Processing RRBS samples: a User Guide

Peter A. Stockwell

Dept. of Biochemistry, University of Otago.

Introduction: This guide describes the steps needed to use locally written and other software tools for analysing differential methylation on multiple individuals after RRBS runs. Much of this is also adaptable to whole genome bisulphite sequence data. The description starts with obtaining genomic files for the mapping stages.

1. Software required and installation:

Obtain the bismark distribution by downloading from:

http://www.bioinformatics.babraham.ac.uk/projects/bismark/

which should produce a download like bismark_v0.7.6.tar.gz. Unpack this with

```
gzip –dc bismark_v0.7.6.tar.gz | tar –xvf –
```

to produce a directory `bismark_v0.7.6` containing documentation as a .pdf and perl executables for bismark, bismark_genome_preparation and methyation_extractor. The executables should be put into an appropriate directory on your defined path – the usual choice is `/usr/local/bin`.

Obtain the sources for bowtie from:

https://sourceforge.net/projects/bowtie-bio/files/bowtie/1.0.0

or an equivalent and install or build depending on what is obtained. For source distribution:

```
unzip bowtie-1.0.0.zip
cd bowtie-1.0.0
make
sudo cp bowtie bowtie-inspect bowtie-build /usr/local/bin
```

or for a pre-compiled form:

```
unzip bowtie-1.0.0-linux-x64_64.zip
cd bowtie-1.0.0
sudo cp bowtie bowtie-inspect bowtie-build /usr/local/bin
```

noting that the 64-bit x86_64 form is required for large genomes and data sets.

Later versions of bismark will optionally use bowtie2 as the aligner. We have not found significant mapping improvements with bowtie2, but if it is required then steps similar to those above for bowtie should be followed for http://bowtie-bio.sourceforge.net/bowtie2/

The fastx-toolkit requires libgtextutils and pkg-config. The latter is available from http://pkgconfig.freedesktop.org/releases/, obtaining version pkg-config-0.27 or later and is built with:

```
gzip –dc pkg-config-0.28.tar.gz | tar –xvf –
cd pkg-config-0.28
./configure --with-internal-glib
make
sudo make install
cd ../
```

Obtain the fastx-toolkit and libgtextutils from http://hannonlab.cshl.edu/fastx_toolkit/ and build with:

```
bzip2 –dc libgtextutils-0.6.1.tar.bz2 | tar –xvf –
cd libgtextutils-0.6.1
./configure
make
sudo make install
cd ../
bzip2 –dc fastx_toolkit-0.0.13.2.tar.bz2
cd fastx_toolkit-0.0.13.2
./configure
make
sudo make install
cd ../
```

Obtain fastqc from http://www.bioinformatics.babraham.ac.uk/projects/fastqc/ and unpack with:

```
unzip fastqc_v0.10.1.zip
cd FastQC
chmod a+x fastqc
sudo ln –s fastqc /usr/local/bin/fastqc
cd ../
```

The cleanadaptors and diffmeth programs require zlib, preferably 1.2.5 or later, to be installed in order to perform compression/decompression on the fly. This is available from http://www.zlib.net and can be unpacked and built with commands like:

```
gzip -dc zlib-1.2.8.tar.gz | tar -xvf -
cd zlib-1.2.8
./configure
make
sudo make install
cd ../
```

which will install files in /usr/local. The option exists for installing elsewhere by using

```
./configure --prefix=<somewhereIcanwrite>
```

if you don't have sudo access to /usr/local: `<somewhereIcanwrite>` being some directory to which you have access. Some change will be needed in the make scripts in order to accommodate this.

The DMAP programs are now distributed as a compressed tar file. Build the programs by:

```
gzip -dc meth_progs_dist.tar.gz | tar -xvf -
cd meth_progs_dist/src
make
```

which should generate a number of executables. Those most useful here are `cleanadaptors, diffmeth and identgeneloc` which should be put into `/usr/local/bin.` If the diffmeth compile fails through the lack of zlib, then this should be obtained and installed (see above). If that is not possible then you can build the program without zlib, but it will not be able to read bam files. To do this:

```
make diffmeth_nozlib
```

and rename the diffmeth_nozlib executable to diffmeth if required.

