AlphaBeta

Y.Shahryary, Rashmi Hazarika, Frank Johannes

2023-10-26

Contents

1 Introduction

AlphaBeta is a computational method for estimating epimutation rates and spectra from high-throughput DNA methylation data in plants. The method can be generally applied to study 'germline' epimuations in mutation accumulation lines (MA-lines), as well as 'somatic' epimutations in long-lived perennials, such as trees. Details regarding the inference approach and example applications can be found in Shahryary et al. 2020.

1.1 Experimental systems

A key challenge in studying epimutational proceses in multi-generational experiments is to be able to distinguish 'germline' epimutations from other types of methylation changes, such as those that are associated with segregating genetic variation or transient environmental perturbations. Mutation accumulation lines (MA-lines) grown in controlled laboratory conditions are a powerful experimental system to achieve this. MA-lines are derived from a single isogenic founder and are independently propagated for a large number of generations. The lines can be advanced either clonally or sexually, i.e. self-fertilization or sibling mating (Fig. 1A). In clonally produced MA lines the isogenicity of the founder is not required because the genome is 'fixed' due to the lack of genetic segregation.

The kinship among the different MA lineages can be presented as a pedigree (Fig. 1A). The structure (or topology) of these pedigrees is typically known, a priori, as the branch-point times and the branch lengths are deliberately chosen as part of the experimental design. In conjunction with multi-generational methylome measurements MA lines therefore permit 'real-time' observations of 'germline' epimutations against a nearly invariant genomic background, and can facilitate estimates of the per generation epimutation rates.

Beyond experimentally-derived MA lines, natural mutation accumulation systems can also be found in the context of plant development and aging. An instructive example are long-lived perennials, such as trees, whose branching structure can be interpreted as a pedigree (or phylogeny) of somatic lineages that carry information about the epimutational history of each branch. In this case, the branch-point times and the branch lengths can be determined ad hoc using coring data, or other types of dating methods (Fig. 1B). By combining this information with contemporary leaf methylome measurements it is possible to infer the rate of somatic epimutations as a function of age (see Hofmeister et al.).

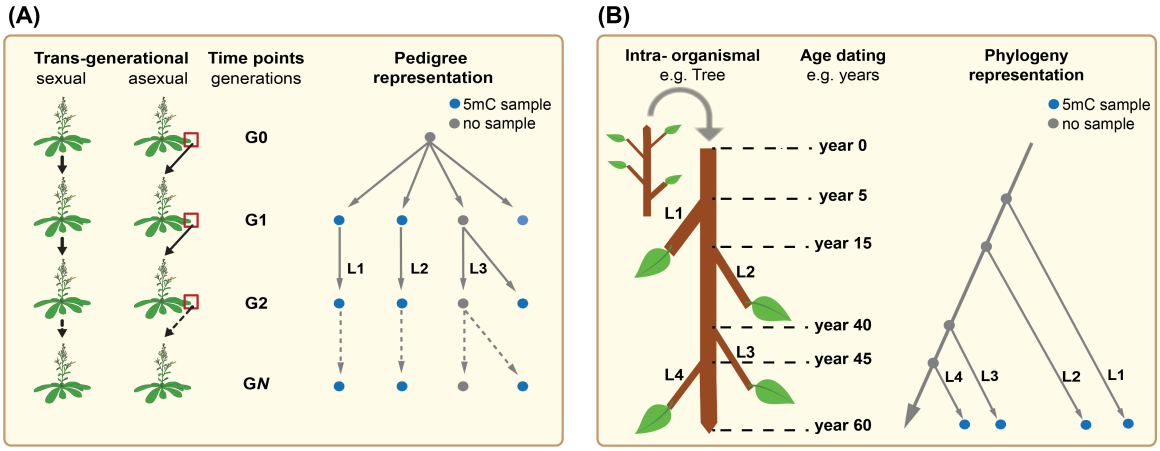


Figure 1: Experimental systems

1.2 DNA methylation sampling strategies

In the analysis of selfing or clonally-derived MA-lines, DNA methylation samples are typically obtained from the final generation (Fig. 2, endpoint sampling) and/or from intermediate generations (Fig. 2, intermediate sampling). With intermediate sampling, the samples can be obtained either directly from the progenitors of each generation (progenitor design, Fig. 2) or else from siblings of those progenitors (sibling design, Fig. 2). The sibling design is feasible in plants as seeds from early generations can be stored and grown out later. This is advantageous because plant material from all generations can be sampled simultaneously under identical conditions. An advantage of the progenitor design is that the methylation status of the pedigree founder is known, and that the methylation status of individual cytosines can be traced back in time through the entire pedigree. However, progenitor samples are taken in real-time (i.e. at selected generations) and thus growth conditions may vary over the experiment and introduce unwanted sources of noise. In the analysis of trees, DNA methylation samples come from contemporary leafs, as the methylomes of earlier developmental time-points are inaccessible (at least not easily accessible). Endpoint sampling is therefore the most obvious sampling strategy.

Irregardless of sampling design, AlphaBeta interprets the underlying pedigree as a sparse directed network. The nodes of the network correspond to 'individuals' whose methylomes have been sampled (i.e. type S* nodes), or of the common ancestors of these individuals, whose methylomes have typically not been sampled (i.e. type S nodes) (Fig.2).

