

Travail de fin d'études

**Exploration des Variations Méthyliques
dans le Méthylome ADN des Bourdons en
fonction de leur âge**

Camille Vande Walle

Bachelier en Biotechnique Bloc 3
HEH Département des sciences et technologies
HEPH – Condorcet
Année académique 2022 - 2023



Travail de fin d'études

**Exploration des Variations Méthyliques dans
le Méthylome ADN des Bourdons en
fonction de leur âge**

Camille Vande Walle

Bachelier en Biotechnique Bloc 3
HEH Département des sciences et technologies
HEPH – Condorcet
Année académique 2022 - 2023

Remerciements

Je tiens à exprimer ma sincère reconnaissance envers Monsieur Coornaert pour son soutien indéfectible tout au long de mes études, et plus spécifiquement pendant mon stage. Sa précieuse aide pour l'installation de programmes complexes et pour résoudre les problèmes liés aux incompatibilités de version a été d'une grande importance. Je regrette seulement de ne jamais pouvoir lui rendre le soutien qu'il m'a généreusement apporté.

Je souhaite également exprimer ma gratitude envers Thibaut Renard et Baptiste Martinet pour leur précieuse assistance dans le domaine de la biologie. Leur aide a été essentielle pour comprendre le projet dans son ensemble, y compris les aspects techniques tels que l'impact de l'inhibiteur sur les enzymes. Ils ont toujours pris le temps de m'expliquer en détail les aspects biologiques.

Merci à ma co-superviseuse de stage, Natalia De Souza Araujo pour son accompagnement tout au long de mon stage, ainsi que pour sa méthode d'enseignement qui m'a encouragé à développer mon autonomie et à réfléchir de manière indépendante. Ses nombreuses corrections et suggestions concernant mes résultats, les paramètres des tests et les choix de programmes ont été extrêmement précieuses.

Je suis profondément reconnaissant envers Serge Aron, mon maître de stage, pour m'avoir offert l'opportunité de réaliser mon stage au sein de l'Université Libre de Bruxelles et de m'avoir confié un travail aussi captivant.

Je suis également reconnaissant pour son intérêt constant à suivre mes progrès et mon épanouissement au sein de l'université. Je tiens à exprimer ma gratitude envers Monsieur Branders pour son précieux soutien sur RStudio. Il m'a apporté une aide précieuse dans la création d'un script permettant de modifier la structure de mes fichiers d'appel de méthylation, afin d'y inclure les numéros de chromosomes plutôt que les identifiants du NCBI.

Résumé

Le processus d'analyse des données a été méthodique et rigoureux. Dans un premier temps, les séquences brutes ont été nettoyées à l'aide de l'outil Trim_Galore, qui a éliminé les séquences de faible qualité. Ensuite, les lectures ont été alignées sur le génome de référence à l'aide de Bismark, fournissant des informations essentielles sur les taux d'alignement et les motifs de méthylation. Des étapes de déduplication et de tri des lectures ont été entreprises pour optimiser les données de séquençage. Des biais de conversion de bisulfite, qui peuvent fausser les résultats de méthylation, ont également été identifiés et corrigés.

L'analyse proprement dite a été effectuée dans l'environnement Rstudio. Cette phase impliquait le filtrage et la normalisation des données, ainsi que des calculs de pourcentage de méthylation et de couverture. Ces analyses ont permis de saisir en détail les motifs de méthylation et les variations entre les différents échantillons. Une comparaison du méthylome en fonction de l'âge a été entreprise. Des étapes de filtrage ont été utilisées pour éliminer les bases présentant une couverture excessive ou insuffisante. La normalisation a été appliquée pour harmoniser les distributions de couverture de lecture. Les échantillons ont été regroupés pour des analyses comparatives, utilisant des méthodes telles que l'analyse en composante principale et le regroupement pour mieux appréhender les similarités entre les échantillons.

La recherche a impliqué l'identification de régions présentant des différences significatives en matière de méthylation. Pour évaluer les variations de méthylation entre les groupes d'échantillons. Un aspect notable a été l'annotation des SNP différemment méthylos, pour lequel un programme personnalisé en Python a été développé. Ce programme interagit avec les fichiers de données et d'annotation génétique, annotant les SNP en fonction de leur position par rapport aux gènes.

Abstract

The data analysis process was methodical and rigorous. First, the raw sequences were cleaned using the Trim_Galore tool, which eliminated low-quality sequences. Next, the reads were aligned to the reference genome using Bismark, providing essential information on alignment rates and methylation patterns. Read de-duplication and sorting steps were undertaken to optimize the sequencing data. Bisulfite conversion biases, which can distort methylation results, were also identified and corrected.

The actual analysis was carried out in the Rstudio environment. This phase involved data filtering and normalization, as well as percentage methylation and coverage calculations. These analyses enabled detailed capture of methylation patterns and variations between different samples. A comparison of methylome as a function of age was undertaken. Filtering steps were used to eliminate bases with excessive or insufficient coverage. Normalization was applied to harmonize read coverage distributions. Samples were pooled for comparative analyses, using methods such as principal component analysis and clustering to better apprehend similarities between samples. The research involved identifying regions with significant differences in methylation. To assess methylation variations between sample groups. A notable aspect was the annotation of differentially methylated SNPs, for which a custom Python program was developed. This program interacts with data and gene annotation files, annotating SNPs according to their position relative to genes.

Table des Tableaux

Introduction	1
Bombus terrestris.....	1
La méthylation de l'ADN	1
Les modifications chimiques de l'ADN.....	1
Transmission de la méthylation	2
Influence de l'expression génique	3
Traitement bisulfite	4
Objectif du travail.....	5
Logiciels utilisés	5
Logiciels de traitement de données.....	5
Trim_Galore	5
Bismark	5
MethylDackel.....	6
Vérification des données.....	6
Fastqc	6
Qualimap.....	6
Langage de programmation	6
R sur Rstudio	6
Python.....	7
Ordinateur de travail	7
Ordinateur personnel.....	7
Ordinateur de l'ULB	7
Résultats	7
Trim_Galore.....	8
Alignement des reads	9
Rapport d'alignement	10
Déduplication des <i>reads</i>	11
Tri par coordonnées.....	12

MethylDackel	12
MethylDackel mbias	12
MethylDackel extract	14
Analyse de la méthylation	14
Librairies	14
Importation des fichiers	15
Préparation des données	15
Filtre et normalisation des données	16
Statistiques descriptions des échantillons.....	16
Pourcentage de méthylation	16
Couverture du contexte dans l'échantillon.....	17
Filtre des échantillons.....	19
Filtre des échantillons sur la couverture	19
Normalisation sur la couverture	20
Fusion des échantillons.....	20
Analyse en composante principale	21
Normaliser les échantillons par taille.....	21
Normalize	22
PCASamples.....	22
Procom.....	22
PCASamples choix final	22
Étude des clusters.....	25
Argument distance.....	25
Argument méthode	25
Regroupement des échantillons.....	26
Analyse des régions différentiellement méthylées.....	27
Régions hyper/hypo méthylées	28
Annotation régions différentiellement méthylées	29
Information des informations d'annotation	29

Conversion fichier GFF en format BED	30
Gffread.....	31
Gff2bed.....	31
GenomeTools	31
Awk.....	31
Utilisation du GFF3 directement	32
Base de données de transcrits (TxDB)	32
Récupération des informations dans TxDB	32
Objet GRanges sur les informations génomiques.....	33
Objet GRanges sur les régions différemment méthylées	33
Pourcentage des régions différemment méthylées annotée	33
Annotation des régions différemment méthylées	33
Commande classique d'annotation	34
Programme d'annotation.....	34
Préparation des données	34
Code python : Vérification des arguments	35
Code python : analyses des arguments de la commande	36
Code python : création de variables	37
Code python : stocker les informations des gènes	37
Code python : obtenir l'annotation	37
Affichage des résultats	38
Conclusion	39
Perspectives.....	40
Fiabilité des résultats.....	40
Ontologie des gènes	40

Table des Tableaux

Tableau 1 : visualisation des données sous forme de tableau.....	7
Tableau 2 : récapitulatif de la taille des fichiers de séquençage.....	8
Tableau 3 : récapitulatif du nombre de <i>read</i> supprimés lors du <i>trimming</i>	8
Tableau 4 : récapitulatif des résumés d'alignement de tous les échantillons.....	9
Tableau 5 : récapitulatif des déuplications des échantillons.....	11
Tableau 6 : récapitulatif des fichiers dédupliqués.....	12
Tableau 7 : récapitulatif des résultats des biais de séquençage des échantillons.....	13
Tableau 8 : Récapitulatif des suppressions après filtre contexte CpG.....	19
Tableau 9 : Récapitulatif des suppressions après filtre contexte CHG.....	19
Tableau 10 : Récapitulatif des suppressions après filtre contexte CHH.....	20
Tableau 11 : Nombre de méthylation dans les échantillons fusionnés.....	21
Tableau 12 : différence de méthylation pour les échantillons.....	28
Tableau 13 : récapitulatif des régions hyper / hypo méthylées.....	29
Tableau 14 : récapitulatif des gènes annotés.....	39