2. Files needed:

(a) Fasta files for each genomic chromosome, named <Header>1.fa, <Header>2.fa to <Header>X.fa & <Header>Y.fa. These programs don't use mitochondrial sequences. Various means of downloading these files are available, but an example is:

```
ftp ftp.ensembl.org
```

log on as `anonymous` using email as password, then:

```
cd pub/release-71/ftp/homo_sapiens
```

obtain compressed fasta files with:

```
mget Homo_sapiens.GRCh37.71.dna*.fa.gz
```

and wait, informational messages should appear, then finally disconnect with:

```
bye
```

Unpack the files with a command like:

```
gunzip *.fa.gz
```

which will inflate them by about 3 times, producing `.fa` files.

(b) Build bismark libraries in the directory where you have the genome sequence files with:

```
bismark_genome_preparation ./
```

which will take a while to run: best done overnight. It will create a directory with the name `Bisulfite_Genome` containing bowtie index files. If you intend to use bowtie2, it will be necessary to generate a separate index for it.

(c) Files of feature table information. Various forms can be used, but we have found the SeqMonk form to be most useful. The best method of obtaining these is within the

graphical SeqMonk application by **`File > New Project…`** then on the **Import Genome** pane do **`<Import New>`** which should prompt for a location to download and write the data files. E.g. `/Volumes/Data2/SeqMonk_Genomes`.

It is not necessary at this stage to use SeqMonk any further. The feature table files then become available at the command line as:

```
/Volumes/Data2/SeqMonk_Genomes/Homo\ sapiens/1.dat
```

etc. Feature table information can also be taken from EMBL, GenBank, GFF3 and GTF files, but the SeqMonk form has been better tested and returns transcription start sites and CpG Island positions which are not provided as conveniently by the other formats.

3. NGS Files delivered from Illumina systems:

since an entire flowcell must be run with the same protocol, sequencing is typically done in paired-end mode, where each read is primed from the 5' adaptor and sequenced in the forward direction, then reprimed from the 3' adaptor and sequenced back from the 3' end. The process also involves sequencing the index on the 3' adaptor in order to demultiplex multiple samples run in the same lane. For RRBS work, it is usual to work only with the first 5' read, since the short fragments will cause the forward and reverse reads to overlap frequently, biasing CpG mapping. It has been proposed that overlapping reads could be identified and the overlap region joined to generate longer reads for these, while permitting forward and reverse reads to be used in mapping, but we have not investigated this to date and we ignore the second read where it has been done.

(a) Decompression: the bulk of data files means that they are frequently distributed as compressed data. This will be indicated by having the suffix **`.gz`** appended to the name. Some processing (`fastqc`) can work directly with compressed files, but most work requires them to be uncompressed. It is assumed here that the base file name is **`mydata_R1.fastq`**. Decompress with:

```
gzip –dc mydata_R1.fastq.gz > mydata_R1.fastq
```

which will inflate the file by about 3x. `mydata_R1.fastq.gz` could be deleted at this point unless it is retained for further work or backup.

`(b) Quality trimming:` typical Illumina runs are now taken to 100 cycles or more but we note that the quality in RRBS runs often deteriorates signficantly before then so that a check of the quality is needed. It is usual for **`fastqc`** to be run as part of the sequencing operation, and this information may be provided along with the data. If not **`fastqc`** can be run graphically or from the command line with:

```
fastqc --outdir qc mydata_R1.fastq
```

which will write a range of quality assessments to a pre-existing directory `qc`. Within the resulting data will be a file `fastqc_report.html` which can be opened with a web browser. The page contains a summary of the run, including the number of reads and a graphical display of the run quality. You can assess an appropriate length for hard trimming by looking at this graphic. Let's assume that we want to trim to 90bp for this example, so that the command:

```
fastx_trimmer –Q 33 –l 90 –i mydata_R1.fastq > mydata_tr90_R1.fastq
```

(all one line) will generate a file of trimmed data or see `cleanadaptors` below. Note that it is possible to concatenate a series of commands together to avoid creating a whole series of intermediate files, so it is worth looking ahead to find how to perform a whole series of steps at once.

