bio-101

Utilisation of Methylome Data to Identify Stably Unmethylated Regions in Plant Genomes

Judith I. M. Eglitis-Sexton1, Leroy M. Mangila1, 2, Haylie L. Andrews1, Lee T. Hickey3, and Peter A. Crisp1, *

[1]School of Agriculture and Food Sustainability, The University of Queensland, Brisbane, Australia
[2]School of the Environment, The University of Queensland, Brisbane, Australia
[3]Centre for Crop Science, Queensland Alliance for Agriculture and Food Innovation, The University of Queensland, Brisbane, Australia
*For correspondence: p.crisp@uq.edu.au

Abstract

DNA methylation is a key chromatin modification that provides a mechanism for epigenetic inheritance. However, DNA methylation profiles can also be used to annotate or filter plant genomes by partitioning a genome into methylated and unmethylated regions (UMRs). UMRs comprise only a very small fraction of moderate to large plant genomes, yet these regions are known to be highly enriched in functionally significant genomic sequences, including genes and cis-regulatory elements. Therefore, methods to efficiently and accurately identify UMRs in plant genomes are useful for genome annotation and functional genomics and potentially for crop improvement. In this protocol, we provide a reproducible vignette to identify UMRs in the maize methylome, starting from raw fastq files obtained by whole-genome bisulfite sequencing. This method determines the average methylation per 100 bp tile of the genome and classifies tiles as methylated and unmethylated. To support training and learning, this step-by-step guide uses a small data subset corresponding to a 20 Mb region of the maize genome so that this analysis could be completed on a standard desktop computer with minimal computational resources.

Keywords: DNA methylation, Epigenetics, Unmethylated regions, Plants, Whole-genome bisulfite sequencing, Methylome, Bioinformatics

Copyright: © 2024 The Authors; exclusive licensee Bio-protocol LLC.
This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

Background

Groundbreaking research into epigenetics has opened up possibilities for its application to human diseases, modern agriculture, synthetic biology, and studies of evolution. Covalent attachment of a methyl group to the 5′ carbon of cytosine in DNA (5-methylcytosine) is generally known as DNA methylation, and 5-methylcytosine is sometimes referred to as the fifth base [1]. DNA methylation can provide a mechanism for epigenetic inheritance of phenotypes and, accordingly, is often referred to as an epigenetic modification. DNA methylation plays critical roles in transposon silencing, genome stability and organisation, heterochromatin formation, gene regulation, development, and imprinting [2]. More recently, we have demonstrated that there is great utility in identifying regions of the genome that lack DNA methylation, referred to as unmethylated regions (UMRs) [3]. In this protocol, we provide a step-by-step guide to identify UMRs from DNA methylation sequencing data.

There are a variety of technologies that can be used to detect and quantify the levels of DNA methylation at a particular locus, and many of these methods can also be scaled to genome-wide profiling of the entire methylome [4,5]. Technologies include bisulfite based, digestion based, or affinity based. Whole-genome techniques include whole-genome bisulfite sequencing (WGBS), enzymatic methyl-seq (EM-seq), reduced representation bisulfite sequencing (RRBS-seq), which uses bisulfite coupled with enzymatic digestions, and methyl-DNA immunoprecipitation (MeDIP), which attracts methylation through antibody enrichment [6–9]. In addition, over the last few years, nanopore sequencing has emerged as another alternative, allowing direct detection of DNA and RNA methylation on long reads [10]. Importantly, the ability to identify methylation at single-base-pair resolution has allowed specific understanding of how methylation—or the lack thereof—influences specific regions of the genome, including regulatory regions that may be important for transcriptional changes and phenotypic variation for traits of interest. In this protocol, we analyse WGBS sequence data; we routinely apply the same analysis pipeline to EM-seq data, to which it is equally applicable. Together, these methyl-seq whole-genome sequencing approaches (WGBS and EM-seq) are the current gold standard for DNA methylation profiling, as they provide a whole-genome analysis of regions that are methylated and the types of methylation present [11,12].

The key step in WGBS involves treating genomic DNA with sodium bisulfite, which converts unmethylated cytosines to uracils by deamination (which are converted to thiamine following PCR amplification), while 5’-methylcytosines remain unaffected [8]. After bisulfite treatment is complete, converted DNA can be prepared for sequencing, commonly on the Illumina platforms, allowing inference of methylation at single-base resolution by comparison to a reference genome. Due to the cytosine conversion step in WGBS (and in EM-seq), a key requirement in the bioinformatic analysis is to use bisulfite aware mapping software to identify the T-converted unmethylated cytosines, which will appear as polymorphisms compared to a reference genome. Commonly used mapping software includes BSMAP, BWA-meth, and Bismark [13,14].

In plants and animals, DNA methylation occurs in different sequence contexts including CG, CHG, and CHH (where H is A, T, or C). Studies have found that the type of methylation can correspond to different functions; for example, CG-only methylation has been correlated with actively transcribed gene bodies, while transcriptionally silenced regions are associated with high levels of methylation in all contexts (CHH, CHG, and CG) [15]. The output of this analysis pipeline includes files for visualisation of each type of methylation. Recently, the research community has demonstrated that regions that lack DNA methylation in all contexts are of particular significance [3,15–23]. Partitioning genomic regions into different categories of methylation types allows us to identify unmethylated regions (UMRs). This approach can be very useful for annotating a genome [24] because UMRs tend to align with regions containing functional genes and also cis-regulatory elements (CREs) [3,25]. CREs are non-coding elements and include enhancers, promoters, and silencers, which can influence gene expression. These have the potential to be important targets for selection and breeding and also as targets for genetic engineering. Researchers have yielded promising results from genetic modification of CREs for tailoring gene expression without causing developmental problems [26,27]. For example, improvements in yield in rice [28,29], maize [30], and tomato [31–33] have been produced by gene editing non-coding cis-regulatory promoter alleles. However, efficiently identifying functional regions of regulatory importance is challenging [3]; identification of UMRs can narrow the search for genetic targets. As genomic sequencing and tools for genome annotation improve, our ability to understand the functionality of the genome is made increasingly powerful when combining knowledge of transcription binding sites, conservation of sequences through evolution, epigenomic markers for transcriptional activity, and other emerging technologies.