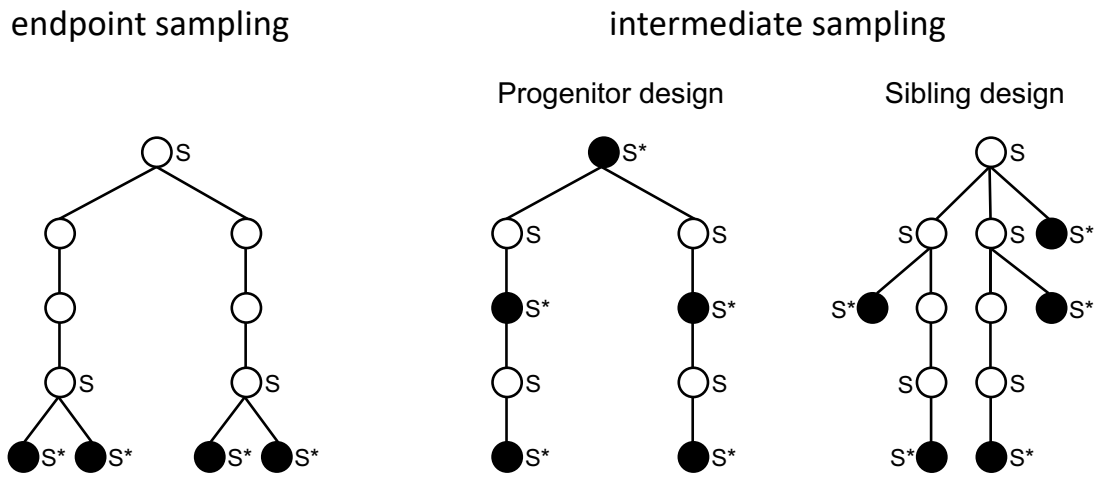


Figure 2: Pedigrees and DNA methylation sampling strategies. S* denotes sampled individuals and S are their (typically unsampled) most recent ancestors.

2 Input files

2.1 Pedigree files of MA lineages

To be able to re-construct the topology of the underlying pedigree, AlphaBeta requires two types of input files: "nodeslist.fn" and "edgelist.fn". The structure of these files follows the standard file format required by the R network package igraph.

```
# Load 'nodelist.fn' file
sampleFile <- system.file("extdata/vg", "nodelist.fn", package = "AlphaBeta")
```

"nodelist.fn" has the following structure

```
                                    filename     node gen meth
 1:                                        -      0_0   0    N
 2:                                        -  2_26_r1   2    N
 3:                                        -  2_87_r1   2    N
 4: /data/methylome_GSE64463_MA1_1_G3_26_r1.txt  3_26_r1   3    Y
 5: /data/methylome_GSE64463_MA1_1_G3_87_r1.txt  3_87_r1   3    Y
 6: /data/methylome_GSE64463_MA1_1_G3_87_r2.txt  3_87_r2   3    Y
 7:                                        - 30_29_r1  30    N
 8:                                        - 30_39_r1  30    N
 9:                                        - 30_49_r1  30    N
10:                                        - 30_59_r1  30    N
```

filename: Lists the filenames corresponding to nodes (S and S*). Filenames of type S* nodes should be identical with the names of their corresponding methylome files (see below). Type S nodes lacking corresponding methylome measurements, and should be designated by (-).

node: An arbitrary but unique name given to each node.

gen: Specifies the generation time of the nodes in the underlying pedigree.

meth: Indicates if methylome measurements are available for a given node (Y = yes, N = no).

The "edgelist.fn" file specifies the ancestral (i.e. lineages) relationship between nodes.

```
# Load 'edgelist.fn' file
edgesFile <- system.file("extdata/vg", "edgelist.fn", package = "AlphaBeta")
```

"edgelist.fn" has the following structure

```
        from       to gendiff group
  1:     0_0  2_26_r1       1     A
  2:     0_0  2_87_r1       1     A
  3: 2_26_r1  3_26_r1       1     A
  4: 2_87_r1  3_87_r1       1     A
  5: 2_87_r1  3_87_r2       1     A
  6:     0_0 30_29_r1      30     B
  7:     0_0 30_39_r1      30     B
  8:     0_0 30_49_r1      30     B
  9:     0_0 30_59_r1      30     B
 10:     0_0 30_79_r1      30     B
```

from and to: Specifies the network edges, which are any direct connections between type S and S* nodes in the pedigree. Only unique pairs of nodes need to be supplied (Fig.1). These 2 columns are mandatory.

gendiff (optional): Specifies the number of generations that separate the two nodes. This column is useful only for plotting purpose and it can be omitted for epimutation rate estimation. However, we recommend that this column be supplied because it is useful for accurately scaling the edge lengths when plotting certain pedigrees with progenitor.endpoint and sibling design (see 4.1).

group (optional): Along with "gendiff" column, groupings supplied in this column will help in scaling the edge lengths when plotting the pedigree.

2.2 Pedigree files of Trees

```
# Load 'SOMA_nodelist.fn' file
treeSamples <- system.file("extdata/soma", "SOMA_nodelist2.fn", package = "AlphaBeta")
```

"SOMA_nodelist.fn" has the following structure

```
        filename node Branchpoint_date meth
  1:           -  330                0    N
  2:           -  113              217    N
  3: data/myfile 13_5              297    Y
  4:           -   72              258    N
  5: data/myfile 13_3              328    Y
  6:           -   44              286    N
  7: data/myfile 13_2              327    Y
  8:           -   31              299    N
  9: data/myfile 13_1              328    Y
 10:           -  115              215    N
```

filename: Lists the filenames corresponding to nodes S. Type S nodes lacking corresponding methylome measurements, and should be designated by (-).

node: An arbitrary but unique name given to each node.

Branchpoint_date: Specifies the branchpoint time of the nodes in the underlying pedigree.

meth: Indicates if methylome measurements are available for a given node (Y = yes, N = no).

The "SOMA_edgelist.fn" file specifies the ancestral (i.e. lineages) relationship between nodes.

```
treeEdges <- system.file("extdata/soma", "SOMA_edgelist2.fn", package = "AlphaBeta")
```

"SOMA_edgelist.fn" has the following structure

```
      from   to Stem
 1:  330  113   13
 2:  113 13_5   13
 3:  113   72   13
 4:   72 13_3   13
 5:   72   44   13
 6:   44 13_2   13
 7:   44   31   13
 8:   31 13_1   13
 9:  330  115   14
10:  115 14_2   14
```

from and to: Specifies the network edges, which are any direct connections between type nodes in the pedigree. Only unique pairs of nodes need to be supplied. These 2 columns are mandatory.

stem (optional): To be provided only for trees with 2 or more stems (as.in our example). This column should be left blank for a tree with a single stem.