(c) Adaptor trimming:

100bp reads on a 40-220 RRBS library will frequently read into the 3' adaptor. Such reads need to have the adaptor trimmed from them or they will not map to the genome. A file of adaptor sequences is needed and we shall assume that this file is available locally for now: two formats can be used: (1) a simple text file with one adaptor sequence/line or (2) a fasta file. `cleanadaptors` v1.22 and later have many additional options including the ability to read and write gzip-compressed fastq files, to hard trim reads and to quality trim. Adaptor trimming is usually complete now with a single pass through the `cleanadaptors` program with the following:

```
cleanadaptors –i ill_inds_adapt.txt –t 3 -x 4 –F mydata_tr90_R1.fastq >
mydata_tr90ad3pp_R1.fastq
```

(all complete on one line). `–t 3` trims 3 bases back from the adaptor to delete the C incorporated during library preparation and `–x 4` rejects any reads which are trimmed to less than 4 bp. Obviously, the latter limit could be set higher, but 4 is the minimum that `bismark` expects without complaint. Later versions of `cleanadaptors` could simplify the required commands for decompressing and hard trimming to something like:

```
cleanadaptors -I contam.fa -t 3 -x 4 -l 90 -z -F mydata_R1.fastq.gz -Z
mydata_tr90ad3pp_R1.fastq
```

which will hard trim all reads to 90bp, check for adaptors in fasta formatted contam.fa, rejecting reads which would be less than 4bp after these operations. Adaptor matching trims are further shortened by 3 bp. The input sequence is gzip compressed while the output is uncompressed.

A further issue sometimes arises with RRBS data where adaptors sometimes mismatch in the first two base positions which will prevent `cleanadaptors` from finding them, although `FastQC` scans will still observe them. This behaviour can be avoided by using the `-T` option which 5' trims the adaptor sequences before the run, and then increases the `-t` trim back by the same amount. `-T 2` seems to work appropriately for this.

(d) Concatenating commands:

Unix-type operating systems allow us to concatenate a series of commands together to avoid creating intermediate files. It is possible to combine commands in the following way, noting that the entire thing has to be complete on a single line or have '\' characters to indication continuation. Assuming we already knew to trim to 90bp, we could perform the following altogether:

```
gzip –dc mydata_R1.fastq.gz | fastx_trimmer –Q 33 –l 90 |  cleanadaptors
–i ill_inds_adapt.txt –t 3 -x 4 –F - > mydata_tr90ad3pp_R1.fastq
```

(all complete on one line). The content of the output file can be checked by:

```
more mydata_tr90ad3pp_R1.fastq
```

to ensure that things have worked sensibly.

4. Mapping:

we now use `bismark` to map the reads with the following:

```
bismark –n 1 <Path_to_Genome_directory> mydata_tr90ad3pp_R1.fastq
```

which will generate a SAM file `mydata_tr90ad3pp_R1.fastq_bismark.sam` and a report file. Versions of **`bismark`** from v0.14.0 produce BAM files by default. While it is possible to convert BAM files to SAM with **`samtools`** it is more efficient for diffmeth to read them directly, which is now possible with **`diffmeth`** v1.60 or later.

Bismark may generate messages like:

```
Chromosomal sequence could not be extracted for   HWI-ST871:252:C21CFACXX:5:2102:16275:48963_1:N:0:ATTCCT
```

which indicate that a read had mapped at the very end of a chromosome, overlapping beyond the end. These messages can be ignored.