Table des Figures

Figure 1 : structure moléculaire d'une cytosine et de la 5-méthylcytosine.....	2
Figure 2 : transfère de la méthylation lors de la réPLICATION.....	2
Figure 3 : L'Impact de la Méthylation de l'ADN sur le Blocage des protéines.....	3
Figure 4 : La méthylation et son impact sur les histones.....	3
Figure 5 : RG108.....	5
Figure 6 : Rapport d'alignement final de l'échantillon S4_CT_3A.....	10
Figure 7 : Résultats des potentiels biais de séquençage.....	13
Figure 8 : information de méthylation séparé par contexte.....	14
Figure 9 : liste des fichiers importés dans Rstudio.....	15
Figure 10 : lecture de méthylation de chaque échantillon.....	15
Figure 11 : Contenu du fichier MethRead.....	16
Figure 12 : les commandes pour filtrer et normaliser les échantillons.....	16
Figure 13 : statistique du pourcentage de méthylation CT_cpg.....	17
Figure 14 : Graphique du pourcentage de méthylation CT_CpG.....	17
Figure 15 : Statistiques du pourcentage de la couverture.....	18
Figure 16 : histogramme du pourcentage de couverture S4_CT_3A.....	18
Figure 17 : fusion des échantillons de contrôle.....	21
Figure 18 : analyse en composante principale pour les contrôles dans le contexte CpG.....	23
Figure 19 : distribution en composante principale pour échantillons de contrôle CpG.....	23
Figure 20 : analyse en composante principale pour les traités dans le contexte CpG.....	24
Figure 21 : distribution en composante principale pour échantillons traités contexte CpG.....	24
Figure 22 : répartitions en groupes pour le contexte de méthylation en CpG.....	26
Figure 23 : répartitions en groupes échantillon traités pour le contexte CpG.....	27
Figure 24 : répartition des différences de méthylation contexte CpG.....	29
Figure 25 : téléchargement des fichiers possibles.....	30
Figure 26 : contenu du fichier GFF3.....	30
Figure 27 : résultat du validateur de fichier GFF3.....	31
Figure 28 : résultat du awk.....	32
Figure 29 : Pourcentage d'annotation entre diff10 et les infos.....	33
Figure 30 : Vérification des arguments.....	35
Figure 31 : erreur lorsqu'un argument non répertorié est inscrit.....	36
Figure 32 : Affichage de l'aide du programme.....	36
Figure 33 : erreur quand il n'y a pas assez d'argument.....	36
Figure 34 : déclaration de variables.....	37
Figure 35 : résultat d'annotation.....	38

Introduction

Bombus terrestris

Le bourdon terrestre (*Bombus terrestris*) est un insecte hyménoptère fascinant appartenant à l'ordre des hyménoptères, tout comme les abeilles, les guêpes et les fourmis. Les bourdons terrestres jouent un rôle essentiel dans l'écosystème en tant que polliniseurs, contribuant ainsi à la reproduction des plantes à fleurs en transportant le pollen entre les fleurs.

Chaque groupe de bourdons terrestres trouve son origine dans une femelle fondatrice, appelée reine, qui érige un nid. Ces ensembles peuvent réunir jusqu'à 500 individus. Les reines entrent en quiescence estivale, une période de ralentissement métabolique en réponse aux conditions environnementales défavorables, et établissent de nouvelles colonies à l'automne. Ces colonies engendrent des mâles et une nouvelle génération de reines, qui après fécondation, hibernent. [1]

De manière similaire aux abeilles, les bourdons présentent un système haplodiploïde. Les femelles bourdons sont diploïdes, ayant deux ensembles de chromosomes provenant d'œufs fécondés. En revanche, la grande majorité des mâles bourdons sont haploïdes, possédant un seul ensemble de chromosomes issus d'œufs non fécondés. Les haploïdes ont un seul exemplaire de chaque chromosome, tandis que les diploïdes ont des paires de chromosomes. [2]

Les castes de femelles, les castes sont différentes catégories ou types de membres d'une colonie, qui se spécialisent dans des tâches particulières. Les reines reproductrices et ouvrières non reproductrices, fournissent une illustration spectaculaire de ce point. Les reines et les ouvrières des hyménoptères sociaux présentent la plus grande différence intraspécifique de durée de vie jamais observée chez les animaux. Par exemple, chez certaines espèces de fourmis, les reines peuvent vivre plus de 20 ans, alors que les ouvrières meurent après seulement quelques mois.

La méthylation de l'ADN

La méthylation de l'ADN est un processus chimique naturel essentiel qui implique le transfert d'un groupe méthyle, composé d'un atome de carbone lié à trois atomes d'hydrogène (CH₃), sur les bases d'ADN.

Les modifications chimiques de l'ADN

Le processus de méthylation de l'ADN est orchestré par une enzyme cruciale appelée ADN méthyltransférase (DNMT), également connue sous le nom de méthyltransférase. Cette enzyme joue un rôle essentiel en catalysant l'ajout d'un groupement méthyle (CH₃) sur le carbone 5 de la cytosine.

Ce mécanisme chimique aboutit à la formation de la 5-méthylcytosine, une modification spécifique de la base cytosine de l'ADN. Il est important de noter que la méthylation de l'ADN est un processus dynamique et réversible. Des mécanismes enzymatiques spécifiques sont en place pour éliminer les groupements méthyle.

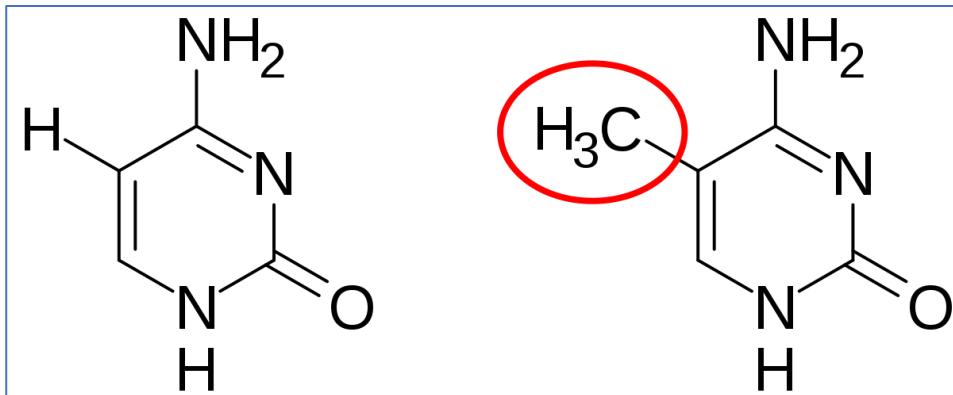


Figure 1 : structure moléculaire d'une cytosine et de la 5-méthylcytosine

La figure 1 montre la methyltransferase ajouter un groupement méthyle sur la cytosine. Les ADN polymérasées, également appelées ADNtransférases, jouent un rôle crucial dans le processus de réplication de l'ADN. Leur fonction essentielle consiste à synthétiser de nouveaux brins d'ADN complémentaires en utilisant un brin matrice préexistant comme guide.

Transmission de la méthylation

La méthylation de l'ADN commence par une instauration de novo, puis est préservée au fil des divisions cellulaires grâce à l'intervention des enzymes appartenant à la famille des ADN méthyltransférases. Les recherches récentes ont apporté des éclaircissements quant à l'implication de DNMT3a et DNMT3b dans la méthylation de novo de l'ADN, tout en excluant leur participation dans le maintien lors des divisions cellulaires. Ce dernier rôle est exclusivement assuré par DNMT1. [3]

La méthylation de la cytosine s'effectue sur les 2 brins de l'ADN lorsqu'elle est située à côté d'une guanine. Cette symétrie permet ainsi la transmission semi-conservatrice de la méthylation de l'ADN.

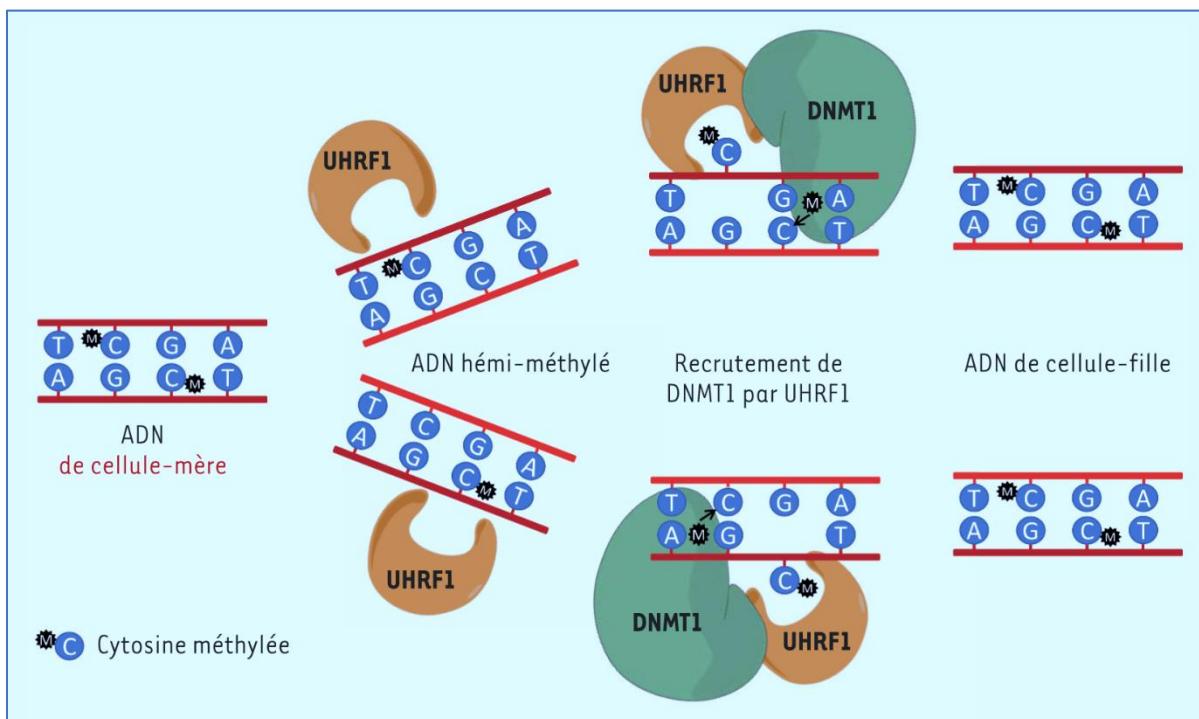


Figure 2 : transfère de la méthylation lors de la réplication

La figure 2 représente la coopération entre UHRF1 et DNMT1 pour la réPLICATION de la méthylation de l'ADN. Dans la cellule mère, l'ADN présente des méthylations sur ses deux brins au niveau des sites CpG. Lorsque l'ADN est dupliqué, un nouveau brin est synthétisé, mais ses cytosines ne sont pas méthylées. Cela entraîne la formation d'un ADN hémiméthylé, c'est-à-dire un seul des deux brins est méthylé.