2.3 Methylome files

Type S* nodes in the pedigree have corresponding methylome data. In its current implementation, AlphaBeta expects methylome files that have been produced by *methimpute* Methimpute package. *methimpute* is a HMM-based methylation state caller for whole-genome bisulphite sequencing (WGBS) data. It can produce cytosine-level methylation state as well as region-level methylation methylation state calls. The former calls are required to obtain cytosine-level epimutation rates, while the latter calls are required to obtain region-level epimuation rates. Methylome files from alternative callers and/or measurement technologies are possible but should be converted to the *methimpute* file structure (see below).

2.3.1 Cytosine-level calls

"cytosine-level methylome files" have the following structure

seqnames	start	strand	context	counts.methylated		counts.total	
1	1	+	CHH	0	0	0.9293	I	0.035	CCC
1	2	+	CHH	0	0	0.8595	I	0.0493	CCT
1	3	+	CHH	0	0	0.8424	I	0.0528	CTA
1	8	+	CHH	0	1	0.8627	I	0.0486	CCC
1	9	+	CHH	0	1	0.8718	I	0.0466	CCT
1	10	+	CHH	0	1	0.8815	I	0.0446	CTA
1	15	+	CHH	0	2	0.9043	I	0.0398	CCC
1	16	+	CHH	0	2	0.9106	I	0.0384	CCT
1	17	+	CHH	0	2	0.9122	I	0.0381	CTA

seqnames, start and strand: Chromosome coordinates

context: Sequence context of cytosine i.e CG,CHG,CHH

counts.methylated: Counts for methylated reads at each position

counts.total: Counts for total reads at each position

posteriorMax: Posterior value of the methylation state call

status : Methylation status

rc.meth.lvl: Recalibrated methylation level calculated from the posteriors and fitted parameters

context.trinucleotide: Trinucleotide context of the cytosine context

2.3.2 Region-level calls

"region-level methylome files" have the following structure

seqnames	start	end	context	posteriorMax	status	rc.meth.lvl
1	3696	3856	CG	0.99999	U	0.01229
1	12100	12155	CG	0.99999	U	0.01229
1	20991	21026	CG	0.99999	U	0.01229
1	21257	21293	CG	0.99999	U	0.01229
1	29966	30008	CG	0.99999	M	0.85203
1	46099	46141	CG	0.99999	U	0.01229
1	46903	46941	CG	0.99999	U	0.01229
1	48988	49031	CG	0.99999	U	0.01229
1	50882	50922	CG	0.99999	U	0.01229

seqnames, start and strand: Chromosome coordinates

posteriorMax: Posterior value of the methylation state call

status : Methylation status

rc.meth.lvl: Recalibrated methylation level calculated from the posteriors and fitted parameters

context: Sequence context of cytosine i.e CG,CHG,CHH

2.3.3 Tips for converting files from alternative callers and/or technologies

Methylome files generated by alternative callers and/or measurement technologies should be converted to meet the *methimpute* file structure. We have the following tentative recommendations.

NGS-based technologies: For NGS-based technologies including whole-genome bisulphite sequencing (WGBS), reduced presentation bisulphite sequencing (RRBS) and epigenotyping by sequencing (epiGBS), columns seqnames, start, strand, context, counts.methylated, counts.toal and status are typically available as they are part of standard outputs of BS-seq aligment software, such as Bismark and BSseeker. In this case, we recommend removing all rows where counts.total = 0, and set posteriorMax = 1, rc.meth.lvl = counts.methylated/counts.total, context.trinucleotide = NA.

Array-based technologies: Although we have not directly tested this, it should also be possible to convert methylome data from array-based technologies, including MeDIP-chip, to the methylome file structure required by AlphaBeta. In this case, we recommend to set seqnames = probe chromosome position, start = probe start position, strand = arbitrarily fix to (+), context = arbitrarily fix to context "CG", counts.methylated = NA, count.total = NA, posteriorMax = 1, rc.meth.lvl = array methylation signal, context.trinucleotide = NA.

3 Building pedigree

The function "buildPedigree" builds the pedigree from the input files. Divergence time (delta.t) is calculated as follows: delta.t = t1 + t2 - 2*t0, where t1 is the time of sample 1 (in generations), t2 is the time of sample 2 (in generations) and t0 is the time (in generations) of the most recent common founder of samples 1 and 2.

3.1 Building MA-lines Pedigree

```
output <- buildPedigree(nodelist = sampleFile, edgelist = edgesFile, cytosine = "CG",
    posteriorMaxFilter = 0.99)
```

Divergence values (D.value):

```
head(output$Pdata)
```

```
      time0 time1 time2 D.value
 [1,]     0     3     3 0.00551
 [2,]     0     3     3 0.00585
 [3,]     0     3    31 0.01210
 [4,]     0     3    31 0.01266
 [5,]     0     3    31 0.01206
 [6,]     0     3    31 0.01235
```

time 1: Generation time of sample i

time 2: Generation time of sample j

time 0: Generation time of most recent common ancestor of samples i and j

D.value: Mean absolute divergence in DNA methylation states between samples i and j

3.2 Building Tree Pedigree

```
outputTree <- buildPedigree(nodelist = treeSamples, edgelist = treeEdges, cytosine = "CG",
    posteriorMaxFilter = 0.99)
```

Divergence values (D.value):

```
head(outputTree$Pdata)
```

```
      time0 time1 time2     D.value
 [1,]     0   297   287 0.003796614
 [2,]     0   297   324 0.003974756
 [3,]     0   327   287 0.003995156
 [4,]     0   297   287 0.004040671
 [5,]     0   328   287 0.004046553
 [6,]     0   328   287 0.004048672
```

4 Diagnostic plots

The correct specification of the pedigree topology and the removal of influential outlier data points are critical aspects for epimutation rate estimation. Visual inspection of the pedigree is provided through the function "plotPedigree".