In the case study outlined below, we have provided a subset of raw WGBS reads extracted from SRX5532987, a

published study of DNA methylation in maize leaf tissue [3]. This subset of reads aligns to a small 20 Mb region of maize chromosome 1; we provide this subset of the genome as a reference sequence for mapping. These files enable processing this example data with minimal computational resources and could be completed on a basic laptop computer with the appropriate software installed. The pipeline was originally optimised for maize, but it is generally applicable to other species without modification [34]; however, users could consider modifying the thresholds in the UMR-calling step if applied to a plant genome that has particularly unusual levels or distribution of DNA methylation. The key steps of the workflow are outlined in Figure 1.

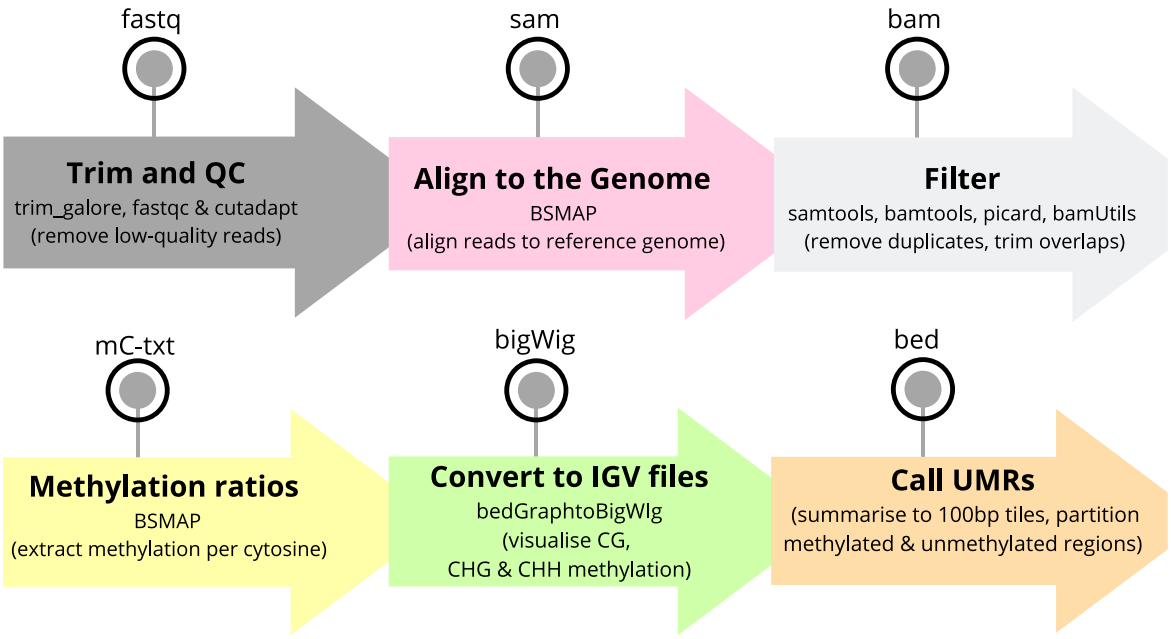

Figure 1. Overview of the workflow to identify unmethylated regions (UMRs) in a plant genome. The bioinformatic workflow presented in the protocol includes six major steps: 1) read trimming, 2) read mapping, 3) filtering, 4) extraction and quantification of methylation levels per cytosine, 5) data visualisation, and lastly 6) summarisation and identification of UMRs. The output file formats are indicated above each step and the software tools and their purpose are summarised.

Software and datasets

The required software, references, and websites for download are provided below:

1. trim_galore! [35], v0.6.4_dev (https://github.com/FelixKrueger/TrimGalore)
2. cutadapt [36], v1.8.1 (https://cutadapt.readthedocs.io/en/stable/)
3. fastQC [37], v0.11.5 (https://www.bioinformatics.babraham.ac.uk/projects/fastqc/)
4. BSMAP [38], v2.74 (https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-10-232)
5. samtools [39], v1.16 (https://github.com/samtools/samtools)
6. bamtools [40], v2.4.0 (https://github.com/pezmaster31/bamtools/)
7. Java [41], v1.8.0_45 (https://www.oracle.com/java/technologies/javase/8u45-relnotes.html)
8. Picard, v2.9.0 (https://broadinstitute.github.io/picard/)
9. bamUtil [42], v1.0.13 (https://genome.sph.umich.edu/wiki/BamUtil)
10. Python [43], v2.7.5 (https://www.python.org/)
11. bedGraphToBigWig (UCSC; download from http://hgdownload.soe.ucsc.edu/admin/exe/)
12. perl [44], v5.26.2 (https://www.perl.org/)
13. IGV [45], v2.5.3 (https://software.broadinstitute.org/software/igv/)

14. R, v4.1 (https://www.r-project.org/)

Each of the above software was run in a terminal application on a server running the software Linux but could also be run on any personal machine running Linux or macOS.

Input data

To demonstrate the identification of UMRs from WGBS data, we have provided a small subset of reads (3,396 reads) for analysis in paired fastq files ("B73_chr1_subset_reads_1.fastq" and "B73_chr1_subset_reads_2.fastq"). These reads were extracted from SRR8738272 [3] SRA PRJNA527657, WGBS from a maize B73 seedling, V1 stage, leaf shoot. These reads map to a section of the maize V4 genome between 80 and 100 kb on chromosome 1. We have provided a fasta reference sequence for this portion of the maize genome for mapping the reads ("maize_chr1_reference.fa"). This minimal example will run with minimal hardware requirements and should only take a few seconds per step.
Users interested in analysing their own data should first perform a quality check before proceeding with this pipeline. For example, FastQC can be used to perform basic checks of sequence data quality. Additionally, when analysing bisulfite data, it is critical to check the conversion efficiency. This is not performed in this example for simplicity; however, it should be performed on every dataset. This can be done either by using a spike-in unmethylated DNA sequence such as lambda gDNA and then mapping reads to this reference sequence or, in plants, mapping to the unmethylated chloroplast genome; good conversion rates should preferably be >99%. In this example, the WGBS data is paired end data; however, this pipeline can equally be run on single-end data; paired end data is not a requirement for UMR analysis.
All steps in this pipeline are run in the same folder that contains the provided input data and other required files. The output files are written to the same folder.
Link to the input data and scripts:

- Repository: https://github.com/Bio-protocol/unmethylated-regions_UMR-extractor-WGBS/tree/master
- Input data located in `/input`
- Other required scripts in `/lib`

Procedure

Case study

1. Trim the reads.
   In the first step, we trim the reads for quality and remove any contaminating adapter sequences. This step requires the software packages trim_galore!, cutadapt, and fastQC. Once these software and the fastq reads are loaded, the code laid out below can be run in the same folder that contains the fastq files. The parameter "phred33" instructs cutadapt to use ASCII+33 quality scores as Phred scores for quality trimming. The parameter "clip" removes 20 base pairs from the 5' end of both reads one and two; this is required for some library preparation methods, for example if using the ACCEL-NGS Methyl-Seq DNA Library Kit (SWIFT Biosciences). The parameter "o" indicates that the output will be placed in the current directory, and the paired end fastq files are provided following the "paired" argument. Please note that the length of adapters is also dependent on the library preparation methods.