Mapping is performed for each of the different samples in the study – usually we keep each of the sample files in its own directory, relying on Unix symbolic links (see `man ln`) to make the fastq data available locally without copying it, if appropriate. Examining the report files will show if the bisulphite chemistry had been successful: specifically the number and percentage of unique alignments and the C methylated in a CpG context should indicate this.

5. Differential methylation:

It is not necessary for mapping files all to be in the same place for the next stage of analysis. Optionally, symbolic links can be used to provide convenient access to bismark output files from elsewhere. Run diffmeth with commands like:

```
diffmeth -F 2 -t 10 -X 40,220 -I 9  \
-g /Volumes/Data2/HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome.
\
-R ../X12/bismark_at3/X12_ad3tr85.fastq_bismark.sam \
-R ../X14/bismark_at3/X14_ad3tr90.fastq_bismark.sam \
-R ../X16_twinA/bismark_at3/X16_ad3tr80.fastq_bismark.sam \
-R ../X18/bismark_at3/X18_ad3tr85.fastq_bismark.sam \
-R ../X19/bismark_at3/X19_ad3tr80.fastq_bismark.sam \
-R ../X20/bismark_at3/X20_ad3tr80.fastq_bismark.sam \
-R ../X21/bismark_at3/X21_ad3tr80.fastq_bismark.sam \
-R ../X9006/fastq/bismark_at3/s6_ad3tr65.fastq_bismark.sam \
-R ../X9007/bismark_at3/X9007_ad3tr100.fastq_bismark.sam \
-R ../X9010/bismark_at3/X9010_ad3tr80_40.fastq_bismark.sam \
-R ../X9015/fastq/bismark_at3/s1_ad3tr75.fastq_bismark.sam \
> 11ind_3pp_allpr_F2t10.txt
```

Noting that putting such commands into a shell script is often useful, since it allows various options to be tried with minimal risk of errors. The options used above can be found with:

```
diffmeth –h
```

or in the program documentation. To describe those used here:

`-F 2` indicates that 2 CpGs in each fragment must qualify for the criteria…

`-t 10` withq 10 or more hits

`-X 40,220` requires a X^2 statistic on 40-220bp fragments

`-I 9` requires that 9 of the 11 individuals in this run have valid fragments for that fragment to be considered

`-g` indicates the location of the chromosomal fasta files `1.fa, 2.fa,` etc.

`-R` is the series of sample SAM files from bismark alignment.

-n use a file of restriction sites instead of the default MspI site for cleavage. The file should only contain a list of cleavage sites (case independent) one per line, with the cutting position indicated by a '^' character. The following would be for a MspI and TaqI combined digest:

```
C^CGG
T^CGA
```

diffmeth writes output to the Unix `stdout` so the `> file.txt` construct catches that into a named file. The output is a tab-delimited file of chromosome, fragment start and stop, CpG counts and X^2 probability and statistic. It is suitable for importation into Excel or for further processing. The above run saved all fragments but it is possible to filter for low probabilities.

From v1.50 diffmeth will optionally take chromosome information from a file rather than automatically generating file names as determined by the -k and -z options. The file, specified with -G should be used when: (1) files are in different directories, (2) naming is not consistent for -g to work, (3) genome has chromosome IDs other than numbers, X & Y (e.g. avian or mitochondrial) or (4) you want a subset of chromosomes. Format of the file is lines of <ChromosomeID> <Filename> e.g.:

```
Y /Genomes/hs_GRCh37/HomoSapiens_Chr_Y.fasta
```

Filenames containing spaces should be enclosed in double quotes '"'.

`diffmeth` from v1.60 can read either BAM and SAM files with the use of the -z and -Z options which are positional and affect the inputs following them on the command line. A mixture of sam and bam files can be processed by multiple occurrences of -z and -Z. The default is sam file input (= -Z). The use of these options is similar to that in **`cleanadaptors`**.