Les sites CpG hémiméthylés sont reconnus par la protéine UHRF1. À la suite de cette reconnaissance, un processus est enclenché pour déplacer la cytosine méthylée hors de la double hélice de l'ADN. Cette étape permet à la protéine UHRF1 de ralentir ou de marquer une pause, probablement dans le but de permettre une interaction ou un dialogue avec l'enzyme DNMT1. Ce dialogue vise à indiquer à DNMT1 quelles cytosines doivent être méthylées sur l'autre brin d'ADN. [4]

Les histones sont des protéines basiques s'associant à l'ADN pour former la structure de base de la chromatine. Les histones jouent un rôle important dans l'empaquetage et le repliement de l'ADN. Chromatine Substance de base des chromosomes constituée de la molécule d'ADN associée à des protéines nommées histones, autour desquelles elle s'enroule.

Influence de l'expression génique

On sait aujourd'hui que les gènes peuvent être « allumés » ou « éteints » par plusieurs types de modifications chimiques qui ne changent pas la séquence de l'ADN comme des méthylations de l'ADN et des modifications des histones, ces protéines sur lesquelles s'enroule l'ADN pour former la chromatine. Toutes ces modifications constituent autant de « marques épigénétiques » regroupées sous le terme d'épigénome.

La méthylation de l'ADN implique l'ajout de groupes chimiques CH₃ sur certains points du brin d'ADN. Ces groupes chimiques agissent comme des obstacles en empêchant les protéines de se lier à l'ADN et de le déchiffrer. En conséquence, les gènes concernés sont inactivés lors du processus de méthylation. [5]

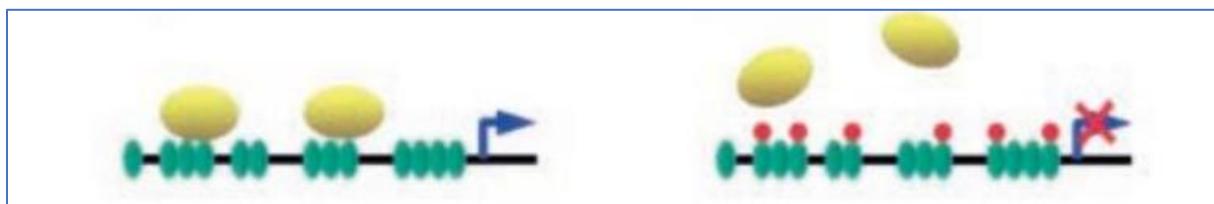


Figure 3 : L'Impact de la Méthylation de l'ADN sur le Blocage des protéines

La figure 3 représente le blocage des protéines lorsqu'il y a des méthylations sur le brin (en rouge). Un autre exemple de modification épigénétique concerne les histones. La manière dont l'ADN est enroulé autour des histones joue un rôle crucial dans la régulation génique. Lorsque l'ADN est fortement enroulé, les gènes sont en position éteinte et ne peuvent pas être activés. En revanche, lorsque l'ADN est moins enroulé, les gènes peuvent être activés. Cette configuration est influencée par l'ajout ou le retrait de groupes chimiques sur les histones, ce qui modifie leur densité. Ces changements dans la structure des histones peuvent activer ou désactiver les gènes en facilitant ou en entravant la liaison des protéines à l'ADN.

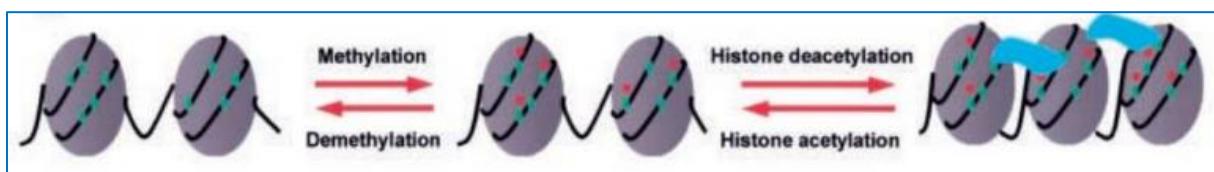


Figure 4 : La méthylation et son impact sur les histones

La régulation de l'expression génique chez les eucaryotes implique des changements dans la structure de la chromatine, qui est le matériau génétique compacté. Les nucléosomes, composés d'ADN enroulé autour de protéines appelées histones, forment l'unité de base de cette structure. Des enzymes modifient les histones par des ajouts chimiques, créant un code épigénétique.

La figure 4 montre les protéines de liaison à la méthylcytosine liées aux CpG dans la région promotrice, également appelées MBPs (*methyl-CpG-binding domain*), forment des complexes avec des histones désacétylases, c'est-à-dire qu'ils ont perdu un groupement acétyle, ainsi qu'avec des corépresseurs (protéines interagissant avec les facteurs de transcription et les complexes protéiques pour réprimer l'expression des gènes). Cette association entraîne la désacétylation des histones, ce qui conduit à la condensation de la chromatine et à la formation d'une structure chromatinienne transcriptionnellement inactive.

Ces altérations chimiques constituent l'une des composantes essentielles de l'épigénétique. L'épigénétique est le domaine d'étude qui se penche sur les mécanismes régissant de manière réversible, transmissible et adaptative l'expression des gènes, indépendamment de la séquence d'ADN.

Cette régulation de l'expression génique peut intervenir à divers niveaux qui interagissent entre eux : d'une part, par le biais de l'organisation tridimensionnelle de l'ADN à l'intérieur du noyau cellulaire, et d'autre part, via la compaction locale de l'ADN au sein de la chromatine, enfin par la modification chimique de l'ADN.

Traitement bisulfite

En préparation de l'étape de traitement au bisulfite, plusieurs phases sont nécessaires. Tout d'abord, il y a l'extraction de l'ADN, qui dans cette étude, a consisté en une fragmentation complète de l'ADN du bourdon. Ensuite, on réalise la dénaturation de l'ADN, ce qui signifie séparer les brins d'ADN en brins simples, facilitant ainsi l'interaction du bisulfite avec les bases de l'ADN. Cette dénaturation peut être accomplie en chauffant l'ADN à une température élevée ou en utilisant des agents chimiques.

Une fois l'ADN dénaturé, il est exposé à une solution de bisulfite. Le bisulfite pénètre les brins simples de l'ADN et réagit spécifiquement avec les cytosines non méthylées, les transformant en uraciles. Les cytosines méthylées résistent au traitement au bisulfite. Après la dénaturation, l'ADN subit des étapes de purification et de récupération pour éliminer les résidus de bisulfite et les contaminants. L'ADN ainsi préparé est ensuite prêt à être utilisé dans une réaction de PCR.

A la suite du traitement au bisulfite et la PCR, une molécule d'ADN peut se scinder en deux molécules distinctes. Dans l'une de ces molécules, les cytosines sont converties en thymines, tandis que dans l'autre molécule, les guanines sont converties en adénines.

Dans le contexte du projet de recherche portant sur la méthylation de l'ADN et sa relation avec le vieillissement, une technique de séquençage appelée BSWGS (*Bi-Sulfite Whole Genome Sequencing*) est employée afin d'évaluer l'effet d'un traitement particulier sur la méthylation de l'ADN. Cette approche permet d'analyser l'intégralité du génome après un traitement au bisulfite, permettant ainsi de détecter d'éventuelles modifications de méthylation qui pourraient se produire.

Objectif du travail

Les effets de la méthylation de l'ADN sur la longévité du bourdon *Bombus terrestris* ont été évalués au cours de cette étude. Pour ce faire, des modifications de la méthylation de l'ADN ont été induites chez les jeunes bourdons, et les conséquences sur la survie, les motifs de méthylation à l'échelle du génome ont été analysées. Les biologistes ont observé que le traitement expérimental augmentait la durée de vie moyenne des ouvrières de 43%. Des recherches récentes ont montré que l'accumulation d'altérations épigénétiques au cours de la vie est un moteur particulièrement important du processus de vieillissement. [6]

L'objectif principal consiste à analyser les changements potentiels dans les schémas de méthylation de l'ADN, mettant en évidence les différences qui pourraient découler du traitement spécifique.

Les résultats des essais expérimentaux indiquent de manière significative que lorsqu'un bourdon âgé d'une semaine est traité avec l'inhibiteur de méthylation RG108, également connu sous le nom de N-phényl-L-tryptophane, et que cet inhibiteur est appliqué sur l'abdomen de l'ouvrière, la durée de vie de ces bourdons est considérablement augmentée. En comparaison avec les individus témoins, dont la durée de vie est de 30 jours, les bourdons traités atteignent une longévité remarquable de 46 jours. C'est cette observation qui a motivé mon travail de recherche.

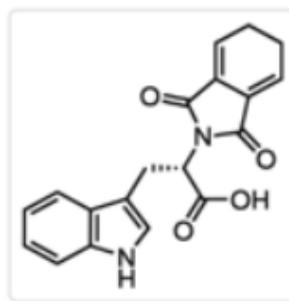


Figure 5 : RG108

L'agent hypométhylant RG108 à la figure 5 prolonge la durée de vie des ouvrières. Dans des conditions de laboratoire, les ouvriers des bourdons ont une durée de vie de 4 à 5 semaines. Pour étudier le rôle du DNA méthylé dans la régulation de la durée de vie, nous avons traité de jeunes ouvrières de *B. terrestris* avec RG108, un agent pharmacologique hypométhylant connu pour ses effets antivieillissement.

Logiciels utilisés

Logiciels de traitement de données

Trim_Galore

Trim-Galore est un outil essentiel pour le prétraitement des données de séquençage, ayant la capacité de retirer les adaptateurs, les séquences de faible qualité, et de découper les lectures en fonction de leur longueur et qualité. Cet outil repose sur le programme Cutadapt et se distingue par sa pertinence pour les séquences bisulfites, ce qui le rend particulièrement adapté à l'analyse de la méthylation de l'ADN. En somme, Trim-Galore fournit une solution complète visant à améliorer la qualité des données de séquençage en amont des étapes analytiques subséquentes.

Bismark

Bismark est un outil informatique essentiel dans le domaine du séquençage bisulfite, utilisé pour aligner les séquences issues du séquençage sur un génome de référence. De plus, il permet de quantifier la méthylation de l'ADN en utilisant les données d'alignement obtenues. Bismark s'appuie sur des ressources telles que Bowtie2 et Samtools pour accomplir ces

tâches. En résultat, il produit des fichiers de sortie qui sont ensuite employés pour des analyses approfondies de la méthylation.