4.1 Plotting pedigrees

4.1.1 Pedigree of MA-lines

```
## Progenitor-endpoint design
plotPedigree(nodelist = sampleFile, edgelist = edgesFile, sampling.design = "progenitor.endpoint",
    output.dir = out.dir, plot.width = 5, plot.height = 5, aspect.ratio = 1, vertex.size = 6,
    vertex.label = FALSE, out.pdf = "MA1_1")

## Sibling design
plotPedigree(nodelist = system.file("extdata/vg", "nodelist_MA2_3.fn", package = "AlphaBeta"),
    edgelist = system.file("extdata/vg", "edgelist_MA2_3.fn", package = "AlphaBeta"),
    sampling.design = "sibling", output.dir = out.dir, plot.width = 5, plot.height = 5,
    aspect.ratio = 2.5, vertex.size = 12, vertex.label = FALSE, out.pdf = "MA2_3")

## Progenitor-intermediate design
plotPedigree(nodelist = system.file("extdata/vg", "nodelist_MA3.fn", package = "AlphaBeta"),
    edgelist = system.file("extdata/vg", "edgelist_MA3.fn", package = "AlphaBeta"),
    sampling.design = "progenitor.intermediate", output.dir = out.dir, plot.width = 5,
    plot.height = 8, aspect.ratio = 2.5, vertex.size = 13, vertex.label = FALSE,
    out.pdf = "MA3")
```

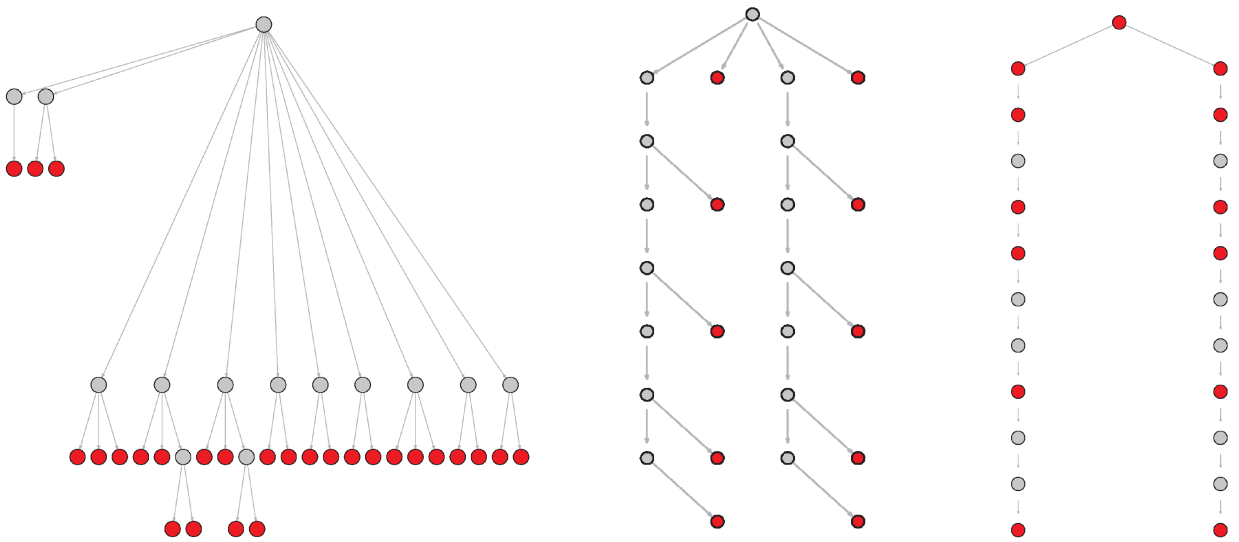


Figure 3: Pedigrees of MA lines with progenitor.endpoint (left), sibling (middle) and progenitor.intermediate design (right)

4.1.2 Tree pedigrees

```
plotPedigree(nodelist = treeSamples, edgelist = treeEdges, sampling.design = "tree",
    output.dir = out.dir, plot.width = 5, plot.height = 5, aspect.ratio = 1, vertex.size = 8,
    vertex.label = FALSE, out.pdf = "Tree")
```

nodelist: file containing list of nodes

edgelist: files containing list of edges

sampling.design: set sampling design according to pedigree

output.dir: set output directory path

plot.width, plot.height: set width and height of the output pdf

aspect.ratio, vertex.size: for adjusting the ratio of height to width of the pedigree plot