Some specific examples:

i) Identifying differentially methylated regions (DMRs), pairwise: the following command applies Fisher's Exact statistic (-P 40,220) to a pair of sam files from the above. The options given require that at least 2 CpGs in each fragment have 10 or more hits (-F

2 -t 10) and that the leading CpG of 3' mapped reads is assigned to the previous fragment (-N). Only chromosome 21 is to be used (-c 21):

```
diffmeth -g /HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome. \
-c 21 -P 40,220 -F 2 -t 10 -N -R ctrl_1.fastq_bismark.sam -R mds_1.fastq_bismark.sam
```

ii) Identifying DMRs - ChiSQ test: on a cohort of 6 individuals. The following command runs diffmeth using the ChiSQ statistic (-X 40,220) requiring that at least 2 CpGs in each fragment have 10 or more hits(-F 2 -t 10), that at least 4 individuals contribute to the statistic(-I 4) and that the leading CpG of 3' mapped reads is assigned to the previous fragment (-N). Only chromosome 21 is to be used (-c 21). The comparison including control and MDS individuals is intended to illustrate the use of this statistic with the test data set, not to imply that there is no difference between the groups:

```
diffmeth -g /HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome. \
 -c 21 -X 40,220 -F 2 -t 10 -N -I 4 -R ctrl_1.fastq_bismark.sam \
 -R ctrl_2.fastq_bismark.sam -R ctrl_3.fastq_bismark.sam \
 -R mds_1.fastq_bismark.sam -R mds_2.fastq_bismark.sam \
 -R mds_3.fastq_bismark.sam
```

iii) Comparing methylation between two groups: this applies the ANOVA F ratio test (-a 40,220), requiring that at least 4 individuals show counts for a fragment to be included (-I 4) and the leading CpG of 3' mapped reads is assigned to the preceding fragment (-N). Sam data for the treatment/disease group is identified with -S:

```
diffmeth -g /HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome. \
-c 21 -a 40,220 -N -I 4 -R ctrl_1.fastq_bismark.sam \
-R ctrl_2.fastq_bismark.sam -R ctrl_3.fastq_bismark.sam \
-S mds_1.fastq_bismark.sam -S mds_2.fastq_bismark.sam \
-S mds_3.fastq_bismark.sam
```

iv) As for 3, but indicate the more methylated group (-A 40,220) and restrict the output to Pr < 0.01 (-U 0.01). Each line is suffixed with a 'R' or 'S' character to indicate which group had higher methylation and a summary of the valid sample counts for R & S groups.

```
diffmeth -g /HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome. \
-c 21 -A 40,220 -U 0.01 -N -I 4 -R ctrl_1.fastq_bismark.sam \
-R ctrl_2.fastq_bismark.sam -R ctrl_3.fastq_bismark.sam \
-S mds_1.fastq_bismark.sam -S mds_2.fastq_bismark.sam \
-S mds_3.fastq_bismark.sam
```

v) Compare two individuals with Fisher's Exact Test, using the `-R` and `-S` group formality to make `diffmeth` generate columns showing the methylation proportion for each and which is the more methylated:

```
diffmeth -g/HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome. \
     -P 40,220 -N -F 2 -t 10 -R Ind_1.fastq_bismark.sam  -S Ind_2.fastq_bismark.sam
```

vi) For WGBS analysis: for tiled fixed window analysis. The option –W <windowlength> is added to the command. E.g. as for 4, but with fixed width windows of 1000 bp rather than fragments (-W 1000):

```
diffmeth -g /HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome. \
-c 21 -A 40,220 -W 1000 -U 0.01 -N -I 4 -R ctrl_1.fastq_bismark.sam \
-R ctrl_2.fastq_bismark.sam -R ctrl_3.fastq_bismark.sam \
-S mds_1.fastq_bismark.sam -S mds_2.fastq_bismark.sam \
-S mds_3.fastq_bismark.sam
```

vii) Generate a list of CpG counts for a combined Taq1 & Msp1 digest, showing CpGs with > 0 counts. Do this for H. sapiens chromosome 1 only. rstsites.txt contains the sites as indicated above (-n), and the mapping is for a sam file:

```
diffmeth -G c1_chrinfo.txt -n rstsites.txt -L 40,220 \
  -R Ind_2.fastq_bismark.sam
```

