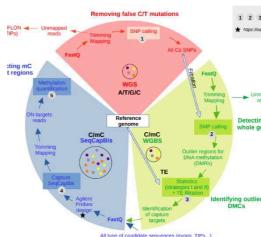


Jun 20, 2024

Bioinformatics manual for population epigenomics combining whole-genome and target genome sequencing

DOI

[dx.doi.org/10.17504/protocols.io.8epv5xw4ng1b/v1](https://doi.org/10.17504/protocols.io.8epv5xw4ng1b/v1)



Odile Rogier¹, Isabelle Lesur Kupin^{2,3}, Mamadou Dia Sow^{4,5}, Christophe Boury², Alexandre Duplan^{1,5}, Abel Garnier⁶, Abdeljalil Senhaji rachik^{1,2}, Peter Civan⁴, Josquin Daron⁷, Alain Delaunay⁵, Ludovic Duvaux², Vanina Benoit¹, Erwan Guichoux², Gregoire Le Provost², Edmond Sanou⁸, Christophe Ambroise⁸, Christophe Plomion², Jérôme Salse⁴, Vincent Segura^{1,9}, Jorg Tost⁶, Stéphane Maury⁵

¹INRAE, ONF, BioForA, F-45075 Orléans, France.; ²INRAE, Univ. Bordeaux, BIOGECO, F-33610 Cestas, France.;

³HelixVenture, F-33700 Mérignac, France.;

⁴INRAE/UCA UMR GDEC 1095. 5 Chemin de Beaulieu, F-63100 Clermont Ferrand, France.;

⁵LBLGC, INRAE, Université d'Orléans, EA 1207 USC 1328, F-45067 Orleans, France.;

⁶Centre National de Recherche en Génomique Humaine, CEA-Institut de Biologie François Jacob, Université Paris-Saclay, F-91000 Evry, France.;

⁷Institut Pasteur, Université Paris Cité, CNRS UMR2000, Insect-Virus Interactions Unit, F- 75724 Paris, France.;

⁸LaMME, 23 Bd. de France, F-91037 Evry Cedex, France.;

⁹UMR AGAP Institut, Univ Montpellier, CIRAD, INRAE, Institut Agro Montpellier, F- 34398 Montpellier, France.



Odile Rogier

INRAE

OPEN  ACCESS



DOI: [dx.doi.org/10.17504/protocols.io.8epv5xw4ng1b/v1](https://doi.org/10.17504/protocols.io.8epv5xw4ng1b/v1)

External link: <https://epitree-project.hub.inrae.fr/>

Protocol Citation: Odile Rogier, Isabelle Lesur Kupin, Mamadou Dia Sow, Christophe Boury, Alexandre Duplan, Abel Garnier, Abdeljalil Senhaji rachik, Peter Civan, Josquin Daron, Alain Delaunay, Ludovic Duvaux, Vanina Benoit, Erwan Guichoux, Gregoire Le Provost, Edmond Sanou, Christophe Ambroise, Christophe Plomion, Jérôme Salse, Vincent Segura, Jorg Tost, Stéphane Maury 2024. Bioinformatics manual for population epigenomics combining whole-genome and target genome sequencing.

protocols.io <https://dx.doi.org/10.17504/protocols.io.8epv5xw4ng1b/v1>

Manuscript citation:

A Strategy for Studying Epigenetic Diversity in Natural Populations: Proof of Concept in Poplar and Oak
Lesur I., Rogier O., Sow M-D., Boury C., Duplan A., Garnier A., Senhaji-Rachik A., Civan P., Daron J., Delaunay A., Duvaux L., Benoit V., Guichoux E., Le Provost G., Sanou E., Ambroise C., Plomion C., Salse J., Segura V., Tost J., Maury S. 2024. *J. Exp. Bot.*

License: This is an open access protocol distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

Protocol status: Working

We use this protocol and it's working

Created: March 14, 2024

Last Modified: June 20, 2024

Protocol Integer ID: 96705

Keywords: DNA Methylation, Epigenetics, Epigenomics, Methylome, Natural population, Oak, Poplar, Transposon Insertion Polymorphism, SeqCapBis, WGS, WGBS

Funders Acknowledgements:

ANR EPITREE

Grant ID: ANR-17-CE32-0009-01

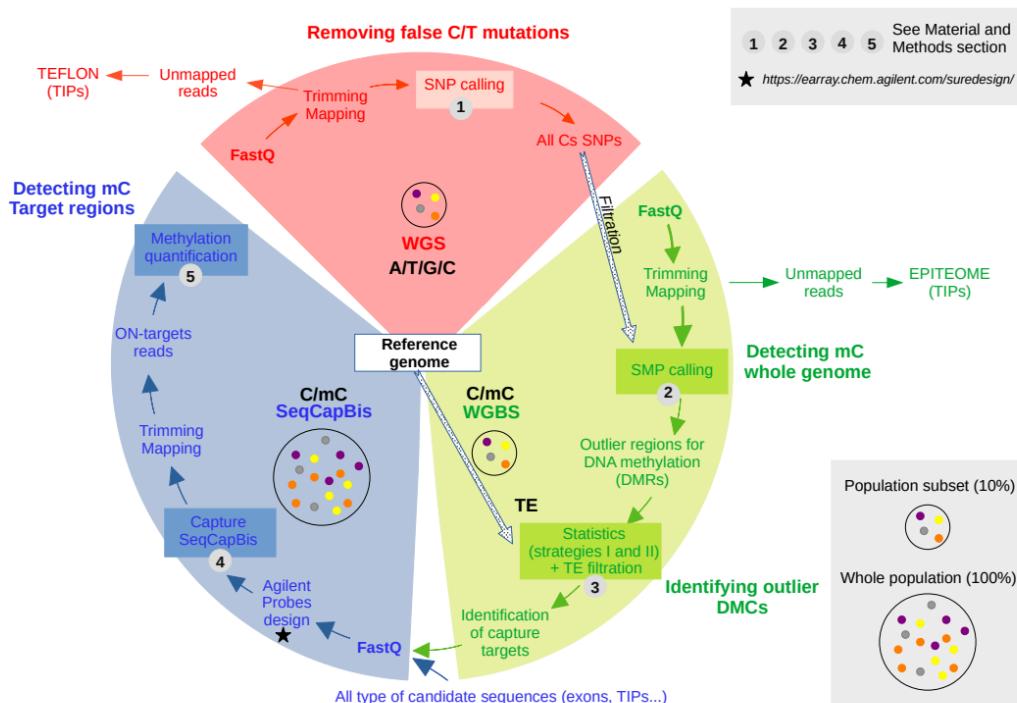
Abstract

We developed a strategy and a workflow for quantifying epigenetic diversity in natural populations combining whole genome and targeted capture sequencing for DNA methylation.

We first identified regions of highly variable DNA methylation in a representative subset of genotypes representative of the biological diversity in the population by WGBS. We then analysed the variations of DNA methylation in these targeted regions at the population level by Sequencing Capture Bisulphite (SeqCapBis).

Whole Genome Sequencing - Removing false C/T mutations

- 1 A preliminary Whole Genome Sequencing (WGS) step was considered for filtering purposes, to prevent C/T Single Nucleotide Polymorphisms (SNP) being interpreted as bisulfite conversions of unmethylated sites (i.e. false-positive calls). However, this C/T SNPs identification step is not required to study epigenetics levels along genomes.



Strategy for population epigenomics combining whole-genome and target genome sequencing.

2 Trimming

Software

Trimmomatic

NAME

<https://doi.org/10.1093/bioinformatics/btu170>

DEVELOPER

<http://www.usadellab.org/cms/?page=trimmomatic>

SOURCE LINK

Publication: Bolger et al., 2014
Version: 0.38
Github: <https://github.com/usadellab/Trimmomatic>

CITATION

Bolger AM, Lohse M, Usadel B (2014). Trimmomatic: a flexible trimmer for Illumina sequence data..