vertex.label: vertex labels can be printed with either TRUE or FALSE

out.pdf: set NULL to print plot on screen or set name to output as pdf

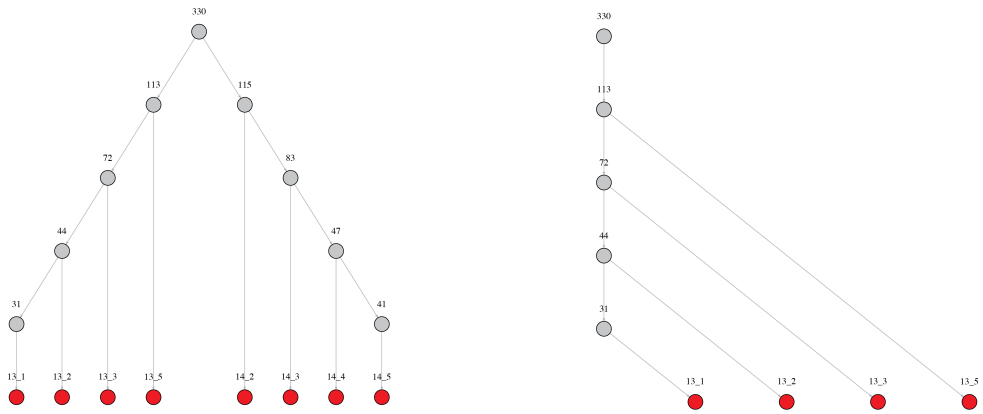
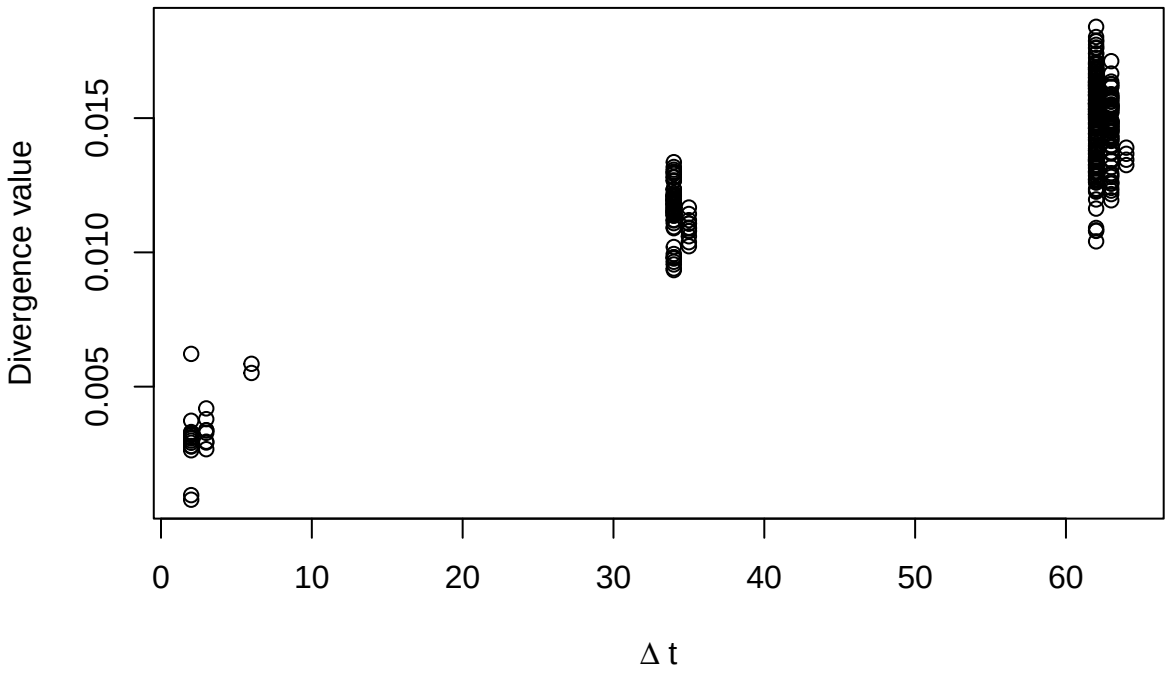


Figure 4: Pedigree of a tree with 2 stems (left) and a single stem (right)

4.2 Plotting divergence time (delta.t) versus methylome divergence (D.value)

This is an interactive plot for inspecting the divergence data and removing outlier samples (if any):

```
pedigree <- output$Pdata
dt <- pedigree[, 2] + pedigree[, 3] - 2 * pedigree[, 1]
plot(dt, pedigree[, "D.value"], ylab = "Divergence value", xlab = expression(paste(Delta,
    " t")))
```



5 Epimutation rate estimation in selfing-systems

Models ABneutral, ABselectMM, ABselectUU and ABnull can be used to estimate the rate of spontaneous epimutations in selfing-derived MA-lines. The models are currently restricted to diploids.

5.1 Run Models

5.1.1 Run Model with no selection (ABneutral)

ABneutral fits a neutral epimutation model. The model assumes that epimutation accumulation is under no selective constraint. Returned are estimates of the methylation gain and loss rates and the proportion of epi-heterozygote loci in the pedigree founder genome.

Initial proportions of unmethylated cytosines:

```
p0uu_in <- output$tmpp0
p0uu_in
```

```
  [1] 0.7435119
```

```
pedigree <- output$Pdata
```

```
# output directory
output.data.dir <- paste0(getwd())

output <- ABneutral(pedigree.data = pedigree, p0uu = p0uu_in, eqp = p0uu_in, eqp.weight = 1,
    Nstarts = 2, out.dir = output.data.dir, out.name = "ABneutral_CG_global_estimates")
```

```
  Progress: 0.5
  Progress: 1
```

NOTE: it is recommended to use at least 50 Nstarts to achieve best solutions

Showing summary output of only output:

```
summary(output)
```

```
                   Length Class      Mode
  estimates           20   data.frame list
  estimates.flagged   20   data.frame list
  pedigree          2457   -none-     numeric
  settings             2   data.frame list
  model                1   -none-     character
  for.fit.plot       315   -none-     numeric
```

```
head(output$pedigree)
```

```
      time0 time1 time2 div.obs delta.t    div.pred    residual
  [1,]     0     3     3 0.00551       6 0.004477275 0.001032725
  [2,]     0     3     3 0.00585       6 0.004477275 0.001372725
  [3,]     0     3    31 0.01210      34 0.009834058 0.002265942
  [4,]     0     3    31 0.01266      34 0.009834058 0.002825942
  [5,]     0     3    31 0.01206      34 0.009834058 0.002225942
  [6,]     0     3    31 0.01235      34 0.009834058 0.002515942
```

Plot estimates of ABneutral model:

```
ABfile <- system.file("extdata/models/", "ABneutral_CG_global_estimates.Rdata", package = "AlphaBeta")
# In 'ABplot' function you can set parameters to customize the pdf output.
ABplot(pedigree.names = ABfile, output.dir = getwd(), out.name = "ABneutral", plot.height = 8,
    plot.width = 11)
```