   ```
   trim_galore \
   --phred33 \
   --clip_R1 20 --clip_R2 20 \
   -o ./ \
   --paired B73_chr1_subset_reads_1.fastq B73_chr1_subset_reads_2.fastq
   ```

```
```

For each fastq file, two new files are produced: 1) a trimming report and 2) a new fastq with trimmed reads as shown below.

```
B73_chr1_subset_reads_1.fastq_trimming_report.txt
B73_chr1_subset_reads_1_val_1.fq
B73_chr1_subset_reads_2.fastq_trimming_report.txt
B73_chr1_subset_reads_2_val_2.fq
```

2. Align the reads to the genome reference.
In this step, we use BSMAP v2.74 to align the reads from the fastq file with the supplied maize genome reference file "maize_chr1_reference.fa." The input parameter "-v 5" allows up to five mismatches, "-r 0" reports only unique mapping pairs, and "-q 20" performs quality trimming to q20. The output file generated is in SAM format.

```
bsmap \
-a B73_chr1_subset_reads_1_val_1.fq \
-b B73_chr1_subset_reads_2_val_2.fq \
-d maize_chr1_reference.fa \
-o mapped.sam \
-v 5 \
-r 0 \
-q 20
```

Below is an example of the standard error report that should be generated from the code above; these reads should have a paired mapping rate of approximately 97%.

```
BSMAP v2.74
Start at: Tue Jun 14 22:44:54 2022

Input reference file: maize_chr1_reference.fa (format: FASTA)
Load in 1 db seqs, total size 20000 bp. 0 secs passed
total_kmers: 43046721
Create seed table. 1 secs passed
max number of mismatches: 5  max gap size: 0
kmer cut-off ratio: 5e-07
max multi-hits: 100  max Ns: 5  seed size: 16 index interval: 4
quality cutoff: 20  base quality char: '!'
min fragment size:28 max fragemt size:500
start from read #1  end at read #4294967295
additional alignment: T in reads => C in reference
mapping strand (read_1): ++,-+
mapping strand (read_2): +-,--
Pair-end alignment(8 threads)
Input read file #1: B73_chr1_subset_reads_1_val_1.fq (format: FASTQ)
Input read file #2: B73_chr1_subset_reads_2_val_2.fq (format: FASTQ)
Output file: mapped.sam (format: SAM)
```

```
Thread #2:  3395 read pairs finished. 1 secs passed
Total number of aligned reads:
pairs:  3277 (97%)
single a: 21 (0.62%)
single b: 16 (0.47%)
Done.
Finished at Tue Jun 14 22:44:55 2022
Total time consumed: 1 secs
```

3. Fix and sort the mapped reads.
   This step requires samtools (v1.3) to perform fixing and sorting of the BSMAP output provided by the previous step. First, we convert the files to BAM format and then name-sort them. The *fixmate* option of samtools can then be used to ensure that mates have the pair's coordinates and insert sizes (to ensure compliance with downstream software) and mappings are again sorted and also indexed. Indexing is not strictly required but can be performed so the mapping file could be visualised in IGV.

   ```
   samtools view -bS mapped.sam > mapped.bam
   samtools sort -n mapped.bam -o mapped_nameSrt.bam
   samtools fixmate mapped_nameSrt.bam mapped_nameSrt_fixed.bam
   samtools sort mapped_nameSrt_fixed.bam -o mapped_sorted.bam
   samtools index mapped_sorted.bam

   rm mapped.sam mapped_nameSrt.bam mapped_nameSrt_fixed.bam mapped.bam
   ```

   The output files generated are:

   ```
   mapped_sorted.bam
   mapped_sorted.bam.bai
   ```

   To see some summary statistics about the mapping file, we can use *samtools stats*.

   ```
   samtools stats mapped_sorted.bam | grep ^SN | cut -f 2-
   ```

   Some select metrics from *samtools stats* are shown below.

   ```
   raw total sequences: 6591
   reads mapped: 6591
   reads mapped and paired:  6566 # paired-end technology bit set + both mates
   mapped
   reads unmapped: 0
   reads duplicated:  0  # PCR or optical duplicate bit set
   reads QC failed:  0
   average length: 104
   maximum length: 106
   average quality:  36.6
   ```

```
insert size average: 156.3
```

4. Filter the mapping file.
This step removes improperly paired reads (for example, pairs that map to different chromosomes), read duplicates, and any overlapping portion of read pairs. Removing duplicate reads and trimming overlaps is important because these represent redundant information originating from the same DNA molecule; retaining duplicates or overlaps can lead to biassed data.

   This step requires bamtools; we use *bamtools filter* to remove any improperly paired or unmapped reads. The second part of this step also requires an installation of Java to run picard for the removal of duplicate reads.

   ```
   bamtools filter \
   -isMapped true \
   -isPaired true \
   -isProperPair true \
   -in mapped_sorted.bam \
   -out mapped_sorted_pairs.bam
   ```

   Now, we remove duplicate reads using picard. This is an essential step in any DNA methylation analysis; however, in this example there are no duplicate reads, so the output file should contain all the input mappings.

   ```
   java -jar /path/to/picard.jar MarkDuplicates \
   I=mapped_sorted_pairs.bam \
   O=mapped_sorted_MarkDup_pairs.bam \
   METRICS_FILE=mapped_MarkDupMetrics.txt \
   ASSUME_SORTED=true \
   CREATE_INDEX=False \
   REMOVE_DUPLICATES=true
   ```

   Finally, we trim any overlapping portion of paired reads so that overlapping regions are only counted once in the analysis. Clipping is required; otherwise, cytosines in the overlapping regions are counted twice. These cytosines represent the same biological information, measured in technical replication, so should only be counted once.