LINK

<https://doi.org/10.1093/bioinformatics/btu170>

Command

```
java -Xmx4G -jar trimmomatic.jar PE -threads 12 file_R1.fastq.gz
file_R2.fastq.gz
file_trimmed_1.fastq.gz  file_unpaired_1.fastq.gz
file_trimmed_2.fastq.gz
file_unpaired_2.fastq.gz ILLUMINACLIP:TruSeq3-PE.fa:2:30:10 LEADING:3
TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:35
```

3 Mapping

Software

BWA NAME

Unix OS

Li, H., Durbin, R. DEVELOPER

<http://bio-bwa.sourceforge.net/> SOURCE LINK

Publication: Li H, 2013

Version: 0.7.17

CITATION

Heng Li (2013). Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. arXiv:1303.3997 [q-bio.GN].

LINK

<https://doi.org/10.48550/arXiv.1303.3997>

Poplar genome: *Populus trichocarpa* v3.1

Publication: Tuskan GA et al., 2006.

CITATION

Tuskan GA, Difazio S, Jansson S, Bohlmann J, Grigoriev I, Hellsten U, Putnam N, Ralph S, Rombauts S, Salamov A, Schein J, Sterck L, Aerts A, Bhalerao RR, Bhalerao RP, Blaudez D, Boerjan W, Brun A, Brunner A, Busov V, Campbell M, Carlson J, Chalot M, Chapman J, Chen GL, Cooper D, Coutinho PM, Couturier J, Covert S, Cronk Q, Cunningham R, Davis J, Degroeve S, Déjardin A, Depamphilis C, Detter J, Dirks B, Dubchak I, Duplessis S, Ehlting J, Ellis B, Gendler K, Goodstein D, Gribskov M, Grimwood J, Groover A, Gunter L, Hamberger B, Heinze B, Helariutta Y, Henrissat B, Holligan D, Holt R, Huang W, Islam-Faridi N, Jones S, Jones-Rhoades M, Jorgensen R, Joshi C, Kangasjärvi J, Karlsson J, Kelleher C, Kirkpatrick R, Kirst M, Kohler A, Kalluri U, Larimer F, Leebens-Mack J, Leplé JC, Locascio P, Lou Y, Lucas S, Martin F, Montanini B, Napoli C, Nelson DR, Nelson C, Nieminen K, Nilsson O, Pereda V, Peter G, Philippe R, Pilate G, Poliakov A, Razumovskaya J, Richardson P, Rinaldi C, Ritland K, Rouzé P, Ryaboy D, Schmutz J, Schrader J, Seegerman B, Shin H, Siddiqui A, Sterky F, Terry A, Tsai CJ, Uberbacher E, Unneberg P, Vahala J, Wall K, Wessler S, Yang G, Yin T, Douglas C, Marra M, Sandberg G, Van de Peer Y, Rokhsar D (2006). The genome of black cottonwood, *Populus trichocarpa* (Torr. & Gray)..

LINK

<https://doi.org/>

Command

```
bwa mem genome.fa file_trimmed_1.fastq.gz file_trimmed_2.fastq.gz -t  
12 -M > file.sam
```

3.1 Mapping adjustments for *Q. petraea*

Oak genome: *Quercus robur* Haplome V2.3

Publication: Plomion C et al., 2018

CITATION

Plomion C, Aury JM, Amselem J, Leroy T, Murat F, Duplessis S, Faye S, Francillonne N, Labadie K, Le Provost G, Lesur I, Bartholomé J, Faivre-Rampant P, Kohler A, Leplé JC, Chantret N, Chen J, Diévert A, Alaeitabar T, Barbe V, Belser C, Bergès H, Bodénès C, Bogaert-Triboulot MB, Bouffaud ML, Brachi B, Chancerel E, Cohen D, Couloux A, Da Silva C, Dossat C, Ehrenmann F, Gaspin C, Grima-Pettenati J, Guichoux E, Hecker A, Herrmann S, Hugueney P, Hummel I, Klopp C, Lalanne C, Lascoux M, Lasserre E, Lemainque A, Desprez-Loustau ML, Luyten I, Madoui MA, Mangenot S, Marchal C, Maumus F, Mercier J, Michotey C, Panaud O, Picault N, Rouhier N, Rué O, Rustenholz C, Salin F, Soler M, Tarkka M, Velt A, Zanne AE, Martin F, Wincker P, Quesneville H, Kremer A, Salse J (2018). Oak genome reveals facets of long lifespan..

LINK

<https://doi.org/10.1038/s41477-018-0172-3>

3.2 Mapping conversion, sorting & statistics

Software

SAMtools

NAME

Li et al.

DEVELOPER

<https://github.com/samtools/>

SOURCE LINK

Publication: Danecek et al., 2021

Version: 1.8

Github: <https://github.com/samtools/samtools>

CITATION

Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H (2021). Twelve years of SAMtools and BCFtools..

LINK

<https://doi.org/10.1093/gigascience/giab008>

Command

```
samtools view -Sb file_trimmed.sam > file_trimmed.bam  
samtools sort file_trimmed.bam -o file_trimmed_sorted.bam  
samtools flagstat file_trimmed_sorted.bam > file_flagstats.txt  
samtools stats file_trimmed_sorted.bam > file_stats.txt
```

4 Variant calling

4.1 Adjustment for *Q. petraea*: Digital normalization

Computational limitations associated with GATK and FreeBayes due to the very deep sequencing in oak (100X on average) necessitated a reduction of the complexity of each dataset. To reduce redundancy within the WGS dataset, we randomly downsampled sequencing reads over genome regions that are over-covered.

Software

KHMER

Linux

NAME

Titus Brown

OS

DEVELOPER

<https://khmer.readthedocs.io/en/latest/>

SOURCE LINK

Publication: Crusoe et al., 2015

Version: 2.1.2

Github: <https://github.com/dib-lab/khmer>

CITATION

Crusoe MR, Alameldin HF, Awad S, Boucher E, Caldwell A, Cartwright R, Charbonneau A, Constantinides B, Edvenson G, Fay S, Fenton J, Fenzl T, Fish J, Garcia-Gutierrez L, Garland P, Gluck J, González I, Guermond S, Guo J, Gupta A, Herr JR, Howe A, Hyer A, Härpfer A, Irber L, Kidd R, Lin D, Lippi J, Mansour T, McA'Nulty P, McDonald E, Mizzi J, Murray KD, Nahum JR, Nanlohy K, Nederbragt AJ, Ortiz-Zuazaga H, Ory J, Pell J, Pepe-Ranney C, Russ ZN, Schwarz E, Scott C, Seaman J, Sievert S, Simpson J, Skennerton CT, Spencer J, Srinivasan R, Standage D, Stapleton JA, Steinman SR, Stein J, Taylor B, Trimble W, Wiencko HL, Wright M, Wyss B, Zhang Q, Zyme E, Brown CT (2015). The khmer software package: enabling efficient nucleotide sequence analysis..

LINK

<https://doi.org/10.12688/f1000research.6924.1>

Step1: Interleave reads

Parameters: Python-3.6.3

Command

```
interleave-reads.py file_R1.fastq file_R2.fastq -o  
file_interleave_R1_R2.fastq
```

Step2: Digital normalization

Parameters: Python-3.6.3; -k 20 → kmer size = 20bp; -C 30 → maximal coverage; -N 4 -x 4e9 → 16Gb

Command

```
normalize-by-median.py -k 20 -C 30 -N 4 -x 4e9  
file_interleave_R1_R2.fastq -o file_normalize_by_median_R1_R2.fastq
```

Step3: Paired reads extraction

Parameters: Python-3.6.3

Command

```
extract-paired-reads.py file_normalize_by_median_R1_R2.fastq -f --  
output-paired file_diginorm_paired --output-single  
file_diginorm_single
```

4.2 Duplicates removing

Software

picardtools

NAME

Publication: "Picard Toolkit." 2019. Broad Institute, GitHub Repository.