MethylDackel

MethylDackel joue un rôle essentiel dans le domaine du séquençage bisulfite en tant qu'outil clé pour extraire les informations de méthylation à partir des données de séquençage. Son utilité réside dans la détermination du statut de méthylation des cytosines spécifiques. Ce programme repose sur des modèles probabilistes afin d'estimer la probabilité de méthylation associée à chaque site cytosine. Dans l'ensemble, MethylDackel constitue une solution puissante pour analyser la méthylation de l'ADN en se basant sur les données de séquençage bisulfite, fournissant des indications détaillées sur l'état de méthylation des cytosines individuelles.

Vérification des données

Fastqc

Lors des séances de formation en bioinformatique, une attention particulière a été accordée à l'utilisation de la version 0.11.9 du logiciel Fastqc. Cet outil s'est avéré précieux pour évaluer la qualité des données issues des séquençages. À travers une variété de graphiques informatifs, Fastqc permet de visualiser la qualité des données et d'identifier d'éventuelles anomalies. Cette visualisation s'avère cruciale pour déterminer si une étape de *trimming*, impliquant le retrait de parties spécifiques des données, est nécessaire. En fournissant ces informations visuelles, Fastqc facilite grandement la prise de décisions relatives aux étapes ultérieures de l'analyse des données, tout en se basant sur leur qualité originale.

Qualimap

Qualimap constitue un logiciel essentiel dans l'évaluation de la qualité de l'alignement des séquences, en se basant sur l'analyse des fichiers BAM ou SAM. Son rôle consiste à identifier d'éventuels biais dans le séquençage ou le *mapping* des séquences, tout en offrant la possibilité de comparer plusieurs échantillons entre eux. Avant de recourir à Qualimap, une étape préliminaire indispensable consiste à trier les séquences selon leurs coordonnées à l'aide de l'outil Samtools sort. Une fois ce tri effectué, les fichiers résultants sont utilisables au sein de Qualimap pour entreprendre une analyse de qualité approfondie.

Langage de programmation

R sur Rstudio

MethylDackel représente un logiciel essentiel dans le domaine de l'analyse de la méthylation à partir des données de séquençage bisulfite. Son utilité réside dans sa capacité à extraire des informations précieuses concernant la méthylation en vue de déterminer si une cytosine spécifique est méthylée ou non méthylée. Ce programme recourt à des modèles probabilistes pour évaluer la probabilité de méthylation pour chaque site cytosine. En somme, MethylDackel se positionne comme un outil robuste permettant d'analyser en profondeur la méthylation de l'ADN en tirant parti des données issues du séquençage bisulfite, et de fournir des détails pertinents sur l'état de méthylation des cytosines individuelles.

Python

Il s'agit d'un langage de programmation polyvalent, réputé pour sa simplicité et sa popularité, qui trouve des applications dans de nombreux domaines tels que le développement web, l'analyse de données et l'intelligence artificielle. Grâce à sa syntaxe claire et à ses bibliothèques étendues, ce langage encourage la lisibilité du code ainsi que l'adoption de bonnes pratiques de programmation.

Ordinateur de travail

Ordinateur personnel

Une partie du travail a été réalisée sur un ordinateur personnel avec 12 Go de RAM, 4 cœurs et 2 Go de swap. Le système d'exploitation est linux mint.

Ordinateur de l'ULB

Le reste du travail a été effectué en parallèle sur un ordinateur de bureau disposant de 64 Go de RAM, 12 cœurs et 2 Go de swap. Le système d'exploitation est linux mint.

Résultats

L'entreprise B.G.I., basée à Shanghai, a réalisé le traitement bisulfite et le séquençage de 12 génomes de bourdons. Chaque génome était accompagné de deux fichiers distincts : un fichier *forward* et un fichier *reverse*. Cependant, pour certains individus, la taille des fichiers était trop importante pour être traitée par le séquenceur. Cette contrainte a conduit à la division des fichiers *forward* et *reverse* en deux parties distinctes. Cette situation s'est produite pour 4 des 12 individus, ce qui a généré un total de 32 fichiers.

Les échantillons dont les fichiers ont été divisés en deux parties sont identifiables grâce à leur nom de dossier se terminant par la lettre A comme S1_RG_2A, qui indique un échantillon de la semaine 1 traité avec l'inhibiteur RG108 et correspondant à l'échantillon 2A. Les noms de dossiers ont été attribués en fonction de l'âge, du traitement et du numéro d'échantillon, par exemple S1_CT_2B pour un échantillon de la semaine 1 non traité (témoin) et correspondant à l'échantillon 2B.

	Semaine 1	Semaine 4
Traités	3 bourdons	3 bourdons
Non traités	3 bourdons	3 bourdons

Tableau 1 : visualisation des données sous forme de tableau

Le tableau 1 montre les données que nous avons reçues sont structurées en fonction de différentes conditions : deux groupes d'âges distincts (1 ou 4 semaines) et pour chaque groupe d'âge, deux conditions spécifiques correspondant aux individus ayant été traités ou non traités. Chaque condition englobe trois répliques, ce qui totalise 12 bourdons au total. L'ensemble de ces échantillons occupe un espace de 127.6 gigaoctet.

Le tableau 2 montre un récapitulatif de la taille des fichiers de séquençage, il tourne autour des 11G. Tout au long du stage, le travail s'est concentré sur les échantillons de la semaine 1. À ce stade, il est essentiel d'appliquer les mêmes procédures et analyses aux échantillons de la semaine 4. Le choix de travailler avec les échantillons de la semaine une et de la semaine quatre repose sur des considérations biologiques importantes pour les bourdons.

	Semaine 1	Semaine 4
Traités	11,9G 11,0G 10,6G	14,3G 9,2G 9,2G
Non traités	8,2G 10,1G 9,4G	12,1G 9,5G 12,1G

Tableau 2 : récapitulatif de la taille des fichiers de séquençage

Au cours de la première semaine, le bourdon atteint son pic de maturité, ce qui signifie qu'il est à son apogée en termes d'attrait pour les reines. À ce stade, les phéromones émises par le bourdon sont particulièrement attractives, et sa qualité spermatique est à son optimum. D'un autre côté, la semaine quatre correspond à la fin de la vie du bourdon.

À ce moment-là, son odeur n'est plus aussi attrayante pour les reines et sa qualité spermatique se dégrade. Cette période marque le déclin de sa capacité à se reproduire efficacement. En choisissant ces deux moments spécifiques dans la vie du bourdon, on peut mieux comprendre les changements dans les profils de méthylation et leur lien avec les changements biologiques importants tels que la maturité et la qualité spermatique. Cela permet d'explorer comment la méthylation peut être associée à ces transitions clés dans le cycle de vie du bourdon.

Trim_Galore

La commande permettant de réaliser le *trimming* des séquences brutes a été déterminée lors de mon stage au sein de l'Université Libre de Bruxelles. [7]

```
Trim_galore -q 30 --phred33 --fastqc --trim-n --trim1 --paired fichier_1.fq.gz fichier_2.fq.gz
```

Échantillons	Brutes	Supprimées	Restantes	Suppression %
S1_CT_2B	53.329.569	250.246	53.079.323	0,47%
S1_CT_3B	66.236.080	295.986	65.940.094	0,45%
S1_CT_4B	62.169.283	251.661	61.917.622	0,40%
S1_RG_2A	78.200.013	335.332	77.864.681	0,43%
S1_RG_4B	72.587.929	315.171	72.272.758	0,43%
S1_RG_5B	69.233.584	382.149	68.851.435	0,55%
S4_CT_3A	78.098.598	375.756	77.722.842	0,48%
S4_CT_4B	61.812.750	279.026	61.533.724	0,45%
S4_CT_5A	78.067.160	392.627	77.674.533	0,50%
S4_RG_1A	90.176.291	482.385	89.693.906	0,53%
S4_RG_2B	60.363.785	254.914	60.108.871	0,42%
S4_RG_5B	60.087.396	298.274	59.789.122	0,50%

Tableau 3 : récapitulatif du nombre de *read* supprimés lors du *trimming*

Le tableau 3 montre un récapitulatif du nombre de *reads* initiaux c'est-à-dire le nombre de *read* dans chacune des séquences brutes par rapport au nombre de *read* après la phase de *trimming*. Le nombre de séquences supprimées par expérience est indiqués mais également le pourcentage de suppression qu'il représente par rapport aux séquences initiales.

Pour déterminer le pourcentage de séquences supprimées en fonction du nombre de *reads* initiaux. Un calcul basique a été réalisé :

$$\left(\frac{\text{Nombre des séquences supprimées}}{\text{Nombre de séquences initiales}} \right) * 100 = \text{Pourcentage des séquences supprimées}$$

L'échantillon avec le plus de séquences supprimées est l'échantillon S4_CT_3A, cependant, lorsque l'on réalise le pourcentage de séquences supprimées par rapport au nombre de séquences initiales, c'est l'échantillon S1_RG_5B qui supprime le plus de séquences par rapport au nombre de séquences initiales.

Alignement des reads

Comme lors de mon travail de stage, j'utilise le programme Bismark. La commande a également été choisie lors de ce travail [8].

```
Bismark --fastq --un --ambiguous --genome_folder ../genome_ref/
-1 fichier_1.fq.gz -2 fichier_2.fq.gz
```

Après chaque alignement des *reads*, Bismark crée un rapport d'alignement.