   ```
   bam clipOverlap \
   --in mapped_sorted_MarkDup_pairs.bam \
   --out mapped_sorted_MarkDup_pairs_clipOverlap.bam \
   --stats
   ```

   Below is an example of the standard error report from the *clipOverlap* step.

   ```
   Overlap Statistics:
   Number of overlapping pairs: 2922
   Average # Reference Bases Overlapped: 61.3809
   ```

```
Variance of Reference Bases overlapped: 727.063
Number of times orientation causes additional clipping: 176
Number of times the forward strand was clipped: 1420
Number of times the reverse strand was clipped: 1502
Completed ClipOverlap Successfully.
```

5. Extract cytosine methylation levels.
The mapping files must now be analysed to determine the level of methylation at each cytosine. A script *methratio.py* is provided with the BSMAP software to extract methylation data; this requires the installation of python and BSMAP. In addition, samtools is required, which is also provided with BSMAP; importantly, this script requires an older version of samtools (< v1.1.18). The parameter "s" is used to direct *methratio.py* to the correct version of samtools, which can be found in the installation folder of BSMAP. The code displayed first directs python to the location of *methratio.py;* the parameter "o" is the output summary text file. The option "d" is used to indicate the reference sequence file (FASTA format); in this case, "maize_chr1_reference.fa," the 20 Mb subset of maize chromosome 1, which is provided. The option "u" processes only unique mappings and pairs, while "z" reports the loci with zero methylation ratio; the parameter "r" is used to remove duplicates.

```
python2 ~/software/bsmap-2.74/methratio.py \
-o methratio.txt \
-d maize_chr1_reference.fa \
-u -z \
-s ~/software/bsmap-2.74/samtools \
-r mapped_sorted_MarkDup_pairs_clipOverlap.bam
```

** "**~/software/**" should be replaced with the user's path to the location of the BSMAP software installation.*

An example standard error report is provided below.

```
total 5701 valid mappings, 7121 covered cytosines, average coverage: 11.16
fold.
```

An example of the output text file called "methratio.txt" is provided below.

```
chr  pos  strand context ratio eff_CT_count C_count CT_count  rev_G_count
rev_GA_count CI_lower  CI_upper
maize_chr1_reference 8  +  ATCAT 0.000 1.00 0  1  0  0  0.000 0.793
maize_chr1_reference 15  +  TTCAC 0.000 2.00 0  2  1  1  0.000 0.658
maize_chr1_reference 17  +  CACAA 0.000 2.00 0  2  1  1  0.000 0.658
maize_chr1_reference 22  +  AACCA 0.000 3.00 0  3  3  3  0.000 0.562
maize_chr1_reference 23  +  ACCAC 0.000 3.00 0  3  3  3  0.000 0.562
```

6. Parse the output of BSMAP.
The output of *methratio.py* provides the sequence context of each cytosine with the two adjacent nucleotides on either side of the cytosine (column 4 "context"). Here, we use a custom *awk* function to convert this output file into a new file with the general methylation context of each cytosine (CG, CHG, or CHH) and parse

coordinates to zero-based format "BED" type for bedtools. Subsequently, *awk* is used to convert to bedgraph format (columns: chromosome, start, stop, and ratio) for downstream tools.

```
# awk function to parse the output of bsmap methratio.py script
awk_make_bed='BEGIN {OFS = FS} (NR>1){
    if(($3=="-" && $4~/^.CG../ ) || ($3=="+" && $4~/^..CG./))
      print $1, $2-1, $2, $3, "CG", $5, $6, $7, $8, $9, $10, $11, $12;
    else if(($3=="-" && $4~/^C[AGT]G../ ) || ($3=="+" && $4~/^..C[ACT]G/))
      print $1, $2-1, $2, $3, "CHG", $5, $6, $7, $8, $9, $10, $11, $12;
    else   if(($3=="-"   &&   $4~/^[AGT][AGT]G../   )   ||   ($3=="+"   &&
$4~/^..C[ACT][ACT]/))
      print $1, $2-1, $2, $3, "CHH", $5, $6, $7, $8, $9, $10, $11, $12;
    else
      print $1, $2-1, $2, $3, "CNN", $5, $6, $7, $8, $9, $10, $11, $12
    }
    '

 # run the awk function
awk -F$"\\t" "$awk_make_bed" \
  "methratio.txt" > "BSMAP_out.txt"

 # output file:
maize_chr1_reference 7   8   +   CHH   0.000 1.00 0   1   0   0   0.000 0.793
maize_chr1_reference 14   15   +   CHH   0.000 2.00 0   2   1   1   0.000 0.658
maize_chr1_reference 16   17   +   CHH   0.000 2.00 0   2   1   1   0.000 0.658
maize_chr1_reference 21   22   +   CHH   0.000 3.00 0   3   3   3   0.000 0.562
maize_chr1_reference 22   23   +   CHH   0.000 3.00 0   3   3   3   0.000 0.562
```

7. Generate bigWig files for viewing in IGV.
Now we can further process the output files into a format compatible with IGV for inspecting the data. Use *bedGraphToBigWig* to make a bigWig file for IGV. The *awk* function filters by required columns and gives us the average percentage of methylation. We are then able to split this into three files based on methylation context (CG, CHH, CHG).

```
# awk function to filter to only the required columns and to calculate the
average percent methylation
  awk_make_bedGraph='BEGIN {OFS = FS} (NR>1){
   print $1, $2, $3, $8/$9*100, $5
  }
  '

 # awk function to split the bedgraph into three files by methylation
context
  awk_make_bedGraph_context='BEGIN {OFS = FS} (NR>1){
   print $1, $2, $3, $4 > "BSMAP_out_"$5".bedGraph"
  }
  '

# run the two above functions
  awk -F$"\\t" "$awk_make_bedGraph" \
```

```
  "BSMAP_out.txt" | \
  awk -F$"\\t" -v ID=$ID "$awk_make_bedGraph_context" -

# Example for CG context (columns: chromosome, start, end, percent-
methylation)
maize_chr1_reference 24  25  100
maize_chr1_reference 25  26  75
maize_chr1_reference 95  96  85.7143
maize_chr1_reference 96  97  64.7059
maize_chr1_reference 125  126  60

# Make bigWigs files for IGV per context
bedGraphToBigWig BSMAP_out_CG.bedGraph maize_chr1_reference.chrom.sizes
BSMAP_out_CG.bigWig

 bedGraphToBigWig BSMAP_out_CHG.bedGraph maize_chr1_reference.chrom.sizes
BSMAP_out_CHG.bigWig

bedGraphToBigWig BSMAP_out_CHH.bedGraph maize_chr1_reference.chrom.sizes
BSMAP_out_CHH.bigWig

# remove intermediate files
rm -rv BSMAP_out*.bedGraph
```