<https://broadinstitute.github.io/picard/>; Broad Institute

Version: 2.18.2

Github: <https://github.com/broadinstitute/picard>

Command

```
java -Xmx16g -jar picard.jar MarkDuplicates I=file_trimmed_sorted.bam  
O=file_trimmed_rmdup.bam CREATE_INDEX=true  
REMOVE_DUPLICATES=true M=file_output.metrics
```

4.3 Variant Caller 1: GATK (Genome Analysis ToolKit)

Software

GATK

NAME

Publication: McKenna et al., 2010

Version: 4.0.11.1

Github: <https://github.com/broadinstitute/gatk>

Poplar genome: *Populus trichocarpa* v3.1

CITATION

McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, DePristo MA (2010). The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data..

LINK

<https://doi.org/10.1101/gr.107524.110>

Command

```
## HaplotypeCaller
gatk --java-options "-Xmx16G" HaplotypeCaller -R genome.fa -I
file_trimmed_sorted_rmdup.bam -ERC GVCF -O
file_trimmed_sorted_rmdup.g.vcf
## GenomicsDBImport
gatk --java-options "-Xmx96G -Xms96G" GenomicsDBImport -V
file1_trimmed_sorted_rmdup.g.vcf -V file2_trimmed_sorted_rmdup.g.vcf -
-genomicsdb-workspace-path my_database -L list_Chrtscuff.list --batch-
size 50 -ip 500
## GenotypeGVCFs
gatk GenotypeGVCFs -R genome.fa -V gendb://my_database -new-qual
true -O all_trimmed_sorted_rmdup_gVCF_GATK.snps.indels.vcf
```

4.4 GATK adjustments for *Q. petraea*

Version: GATK 3.8

Download: https://console.cloud.google.com/storage/browser/_details/gatk-software/package-archive/gatk/GenomeAnalysisTK-3.8-0-ge9d806836.tar.bz2;tab=live_object

Oak reference genome: *Quercus robur* Haplome V2.3

Parameters: java 1.8.0_72 ; HaplotypeCaller; GenotypeGVCFs

Command

```
#HaplotypeCaller  
GATK -R haplome_v2.3.fa -T HaplotypeCaller -nct 20 -I  
sample1_trimmed_vs_haploV23.bam -I sample2_trimmed_vs_haploV23.bam -I  
sample3_trimmed_vs_haploV23.bam -I sample4_trimmed_vs_haploV23.bam -I  
sample5_trimmed_vs_haploV23.bam -I sample6_trimmed_vs_haploV23.bam -  
I sample7_trimmed_vs_haploV23.bam -I sample8_trimmed_vs_haploV23.bam -  
I sample9_trimmed_vs_haploV23.bam -I sample9_trimmed_vs_haploV23.bam  
--emitRefConfidence GVCF -o gatk_nct20_slurm_1node-c20_snps.vcf  
  
#GenotypeGVCFs  
GATK -T GenotypeGVCFs -R haplome_v2.3.fa --variant sample1.vcf --  
variant sample2.vcf --variant sample3.vcf --variant sample4.  
vcf --variant sample5.vcf --variant sample6.vcf --variant sample7.vcf  
--variant sample8.vcf --variant sample9.vcf --variant sample10.vcf -o  
gatk_all10samples_SNPs.vcf
```

4.5 Variant Caller 2: samtools / bcftools

Software

SAMtools

NAME

Linux

OS

Wellcome Trust Sanger Institute

DEVELOPER

<https://github.com/samtools/samtools>

SOURCE LINK

Publication: Danecek et al., 2021

Version: 1.8

Github: <https://github.com/samtools/samtools>Poplar genome: *Populus trichocarpa* v3.1

CITATION

Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H (2021). Twelve years of SAMtools and BCFtools..

LINK

<https://doi.org/10.1093/gigascience/giab008>

Software

bcftools

NAME

<https://github.com/samtools/bcftools>

SOURCE LINK

Publication: Li H, 2011

Version: 1.8

Github: <https://github.com/samtools/bcftools>

CITATION

Li H (2011). A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data..

LINK

<https://doi.org/10.1093/bioinformatics/btr509>

Command

```
samtools mpileup -uf genome.fa  
mapping_file_sort_without_duplicate.bam | bcftools call -mv -Oz >  
file_bcftools_noduplicate.vcf.gz
```

4.6 **bcftools adjustments for *Q. petraea***

Oak genome: *Q. robur* haplome V2.3

bcftools version: 1.6

Download: <https://sourceforge.net/projects/samtools/files/samtools/1.6/>

4.7 **Variant Caller 3: FreeBayes**

Software

freebayes

NAME

Garrison and Marth

DEVELOPER

<https://github.com/freebayes/freebayes>

SOURCE LINK

Publication: Garrison and Marth, 2012

Version: 1.2.0-2

Github: <https://github.com/freebayes/freebayes>

CITATION

Erik Garrison and Gabor Marth (2012). Haplotype-based variant detection from short-read sequencing. arXiv preprint arXiv:1207.3907 [q-bio.GN] 2012.

LINK

<https://doi.org/10.48550/arXiv.1207.3907>

Poplar genome: *Populus trichocarpa* v3.1

Oak genome: *Q. robur* haplome V2.3

Command

```
freebayes -f genome.fa all_samples.bam > freebayes_all_samples.vcf
```

4.8 SNP filtering

For poplar, we considered only biallelic intra-nigra SNPs with quality threshold ≥ 30 .

Software

VCFtools

NAME

Adam Auton, Petr Danecek, Anthony Marcketta DEVELOPER

https://vcftools.github.io/man_latest.html

SOURCE LINK

Publication: Danecek et al., 2011

Version: 0.1.15

Github: https://vcftools.github.io/man_latest.html

CITATION

Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, Handsaker RE, Lunter G, Marth GT, Sherry ST, McVean G, Durbin R, 1000 Genomes Project Analysis Group (2011). The variant call format and VCFtools..

LINK

<https://doi.org/10.1093/bioinformatics/btr330>

Command

```
vcftools --vcf all_tool.snps.indels.vcf --out all_filtered_tool.vcf --remove-indels --max-alleles 2 --min-alleles 2 --minQ 30 --recode --recode-INFO-all
```

For oak, we considered bi-allelic SNPs, depth ≥ 20 , maf $\geq 30\%$ and $\leq 70\%$

4.9 SNP identification

Only SNPs identified by at least 2 callers were selected to obtain the final set of SNPs.

Software

bcftools

NAME

<https://github.com/samtools/bcftools>

SOURCE LINK

Publication: Danecek P, et al. 2021

Version: 1.8

Github: <https://github.com/samtools/bcftools>

CITATION

Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwam A, Keane T, McCarthy SA, Davies RM, Li H (2021). Twelve years of SAMtools and BCFtools..

LINK

<https://doi.org/10.1093/gigascience/giab008>

Parameters: tabix-0.2.5, samtools-1.8, bcftools-1.8

Command

```
bcftools index sample1_diginorm_gatk3.8_depth20_maf30.vcf.gz  
bcftools index sample1_diginorm_FreeBayes_depth20_maf30.vcf.gz  
bcftools index sample1_samtools_depth20_maf30.vcf.gz  
  
bcftools isec -n +3 sample1_diginorm_gatk3.8_depth20_maf30.vcf.gz  
sample1_diginorm_FreeBayes_depth20_maf30.vcf.gz  
sample1_samtools_depth20_maf30.vcf.gz -O v -o  
common_SNPs_sample1_GATK_FreeBayes_samtools_depth20_maf30_bcftools.txt
```

5 Selection of C/T SNP

SMPs colocalizing with a C/T SNP (see the WGS and SNP detection section of the manuscript) will be removed at step #7 "SMPs filtering".