Échantillons	Mapping	Ambigus	Non-mappés	CpG	CHG	CHH	Inco
S1_CT_2B	61,5%	8.544.818	11.911.456	0,8%	0,6%	0,7%	0,6%
S1_CT_3B	58,9%	11.380.013	15.732.893	0,8%	0,5%	0,5%	0,5%
S1_CT_4B	62,9%	8.764.013	14.238.230	0,7%	0,5%	0,5%	0,5%
S1_RG_2A	66,4%	13.180.396	12.994.884	0,7%	0,5%	0,5%	0,4%
S1_RG_4B	66,0%	12.203.929	12.375.561	0,7%	0,4%	0,4%	0,4%
S1_RG_5B	66,4%	11.806.251	11.306.771	0,7%	0,4%	0,5%	0,4%
S4_CT_3A	67,2%	14.094.140	11.377.001	0,7%	0,5%	0,5%	0,5%
S4_CT_4B	61,5%	12.599.675	8.855.041	0,8%	0,5%	0,5%	0,5%
S4_CT_5A	64,2%	14.403.130	13.431.291	0,7%	0,5%	0,5%	0,5%
S4_RG_1A	61,5%	22.395.237	15.276.313	0,7%	0,5%	0,6%	0,5%
S4_RG_2B	50,5%	21.741.205	7.994.165	0,8%	0,5%	0,5%	0,5%
S4_RG_5B	65,7%	10.848.322	9.687.444	0,8%	0,5%	0,5%	0,5%

Tableau 4 : récapitulatif des résumés d'alignement de tous les échantillons

Le tableau 4 constitue un résumé des taux d'alignement pour tous les échantillons. Plusieurs informations cruciales y figurent. Parmi elles, le pourcentage de séquences alignées sur le génome de référence. Les séquences ambiguës, désignent les *reads* qui peuvent être alignées à plusieurs emplacements dans le génome et qui, de ce fait, appartiennent à l'ensemble des séquences de bourdons.

Les séquences non alignées, cependant, ne présentent aucune correspondance avec le génome de référence. Cette situation pourrait découler d'une contamination ou de la présence de bactéries à la surface des bourdons broyés. Cette issue est attribuée à la méthode d'extraction de l'ADN. À ce stade, des données sur la méthylation sont déjà disponibles, séparées en différents contextes de méthylation. Le contexte de méthylation inconnu reflète des cas où le programme n'est pas en mesure d'identifier spécifiquement le contexte de méthylation. Ultérieurement, la méthylation inconnue reste irrécupérable.

Rapport d'alignement

À titre illustratif, le choix s'est porté sur le rapport d'alignement qui a démontré le plus haut taux de *mapping* avec le génome de référence. Certaines séquences chromosomiques ne peuvent pas être extraites correctement dans l'exemple à la figure, 17 n'ont pas été extraits. Cela peut être dû à plusieurs raisons, telles que des régions génomiques complexes ou répétitives, des erreurs de séquençage ou d'autres facteurs techniques. Certains segments du génome peuvent présenter une forte répétitivité ou une complexité élevée, ce qui complique l'alignement précis des séquences brutes. Les outils d'alignement peuvent rencontrer des difficultés pour déterminer la position adéquate de ces séquences. Dans le cas où les appariements des séquences sont incorrects ou de médiocre qualité lors de la mise en paire, cela peut engendrer des problèmes durant l'alignement.

```
Final Alignment report
=====
Sequence pairs analysed in total:      77722842
Number of paired-end alignments with a unique best hit: 52251701
Mapping efficiency:      67.2%
Sequence pairs with no alignments under any condition: 14094140
Sequence pairs did not map uniquely:    11377001
Sequence pairs which were discarded because genomic sequence could not be extracted:    17

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      26226412      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      26025272      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:    0

Final Cytosine Methylation Report
=====
Total number of C's analysed: 1839098222

Total methylated C's in CpG context: 4002508
Total methylated C's in CHG context: 1406929
Total methylated C's in CHH context: 5628478
Total methylated C's in Unknown context:      2164

Total unmethylated C's in CpG context: 538537128
Total unmethylated C's in CHG context: 271092394
Total unmethylated C's in CHH context: 1018430785
Total unmethylated C's in Unknown context:      450481

C methylated in CpG context: 0.7%
C methylated in CHG context: 0.5%
C methylated in CHH context: 0.5%
C methylated in unknown context (CN or CHN): 0.5%

Bismark completed in 0d 7h 15m 8s
```

Figure 6 : Rapport d'alignement final de l'échantillon S4_CT_3A

La figure 6 montre le rapport obtenu à la fin de l'alignement avec Bismark. Pour bien comprendre comment le programme calcul le pourcentage de méthylation que réalise Bismark. Le nombre total de cytosines analysées dans l'échantillon : 1 839 098 222. Le nombre de cytosines analysées dans un certain contexte. Ensuite on multiplie par 100, pour obtenir le pourcentage de méthylation. L'annexe 1 contient les rapports d'alignement de tous les échantillons.

$$\left(\frac{\text{Nombre de cytosines par contexte}}{\text{Nombre de cytosines totales}} \right) * 100 = \text{Pourcentage de méthylation par contexte}$$

Le programme fournit également une indication sur la durée de traitement, laquelle est influencée par le nombre de cœurs disponibles sur l'ordinateur et donc la possibilité de travailler en parallèle.

Déduplication des reads

La déduplication se réalise également avec le programme Bismark. La déduplication est un processus dans lequel les lectures dupliquées (identiques) dans les données de séquençage sont supprimées ou comptées une seule fois.

```
Deduplicate_bismark -p --bam fichier_bismark_pe.bam
```

Lors de la PCR, des duplications peuvent exister il est fortement conseillé de supprimer les séquences dupliquées. Dans le cas contraire, lors de l'analyse ultérieures les calculs pourraient être faussés. En comptant par exemple, la même cytosine méthylée dupliquée plusieurs fois.

Échantillons	Brutes	Supprimées	Positions ≠	Restantes	Suppression
S1_CT_2B	32.623.035	416.931	400.616	32.206.104	1,28%
S1_CT_3B	38.827.171	729.823	705.132	38.097.348	1,88%
S1_CT_4B	38.915.371	382.136	372.700	38.533.235	0,98%
S1_RG_2A	51.689.420	675.427	657.165	51.013.993	1,31%
S1_RG_4B	47.693.246	556.286	543.027	47.136.960	1,17%
S1_RG_5B	45.738.390	674.995	662.104	45.063.395	1,48%
S4_CT_3A	52.251.684	721.727	685.514	51.529.957	1,38%
S4_CT_4B	40.077.996	478.196	460.614	39.599.800	1,19%
S4_CT_5A	49.840.097	949.526	894.267	48.890.571	1,91%
S4_RG_1A	60.160.627	674.700	674.700	59.485.927	1,12%
S4_RG_2B	30.373.486	330.904	319.182	30.042.582	1,09%
S4_RG_5B	39.253.345	445.701	445.701	38.807.644	1,14%

Tableau 5 : récapitulatif des déuplications des échantillons

Lors de la déduplication de l'échantillon S4_RG_2B, il n'y avait que 2 millions de séquences restantes pourtant je devais en avoir 39 millions, j'ai donc pu me rendre compte qu'une erreur c'était produite en transférant mon fichier de l'ordinateur fixe à mon ordinateur personnel. J'ai donc rectifié le problème. Grâce au pourcentage d'alignement, il est facile de connaître le nombre de séquences qu'il doit y avoir pour la prochaine étape.

Le tableau 5 est un récapitulatif des rapports de déduplication de tous les échantillons. Le taux de déduplication tourne autour des 1% pour l'ensemble des échantillons. Ce pourcentage a été réalisé en utilisant la même équation citée précédemment qui tient compte du nombre de séquences supprimées en fonction du nombre de séquences initiales :

$$\left(\frac{\text{Nombre de séquences supprimées}}{\text{Nombre de séquences initiales}} \right) * 100 = \text{Pourcentage de séquences supprimées}$$

Un taux moyen de déduplication d'environ 1%, ce qui suggère une efficacité notable du séquençage et du traitement des données. Ce constat indique qu'environ 1% seulement des lectures dans les échantillons étaient des duplicates, une observation généralement associée à des données de qualité satisfaisante.

Tri par coordonnées

Pour trier un fichier BAM par coordonnées, il faut utiliser Samtools. Aucune recherche n'a été nécessaire pour ce programme. Nous l'avons vu en classe avec Monsieur Coornaert. Le tri par coordonnées réduit la taille des fichiers.

```
Samtools sort fichier_dedup.bam > fichier_dedup.s.bam
```

Échantillons	Taille BAM	Taille s.bam	Différence
S1_CT_2B	8,2 G	5,6 G	2,6 G
S1_CT_3B	9,8 G	6,5 G	3,3 G
S1_CT_4B	9,8 G	6,5 G	3,3 G
S1_RG_2A	14 G	8,5 G	5,5 G
S1_RG_4B	12 G	7,8 G	4,2 G
S1_RG_5B	12 G	7,7 G	4,3 G
<hr/>			
S4_CT_3A	14 G	8,6 G	5,4 G
S4_CT_4B	11 G	6,6 G	4,4 G
S4_CT_5A	13 G	8,1 G	4,9 G
S4_RG_1A	16 G	9,9 G	6,1 G
S4_RG_2B	7,7 G	5,2 G	2,5 G
S4_RG_5B	10 G	6,6 G	3,4 G

Tableau 6 : récapitulatif des fichiers dédupliqués

Le tableau 6 présente une synthèse des tailles des fichiers de séquençage au format BAM, comparées à celles des fichiers triés par coordonnées au format s.bam.

MethylDackel

MethylDackel mbias

Ce sous-programme de MethylDackel [9] permet de l'identification et la correction des biais de conversion de bisulfite, qui peuvent fausser les résultats de l'analyse de méthylation. C'est-à-dire que lors du traitement bisulfite, les cytosines non méthylées sont converties en uracile. Cependant, ce processus de conversion n'est pas parfait et peut introduire des biais de conversion.

L'option mbias examine les cytosines non méthylées et évalue combien d'entre elles sont converties en thymine pendant le processus de séquençage. Par le biais de ce calcul, le sous-programme élabore un profil de biais de conversion.

Ce profil met en lumière le degré de conversion des différents contextes de cytosine en thymine, offrant ainsi une indication quant à l'existence potentielle de biais systématiques de conversion dans les données.