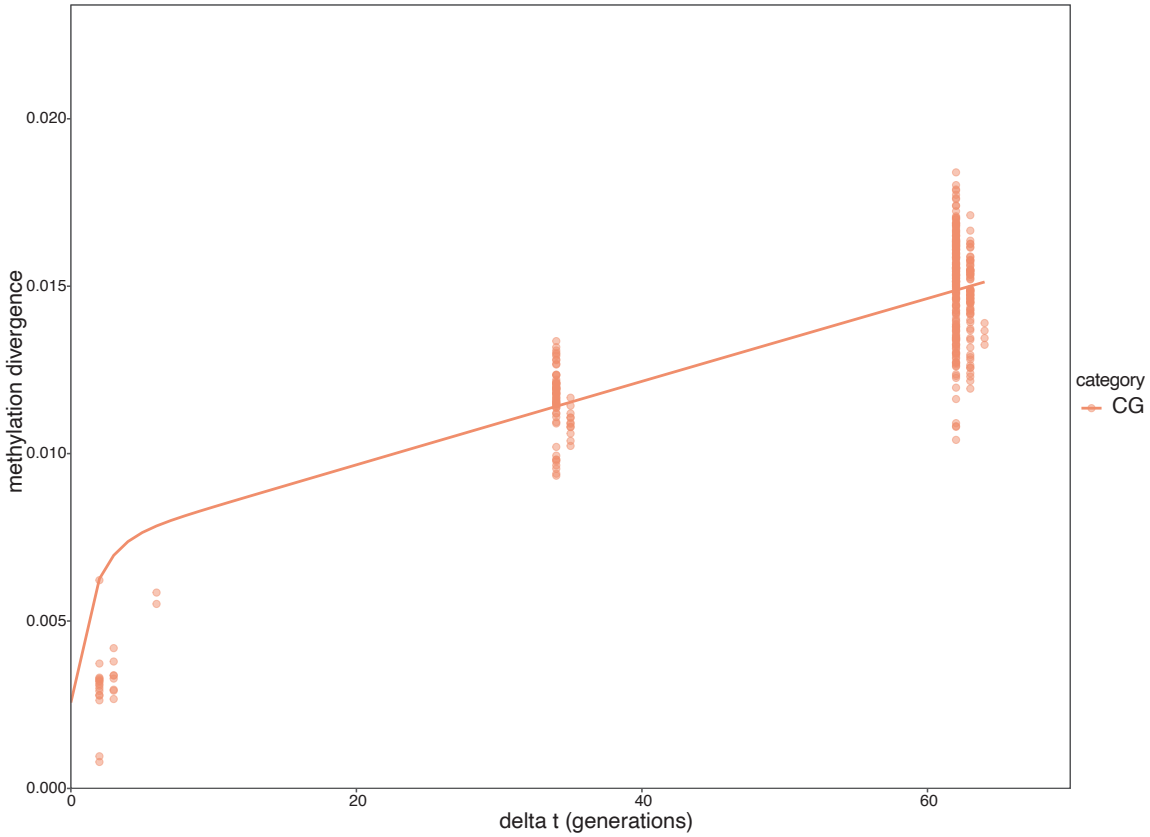


Figure 5: Divergence versus delta.t

5.1.2 Run model with selection against spontaneous gain of methylation (ABselectMM)

ABselectMM fits an epimutation model with selection. The model assumes that epimutation accumulation is in part shaped by selection against spontaenous losses of cytosine methylation. Returned are estimates of the methylation gain and loss rates, the selection coeficient, and the proportion of epi-heterozygote loci in the pedigree founder genome.

```
outputABselectMM <- ABselectMM(pedigree.data = pedigree, p0uu = p0uu_in, eqp = p0uu_in,
    eqp.weight = 1, Nstarts = 2, out.dir = output.data.dir, out.name = "ABselectMM_CG_global_estimates")
```

```
  Progress: 0.5
```

```
  Progress: 1
```

5.1.3 Run model with selection against spontaneous loss of methylation (ABselectUU)

ABselectUU fits an epimutation model with selection. The model assumes that epimutation accumulation is in part shaped by selection against spontaenous gains of cytosine methylation. Returned are estimates of the methylation gain and loss rates, the selection coeficient, and the proportion of epi-heterozygote loci in the pedigree founder genome.

```
outputABselectUU <- ABselectUU(pedigree.data = pedigree, p0uu = p0uu_in, eqp = p0uu_in,
    eqp.weight = 1, Nstarts = 2, out.dir = output.data.dir, out.name = "ABselectUU_CG_global_estimates")
```

```
  Progress: 0.5
```

```
  Progress: 1
```

5.1.4 Run model that considers no accumulation of epimutations (ABnull)

ABnull fits a model of no epimutation accumulation. This model serves as the Null model.

```
outputABnull <- ABnull(pedigree.data = pedigree, out.dir = output.data.dir, out.name = "ABnull_CG_global_estim
```

5.2 Comparison of different models and selection of best model

5.2.1 Testing ABneutral vs. ABnull

```
file1 <- system.file("extdata/models/", "ABneutral_CG_global_estimates.Rdata", package = "AlphaBeta")
file2 <- system.file("extdata/models/", "ABnull_CG_global_estimates.Rdata", package = "AlphaBeta")

out <- FtestRSS(pedigree.select = file1, pedigree.null = file2)

out$Ftest
```

```
         RSS_F         RSS_R          df_F          df_R        Fvalue        pvalue
  6.508955e-04  4.125201e-03  3.460000e+02  3.500000e+02  4.617138e+02 2.703258e-137
```

5.2.2 Testing ABselectMM vs.ABneutral

```
file1 <- system.file("extdata/models/", "ABselectMM_CG_global_estimates.Rdata", package = "AlphaBeta")
file2 <- system.file("extdata/models/", "ABnull_CG_global_estimates.Rdata", package = "AlphaBeta")

out <- FtestRSS(pedigree.select = file1, pedigree.null = file2)

out$Ftest
```

```
         RSS_F         RSS_R          df_F          df_R        Fvalue        pvalue
  6.507056e-04  4.125201e-03  3.460000e+02  3.500000e+02  4.618738e+02 2.570279e-137
```

5.2.3 Testing ABselectUU vs.ABneutral

```
file1 <- system.file("extdata/models/", "ABselectUU_CG_global_estimates.Rdata", package = "AlphaBeta")
file2 <- system.file("extdata/models/", "ABnull_CG_global_estimates.Rdata", package = "AlphaBeta")

out <- FtestRSS(pedigree.select = file1, pedigree.null = file2)

out$Ftest
```

```
         RSS_F         RSS_R          df_F          df_R        Fvalue        pvalue
  6.499228e-04  4.125201e-03  3.460000e+02  3.500000e+02  4.625343e+02 2.087582e-137
```

