < 0.
mafcut	Minimum minor allele frequency of probes to be filtered out.
and	If TRUE, the probe must have at least 1 SNP binding to it that satisfies **both** requirements in dist and mafcut for it to be filtered out. If FALSE, it will be filtered out if either requirement is satisfied. Default is TRUE.
rmcrosshyb	If TRUE, filters out probes found by Chen et al. (2013) to be cross-reactive with areas of the genome not at the site of interest. Many of these sites are on the X-chromosome, leading to potential confounding if the sample group is a mix of males and females. There are 30,969 probes in total in this list. Default is TRUE.
rmXY	If TRUE, filters out probe hybridising to sex chromosomes. Or-operator applies when combined with other 2 filters.

Details

Probes in -1:dist will be filtered out for any integer specification of dist. When a probe is listed as being "-1" nucleotides from a SNP (7 in total of the 153,113), that SNP is immediately adjacent to the end of the probe, and is likely to confound the measurement, in addition to those listed as 0, 1 or 2 nucleotides away. See vignette for further details.

Value

A matrix, attenuated from object, with rows corresponding to probes matching user input filtered out.

Author(s)

Tim J. Peters <Tim.Peters@csiro.au>

References

Chen YA, Lemire M, Choufani S, Butcher DT, Grafodatskaya D, Zanke BW, Gallinger S, Hudson TJ, Weksberg R. Discovery of cross-reactive probes and polymorphic CpGs in the Illumina Infinium HumanMethylation450 microarray. *Epigenetics*. 2013 Jan 11;8(2).

http://supportres.illumina.com/documents/myillumina/88bab663-307c-444a-848e-0ed6c338ee4d/humanmethylation450_15017482_v.1.2.snpupdate.table.v3.txt

Examples

```
## Not run:
data(dmrcatedata)
myMs <- logit2(myBetas)
myMs.noSNPs <- rmSNPandCH(myMs, dist=2, mafcut=0.05)

## End(Not run)
```

Index