8. Summarise methylation levels into 100 bp tiles.
The output of the previous step is a text file per context with the average methylation level of each individual cytosine. DNA methylation can either be analysed at the single-cytosine level or at a regional level. Single-cytosine analysis can be useful for investigating rates of epimutation; however, it is generally agreed that methylation over a contiguous region is of most biological relevance because this can affect chromatin conformation, chromatin accessibility, and gene expression. A powerful and simple (and computationally cheap) way to determine region-level methylation is to divide the genome into small equal-sized tiles (also sometimes called windows or bins). We have found that 100 bp tiles are an optimal compromise between resolution and computational efficiency. We note that there are many different software tools that provide alternative (often more complex) algorithms for defining regional methylation and partitioning the genome into different methylation states, for example *DSS* [46], *methylkit* [47] or *MethylScore* [48]. Here, we provide a simple perl script *met_context_window.pl* to parse and summarise the BSMAP output into the average methylation per context per 100 bp tile. The argument "100" sets the tile size; users can also parse the data into different sized tiles by changing the last argument.

```
# summarise into 100bp tiles
perl met_context_window.pl BSMAP_out.txt 100

# output (columns: chromosome, start, end, sites, "Cs", "C+Ts", percent-
methylation)
maize_chr1_reference 0  100  4  23  31  0.741935483870968
maize_chr1_reference 100  200  6  52  70  0.742857142857143
maize_chr1_reference 200  300  6  70  95  0.736842105263158
maize_chr1_reference 300  400  2  27  34  0.794117647058823
maize_chr1_reference 400  500  6  94  114  0.824561403508772
```

9. Identify unmethylated regions.

   The final step requires classifying each 100 bp tile into one of six methylation categories. This step requires R and the R package *tidyverse*. The methylation categories include "missing data" (including "no data" and "no sites"), "RdDM," "heterochromatin," "CG-only," "unmethylated," or "intermediate". In this analysis, we are primarily interested in using this classification to identify the UMRs; however, users might also be interested in other types of methylation, which could be extracted from this same analysis strategy. We also suggest removing organelles from the data before proceeding with this step; however, in this example data, the organelle genomes have already been removed.

   This analysis is performed using a custom R script that we have provided: *Call-umrs.R*. Regions are classified according to the following hierarchy: tiles are classified as missing data if they have less than two cytosines in the relevant context or if there is less than the specified coverage threshold of reads (e.g., 3–5× coverage); RdDM if CHH methylation is greater than 15%; heterochromatin if CG and CHG methylation is 40% or greater; CG-only if CG methylation is greater than 40%; unmethylated if CG, CHG, and CHH are less than 10%; and intermediate if methylation is 10% or greater but less than 40%. Note that the levels of CHH methylation are hard coded in this script, while the level of CG and CHG are specified when calling the script. We have found these levels to be appropriate for a range of species; however, they could be adjusted if a genome has a different or unusual distribution, for example if CHH methylation is known to be higher.

   This script also requires a genome reference cytosine tile file that provides the number of cytosines that occur in each context for each tile. We have provided this file, *maize_chr1_reference_100 bp_tiles.bed*, for the analysis of the 20 Mb region in this example. Users will need to create this file to analyse a different plant genome; some reference genome files are linked in the git repository https://github.com/Bio-protocol/unmethylated-regions_UMR-extractor-WGBS/tree/master. The format of the file is shown below (columns: chromosome, start, end, #CG sites, #CHG sites, #CHH sites):

   ```
   maize_chr1_reference 0  100  4  6  29
   maize_chr1_reference 100  200  6  7  25
   maize_chr1_reference 200  300  6  4  36
   maize_chr1_reference 300  400  2  10  28
   maize_chr1_reference 400  500  6  2  31
   ```

   When calling the R script, the following arguments are required in this order; suggested default settings are indicated in the brackets:

   1. The reference genome cytosine tile file ("maize_chr1_reference_100 bp_tiles_sites_counts.txt").
   2. Minimum coverage (suggestion 3× or 5×).
   3. Minimum number of sites (suggestion 2).
   4. Minimum percent to be considered methylated (suggestion 40%).
   5. Maximum percent to be considered unmethylated (suggestion 10%).

   ```
   R -f Call-umrs.r \
   --args maize_chr1_reference_100bp_tiles_sites_counts.txt \
   3 \
   2 \
   0.4 \
   0.1
   ```

   An example of the output bed file is below (columns: chromosome, start, stop, methylation category).

   ```
   maize_chr1_reference 2900 3000 Unmethylated
   ```

```
maize_chr1_reference 3000 3100 Unmethylated
maize_chr1_reference 3100 3200 Unmethylated
maize_chr1_reference 3200 3300 Unmethylated
maize_chr1_reference 12500 12600 Unmethylated
```

The output of the R script is a bed file *UMTs.bed* that lists the coordinates of all the tiles that were categorised as unmethylated. In addition, a file called *mC_domains_cov_3_sites_2_MR_0.4_UMR_0.1_tiles_with_data.bed* is also produced by this script and provides a list of all regions that had sufficient data for UMR testing. The UMR file can now be sorted, and then adjacent tiles are merged to yield the final unmethylated regions.

```
sort -k1,1 -k2,2n UMTs.bed > UMTs_sorted.bed

bedtools merge -i UMTs_sorted.bed > all_UMRs.bed

awk '($3-$2) >= 299' all_UMRs.bed > UMRs.bed
```