5.3 Bootstrap analysis with the best fitting model(BOOTmodel)

i.e ABneutral in our case

NOTE: it is recommended to use at least 50 Nboot to achieve best solutions

```
inputModel <- system.file("extdata/models/", "ABneutral_CG_global_estimates.Rdata",
    package = "AlphaBeta")

# Bootstrapping models CG

Boutput <- BOOTmodel(pedigree.data = inputModel, Nboot = 2, out.dir = getwd(), out.name = "ABneutral_Boot_CG_g
```

```
  Bootstrap interation: 0.5
  Bootstrap interation: 1
```

```
summary(Boutput)
```

```
                 Length Class      Mode
  standard.errors 24     -none-     numeric
  boot.base       20     data.frame list
  settings         2     data.frame list
  N.boots          1     -none-     numeric
  N.good.boots     1     -none-     numeric
  boot.results    19     data.frame list
```

```
  model            1     -none-     character

                     SE         2.5%        97.5%
  alpha      9.489868e-07 9.692187e-05 9.819684e-05
  beta       2.759288e-06 2.813835e-04 2.850906e-04
  beta/alpha 4.267382e-05 2.903199e+00 2.903256e+00
  weight     2.975373e-04 2.910079e-02 2.950054e-02
  intercept  1.212447e-04 2.378426e-03 2.541319e-03
  PrMMinf    5.620157e-06 2.559044e-01 2.559120e-01
  PrUMinf    5.638306e-06 5.761789e-04 5.837540e-04
  PrUUinf    1.814899e-08 7.435118e-01 7.435118e-01
```

6 Epimutation rate estimation in clonal, asexual and somatic systems

Models ABneutralSOMA, ABselectMMSOMA and ABselectUUSOMA can be used to estimate the rate of spontaneous epimutations from pedigree-based high-throughput DNA methylation data. The models are generally designed for pedigree data arising from clonally or asexually propagated diploid species. The models can also be applied to long-lived perrenials, such as trees, using leaf methylomes and coring data as input. In this case, the tree branching structure is treated as an intra-organismal pedigree (or phylogeny) of somatic lineages.

6.1 Run Models

6.1.1 Run Model with no selection (ABneutralSOMA)

This model assumes that somatically heritable gains and losses in cytosine methylation are selectively neutral.

Initial proportions of unmethylated cytosines:

```
Tree_p0uu_in <- outputTree$tmpp0
Tree_p0uu_in
```

```
  [1] 0.45245
```

```
pedigree.Tree <- outputTree$Pdata
```

```
outputABneutralSOMA <- ABneutralSOMA(pedigree.data = pedigree.Tree, p0uu = Tree_p0uu_in,
    eqp = Tree_p0uu_in, eqp.weight = 0.001, Nstarts = 2, out.dir = getwd(), out.name = "ABneutralSOMA_CG_globa
```

```
  Progress: 0.5
  Progress: 1
```

```
summary(outputABneutralSOMA)
```

```
                  Length Class      Mode
  estimates           20   data.frame list
  estimates.flagged   20   data.frame list
  pedigree           196   -none-     numeric
  settings             2   data.frame list
  model                1   -none-     character
  for.fit.plot      3275   -none-     numeric
```

```
head(outputABneutralSOMA$pedigree)
```

```
      time0 time1 time2     div.obs delta.t    div.pred      residual
  [1,]     0   297   287 0.003796614     584 0.004002421 -2.058072e-04
  [2,]     0   297   324 0.003974756     621 0.004115551 -1.407947e-04
  [3,]     0   327   287 0.003995156     614 0.004094157 -9.900055e-05
  [4,]     0   297   287 0.004040671     584 0.004002421  3.824981e-05
  [5,]     0   328   287 0.004046553     615 0.004097214 -5.066110e-05
  [6,]     0   328   287 0.004048672     615 0.004097214 -4.854210e-05
```

Plot estimates of ABneutralSOMA model:

```
ABfilesoma <- system.file("extdata/models/", "ABneutralSOMA_CG_global_estimates.Rdata",
    package = "AlphaBeta")
```

```
ABplot(pedigree.names = ABfilesoma, output.dir = getwd(), out.name = "ABneutralSOMA",
    plot.height = 8, plot.width = 11)
```

