metadamage(?) formats

tsk

November 22, 2023

This document describe some of the internal formats used by the metadamage software. These are at the current time.

- .bdamage.gz
- .lca
- .stat
- .dfit.txt.gz
- .agggregate.stat.txt.gz
- .rlens.gz

1 bdamage format

bdamage files are files that contain counts of mismatchs conditional on strand and cycle (position within read). These are generated with metadamage lca or metadamage getdamage. The first 8 bytes magic number determines which bdamage version. If no magic number is present then version0 is assumed.

1.1 version 0

First version of the bdamage file is a single bgzf compressed file. MAXLENGTH occurs once in the beginning of the file, followed by succesive blocks of data[1-8]. Block[3-5] indicates the actual mismatch counts for the forward which we have MAXLENGTH times. Block[6-8] indicates the actual mismatch counts for the reverse strand which will also occur MAXLENGTH times.

Col	Field	Type	Brief description
0	MAXLENGTH	int	Number of cycles
1	ID	int	Id for mismatch type[1]
2	NREADS	size_t	Number of reads used supporting the mismatch matrix
3	1	int[16]	mismatch rate for first cycle from the 5prime
4	i	int[16]	mismatch rate for the i'th cycle from the 5prime
5	MAXLENGTH	int[16]	mismatch rate for the last cycle from the 5prime
6	1	int[16]	mismatch rate for first cycle from the 3prime
7	i	int[16]	mismatch rate for the i'th cycle from the 3prime
8	MAXLENGTH	int[16]	mismatch rate for the last cycle from the 3prime

Table 1: Content of bdamage.gz file. Note[1] This is either the taxidID or the referenceID relative to the SAM/BAM header for single species resequencing projects or it is the *taxid* if output has been generated with metadamage lca 3) Order is given by AA,AC,AG,AT,CA,CC,CG,CT,GA,GC,GG,GT,TA,TC,TG,TT, with first base indicatting reference nucleotide and second base indicating observed nucleotide

.

2 .lca

This section describes the test output generated by a metadamage lca subfunctionality and contains information at the readlevel regarding both taxonomic information and statistics pertaining usefull readinformation.

First line of the file begins with a hashtag followed by the actual command used for generating the file. Last entry of the line is again a hashtag followed by the git commit id which will serve as a primitive versioncontrol.

Each line consists of a number of items seperated by tabspace. First entry contains readID together with other information seperated by colon. After the first entry succesive blocks of the type taxid:name:"taxlevel" from the lca toward the root. The complete specification is seen in table below.

Col	Brief description	
1	readID	readID, this might contains colon
2	seq	The actual sequence
3	length(seq)	The length of the sequence
4	nAlignments	The number of alignments used for inferring the lca
5	gc-content	The GC content for the sequence
6	lca taxid	the taxomic id (integer)
7	lca taxid	the taxonomic name(string)
8	lca "taxlevel"	the taxonomic level
9	taxid	the taxomic id (integer)
10	taxid	the taxonomic name(string)
11	"taxlevel"	the taxonomic level

Table 2: Content of a .lca file. Note that 1) seperate between fields[1-6,8-9] is tab, readID might also contain colon. 2) the quotes around field[7,8] is intentional since taxlevels and names might contain spaces andor colons. 3) Number of tab seperated entries is consistent across different reads, since seperator between block 9-11 is semicolon. Seperator between 6-9,9-11 is colon

3 .stat

Very simple tabseperated flatfile

1. taxid
2. Number of supporting reads
3. Mean lengths of supporting reads
4. Variance of the lengths of the supporting reads
5. Mean gccontent of supporting reads
6. Variance of the gccontent of supporting reads
7. name of lca in quotes (if relevant, otherwise NA)
8. name of taxomic level of lca in quotes (if relevant, otherwise NA)

4 .dfit.txt.gz

The output format will be heavily dependent on the provided runmode and supplied parameters. It might contain the per file, the per reference or the per species estimate of damage.
Only the columns described in *showfits 0* remains the same across the runmode, with the *showfits 1* or *showfits 2* describes the unique columns and order which is appended to the *showfits 0* columns.
Non-boostrapping optimization:

Column name	Brief description
	–showfits 0
id	identifier see paragraph for details
A	Dfit statistic. Damage at position one, taking into account offset
q	per cycle decrease
c	background substitution rate or noise baseline
ϕ	Variance between beta-binomial and binomial model
llh	Likelihood for our MLE
ncall	Number of optimization calls used for obtaining our MLE
σ_D	Z value
Zfit	significance
	–showfits 1, *i* signifies position inferred from *.bdamage.gz*
$fwdx i$	damage estimates forward strand
$fwdConf i$	forward strand confidence interval ±
$bwdx i$	damage estimates backward strand
$bwdConf i$	backward strand confidence interval ±
	–showfits 2, *i* signifies position inferred from *.bdamage.gz*
$fwK i$	Number of deamination substitution (C→T) observations forward strand
$fwN i$	Total number of C for forward strand
$fwdx i$	Damage estimates forward strand
$fwf i$	Calculated damage frequency $fwK i / fwN i$
$fwdConf i$	forward strand confidence interval ±
$bwK i$	Number of deamination substitution (G→A for ds, C→T for ss) observations forward strand
$bwN i$	Total number of G for forward strand (ds) and C (ss)
$bwdx i$	Damage estimates backward strand
$bwf i$	Calculated damage frequency $fwK i / fwN i$
$bwdConf i$	backward strand confidence interval ±

Table 3: Content of a .dfit.txt.gz file. Note that entry nine to 13 is repeated for each cycle first of the 5ánd then from the 3ź With the total number of times repeated is given by 1.1.