Whole Genome Bisulfite Sequencing - Detecting mC whole genome and Identifying outlier DMCs

6 Galaxy pipeline

SMPs were identified with the GALAXY (The Galaxy Community, 2022) pipeline (Dugé de Bernonville et al., 2022; Sow et al., 2023).

CITATION

Dugé de Bernonville T, Daviaud C, Chaparro C, Tost J, Maury S (2022). From Methylome to Integrative Analysis of Tissue Specificity..

LINK

https://doi.org/10.1007/978-1-0716-2349-7_16

CITATION

Sow MD, Rogier O, Lesur I, Daviaud C, Mardoc E, Sanou E, Duvaux L, Civan P, Delaunay A, Lesage-Descausses MC, Benoit V, Le-Jan I, Buret C, Besse C, Durufle H, Fichot R, Le-Provost G, Guichoux E, Boury C, Garnier A, Senhaji-Rachik A, Jorge V, Ambroise C, Tost J, Plomion C, Segura V, Maury S, Salse J (2023). Epigenetic Variation in Tree Evolution: a case study in black poplar (*Populus nigra*). bioRxiv 2023.07.16.549253.

LINK

<https://doi.org/10.1101/2023.07.16.549253>

Following Sow et al., 2023:



mC detection using the Galaxy pipeline

6.1 Trimming

Software

TrimGalore

NAME

Felix Krueger

DEVELOPER

<https://github.com/FelixKrueger/TrimGalore>

SOURCE LINK

Publication: Krueger F et al., 2023. FelixKrueger/TrimGalore: v0.4.3.1

Version: v0.4.3.1

Github: <https://github.com/FelixKrueger/TrimGalore>

Parameters: --paired read1.fastq read2.fastq --clip_R1 10 --clip_R2 30

CITATION

Felix Krueger; Frankie James; Phil Ewels; Ebrahim Afyounian; Michael Weinstein; Benjamin Schuster-Boeckler; Gert Hulselmans; sclamons (2023).

FelixKrueger/TrimGalore: v0.6.10. Zenodo.

LINK

<https://doi.org/10.5281/zenodo.5127898>

6.2 **Mapping**

Software

BSMAP

NAME

<https://github.com/genome-vendor/bsmap/>

SOURCE LINK

Publication: Xi Y and Li W, 2009

Version: v1.0.0

Github: <https://github.com/genome-vendor/bsmap/>

Parameters: default options

CITATION

Xi Y, Li W (2009). BSMAP: whole genome bisulfite sequence MAPping program..

LINK

<https://doi.org/10.1186/1471-2105-10-232>

Poplar genome: *Populus trichocarpa* v3.1

Mapping adjustments for *Q. petraea*

Oak genome: *Quercus robur* Haplome V2.3

6.3 **Methylation calling (SMP)**

Software

BSMAP methylation caller

NAME

Greg Zynda

DEVELOPER

Publication: Xi Y and Li W, 2009

Version: v1.0.0

Github: <https://github.com/genome-vendor/bsmap/>

CITATION

Xi Y, Li W (2009). BSMAP: whole genome bisulfite sequence MAPping program..

LINK

<https://doi.org/10.1186/1471-2105-10-232>Poplar genome: *Populus trichocarpa* v3.1

Command

```
methratio.py --ref ref_genome.fa --zero-meth TRUE --trim-fillin 2 --
combine-CpG --min-depth 8 --context all bsmap_sample*.sam
```

Mapping adjustments for *Q. petraea*Oak genome: *Quercus robur* Haplome V2.3

7

SMP filtering

Each methylation context (CpG, CHG, CHH) was considered separately.

Software

methylKit

NAME

Alexander Blume

DEVELOPER

<https://github.com/al2na/methylKit/releases>

SOURCE LINK

Publication: Akalin et al., 2012

Version: Methylkit R package v0.99.2

Github: <https://github.com/al2na/methylKit/releases>Site: <https://bioconductor.org/packages/release/bioc/html/methylKit.html>

Parameters: R (v3.5.1), library(methylKit)

CITATION

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE (2012). methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles..

LINK

<https://doi.org/10.1186/gb-2012-13-10-r87>

Step1: Forward and reverse strands were merged for the CG context only and 30% missing data were tolerated for each context.

Command

```
meth.CpG <- unite(CpG, destrand = TRUE, min.per.group = 7L)
meth.CHG <- unite(CHG, destrand = FALSE, min.per.group = 7L)
meth.CHH <- unite(CHH, destrand = FALSE, min.per.group = 7L)
```

Step2: Positions corresponding to C/T SNPs were removed.

Command

```
SNPdat <- read.delim("SNP_file.txt", header = F)

#with SNP_file.txt:
#  ScaffoldID      position      allele1      allele2

SNPdat$Scaff_Pos <- paste(SNPdat$Scaff, SNPdat$Pos, sep = "_")
SNPdat$SNP <- paste(SNPdat$Ref, SNPdat$Alt, sep = "/")
MethPos2 <- paste(meth.CpG2$chr, meth.CpG2$start, sep = "_")
MethPosMatchSNP2 <- which(MethPos2 %in% SNPdat$Scaff_Pos)
SNPMeth2 <- subset(SNPdat, Scaff_Pos %in% MethPos2[MethPosMatchSNP2])
SNPMeth0k <- subset(SNPMeth2, SNP == "C/T")
CpG.posOK2 <- select(meth.CpG2, which (!MethPos2 %in%
SNPMeth0k$Scaff_Pos))
```

Step3: A minimum coverage of 7X per sample was considered.

Command

```
for (i in 1:19) {
cov <- getData(meth.CHG.filtindfiltSNP.filtCov)
[, colnames(meth.CHG.filtindfiltSNP.filtCov) == paste0("coverage", i)]
cov_filt <- sort(c(which(cov < 7), which(is.na(cov))))
meth.CHG.filtindfiltSNP.filtCov[cov_filt,
colnames(meth.CHG.filtindfiltSNP.filtCov) == paste0("numCs", i)] <-
NA meth.CHG.filtindfiltSNP.filtCov[cov_filt,
colnames(meth.CHG.filtindfiltSNP.filtCov) == paste0("numTs", i)] <-
NA
rm(cov, cov_filt)
}
```

8 Identification of target regions for the SeqCapBis design

We first grouped SMPs into 1kb sliding windows of 250bp for each methylation context. Following the calculation of the methylation levels in each window, the outlier DMRs were identified using two strategies (see 8.2 and 8.3) with homemade scripts (given as examples). Finally, target sequences correspond to outlier DMRs identified by the two strategies.

8.1 Grouping SMPs in windows and DMRs identification

Software

methylKit

Alexander Blume

NAME

DEVELOPER

<https://github.com/al2na/methylKit/releases>

SOURCE LINK

Publication: Akalin et al., 2012

Version: 1.18.0

Github: <https://github.com/al2na/methylKit/releases>

Site: <https://bioconductor.org/packages/release/bioc/html/methylKit.html>

Parameters: MethylKit package

CITATION

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE (2012). methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles..

LINK

<https://doi.org/10.1186/gb-2012-13-10-r87>

Input files: pre-filtered SMPs in each context.

Command

```
meth.CpG.window <-
tileMethylCounts(meth.CpG.filtindfiltSNP.filtTE.filtCov.filtNA,win.size = 1000, step.size = 250)
meth.CHG.window <-
tileMethylCounts(meth.CHG.filtindfiltSNP.filtTE.filtCov.filtNA,win.size = 1000, step.size = 250)
meth.CHH.window <-
tileMethylCounts(meth.CHH.filtindfiltSNP.filtTE.filtCov.filtNA,win.size = 1000, step.size = 250)
```

8.2 Strategy I: STANDARD DEVIATION OF THE MEANS

Calculate average C-methylation by averaging the methylation level across all (pre-filtered) cytosines in each window for each individual. Then calculate standard deviation of this average across individuals.