6. Proximal gene location:

Takes the output file from diffmeth and compares the positions of fragments or any region of interest with feature table information. Typically this would be processed with:

```
identgeneloc -i -Q -U -R -B "protein_coding" -p
/Volumes/Data2/SeqMonk_Genomes/Homo\ sapiens/GRCh37/ -s ".dat" -r
dmeth_10ind_lopr.txt
```

(all complete on 1 line) where the options in this run are as follows:

`-i` relates fragments to intron/exon boundaries and looks internally within genes

`-Q` expects feature table information from SeqMonk data files

`-U` scans for nearest upstream CpG Island

`-R` shows ranges for CpG Islands

`-B "protein_coding"` restricts the search to genes with /biotype="protein_coding" (noting that the SeqMonk Human data includes this, but not for zebrafish)

`-p` defines the header for numbered feature table files (i.e. where you put the files)

`-s` defines the suffix for numbered feature table files (e.g. 1.dat)

`-r` the `diffmeth` output file

Like other programs, the output is to the Unix `stdout` stream which defaults to the terminal or which can be redirected into a file. The output is tab-delimited and can be imported into Excel or used for other processing.

7. Sorting SAM files for IGV use:

IGV requires sorted files and this can be done with Bismark output files using the following:

```
grep –v '^@' mydata_R1.fastq_bismark.sam | sort –n –k3,3 | sort –n –k4,4
> mydata_R1.fastq_bismark_sorted.sam
```

(all complete on one line) where the grep statement is used to exclude header lines from the sam file. It is possible to have Bismark exclude these, but in typical runs, they may have been left in place or have been retained for some other purpose. Other sorted inputs can be used for IGV, but the strategy shown here seems effective.

8. %methylation for individual CpGs:

Can be performed with a combination of diffmeth and the awk script getcpgpcmeth.awk. diffmeth is run using the –e or –E options to generate a detailed CpG list, then the script decomposes the output into a tabbed list of chromosome, position and %methylation. CpGs with no hits are shown as '-'. An example:

```
diffmeth –e 40,220 -g
/Volumes/Data2/HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome. -
R ../X12/bismark_at3/X12_ad3tr85.fastq_bismark.sam | awk –f getcpgpcmeth.awk
```

(all complete on one line) will write the resulting list to <stdout>.

9. Comparing Samples for control/treatment:

diffmeth v1.45 enables differential methylation comparison between two different groups of samples. E.g.:

```
diffmeth -c 21 \
-g /Volumes/Data2/HomoSapiens_genome/hs_ref_GRCh37/Homo_sapiens.GRCh37.65.dna.chromosome.
\
-a 40,220 -R ctrl_1.sam -R ctrl_2.sam -R ctrl_3.sam \
-S mds_1.sam -S mds_2.sam -S mds_2.sam
```

will perform Analysis of Variance (ANOVA) on 3 control and 3 MDS (disease) samples, in this case only for chromosome 21 (**-c 21**). Note that diffmeth does not distinguish which set is disease or control, the **-R** and **-S** options are only used to identify which samples belong to one or the other group. Probabilities for the resulting F statistics are calculated using a continued fraction method modified from 'Numerical Recipes in C: the Art of Scientifc Computing' (ISBM 0-521-43108-5). A **-A** option returns additional information about which of the **-R/r** or **-S/s** sample groups has greater methylation and gives counts of the samples which contributed to the Anova statistic.

10. Conclusion:

These notes are intended as a guide, they do not give all the possible options of the programs mentioned. Further, changes in sequencing systems and downstream processing will date some of this material.

Peter A. Stockwell
Dept. of Biochemistry, University of Otago, Dunedin, New Zealand.
19-Oct-2015.