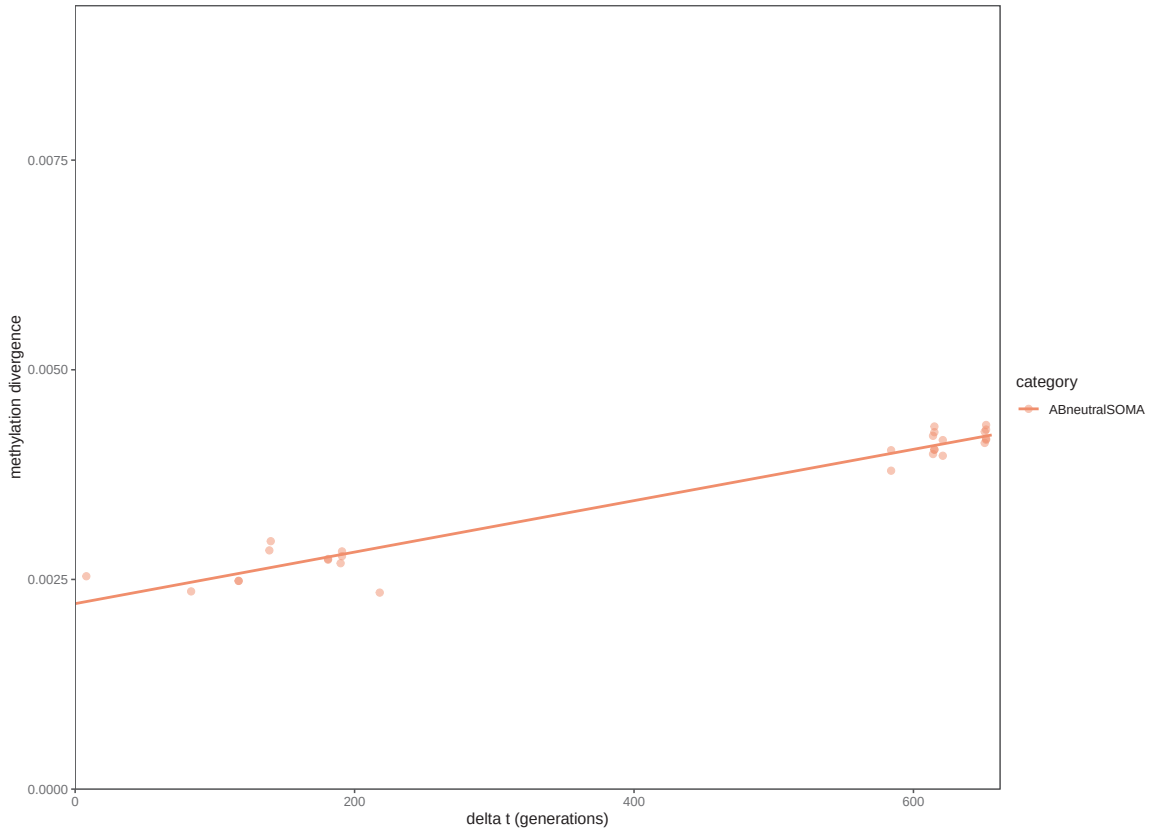


Figure 6: Divergence versus delta.t of Tree

6.1.2 Run model with selection against spontaneous gain of methylation (ABselectMMSOMA)

This model assumes that somatically heritable losses of cytosine methylation are under negative selection. The selection parameter is estimated.

```
outputABselectMMSOMA <- ABselectMMSOMA(pedigree.data = pedigree.Tree, p0uu = Tree_p0uu_in,
    eqp = Tree_p0uu_in, eqp.weight = 0.001, Nstarts = 2, out.dir = getwd(), out.name = "ABselectMMSOMA_CG_glob
```

```
  Progress: 0.5
```

```
  Progress: 1
```

6.1.3 Run model with selection against spontaneous loss of methylation (ABselectUUSOMA)

This model assumes that somatically heritable gains of cytosine methylation are under negative selection. The selection parameter is estimated.

```
outputABselectUUSOMA <- ABselectUUSOMA(pedigree.data = pedigree.Tree, p0uu = Tree_p0uu_in,
    eqp = Tree_p0uu_in, eqp.weight = 0.001, Nstarts = 2, out.dir = getwd(), out.name = "ABselectUUSOMA_CG_glob
```

```
  Progress: 0.5
```

```
  Progress: 1
```

6.2 Bootstrap analysis with the best fitting model (BOOTmodel)

```
inputModelSOMA <- system.file("extdata/models", "ABneutralSOMA_CG_global_estimates.Rdata",
    package = "AlphaBeta")
```

```r
# Bootstrapping models CG

Boutput <- BOOTmodel(pedigree.data = inputModelSOMA, Nboot = 2, out.dir = getwd(),
    out.name = "ABneutral_Boot_CG_global_estimates")
```

```
  Bootstrap interation: 0.5

  Bootstrap interation: 1
```

```r
summary(Boutput)
```

```
                 Length Class      Mode
  standard.errors 24     -none-     numeric
  boot.base       20     data.frame list
  settings         2     data.frame list
  N.boots          1     -none-     numeric
  N.good.boots     1     -none-     numeric
  boot.results    19     data.frame list
  model            1     -none-     character

                      SE         2.5%        97.5%
  alpha      9.278903e-08 1.953307e-06 2.077970e-06
  beta       1.906601e-07 4.013614e-06 4.269767e-06
  beta/alpha 3.946063e-07 2.054778e+00 2.054779e+00
  weight     4.185952e-06 4.828661e-02 4.829223e-02
  intercept  8.625476e-05 2.131499e-03 2.247382e-03
  PrMMinf    2.768566e-08 1.071619e-01 1.071619e-01
  PrUMinf    2.920224e-08 4.403881e-01 4.403881e-01
  PrUUinf    5.688789e-08 4.524500e-01 4.524500e-01
```

7 R session info

```r
sessionInfo()
```

```
  R version 4.3.1 (2023-06-16)
  Platform: aarch64-apple-darwin20 (64-bit)
  Running under: macOS Ventura 13.6.1

  Matrix products: default
  BLAS:   /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRblas.0.dylib
  LAPACK: /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRlapack.dylib;  LAPACK version 3

  locale:
  [1] C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

  time zone: America/New_York
  tzcode source: internal

  attached base packages:
  [1] stats     graphics  grDevices utils     datasets  methods   base

  other attached packages:
  [1] ggplot2_3.4.2     igraph_1.5.0      data.table_1.14.8 AlphaBeta_1.16.0

  loaded via a namespace (and not attached):
   [1] plotly_4.10.2       utf8_1.2.3          generics_0.1.3      tidyr_1.3.0         gtools_3.9.4
   [6] stringi_1.7.12      lattice_0.21-8      digest_0.6.33       magrittr_2.0.3      evaluate_0.21
  [11] grid_4.3.1          fastmap_1.1.1       jsonlite_1.8.7      Matrix_1.6-0        formatR_1.14
  [16] httr_1.4.6          purrr_1.0.1         fansi_1.0.4         viridisLite_0.4.2   scales_1.2.1
  [21] codetools_0.2-19    numDeriv_2016.8-1.1 lazyeval_0.2.2      cli_3.6.1           rlang_1.1.1
  [26] expm_0.999-7        munsell_0.5.0       withr_2.5.0         yaml_2.3.7          tools_4.3.1
```

```
[31] parallel_4.3.1      BiocParallel_1.36.0 dplyr_1.1.2         colorspace_2.1-0    optimx_2022-4.30
[36] png_0.1-8           vctrs_0.6.3         R6_2.5.1            lifecycle_1.0.3     stringr_1.5.0
[41] htmlwidgets_1.6.2   pkgconfig_2.0.3     pillar_1.9.0        gtable_0.3.3        glue_1.6.2
[46] highr_0.10          xfun_0.39           tibble_3.2.1        tidyselect_1.2.0    knitr_1.43
[51] htmltools_0.5.5     rmarkdown_2.23      compiler_4.3.1
```