Command

```
#Identification of windows to remove
percmeth.CpG.window.sd <- rowSds(percmeth.CpG.window, na.rm = TRUE)
sum(percmeth.CpG.window.sd == 0)

# Removal of windows showing the less variable levels of methylation
percmeth.CpG.window <-
percmeth.CpG.window[which(percmeth.CpG.window.sd != 0), ]
dim(percmeth.CpG.window)

#Identification of the windows associated with the most variable
methylation levels
percmeth.CpG.window.sd <- rowSds(percmeth.CpG.window, na.rm = TRUE)
layout(matrix(c(rep(1, 2), 2), nrow = 1))
hist(percmeth.CpG.window.sd, col = "grey", main = "")
bp <- boxplot(percmeth.CpG.window.sd, col = "grey")
length(bp$out)
bp$stats
```

8.3 Strategy II: MEAN OF THE STANDARD DEVIATIONS

For each (pre-filtered) cytosine, calculate the standard deviation of methylation across individuals. Then calculate the mean standard deviation from all cytosines in a window.

Command

```

dag_window_size=1000
dag_step=250

load("meth.CHG.filtind.filtSNP.filtTE.filtCov.filtNA.Rdata")
y<-x[,c("chr","start","end","strand")]

for (i in 1:length(colnames(x)[colnames(x) %like% "coverage"])){ # To recover the C/coverage values
  j=5+3*(i-1)
  print(paste0(j, " ", j+1))
  y[,paste0("in",i)]<-x[,j+1]/x[,j]
}
yy<-x[,c("chr","start","end","strand")]
rm(x)

z<-rowSds(as.matrix(y[,5:ncol(y)]),na.rm=TRUE) # Calculate row standard deviations
yy$STDEV<-z
rm(z)
y<-yy
rm(yy)

# Do last adaptations and launch
dag_window=dag_window_size/dag_step
colnames(y)<-c("CHR","START","END","STRAND","STDEV")
y$MEAN<-(y$START+y$END)/2
y$CHR<-gsub("Chr0","Chr",y$CHR,perl=TRUE)
y$WINDOW<-floor(y$MEAN/dag_step)+1

stdev_counts = data.table(
  CHR = character(),
  WIN = numeric(),
  POS = numeric(),
  STDEV = numeric()
)

count=0
for (i in unique(y[y$CHR %like% "Chr" | y$CHR %like% "scaffold",]$CHR)){
  window size=dag window size
}

```

```

step=dag_step
#i<-paste0("Chr",i)
z<-y[y$CHR==i,]
min=0
max=max(z$WINDOW)
#print(paste(i,min,max,min(z$MEAN),max(z$MEAN)))
count=count+1

print(paste(i,min,max,min(z$MEAN),max(z$MEAN),count,length(unique(y[y$CHR %like% "Chr" | y$CHR %like% "scaffold",]$CHR))))
zz<-data.frame(matrix(ncol=2,nrow=max*step))
colnames(zz)<-c("MEAN","STDEV")
zz$MEAN<-rownames(zz)

zz[zz$MEAN %in% z$MEAN,]$STDEV<-z[z$MEAN %in% zz$MEAN,]$STDEV

# Sliding window
total <- nrow(zz)
if (max(z$MEAN)<window_size){ # Adapted to avoid problems with scaffolds smaller than window_size
  spots <- 1
}
else {
  spots <- seq(from=1, to=(total-window_size), by=step)
}

if (spots[length(spots)]<=total-window_size){spots<-c(spots,
(spots[length(spots)]+step))} # Adapted to recover the last bits inside smaller window
result <- vector(length = length(spots))
for(j in 1:length(spots)){
  if (j%%50000==0){print(paste(j,length(spots)))}
  if ((spots[j]+window_size)>=total){window_size=(total-spots[j])}
# Adapted to recover the last bits inside last smaller window
  result[j] <- mean(zz[spots[j]:(spots[j]+window_size-1),"STDEV"],na.rm=TRUE)
}

stdev_counts<-
rbind(stdev_counts,data.frame(CHR=i,WIN=1:length(spots),POS=spots,STDE
V=result))
}

x<-stdev_counts
write.table(x,file=paste0(save_file_name))

```

8.4 Outlier threshold

The threshold for DMRs is defined as $(Q3 + 1.5 * (Q3 - Q1))$ where Q1 and Q3 are the first and third quartiles (i.e. the threshold is not defined by a percentile, but instead depends on the length of the boxplot box)

* Strategy I

Parameters: Python 3.7

Command

```
#$Id$  
  
###run with python get_threshold_over_all_windows_calc1.py  
OUTPUT_FILE_from_calc1_get_mean_and_stdv_for_each_window.py >  
threshold_calc1.txt  
  
import os  
import re  
import string  
import sys  
import glob  
import numpy  
  
file1 = sys.argv[1]  
file1_stream = open(file1)  
list_of_means = []  
  
for line1 in file1_stream.readlines():  
    if (line1.count('start') == 0):  
        line1 = line1.replace('\n','')  
        splitted_line1 = line1.split('\t')  
        scaffold = splitted_line1[0]  
        start = splitted_line1[1]  
        end = splitted_line1[2]  
  
        mean = splitted_line1[13]  
        mean = float(mean)  
        list_of_means.append(mean)  
  
list_of_means.sort()  
nbre_de_means = len(list_of_means)  
##XXX corresponds to the first half of the dataset  
##YYY corresponds to the second half of the dataset  
Q1 = numpy.median(list_of_means[:XXX])  
Q3 = numpy.median(list_of_means[YYY:])  
  
##for CHH context, hreshold = (Q3 + 3*(Q3- Q1))  
threshold = (Q3 + 1.5*(Q3- Q1))  
threshold = round(threshold,5)  
  
print 'threshold = '.threshold
```

*** Strategy II**

Parameters: Python 3.7

Command

```
#$Id$  
  
###run with python get_threshold_stdv_over_all_windows_calc2.py  
OUTPUT_FILE_from_get_stdv_between_individuals_for_each_window_calc2.py  
> threshold_calc2.txt  
  
  
import os  
import re  
import string  
import sys  
import glob  
import numpy  
  
file1 = sys.argv[1]  
file1_stream = open(file1)  
list_of_stdv = []  
  
for line1 in file1_stream.readlines():  
    if (line1.count('start') == 0):  
        line1 = line1.replace('\n', '')  
        splitted_line1 = line1.split('\t')  
        scaffold = splitted_line1[0]  
        start = splitted_line1[1]  
        end = splitted_line1[2]  
  
        stdv = splitted_line1[4]  
        stdv = float(stdv)  
        list_of_stdv.append(stdv)  
  
list_of_stdv.sort()  
nbre_de_stdv = len(list_of_stdv)  
##XXX corresponds to the first half of the dataset  
##YYY corresponds to the second half of the dataset  
Q1 = numpy.median(list_of_stdv[:XXX])  
Q3 = numpy.median(list_of_stdv[YYY:])  
  
##for CHH context, hreshold = (Q3 + 3*(Q3- Q1))  
threshold = (Q3 + 1.5*(Q3- Q1))  
threshold = round(threshold,5)  
  
print 'threshold = '.threshold
```

8.5 Identification of capture targets

Target sequences correspond to outlier DMRs identified by the two strategies. This is a two-steps strategy where the 3 contexts are first merged and, then, sequence redundancy between the three methylation contexts is removed.

Software

bedtools

NAME

Linux

OS

Publication: Quinlan AR and Hall IM, 2010

Version: 2.27.1

Github: <https://github.com/arq5x/bedtools2>

Parameters: intersect, merge

CITATION

Quinlan AR, Hall IM (2010). BEDTools: a flexible suite of utilities for comparing genomic features..