The entire final output bed files are shown below (columns: chromosome, start, stop). The "all_UMRs.bed" file contains 6 UMRs; however, we highly recommend that the UMRs are further filtered to only retain UMRs that are 300 bp or longer in size. Some unmethylated regions of 300 bp or less could be functionally important. However, we have found that the majority of these small unmethylated regions (and there are a lot) lack evidence of functionality; for example, 99.5% did not have accessible chromatin [3]. However, further research is needed to develop methods to identify the potentially small minority of functionally important small (<300 bp) unmethylated regions.

```
# all_UMRs.bed
maize_chr1_reference 2900 3300
maize_chr1_reference 12500 12800
maize_chr1_reference 12900 13600
maize_chr1_reference 13800 13900
maize_chr1_reference 14000 14800
maize_chr1_reference 15200 15300

# UMRs.bed
maize_chr1_reference   2900    3300
maize_chr1_reference   12500   12800
maize_chr1_reference   12900   13600
maize_chr1_reference   14000   14800
```

Results interpretation

The key output from this analysis workflow is the bed file with the coordinates of the UMRs, "UMRs.bed." The bed file can be used for a number of downstream analyses. The total number, size distribution, and genomic location of UMRs can be analysed. Most diploid genomes analysed to date have approximately 100 Mb of UMRs in total across the whole genome, so it would be expected that the total number of UMRs should be around this number (or greater for polyploids). If no UMRs are detected, this would be surprising and would suggest that an error has occurred; users are advised to step backwards in the protocol to identify which step and output file may be empty or incomplete.

The output file of this example can also be visualised using IGV. In Figure 2 below, we show an IGV screenshot of this UMR bed file (track labelled "UMRs") along with the bigWig files that display the per-cytosine methylation (track labelled "methylation"). In the methylation track, we can see the three different coloured bars, which represent CG, CHG, and CHH methylation in each region of the 20 kb fragment. This has been aligned to the genome annotation for maize, so that we can see where genes are in relation to the methylation. The one gene within this 20 kb region is Zm0001d027232, which corresponds to an mRNA-hypothetical protein; as we can see, there is reduced methylation present where the gene is located. The analysis has identified five UMRs in this gene region and a sixth present downstream of the gene. This singular downstream UMR may represent a regulatory region containing cis-regulatory elements that could influence expression of the nearby gene, either as enhancers or silencers. Importantly, as highlighted in the figure, there are also other gaps in the methylation data upstream of the gene; however, rather than being unmethylated regions, these regions lack data as can be seen by the gaps in the "tiles with data" track.

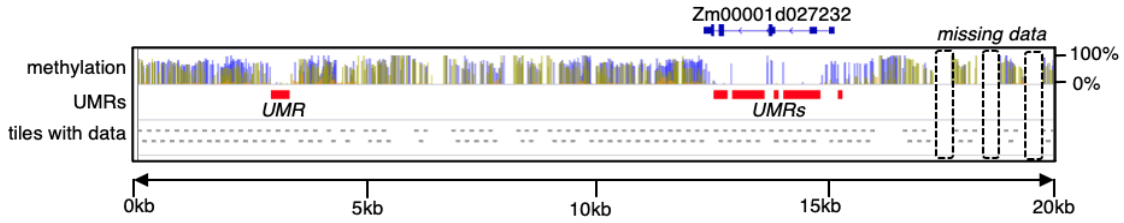

Figure 2. Example output of the unmethylated region (UMR)-calling pipeline. The output is viewed in IGV for a 20 kb section of the maize genome on chromosome one surrounding the gene Zm00001d027232. Bars in the "methylation" track (bigwig files) represent percent methylation (0%–100%) for each cytosine in the CG (blue), CHG (green), and CHG (orange) context. UMRs are marked by the large red rectangles (UMRs.bed file); there are several UMRs overlapping the gene locus and one UMR located 9 kb downstream of the gene. Tiles with sufficient data (coverage and minimum number of cytosines) are marked by the small grey rectangles; the dashed boxes mark examples of gaps in the methylation data that are not UMRs because these regions are instead missing data.

Acknowledgments

We wish to acknowledge the University of Queensland's Research Computing Centre (RCC) for its support in this research. PAC was supported by an ARC Discovery Early Career Researcher Award (DE200101748). This protocol was adapted from our previous work [3].

Competing interests

The authors declare no competing interests.

References

1. Moore, L. D., Le, T. and Fan, G. (2013). DNA methylation and its basic function. *Neuropsychopharmacology* 38(1): 23–38. https://doi.org/10.1038/npp.2012.112
2. Springer, N. M. and Schmitz, R. J. (2017). Exploiting induced and natural epigenetic variation for crop improvement. *Nat. Rev. Genet.* 18(9): 563–575. https://doi.org/10.1038/nrg.2017.45
3. Crisp, P. A., Marand, A. P., Noshay, J. M., Zhou, P., Lu, Z., Schmitz, R. J. and Springer, N. M. (2020). Stable unmethylated DNA demarcates expressed genes and their cis-regulatory space in plant genomes. *Proc. Natl.*

Acad. Sci. U.S.A. 117(38): 23991–24000. https://doi.org/10.1101/2020.05.21.109744