Boostrapping optimization:
Providing *-nbootstrap* > 1 numerical optimizations of the binomial distribution will also be conducted using bootstrapping methods

Column name	Brief description
	–showfits 0
id	identifier see paragraph for details
A	Dfit statistic. Damage at position one, taking into account offset
q	per cycle decrease
c	background substitution rate or noise baseline
ϕ	Variance between beta-binomial and binomial model
llh	Likelihood for our MLE
ncall	Number of optimization calls used for obtaining our MLE
σ_D	Z value
Zfit	significance
A_b	Dfit statistic from bootstrap estimate. Damage at position one, taking into account offset
q_b	per cycle decrease from bootstrap estimate
c_b	background substitution rate or noise baseline from bootstrap estimate
ϕ_b	Variance between beta-binomial and binomial model from bootstrap estimate
A_CI_l	Lower bound of CI for A estimate calculated from all bootstrap values
A_CI_h	Upper bound of CI for A estimate calculated from all bootstrap values
q_CI_l	Lower bound of CI for q estimate calculated from all bootstrap values
q_CI_h	Upper bound of CI for q estimate calculated from all bootstrap values
c_CI_l	Lower bound of CI for c estimate calculated from all bootstrap values
c_CI_h	Upper bound of CI for c estimate calculated from all bootstrap values
ϕ_CI_l	Lower bound of CI for ϕ estimate calculated from all bootstrap values
ϕ_CI_h	Upper bound of CI for ϕ estimate calculated from all bootstrap values
	–showfits 1, *i* signifies position inferred from *.bdamage.gz*
$fwdx i$	damage estimates forward strand
$fwdConf i$	forward strand confidence interval ±
$bwdx i$	damage estimates backward strand
$bwdConf i$	backward strand confidence interval ±
	–showfits 2, *i* signifies position inferred from *.bdamage.gz*
$fwK i$	Number of deamination substitution (C→T) observations forward strand
$fwN i$	Total number of C for forward strand
$fwdx i$	Damage estimates forward strand
$fwf i$	Calculated damage frequency $fwK i/fwN i$
$fwdConf i$	forward strand confidence interval ±
$bwK i$	Number of deamination substitution (G→A for ds, C→T for ss) observations forward strand
$bwN i$	Total number of G for forward strand (ds) and C (ss)
$bwdx i$	Damage estimates backward strand
$bwf i$	Calculated damage frequency $fwK i/fwN i$
$bwdConf i$	backward strand confidence interval ±

Table 4: Content of a .dfit.txt.gz file. Note that entry nine to 13 is repeated for each cycle first of the 5ánd then from the 3ź With the total number of times repeated is given by 1.1.

Depending on which parameters and runmode (local,global or lca)that was supplied to both *getdamage* and *dfit*, the content of the id will be different. If a bamfile is supplied (with –bam) then each line will be the information associated with the different refids in the bam file and the id will be the referenceids from the bam file. The case scenario for this would be either obtaining perchromosome estimates of damage or per reference damage which could be relevant for metagenomic studies. If user are computing the damagesignal in the context of the lca. Then the id column will contain the taxid. If -names has been supplied to the dfit program, then the id column will the taxid:*scientific name*. If -nodes has not been defined the dfit.txt.gz will only contain information for the observed references.

If -nodes has been defined the program will aggregate the summary statitics for the internal nodes.

5 .boot.stat.txt.gz

Providing *dfit* command with *nbootstrap > 1* and *doboot 1* the boostrapping values (id,A_b,q_b,c_b,ϕ_b) for each iteration are stored in separate file

6 .agggregate.stat.txt.gz

Aggregates the information stored within the lca *.stat* format, described in section 2. The aggregated file has the prefix *.aggregate.stat.txt.gz*, with a similar format to the format presented for the lca output in section 2.

Col	Brief description	
1	taxid	The lca taxomic id (integer)
2	name	The lca taxonomic name(string)
3	rank	The lca taxonomic rank/level (string)
4	nalign	The aggregated number of alignments used for inferring the lca across the taxonomic levels
5	nreads	The aggregated number of reads for inferring the lca across the taxonomic levels
6	mean_rlen	The weighted mean of read length when transvering through the taxonomic levels
7	var_rlen	The pooled variance of read length when transvering through the taxonomic levels
8	mean_gc	The weighted mean of gc content when transvering through the taxonomic levels
9	var_gc	The pooled variance of gc content when transvering through the taxonomic levels
10	lca	The lca rank
11	taxa_path	The taxonimcal path from lca to root

Table 5: Content of a .aggregate.stat.txt.gz file. With column 10 containing taxid and name for the lca node separated by colon, and column 11 contains the taxid":"name for all nodes from the lca to the root, with each node separated by comma, i.e. taxid":"name,taxid":"name,taxid":"name

7 .rlens.gz

Readlength distribution. Distribution is count of alignments of specific readlengths. Depending on runmode there might be multiple groups as taxid/refs and there will be group specific distributions. Distributions for different groups are split by newlines. First entry on each line is the identifer (chromosomename,taxid). The remaining entries are the number of times we have observed an alignment of length (columnnumber). Notice that the first 30 column of counts is likely to be zero since reads shorter than 30basepairs are normally discarded.