LINK

<https://doi.org/10.1093/bioinformatics/btq033>

SeqCapBis - Detecting mC Target regions

9 Agilent Probes design and sequencing

A set of 120 bp probes was selected to capture 18 Mb of each genome (Agilent, <https://earray.chem.agilent.com/suredesign/>). The targeted regions corresponded to the regions identified as differentially methylated between populations. Custom targeted genome bisulfite sequencing was performed with SureSelect XT Methyl-Seq Target Enrichment (Agilent, Santa Clara, CA, USA) according to the manufacturer's recommendations.

For poplar, in total, 17.84 Mb of sequence corresponding to the 25,434 DMRs was covered by 339,658 probes. Regarding oak, a set of 140,249 probes (120 bp) was designed by Agilent to cover 16.15 Mb DMRs.

10 Trimming

Software

TrimGalore

NAME

Linux

OS

Publication: Krueger F et al., 2023. FelixKrueger/TrimGalore: v0.6.5

Version: 0.6.5

Github: <https://github.com/FelixKrueger/TrimGalore>

CITATION

Felix Krueger; Frankie James; Phil Ewels; Ebrahim Afyounian; Michael Weinstein; Benjamin Schuster-Boeckler; Gert Hulselmans; sclamons (2023).

FelixKrueger/TrimGalore: v0.6.10. Zenodo.

LINK

<https://doi.org/10.5281/zenodo.5127898>

Command

```
trim_galore input_R1.fastq.gz input_R2.fastq.gz --paired ADAPTER1 -a2  
ADAPTER2 -o output_directory --gzip -j {threads}
```

11 Quality control

Software

FastQC

NAME

Linux

OS

Simon Andrews

DEVELOPER

Publication: Andrews, S. (2010). FastQC: A Quality Control Tool for High Throughput Sequence Data [Online]. Available online at:

<http://www.bioinformatics.babraham.ac.uk/projects/fastqc/>

Version: 0.11.9

Github: <https://github.com/s-andrews/FastQC>

Command

```
fastqc trimmed_reads.fq.gz -o fastQC_output_directory -t {threads}
```

12 **Mapping**

Software

BsmapZ

NAME

Linux

OS

Publications:

- Xi Y, Li W, 2009

CITATION

Xi Y, Li W (2009). BSMAP: whole genome bisulfite sequence MAPping program..

LINK

<https://doi.org/10.1186/1471-2105-10-232>

- Zynda G. 2018. BSMAPz. <https://github.com/zyndagj/BSMAPz>

Version: 1.1.3

Github: <https://github.com/zyndagj/BSMAPz>

Poplar genome: *Populus trichocarpa* v4.1

Command

```
bsmapz -a fileR1.fq.gz -b fileR2.fq.gz -o {output.out} -d  
mapped_file.bam -d ref_genome.fa -p threads
```

Mapping adjustments for *Q. petraea*

Oak genome: *Quercus robur* Haplome V2.3

12.1 Duplicate Removing

Software

samtools

NAME

Linux

OS

Publication: Danecek et al., 2021

Version: 1.11

Github: <https://github.com/samtools/samtools>

Parameters: stat, fixmate, sort, markup

Poplar genome: *Populus trichocarpa* v4.1

CITATION

Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H (2021). Twelve years of SAMtools and BCFtools..

LINK

<https://doi.org/10.1093/gigascience/giab008>

Command

```
samtools stats sample_bsmapz_sorted.bam -r ref_genome.fa -@ {threads}
> sample.statics
samtools fixmate -@ {threads} -O BAM -m sample_bsmapz_sorted.bam
sample_fixmate.bam
samtools sort -@ {threads} -O BAM sample_fixmate.bam -o
sample_fixmate_sort.bam
samtools markdup -r ref_genome.fa -@ {threads} -s -f sample.statics
sample_fixmate_sort.bam sample_fixmate_sort_temp.bam
```

Mapping adjustments for *Q. petraea*

Oak genome: *Quercus robur* Haplome V2.3

13 Detection of methylated cytosines (mC)

Software

BsmapZ

NAME

Linux

OS

Publications:

- Xi Y and Li W, 2009.

CITATION

Xi Y, Li W (2009). BSMAP: whole genome bisulfite sequence MAPping program..

LINK

<https://doi.org/10.1186/1471-2105-10-232>

- Zynda G. 2018. BSMAPz. <https://github.com/zyndagj/BSMAPz>

Version: 1.1.3

Github: <https://github.com/zyndagj/BSMAPz>

Poplar genome: *Populus trichocarpa* v4.1

Parameters: methratio.py, python 2.7, samtools 1.11, pysam 0.16.0.1

Command

```
python methratio.py sample.dedup.bam -o meth_sample.txt -d  
ref_genome.fa -N {threads} -I
```

Mapping adjustments for *Q. petraea*

Oak genome: *Quercus robur* Haplome V2.3

14 **10X sequencing filtering**

Software

methylKit

NAME

Alexander Blume

DEVELOPER

<https://github.com/al2na/methylKit/releases>

SOURCE LINK

Publication: Akalin A et al, 2012.

Version: 1.18.0

Parameters: MethylKit package

Github: <https://github.com/al2na/methylKit/releases>

Site: <https://bioconductor.org/packages/release/bioc/html/methylKit.html>

CITATION

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE (2012). methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles..

LINK

<https://doi.org/10.1186/gb-2012-13-10-r87>

Command

```
SeqCapBis_CHG = methRead(location = path_to_the_files, sample.id =  
sample.ids, assembly = "quercus", mincov = 10, context = "CHG",  
treatment = rep(0,10))
```

15 Splitting context

We set up a homemade bash script (`splitting.sh`) to obtain methylation files for each sample in the three contexts (CG, CHG and CHH).

Command

```
#!/bin/bash
# Splitting context:

usage()
{
cat << EOF
usage: $0 <options>
splitting context.
OPTION:
-h      show this Help message.
-o      Output.
-i      Input.
EOF
}

# Get options
while getopts "ho:i:" OPTION
do
case $OPTION in
h) usage; exit 1;;
o) output=$OPTARG;;
i) input=$OPTARG;;
?) usage; exit;;
esac
done

# Check that all options were passed
if [[ -z $output ]] || [[ -z $input ]]
then
printf "\n=====\\n ERROR: missing
options\\n=====\\n\\n"
usage
exit 1
fi

#in_file = snakemake.input["isoforms"]
#out_file = snakemake.output["plot"]

# Fail on the first error
set -e

#####
```

```
file=$(echo $output|rev|cut -d "/" -f 1 |rev)
path=$(echo $output|rev|cut -d "/" -f 2- |rev)

for context in "CHH" "CG" "CHG"; do

    awk "NR<=1 || \$4~/$context/" $input > $path/$context-$file ;
done
```

16 Methylation quantification

Software

methylKit NAME

Alexander Blume DEVELOPER

<https://github.com/al2na/methylKit/releases> SOURCE LINK

Publication: Akalin A et al, 2012.

Version: 1.18.0

Parameters: MethylKit package

Github: <https://github.com/al2na/methylKit/releases>

Site: <https://bioconductor.org/packages/release/bioc/html/methylKit.html>

CITATION

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE (2012). methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles..

LINK

<https://doi.org/10.1186/gb-2012-13-10-r87>

Functions: getMethylationStats(), getCoverageStats()

Command

```
# Read methylation using methylkit function methRead  
myobj <- methRead(location = files, sample.id = sample_id, assembly =  
"populus tricharpa v3.1", mincov = 1, context = context, treatment =  
rep(0, length(files)), pipeline = list(fraction=TRUE, chr.col=1,  
start.col=2, end.col=2, coverage.col=6, strand.col=3, freqC.col=5 ))  
  
# Concatenate all samples tables into one unique table  
finalFrame <- mergeMethylkitOutput(myobj)  
  
#Write the final table as a csv2 file  
write.csv2(finalFrame,file = table,,)  
  
# head(myobj)  
  
# plots for statistics and coverage simple :  
pdf(file = XXX)  
getMethylationStats(myobj[[1]],plot=TRUE,both.strands=FALSE)  
getCoverageStats(myobj[[1]],plot=TRUE,both.strands=FALSE)  
dev.off()
```

Transposon insertion polymorphisms (TIPs)

17 Trimming

Eliminate unwanted or irrelevant parts of the read. Data trimming may include removing low quality bases or adapters used during sequencing.