L'analyse des profils de biais de conversion peut révéler des erreurs systématiques attribuables à la conversion bisulfite. Si des biais significatifs sont identifiés, il devient possible de les corriger lors de l'extraction de la méthylation via MethylDackel.

```
MethylDackel mbias ../genome_ref/genome.fa fichier.s.bam fichier_output
```

```
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_CT_4B$ MethylDackel mbias ../../genome_ref/genome.fa S4_CT_4B_bismark_pe.deduplicated.s.bam S4_CT_4B_mbias
[E:idx find and load] Could not retrieve index file for S4 CT 4B bismark_pe.deduplicated.s.bam
Couldn't load the index for S4 CT 4B bismark_pe.deduplicated.s.bam, will attempt to build it.
Suggested inclusion options: -OT 0,0,0 --OB 0,0,0
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_CT_4B$ cd ../../genome_ref/genome.fa S4_CT_5A_S4_CT_5B_mbias
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_CT_5A$ MethylDackel mbias ../../genome_ref/genome.fa S4_CT_5A_S4_CT_5B_mbias
S4 CT 5A 1 val 1 bismark_bt2_pe.deduplicated.bam S4 CT 5A 1 val 1 bismark_bt2_pe.deduplicated.s.bam
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_CT_5A$ MethylDackel mbias ../../genome_ref/genome.fa S4_CT_5A_bismark_bt2_pe.deduplicated.s.bam S4_CT_5B_mbias
[E:idx find and load] Could not retrieve index file for S4 CT 5A bismark_bt2_pe.deduplicated.s.bam
Couldn't load the index for S4 CT 5A bismark_bt2_pe.deduplicated.s.bam, will attempt to build it.
Suggested inclusion options: -OT 0,0,0 --OB 0,0,0
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_CT_5A$ cd ../../genome_ref/genome.fa S4_RG_1A_S4_RG_1A_mbias
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_RG_1A$ MethylDackel mbias ../../genome_ref/genome.fa S4_RG_1A_S4_RG_1A_mbias
S4 RG 1A 1 val 1 bismark_bt2_pe.deduplicated.bam S4 RG 1A 1 val 1 bismark_bt2_pe.deduplicated.s.bam
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_RG_1A$ MethylDackel mbias ../../genome_ref/genome.fa S4_RG_1A_S4_RG_1A_mbias
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_RG_1A$ cd ../../genome_ref/genome.fa S4_RG_1A_bismark_deduplicated.s.bam S4_RG_1A_mbias
[E:idx find and load] Could not retrieve index file for S4 RG 1A bismark_deduplicated.s.bam
Couldn't load the index for S4 RG 1A bismark_deduplicated.s.bam, will attempt to build it.
Suggested inclusion options: -OT 0,0,0 --OB 0,0,0
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_RG_1A$ cd ../../genome_ref/genome.fa S4_RG_2B_S4_RG_2B_mbias
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_RG_2B$ MethylDackel mbias ../../genome_ref/genome.fa S4_RG_2B_bismark_deduplicated.s.bam S4_RG_2B_mbias
[E:idx find and load] Could not retrieve index file for S4 RG 2B bismark_deduplicated.s.bam
Couldn't load the index for S4 RG 2B bismark_deduplicated.s.bam, will attempt to build it.
Suggested inclusion options: -OT 0,0,0 --OB 0,0,0
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_RG_2B$ cd ../../genome_ref/genome.fa S4_RG_5B_S4_RG_5B_mbias
(methyldackel) camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Camille/bourdons/deduplication/S4_RG_5B$ MethylDackel mbias ../../genome_ref/genome.fa S4_RG_5B_bismark_deduplicated.s.bam S4_RG_5B_mbias
[E:idx find and load] Could not retrieve index file for S4 RG 5B bismark_deduplicated.s.bam
Couldn't load the index for S4 RG 5B bismark_deduplicated.s.bam, will attempt to build it.
Suggested inclusion options: -OT 0,0,0 --OB 0,0,0
```

Figure 7 : Résultats des potentiels biais de séquençage

La Figure 7 montre le résultat obtenu en exécutant cette commande, montrant les valeurs potentielles de biais de séquençage. Il est important de noter qu'aucun biais de séquençage n'est observé sur l'ensemble des échantillons.

Échantillons	OT	OB	Biais
S1_CT_2B			
S1_CT_3B			
S1_CT_4B			
S1_RG_2A	0,0,0,0	0,0,0,0	/
S1_RG_4B			
S1_RG_5B			
S4_CT_3A			
S4_CT_4B			
S4_CT_5A			
S4_RG_1A	0,0,0,0	0,0,0,0	/
S4_RG_2B			
S4_RG_5B			

Tableau 7 : récapitulatif des résultats des biais de séquençage sur l'ensemble des échantillons

Le tableau 7 regroupe les résultats de l'analyse des biais de séquençage. Il n'y en pas un seul dans les échantillons. Lors de l'extraction de la méthylation, avec un autre sous-programme de MethylDackel, un paramètre supprime ces biais détectés. Dans ce cas, il n'est pas nécessaire d'utiliser ces paramètres étant donné qu'il n'y en a pas.

MethylDackel extract

Ce sous-programme permet de récupérer les informations de méthylation. Le paramètre très intéressant de ce sous-programme est le paramètre: --methylKit, il permet de mettre le fichier de sortie au format spécifique de MethylKit.

```
MethylDackel extract --methylKit --CHG --CHH genome_ref/genome.fa fichier.s.bam
```

Une fois cette étape terminée, il y a 3 fichiers par expérience, CpG CHG et CHH chacun au format spécifique de methylKit. Pour l'analyse ultérieur avoir les contextes séparés est plus simple et le temps de travail individuellement est plus rapide.

- S4_CT_3A_bismark_deduplicated.s_CHG.methylKit
- S4_CT_3A_bismark_deduplicated.s_CHH.methylKit
- S4_CT_3A_bismark_deduplicated.s_CpG.methylKit

Figure 8 : information de méthylation séparé par contexte

La figure 8 montre les 3 fichiers obtenus à la fin de l'extraction de la méthylation avec MethylDackel extract.

Après avoir effectué l'extraction, l'étape suivante consiste à procéder à l'analyse de la méthylation dans Rstudio.

Analyse de la méthylation

Pour aborder la problématique de mon travail de fin d'étude, il est nécessaire d'effectuer une analyse comparative du méthylome de bourdons en fonction de leur âge. Ainsi, la démarche entreprise consiste à confronter les bourdons ayant été exposés à l'inhibiteur de méthylation durant la première semaine avec ceux exposés pendant la quatrième semaine.

De plus, une comparaison est également réalisée entre les groupes témoins de bourdons de la première et de la quatrième semaine. À titre d'illustration dans la partie de l'analyse statistique, les échantillons de contrôle dans le contexte de méthylation CpG seront exposés.

Librairies

Avant d'entamer toute analyse au sein de Rstudio, il est nécessaire d'importer certaines librairies afin de pouvoir utiliser des fonctions spécifiques.

```
library(methylKit)
library(data.table)
library(ggplot2)
library(genomation)
library(GenomicFeatures)
```

La figure X montre la liste des librairies utilisée dans la suite des analyses. Dans le contexte des analyses réalisées dans R, différentes librairies jouent des rôles spécifiques.

MethylKit [10] est essentielle pour l'analyse des données de méthylation de l'ADN, en mettant l'accent sur la détection des régions où les niveaux de méthylation varient. La librairie data.table est couramment utilisée pour manipuler efficacement les données au format tabulaire. En ce qui concerne la création de graphiques et la visualisation de données, la librairie ggplot2 a été utilisée.

La librairie Genomation est utile pour annoter et visualiser les données génomiques. GenomicFeatures offre des outils pour manipuler et analyser diverses annotations génomiques telles que les gènes, les exons, les introns, et bien d'autres.

Pour installer certaines bibliothèques, deux méthodes sont disponibles. La première méthode est l'installation classique, qui consiste à utiliser la fonction `install.packages()`. La seconde méthode implique d'utiliser `BiocManager::install()`.

Importation des fichiers

Pour l'importation des fichiers dans Rstudio, la méthode adoptée consiste à établir une liste renfermant les fichiers à utiliser, séparé en fonction de leur contexte de méthylation.

```
file_CT_cpg <- list("/media/camille/Camille/bourdons/deduplication/Semaine_1/S1_CT_2B/dedup/S1_CT_2B_dedup.s_CpG.methylKit",
  "/media/camille/Camille/bourdons/deduplication/Semaine_1/S1_CT_3B/dedup/S1_CT_3B_dedup.s_CpG.methylKit",
  "/media/camille/Camille/bourdons/deduplication/Semaine_1/S1_CT_4B/dedup/S1_CT_4B_dedup.s_CpG.methylKit",
  "/media/camille/Camille/bourdons/deduplication/Semaine_4/S4_CT_3A/dedup/S4_CT_3A_bismark_deduplicated.s_CpG.methylKit",
  "/media/camille/Camille/bourdons/deduplication/Semaine_4/S4_CT_4B/S4_CT_4B_bismark_pe.deduplicated.s_CpG.methylKit",
  "/media/camille/Camille/bourdons/deduplication/Semaine_4/S4_CT_5A/S4_CT_5A_bismark_bt2_pe.deduplicated.s_CpG.methylKit")
```

Figure 9 : liste des fichiers importés dans Rstudio

La figure 9 montre la commande à effectuer pour stocker les différents fichiers prêts pour être utilisé dans la suite des analyses.

Préparation des données

L'importance de cette étape réside dans la lecture et la préparation des fichiers précédemment importés dans Rstudio. Cette opération aboutit à leur stockage dans une variable nommée `objet_CT_cpg`. L'objet en question est donc un objet `MethylRawList`.

```
objet_CT_cpg <- methRead(file_CT_cpg,
  sample.id = list("S1_CT_2B", "S1_CT_3B", "S1_CT_4B", "S4_CT_3A", "S4_CT_4B", "S4_CT_5A"),
  assembly = "Bombus terrestris",
  treatment = c(0, 0, 0, 1, 1, 1),
  context = "CpG",
  mincov = 10)
```

Figure 10 : lecture de méthylation de chaque échantillon

Dans la figure 10, l'option `methRead` du package `methylKit` utilisée pour lire les données de méthylation dans les fichiers spécifiés. L'option `sample.id` permet de fournir sous forme de liste les identifiants d'échantillons. Avec `assembly` on donne l'assemblage génomique sur lequel les données sont basées.