4. Harris, R. A., Wang, T., Coarfa, C., Nagarajan, R. P., Hong, C., Downey, S. L., Johnson, B. E., Fouse, S. D., Delaney, A., Zhao, Y., et al. (2010). Comparison of sequencing-based methods to profile DNA methylation and identification of monoallelic epigenetic modifications. *Nat. Biotechnol.* 28(10): 1097–1105. https://doi.org/10.1038/nbt.1682
5. Morrison, J., Koeman, J. M., Johnson, B. K., Foy, K. K., Beddows, I., Zhou, W., Chesla, D. W., Rossell, L. L., Siegwald, E. J., Adams, M., et al. (2021). Evaluation of whole-genome DNA methylation sequencing library preparation protocols. *Epigenetics and Chromatin* 14(1): e1186/s13072–021–00401–y. https://doi.org/10.1186/s13072-021-00401-y
6. Meissner, A. (2005). Reduced representation bisulfite sequencing for comparative high-resolution DNA methylation analysis. *Nucleic Acids Res.* 33(18): 5868–5877. https://doi.org/10.1093/nar/gki901
7. Weber, M., Davies, J. J., Wittig, D., Oakeley, E. J., Haase, M., Lam, W. L. and Schübeler, D. (2005). Chromosome-wide and promoter-specific analyses identify sites of differential DNA methylation in normal and transformed human cells. *Nat. Genet.* 37(8): 853–862. https://doi.org/10.1038/ng1598
8. Olova, N., Krueger, F., Andrews, S., Oxley, D., Berrens, R. V., Branco, M. R. and Reik, W. (2018). Comparison of whole-genome bisulfite sequencing library preparation strategies identifies sources of biases affecting DNA methylation data. *Genome Biol.* 19(1): 33. https://doi.org/10.1186/s13059-018-1408-2
9. Vaisvila, R., Ponnaluri, V. C., Sun, Z., Langhorst, B. W., Saleh, L., Guan, S., Dai, N., Campbell, M. A., Sexton, B. S., Marks, K., et al. (2021). Enzymatic methyl sequencing detects DNA methylation at single-base resolution from picograms of DNA. *Genome Res.* 31(7): 1280–1289. https://doi.org/10.1101/gr.266551.120
10. Liu, Y., Rosikiewicz, W., Pan, Z., Jillette, N., Wang, P., Taghbalout, A., Foox, J., Mason, C., Carroll, M., Cheng, A., et al. (2021). DNA methylation calling tools for Oxford Nanopore sequencing: a survey and human epigenome-wide evaluation. *Genome Biol.* 22(1): 295. https://doi.org/10.1101/2021.05.442849
11. Zhang, X., Yazaki, J., Sundaresan, A., Cokus, S., Chan, S. L., Chen, H., Henderson, I. R., Shinn, P., Pellegrini, M., Jacobsen, S. E., et al. (2006). Genome-wide High-Resolution Mapping and Functional Analysis of DNA Methylation in Arabidopsis. *Cell* 126(6): 1189–1201. https://doi.org/10.1016/j.cell.2006.08.003
12. Beck, D., Ben Maamar, M. and Skinner, M. K. (2021). Genome-wide CpG density and DNA methylation analysis method (MeDIP, RRBS, and WGBS) comparisons. *Epigenetics* 17(5): 518–530. https://doi.org/10.1080/15592294.2021.1924970
13. Grehl, C., Wagner, M., Lemnian, I., Glaser, B. and Grosse, I. (2020). Performance of Mapping Approaches for Whole-Genome Bisulfite Sequencing Data in Crop Plants. *Front. Plant Sci.* 11: e00176. https://doi.org/10.3389/fpls.2020.00176
14. Nunn, A., Otto, C., Stadler, P. F. and Langenberger, D. (2021). Comprehensive benchmarking of software for mapping whole genome bisulfite data: from read alignment to DNA methylation analysis. *Briefings Bioinf.* 22(5): e1093/bib/bbab021. https://doi.org/10.1093/bib/bbab021
15. Zhang, Y., Andrews, H., Eglitis-Sexton, J., Godwin, I., Tanurdžić, M. and Crisp, P. A. (2022). Epigenome guided crop improvement: current progress and future opportunities. *Emerging Top. Life Sci.* 6(2): 141–151. https://doi.org/10.1042/etls20210258
16. Olson, A., Klein, R. R., Dugas, D. V., Lu, Z., Regulski, M., Klein, P. E. and Ware, D. (2014). Expanding and Vetting Sorghum bicolor Gene Annotations through Transcriptome and Methylome Sequencing. *The Plant Genome* 7(2): e0025. https://doi.org/10.3835/plantgenome2013.08.0025
17. Oka, R., Zicola, J., Weber, B., Anderson, S. N., Hodgman, C., Gent, J. I., Wesselink, J. J., Springer, N. M., Hoefsloot, H. C. J., Turck, F., et al. (2017). Genome-wide mapping of transcriptional enhancer candidates using DNA and chromatin features in maize. *Genome Biol.* 18(1): 137. https://doi.org/10.1186/s13059-017-1273-4
18. Oka, R., Bliek, M., Hoefsloot, H. C. and Stam, M. (2020). In plants distal regulatory sequences overlap with unmethylated rather than low-methylated regions, in contrast to mammals. *bioRxiv:* 2020.2003.2024.005678. https://doi.org/10.1101/2020.03.24.005678
19. Lu, Z., Marand, A. P., Ricci, W. A., Ethridge, C. L., Zhang, X. and Schmitz, R. J. (2019). The prevalence, evolution and chromatin signatures of plant regulatory elements. *Nat. Plants* 5(12): 1250–1259. https://doi.org/10.1038/s41477-019-0548-z
20. Ricci, W. A., Lu, Z., Ji, L., Marand, A. P., Ethridge, C. L., Murphy, N. G., Noshay, J. M., Galli, M., Mejía-Guerra, M. K., Colomé-Tatché, M., et al. (2019). Widespread long-range cis-regulatory elements in the maize

genome. *Nat. Plants* 5(12): 1237–1249. https://doi.org/10.1038/s41477-019-0547-0