Software

TrimGalore

NAME

Linux

OS

Felix Krueger

DEVELOPER

CITATION

Felix Krueger; Frankie James; Phil Ewels; Ebrahim Afyounian; Michael Weinstein; Benjamin Schuster-Boeckler; Gert Hulselmans; sclamons (2023).
FelixKrueger/TrimGalore: v0.6.10. Zenodo.

LINK

<https://doi.org/10.5281/zenodo.5127898>

Command

```
#Trim the paired sequences
trim_galore -q 30 --paired -o paired_1.fastq paired_2.fastq
```

18 Detection of TIPs on whole genome sequencing (WGS) data with TEFLoN

18.1 Mapping

Alignment of DNA sequences to a reference genome.

Software

BWA	NAME
-----	------

Linux	OS
-------	----

Heng Li	DEVELOPER
---------	-----------

CITATION

Heng Li; Richard Durbin (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. *bioinformatics*.

LINK

<https://doi.org/10.1093/bioinformatics/btp324>

Command

```
#Index Genome  
bwa index genome_ref.fa  
  
#Align  
bwa mem -Y genome_ref.fa paired_trimmed_1.fastq  
paired_trimmed_2.fastq > whole.sam
```

18.2 Extracting unmapped reads

Search for TIPs from reads not aligning with the reference genome. It is interesting to choose non-mapped sequences, because we hypothesize that the insertion of a transposable element is one of the reasons which prevented the alignment of certain reads to their reference genome.

Software

samtools

NAME

<https://github.com/samtools/samtools>

SOURCE LINK

CITATION

Petr Danecek, James K Bonfield, Jennifer Liddle, John Marshall, Valeriu Ohan, Martin O Pollard, Andrew Whitwham, Thomas Keane, Shane A McCarthy, Robert M Davies, Heng Li (2021). Twelve years of SAMtools and BCFtools. *GigaScience*, Volume 10.

LINK

<https://doi.org/10.1093/gigascience/giab008>

Command

```
#From SAM2BAM
samtools view -S -b whole.sam -o whole.bam

#Extract Unmapped reads

#An unmapped read whose mate is mapped.
samtools view -u -f 4 -F264 whole.bam > tmps1.bam

#Both reads of the pair are unmapped
samtools view -u -f 12 -F 256 whole.bam > tmps2.bam

#merge
samtools merge unmapped.bam tmps1.bam tmps2.bam
```

Software

BamToFastq

NAME

Linux

OS

Maxime U Garcia

DEVELOPER

CITATION

Friederike Hanssen, SusiJo, Gisela Gabernet, Maxime U Garcia, Matilda Åslin, nf-core bot (2023). nf-core/bamtofastq: 2.1.0. Zenodo.

LINK

<https://doi.org/10.5281/zenodo.4710628>

Command

```
#Extract the reads in FASTQ format (paired)
bamToFastq -bam unmapped.bam -fq1 unmapped_reads1.fastq -fq2
unmapped_reads2.fastq
```

18.3 TIPs detection

Search for TIPs from reads not aligning with the reference genome. It is interesting to choose non-mapped sequences, because we hypothesize that the insertion of a transposable element is one of the reasons which prevented the alignment of certain reads to their reference genome.

Software

TEFLoN

NAME

Linux

OS

Jeffrey Adrion

DEVELOPER

CITATION

Adrion, J.R., M.J. Song, D.R. Schrider, M.W. Hahn, and S. Schaack (2017). Genome-wide estimates of transposable element insertion and deletion rates in **Drosophila melanogaster**. *Genome Biology and Evolution*.

LINK

<https://doi.org/10.1093/gbe/evx050>

Software

RepeatMasker

NAME

Linux

OS

Robert Hubley

DEVELOPER

Command

```
WD="path/to/working/_directory"
PREFIX="prefix_you_want"

##For each samples
python teflon_prep_custom.py -wd ${WD}reference -g genome_ref -l
path/to/TE_LIBRARY -p ${PREFIX}

bwa index ${WD}reference/${PREFIX}.prep_MP/${PREFIX}.mappingRef.fa

bwa mem -Y ${WD}reference/${PREFIX}.prep_MP/${PREFIX}.mappingRef.fa
${READS1} ${READS2} > ${WD}reference/${PREFIX}.sam

samtools view -Sb ${WD}reference/${PREFIX}.sam | samtools sort -o
${WD}reference/${PREFIX}.sorted.bam

samtools index ${WD}reference/${PREFIX}.sorted.bam

#Run Teflon
#For each samples
python teflon.v0.4.py -wd ${WD} -d ${WD}reference/${PREFIX}.prep_TF/ -
-s path/to/samples -i unique_ID -l1 family -l2 class

#Teflon collapse
##Only once
python teflon_collapse.py -wd ${WD} -d
${WD}reference/${PREFIX}.prep_TF/ -s path/to/samples -n1
minimum_reads_to_support_TE_in_one_sample -n2
minimum_reads_to_support_TE_in_all_samples

#Teflon Count
#For each samples
python teflon_count.py -wd ${WD} -d ${WD}reference/${PREFIX}.prep_TF/
-s path/to/samples -i unique_ID

#Teflon genotype
##Only once
python teflon_genotype.py -wd ${WD} -d
${WD}reference/${PREFIX}.prep_TF/ -s path/to/samples -dt pooled
```

19 Detection of TIPs on whole genome bisulfite sequencing (WGBS) data with epiTEome

19.1 Mapping and extracting unmapped reads

Alignment of DNA sequences to a reference genome. Search for TIPs from reads not aligning with the reference genome. We choose non-mapped sequences, because we hypothesize that the insertion of a transposable element is one of the reasons which prevented the alignment of certain reads to their reference genome.

Software

Bismark

NAME

Felix Krueger

DEVELOPER

CITATION

Felix Krueger, Simon R Andrews (2011). Bismark: a flexible aligner and methylation caller for Bisulfite-Seq applications. Bioinformatics.

LINK

<https://doi.org/10.1093/bioinformatics/btr167>

Command

```
bismark_genome_preparation --verbose genome_ref.fa  
  
bismark --genome genome_ref.fa paired_trimmed_1.fastq  
paired_trimmed_2.fastq --un
```

19.2 TIPs detection

Search for TIPs from reads not aligning with the reference genome. It is interesting to choose non-mapped sequences, because we hypothesize that the insertion of a transposable element is one of the reasons which prevented the alignment of certain reads to their reference genome.

Software

epiT Eome

NAME

Josquin Daron

DEVELOPER

CITATION

Josquin Daron & R. Keith Slotkin (2017). EpiTEome: Simultaneous detection of transposable element insertion sites and their DNA methylation levels. *Genome Biology*.

LINK

<https://doi.org/10.1186/s13059-017-1232-0>

Command

```
idxEpiTEome.pl -l 100 -gff genome_ref.gff -t /path/to/TE_LIBRARY -  
fasta genome_ref.fa
```

```
epiT Eome.pl -gff genome_ref.gff -ref genome_ref.epiT Eome.masked.fasta  
-un unmapped_reads.fastq -t /path/to/TE_LIBRARY
```

Citations

Step 12

Xi Y, Li W. BSMAP: whole genome bisulfite sequence MAPping program.

<https://doi.org/10.1186/1471-2105-10-232>

Step 12.1

Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H. Twelve years of SAMtools and BCFtools.

<https://doi.org/10.1093/gigascience/gjab008>

Step 13

Xi Y, Li W. BSMAP: whole genome bisulfite sequence MAPping program.

<https://doi.org/10.1186/1471-2105-10-232>

Step 14

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE. methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles.