Treatment qui spécifie le traitement appliqué à chaque échantillon ; dans ce cas, les trois premiers échantillons partagent la même valeur d'âge, tandis que les trois derniers, avec la valeur 1, sont âgés de quatre semaines. Context permet simplement de signaler quel contexte de méthylation est utilisé pour cette analyse.

Enfin, l'option `mincov` permet d'établir le seuil minimum de lectures requis pour considérer la méthylation comme valide. Dans mes analyses, j'ai imposé un minimum de couverture à 10.

Voici le contenu de l'objet de contrôle dans le contexte CpG.

```

methylRaw object with 21786227 rows
-----
  chr start   end strand coverage numCs numTs
1 NC_063269.1 53152 53152    -      11     0     11
2 NC_063269.1 53167 53167    -      11     0     11
3 NC_063269.1 53172 53172    -      11     0     11
4 NC_063269.1 53261 53261    -      10     0     10
5 NC_063269.1 54217 54217    +      11     0     11
6 NC_063269.1 54245 54245    +      13     0     13
-----
sample.id: S4_CT_3A
assembly: Bombus terrestris
context: CpG
resolution: base

```

La figure 11 montre à quoi ressemble le fichier S4_CT_3A, on voit qu'il crée un format tabulaire. Chaque ligne est numérotée et il y a également sept colonnes : chromosome, début, fin, brin (+ est le *forward* et – est le *reverse*), la valeur de la couverture, numCs signifie la cytosine observée à cette endroit et numTs signifie le nombre de thymine observée.

Figure 11 : Contenu du fichier MethRead

Filtre et normalisation des données

Ces instructions font également partie du package methylKit. Avant d'entamer l'analyse des données de méthylation, il est essentiel d'éliminer tout biais potentiel de séquençage attribuable à la PCR. Cette démarche implique la suppression des bases présentant une couverture excessive ainsi que celles affichant une couverture insuffisante. Pour ce faire, la fonction `filteredByCoverage` est employée.

```

filtered.objet_CT_cpg=filterByCoverage(objet_CT_cpg, lo.count = 10, lo.perc = NULL, hi.count = NULL, hi.perc = 99.9)
normal_cov_CT_cpg = normalizeCoverage(filtered.objet_CT_cpg)

```

Figure 12 : les commandes pour filtrer et normaliser les échantillons

La seconde instruction présentée dans la figure 12 illustre la normalisation des échantillons en relation avec leur couverture. Cette opération vise à homogénéiser les distributions de couverture de lecture entre les différents échantillons.

Lors de la création de l'objet MethylRawList avec la commande précédente, le minimum de couverture a été imposé à 10. Cependant, la normalisation de la couverture a été réalisée afin de s'assurer que la commande précédente a bien été correctement réalisée.

Statistiques descriptions des échantillons

Pourcentage de méthylation

Avec les données de méthylation, il est possible de visualiser le pourcentage de méthylation d'un MethylRawList, cet objet contient les informations de méthylation pour chacun des échantillons. Pour ce faire, l'option `GetMethylationStats` est employée en invoquant l'objet et en indiquant la position dans la liste entre crochets, permettant ainsi de sélectionner l'échantillon à afficher. Contrairement à Python, les positions débutent à la valeur 1.

Pour afficher uniquement la statistique nécessaire à la création d'un histogramme du pourcentage de méthylation, il est requis de spécifier `plot = FALSE` dans la commande. En ce qui concerne `Both.strands`, cette option indique si les deux brins doivent être pris en compte. Dans ce contexte, un seul brin est pris en compte.

```
GetMethylationStats(objet_CT_cpg[[4]], plot = FALSE, both.strands = FALSE)
```

```
methylation statistics per base
summary:
   Min. 1st Qu. Median Mean 3rd Qu. Max.
0.0000 0.0000 0.0000 0.7378 0.0000 100.0000
percentiles:
    0%     10%     20%     30%     40%     50%     60%     70%     80%     90%     95%     99%     99.5%
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905 10.000000 28.000000
   99.9%    100%
78.260870 100.000000
```

Figure 13 : statistique du pourcentage de méthylation CT_cpg

La figure 13 imprime les statistiques du pourcentage de méthylation pour le quatrième échantillon des données de contrôle, c'est-à-dire l'échantillon S4_CT_3A. On peut déjà remarquer qu'il n'y a pratiquement aucune méthylation mais pour l'apercevoir visuellement, on peut modifier la commande en mettant plot = TRUE, affichant ainsi l'histogramme pour cet échantillon.

```
GetMethylationStats(objet_CT_cpg[[4]], plot = TRUE, both.strands = FALSE)
```

La figure 14 illustre l'histogramme du pourcentage de méthylation du même échantillon. Dans la grande majorité, aucune méthylation n'est détectée, correspondant à un pourcentage de 0%. Toutefois, on peut remarquer la présence de quelques cas de méthylation.

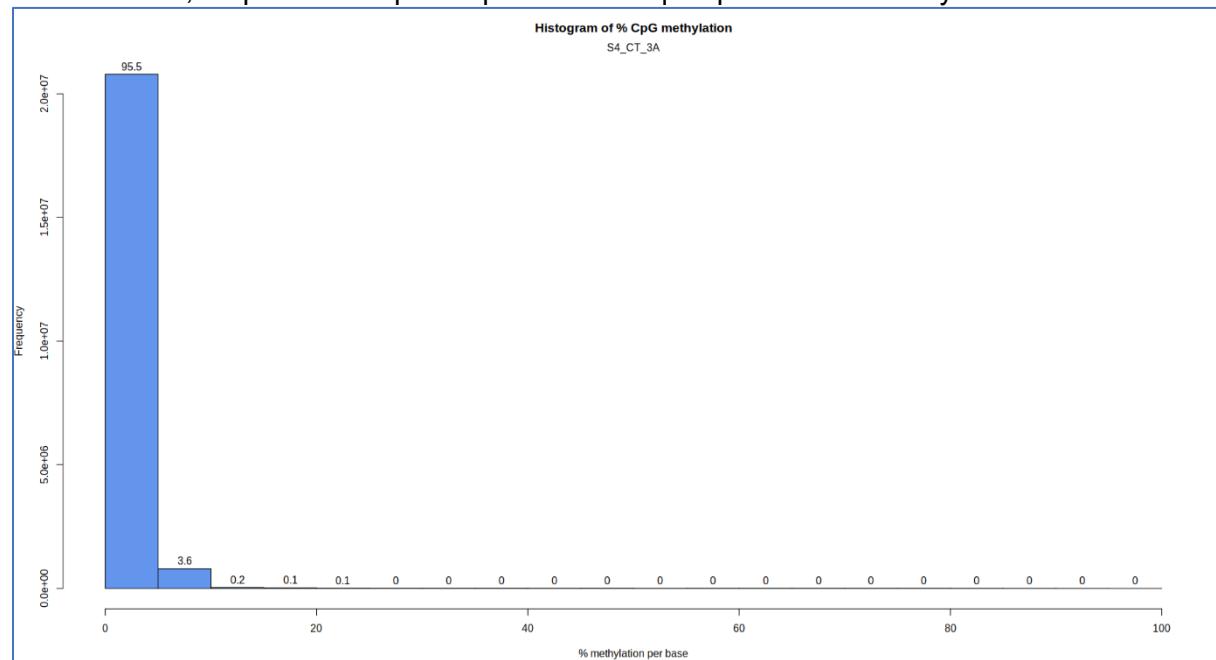


Figure 14 : Graphique du pourcentage de méthylation CT_CpG

Les détails statistiques de chaque échantillon sont disponibles dans l'annexe 2, tandis que les graphiques correspondants sont disponibles en annexe 3, spécifiquement dans le contexte de méthylation. Pour les autres contextes, les graphiques sont similaires, d'où leur absence dans l'écrit.

Couverture du contexte dans l'échantillon

De manière similaire à l'affichage du pourcentage de méthylation, il est possible de représenter le pourcentage de couverture de chaque échantillon. Le nombre de barres reflète le pourcentage d'emplacement dans chaque intervalle. Les échantillons qui présentent un fort biais de déduplication attribuable à la PCR montrent un pic secondaire sur le côté droit du graphique.

```
getCoverageStats(objet_CT_cpg[[4]], plot = FALSE, both.strands = FALSE)
```

De manière similaire, l'ajout du paramètre `plot = FALSE` permet d'obtenir la statistique requise pour générer le graphique du pourcentage de couverture. En annexe 4 se trouve les statistiques de couverture sur les échantillons.

```
read coverage statistics per base
summary:
   Min. 1st Qu. Median     Mean 3rd Qu.    Max.
10.00   14.00   18.00 18.55 22.00 80.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99% 99.5% 99.9% 100%
  10    12    13    15    16    18    19    21    23    26    29    35    38    47    80
```

Figure 15 : Statistiques du pourcentage de la couverture

La figure 15 illustre la statistique du pourcentage de couverture de l'échantillon S4_CT_3A. Quelques informations clés sont disponibles, dont une couverture minimale de 10. Le terme 1st Qu indique que le premier quartile, soit 25% des sites de méthylation, présente une couverture de 14 lectures ou moins. La médiane s'élève à 18, signifiant que la moitié des sites de méthylation ont une couverture de 18 ou moins.

La moyenne de la couverture des lectures pour les sites de méthylation est d'environ 18,55, représentée par Mean. Le troisième quartile affiche une valeur de 22, signifiant que pour 75% des sites de méthylation, la couverture est de 22 ou moins. La plus haute couverture observée pour les sites de méthylation est de 80.