21. Hufford, M. B., Seetharam, A. S., Woodhouse, M. R., Chougule, K. M., Ou, S., Liu, J., Ricci, W. A., Guo, T., Olson, A., Qiu, Y., et al. (2021). De novo assembly, annotation, and comparative analysis of 26 diverse maize genomes. *Science* 373(6555): 655–662. https://doi.org/10.1126/science.abg5289
22. Ricci, W. (2021). Unmethylated Regions Encompass the Functional Space Within The Maize Genome. *bioRxiv*: 2021.2004.2021.425900. https://doi.org/10.1101/2021.04.21.425900
23. Crisp, P. A., Bhatnagar-Mathur, P., Hundleby, P., Godwin, I. D., Waterhouse, P. M. and Hickey, L. T. (2022). Beyond the gene: epigenetic and cis-regulatory targets offer new breeding potential for the future. *Curr. Opin. Biotechnol.* 73: 88–94. https://doi.org/10.1016/j.copbio.2021.07.008
24. Crisp, P. A., Noshay, J. M., Anderson, S. N. and Springer, N. M. (2019). Opportunities to Use DNA Methylation to Distil Functional Elements in Large Crop Genomes. *Mol. Plant* 12(3): 282–284. https://doi.org/10.1016/j.molp.2019.02.006
25. Schmitz, R. J., Grotewold, E. and Stam, M. (2021). Cis-regulatory sequences in plants: Their importance, discovery, and future challenges. *Plant Cell* 34(2): 718–741. https://doi.org/10.1093/plcell/koab281
26. Swinnen, G., Goossens, A. and Pauwels, L. (2016). Lessons from Domestication: Targeting Cis -Regulatory Elements for Crop Improvement. *Trends Plant Sci.* 21(6): 506–515. https://doi.org/10.1016/j.tplants.2016.01.014
27. Zhang, R. X., Li, B. B., Yang, Z. G., Huang, J. Q., Sun, W. H., Bhanbhro, N., Liu, W. T. and Chen, K. M. (2022). Dissecting Plant Gene Functions Using CRISPR Toolsets for Crop Improvement. *J. Agric. Food. Chem.* 70(24): 7343–7359. https://doi.org/10.1021/acs.jafc.2c01754
28. Song, X., Meng, X., Guo, H., Cheng, Q., Jing, Y., Chen, M., Liu, G., Wang, B., Wang, Y., Li, J., et al. (2022). Targeting a gene regulatory element enhances rice grain yield by decoupling panicle number and size. *Nat. Biotechnol.* 40(9): 1403–1411. https://doi.org/10.1038/s41587-022-01281-7
29. Wang, S., Li, S., Liu, Q., Wu, K., Zhang, J., Wang, S., Wang, Y., Chen, X., Zhang, Y., Gao, C., et al. (2015). The OsSPL16-GW7 regulatory module determines grain shape and simultaneously improves rice yield and grain quality. *Nat. Genet.* 47(8): 949–954. https://doi.org/10.1038/ng.3352
30. Liu, L., Gallagher, J., Arevalo, E. D., Chen, R., Skopelitis, T., Wu, Q., Bartlett, M. and Jackson, D. (2021). Enhancing grain-yield-related traits by CRISPR–Cas9 promoter editing of maize CLE genes. *Nat. Plants* 7(3): 287–294. https://doi.org/10.1038/s41477-021-00858-5
31. Muños, S., Ranc, N., Botton, E., Bérard, A., Rolland, S., Duffé, P., Carretero, Y., Le Paslier, M. C., Delalande, C., Bouzayen, M., et al. (2011). Increase in Tomato Locule Number Is Controlled by Two Single-Nucleotide Polymorphisms Located Near WUSCHEL. *Plant Physiol.* 156(4): 2244–2254. https://doi.org/10.1104/pp.111.173997
32. Rodríguez-Leal, D., Lemmon, Z. H., Man, J., Bartlett, M. E. and Lippman, Z. B. (2017). Engineering Quantitative Trait Variation for Crop Improvement by Genome Editing. *Cell* 171(2): 470–480.e8. https://doi.org/10.1016/j.cell.2017.08.030
33. Wang, X., Aguirre, L., Rodríguez-Leal, D., Hendelman, A., Benoit, M. and Lippman, Z. B. (2021). Dissecting cis-regulatory control of quantitative trait variation in a plant stem cell circuit. *Nat. Plants* 7(4): 419–427. https://doi.org/10.1038/s41477-021-00898-x
34. Wrightsman, T., Marand, A. P., Crisp, P. A., Springer, N. M. and Buckler, E. S. (2021). Modeling chromatin state from sequence across angiosperms using recurrent convolutional neural networks. *bioRxiv:* 2021.2011.2011.468292. https://doi.org/10.1101/2021.11.11.468292
35. Krueger, F., James, F., Ewels, P., Afyounian, E. and Schuster-Boeckler, B. (2021). FelixKrueger/TrimGalore: v0.6.7 - DOI via Zenodo.
36. Martin, M. (2011). Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMB net. J.* 17(1): 10. https://doi.org/10.14806/ej.17.1.200
37. Andrews, S. (2010). FastQC: a quality control tool for high throughput sequence data. Babraham Bioinformatics, Babraham Institute, Cambridge, United Kingdom.
38. Xi, Y. and Li, W. (2009). BSMAP: whole genome bisulfite sequence MAPping program. *BMC Bioinf.* 10(1): e1186/1471–2105–10–232. https://doi.org/10.1186/1471-2105-10-232
39. Danecek, P., Bonfield, J. K., Liddle, J., Marshall, J., Ohan, V., Pollard, M. O., Whitwham, A., Keane, T., McCarthy, S. A., Davies, R. M., et al. (2021). Twelve years of SAMtools and BCFtools. *GigaScience* 10(2):

e1093/gigascience/giab008. https://doi.org/10.1093/gigascience/giab008
40. Barnett, D. W., Garrison, E. K., Quinlan, A. R., Strömberg, M. P. and Marth, G. T. (2011). BamTools: a C++ API and toolkit for analyzing and managing BAM files. *Bioinformatics* 27(12): 1691–1692. https://doi.org/10.1093/bioinformatics/btr174
41. Addison, W. (2005). THE Java™ programming language, fourth edition. https://www.acs.ase.ro/Media/Default/documents/java/ClaudiuVinte/books/ArnoldGoslingHolmes06.pdf
42. Jun, G., Wing, M. K., Abecasis, G. R. and Kang, H. M. (2015). An efficient and scalable analysis framework for variant extraction and refinement from population-scale DNA sequence data. *Genome Res.* 25(6): 918–925. https://doi.org/10.1101/gr.176552.114
43. Van Rossum and Drake. (1995). Python reference manual. /project/gwydion-1/OldFiles/OldFiles/python/Doc/ref
44. Wall, L., Christiansen, T. and Orwant, J. (2000). Programming Perl. O'Reilly Media. https://play.google.com/store/books/details?id=xx5JBSqcQzIC
45. Robinson, J. T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E. S., Getz, G. and Mesirov, J. P. (2011). Integrative genomics viewer. *Nat. Biotechnol.* 29(1): 24–26. https://doi.org/10.1038/nbt.1754
46. Feng, H. and Wu, H. (2019). Differential methylation analysis for bisulfite sequencing using DSS. *Quant. Biol.* 7(4): 327–334. https://doi.org/10.1007/s40484-019-0183-8
47. Akalin, A., Kormaksson, M., Li, S., Garrett-Bakelman, F. E., Figueroa, M. E., Melnick, A. and Mason, C. E. (2012). methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles. *Genome Biol.* 13(10): R87. https://doi.org/10.1186/gb-2012-13-10-r87
48. Hüther, P., Hagmann, J., Nunn, A., Kakoulidou, I., Pisupati, R., Langenberger, D., Weigel, D., Johannes, F., Schultheiss, S. J., Becker, C., et al. (2022). MethylScore, a pipeline for accurate and context-aware identification of differentially methylated regions from population-scale plant WGBS data. *bioRxiv :* e475031. https://doi.org/10.1101/2022.01.06.475031

Supplementary information

Data and code availability: All data and code have been deposited to GitHub: https://github.com/Bio-protocol/unmethylated-regions_UMR-extractor-WGBS/tree/master