<https://doi.org/10.1186/gb-2012-13-10-r87>

Step 16

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE. methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles.

<https://doi.org/10.1186/gb-2012-13-10-r87>

Step 17

Felix Krueger; Frankie James; Phil Ewels; Ebrahim Afyounian; Michael Weinstein; Benjamin Schuster-Boeckler; Gert Hulselmans; sclamons. FelixKrueger/TrimGalore: v0.6.10

<https://doi.org/10.5281/zenodo.5127898>

Step 18.1

Heng Li; Richard Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform

<https://doi.org/10.1093/bioinformatics/btp324>

Step 2

Bolger AM, Lohse M, Usadel B. Trimmomatic: a flexible trimmer for Illumina sequence data.

<https://doi.org/10.1093/bioinformatics/btu170>

Step 3

Heng Li. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM

<https://doi.org/10.48550/arXiv.1303.3997>

Step 3

Tuskan GA, Difazio S, Jansson S, Bohlmann J, Grigoriev I, Hellsten U, Putnam N, Ralph S, Rombauts S, Salamov A, Schein J, Sterck L, Aerts A, Bhalerao RR, Bhalerao RP, Blaudez D, Boerjan W, Brun A, Brunner A, Busov V, Campbell M, Carlson J, Chalot M, Chapman J, Chen GL, Cooper D, Coutinho PM, Couturier J, Covert S, Cronk Q, Cunningham R, Davis J, Degroeve S, Déjardin A, Depamphilis C, Detter J, Dirks B, Dubchak I, Duplessis S, Ehltung J, Ellis B, Gendler K, Goodstein D, Gribskov M, Grimwood J, Groover A, Gunter L, Hamberger B, Heinze B, Helariutta Y, Henrissat B, Holligan D, Holt R, Huang W, Islam-Faridi N, Jones S, Jones-Rhoades M, Jorgensen R, Joshi C, Kangasjärvi J, Karlsson J, Kelleher C, Kirkpatrick R, Kirst M, Kohler A, Kalluri U, Larimer F, Leebens-Mack J, Leplé JC, Locascio P, Lou Y, Lucas S, Martin F, Montanini B, Napoli C, Nelson DR, Nelson C, Nieminen K, Nilsson O, Pereda V, Peter G, Philippe R, Pilate G, Poliakov A, Razumovskaya J, Richardson P, Rinaldi C, Ritland K, Rouzé P, Ryaboy D, Schmutz J, Schrader J, Segerman B, Shin H, Siddiqui A, Sterky F, Terry A, Tsai CJ, Überbacher E, Unneberg P, Vahala J, Wall K, Wessler S, Yang G, Yin T, Douglas C, Marra M, Sandberg G, Van de Peer Y, Rokhsar D. The genome of black cottonwood, *Populus trichocarpa* (Torr. & Gray).

<https://doi.org/>

Step 3.1

Plomion C, Aury JM, Amselem J, Leroy T, Murat F, Duplessis S, Faye S, Francillonne N, Labadie K, Le Provost G, Lesur I, Bartholomé J, Faivre-Rampant P, Kohler A, Leplé JC, Chantret N, Chen J, Diévert A, Alaeitabar T, Barbe V, Belser C, Bergès H, Bodénès C, Bogaert-Triboulot MB, Bouffaud ML, Brachi B, Chancerel E, Cohen D, Couloux A, Da Silva C, Dossat C, Ehrenmann F, Gaspin C, Grima-Pettenati J, Guichoux E, Hecker A, Herrmann S, Hugueney P, Hummel I, Klopp C, Lalanne C, Lascoux M, Lasserre E, Lemainque A, Desprez-Loustau ML, Luyten I, Madoui MA, Mangenot S, Marchal C, Maumus F, Mercier J, Michotey C, Panaud O, Picault N, Rouhier N, Rué O, Rustenholz C, Salin F, Soler M, Tarkka M, Velt A, Zanne AE, Martin F, Wincker P, Quesneville H, Kremer A, Salse J. Oak genome reveals facets of long lifespan.

<https://doi.org/10.1038/s41477-018-0172-3>

Step 3.2

Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H. Twelve years of SAMtools and BCFtools.

<https://doi.org/10.1093/gigascience/giab008>

Step 4.1

Crusoe MR, Alameldin HF, Awad S, Boucher E, Caldwell A, Cartwright R, Charbonneau A, Constantinides B, Edvenson G, Fay S, Fenton J, Fenzl T, Fish J, Garcia-Gutierrez L, Garland P, Gluck J, González I, Guermond S, Guo J, Gupta A, Herr JR, Howe A, Hyer A, Härpfer A, Irber L, Kidd R, Lin D, Lippi J, Mansour T, McA'Nulty P, McDonald E, Mizzi J, Murray KD, Nahum JR, Nanlohy K, Nederbragt AJ, Ortiz-Zuazaga H, Ory J, Pell J, Pepe-Ranney C, Russ ZN, Schwarz E, Scott C, Seaman J, Sievert S, Simpson J, Skennerton CT, Spencer J, Srinivasan R, Standage D, Stapleton JA, Steinman SR, Stein J, Taylor B, Trimble W, Wiencko HL, Wright M, Wyss B, Zhang Q, Zyme E, Brown CT. The khmer software package: enabling efficient nucleotide sequence analysis.

<https://doi.org/10.12688/f1000research.6924.1>

Step 4.3

McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, DePristo MA. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data.

<https://doi.org/10.1101/gr.107524.110>

Step 4.5

Li H. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data.

<https://doi.org/10.1093/bioinformatics/btr509>

Step 4.5

Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H. Twelve years of SAMtools and BCFtools.

<https://doi.org/10.1093/gigascience/giab008>

Step 4.7

Erik Garrison and Gabor Marth. Haplotype-based variant detection from short-read sequencing

<https://doi.org/10.48550/arXiv.1207.3907>

Step 4.8

Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, Handsaker RE, Lunter G, Marth GT, Sherry ST, McVean G, Durbin R, 1000 Genomes Project Analysis Group. The variant call format and VCFtools.

<https://doi.org/10.1093/bioinformatics/btr330>

Step 4.9

Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H. Twelve years of SAMtools and BCFtools.

<https://doi.org/10.1093/gigascience/giab008>

Step 6

Dugé de Bernonville T, Daviaud C, Chaparro C, Tost J, Maury S. From Methylome to Integrative Analysis of Tissue Specificity.

https://doi.org/10.1007/978-1-0716-2349-7_16

Step 6

Sow MD, Rogier O, Lesur I, Daviaud C, Mardoc E, Sanou E, Duvaux L, Civan P, Delaunay A, Lesage-Descauses MC, Benoit V, Le-Jan I, Buret C, Besse C, Durufle H, Fichot R, Le-Provost G, Guichoux E, Boury C, Garnier A, Senhaji-Rachik A, Jorge V, Ambroise C, Tost J, Plomion C, Segura V, Maury S, Salse J. Epigenetic Variation in Tree Evolution: a case study in black poplar (*Populus nigra*)

<https://doi.org/10.1101/2023.07.16.549253>

Step 6.2

Xi Y, Li W. BSMAP: whole genome bisulfite sequence MAPping program.

<https://doi.org/10.1186/1471-2105-10-232>

Step 6.3

Xi Y, Li W. BSMAP: whole genome bisulfite sequence MAPping program.

<https://doi.org/10.1186/1471-2105-10-232>

Step 7

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE. methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles.

<https://doi.org/10.1186/gb-2012-13-10-r87>

Step 8.1

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE. methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles.

<https://doi.org/10.1186/gb-2012-13-10-r87>

Step 8.1

Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE. methylKit: a comprehensive R package for the analysis of genome-wide DNA methylation profiles.

<https://doi.org/10.1186/gb-2012-13-10-r87>

Step 8.5

Quinlan AR, Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features.

<https://doi.org/10.1093/bioinformatics/btq033>