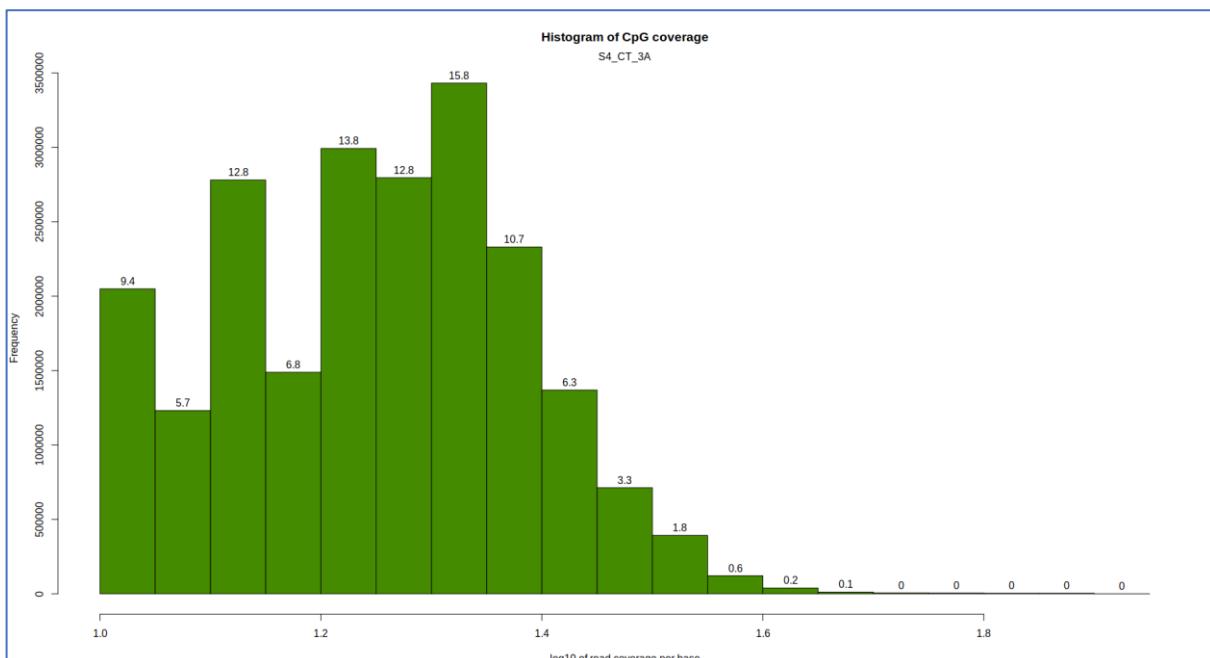


Figure 16 : histogramme du pourcentage de couverture S4_CT_3A

La figure 16 illustre l'histogramme de la couverture pour l'échantillon S4_CT_3A. À droite du graphique, l'absence de pic indique qu'il n'y a pas de biais de duplication dû à la PCR affectant l'échantillon. Pour consulter les graphiques de couverture pour les divers échantillons dans les différents contextes : annexe 5.

Filtre des échantillons

Filtre des échantillons sur la couverture

Ça peut être utile de filtrer les échantillons en fonction de la couverture surtout s'il y a un biais de PCR. Il faut jeter les bases avec une couverture trop élevée. Les valeurs de couverture extrêmement élevées peuvent provenir d'erreurs expérimentales ou de biais techniques, ce qui peut affecter la qualité de vos résultats. Il faut également supprimer les bases de basse couverture cela permettra d'augmenter la puissance des tests ultérieurs.

```
filtered.objet_CT_cpg=filterByCoverage(objet_CT_cpg, lo.count = 10, lo.perc = NULL,
                                         hi.count = NULL, hi.perc = 99.9)
```

Cette instruction permet de filtrer la liste methylRawList en éliminant les bases avec une couverture inférieure à 10X, ainsi que les bases dont la couverture dépasse le 99,9e centile dans chaque échantillon. Le tableau 8, montre les échantillons non filtrés et les échantillons ainsi que le nombre de séquences supprimées dans le contexte CpG.

Échantillons	Non filtrés	Filtrés	Supprimés
S1_CT_2B	12.286.487	12.274.073	12.414
S1_CT_3B	15.960.125	15.943.998	16.127
S1_CT_4B	17.075.982	17.058.895	17.087
S1_RG_2A	22.399.467	22.376.908	22.559
S1_RG_4B	20.746.441	20.725.518	20.923
S1_RG_5B	20.254.086	20.233.757	20.329
<hr/>			
S1_CT_3A	21.786.227	21.764.405	21.822
S1_CT_4B	16.916.387	16.899.432	16.955
S1_CT_5A	20.945.501	20.924.377	21.124
S1_RG_1A	23.225.923	23.202.500	23.423
S1_RG_2B	11.297.683	11.286.267	11.416
S1_RG_5B	16.312.793	16.296.408	16.385

Tableau 8 : Récapitulatif des suppressions après filtre CT CpG

Le tableau 9, montre les échantillons de nombre de séquences supprimées en contexte CHG.

Échantillons	Non filtrés	Filtrés	Supprimés
S1_CT_2B	6.359.855	6.353.444	6.411
S1_CT_3B	8.314.082	8.305.736	8.346
S1_CT_4B	8.712.024	8.703.286	8.738
S1_RG_2A	11.440.817	11.429.316	11.501
S1_RG_4B	10.651.367	10.640.713	10.654
S1_RG_5B	10.408.841	10.398.394	10.447
<hr/>			
S1_CT_3A	11.174.807	11.163.577	11.230
S1_CT_4B	8.707.192	8.698.375	8.817
S1_CT_5A	10.691.399	10.680.605	10.794
S1_RG_1A	11.932.452	11.920.519	11.933
S1_RG_2B	5.795.724	5.789.862	5.862
S1_RG_5B	8.489.088	8.480.575	8.513

Tableau 9 : Récapitulatif des suppressions après filtre CT CHG

Le tableau 10, montre les échantillons non filtrés et les échantillons ainsi que le nombre de séquences supprimées dans le contexte CHH.

Échantillons	Non filtrés	Filtrés	Supprimés
S1_CT_2B	24.668.923	24.644.000	24.923
S1_CT_3B	32.628.019	32.595.232	32.787
S1_CT_4B	32.855.963	32.822.828	33.135
S1_RG_2A	41.722.420	41.680.634	41.786
S1_RG_4B	39.857.183	39.817.208	39.975
S1_RG_5B	39.299.706	39.260.314	39.392
S1_CT_3A	41.463.997	41.422.294	41.703
S1_CT_4B	33.298.448	33.264.732	33.716
S1_CT_5A	39.037.728	38.264.732	39.183
S1_RG_1A	43.969.719	43.925.611	44.108
S1_RG_2B	22.116.142	22.093.905	22.237
S1_RG_5B	32.814.291	32.781.339	32.952

Tableau 10 : Récapitulatif des suppressions après filtre contexte CHH

Pour l'ensemble des échantillons, il y a en moyenne une perte d'un millionième d'information (0,0001%).

Normalisation sur la couverture

La normalisation des distributions de couverture de lecture entre les échantillons est effectuée en utilisant la fonction `normalizeCoverage` de methylKit. Avant d'exécuter cette commande, il est nécessaire de préalablement filtrer les échantillons.

```
normal_cov_CT_cpg = normalizeCoverage(filtered.objet_CT_cpg)
```

À la suite de cette commande, aucune méthylation n'a été supprimée c'est sûrement que les données avaient peut-être déjà une distribution de couverture relativement homogène et ne nécessitaient pas de modifications majeures après la normalisation.

Une méthylation minimale de 10 a été définie, et il est constaté que la majorité des séquences possèdent déjà une couverture dépassant ce seuil. Il est donc envisageable que la normalisation n'ait pas requis la suppression de séquences, car celles-ci satisfont déjà à la condition énoncée.

Étant donné l'absence de distinction entre les échantillons filtrés et ceux normalisés en fonction de la couverture, il n'est pas nécessaire de présenter les tableaux récapitulatifs, car ils affichent des valeurs identiques à celles des échantillons filtrés.

Fusion des échantillons

L'objectif est de réaliser une analyse plus approfondie des données de méthylation en combinant les informations de tous les échantillons en un seul objet. Pour ce faire, j'utilise la fonction de MethylKit `unite`.

```
meth_cpg_normal_CT <- methylKit::unite(normal_cov_CT_cpg, destrand = FALSE)
```

Lorsqu'il s'agit de mener des analyses comparatives entre ces échantillons, il est crucial de ne prendre en compte que les emplacements où les données de méthylation sont disponibles pour tous les échantillons. En employant la fonction unite(), un objet de methylBase est créé. Cet objet est utilisé pour les futures analyses comparatives.

Il contient les informations de méthylation pour les régions ou bases présentes dans tous les échantillons. L'option destrand signifie que les informations de méthylation des deux brins d'ADN seront traitées de la même manière donc si une cytosine méthylée ne l'est que sur un seul des deux brins, elle n'est pas retenue.

```
methylBase object with 7551874 rows
-----
chr start end strand coverage1 numCs1 numTs1 coverage2 numCs2 numTs2 coverage3 numCs3 numTs3 coverage4 numCs4 numTs4 coverage5 numCs5 numTs5 coverage6 numCs6 numTs6
1 NC_063269..1 55861 55861 - 17 0 17 26 0 26 14 1 13 15 0 15 15 0 15 11 0 11
2 NC_063269..1 55868 55868 - 15 0 15 26 3 23 14 0 14 13 1 12 15 0 15 12 0 12
3 NC_063269..1 90278 90278 - 14 0 14 15 0 15 22 0 22 13 0 13 15 0 15 16 0 16
4 NC_063269..1 90288 90288 - 18 0 18 17 0 17 23 0 23 14 0 14 17 0 17 18 0 18
5 NC_063269..1 90298 90298 - 19 0 19 15 0 15 24 0 24 17 0 17 17 0 17 21 0 21
6 NC_063269..1 90308 90308 - 21 0 21 15 0 15 24 1 23 19 0 19 17 0 17 23 1 22
-----
sample.ids: S1_CT_2B S1_CT_3B S1_CT_4B S4_CT_3A S4_CT_4B S4_CT_5A
destranded: FALSE
assembly: Bombus terrestris
context: CpG
treatment: 0 0 1 1 1
resolution: base
```

Figure 17 : fusion des échantillons de contrôle

La figure 17 illustre l'objet fusionné de l'ensemble des échantillons, tant dans le contexte CpG que dans les échantillons non traités, c'est-à-dire les échantillons de contrôle. Cette démarche peut être répétée pour chaque contexte et chaque traitement.

Échantillons	Nombre de ligne	Nombre de colonne
CT CpG	7 551 874	22
CT CHG	3 873 433	
CT CHH	14 975 951	
RG CpG	8 677 236	
RG CHG	4 436 128	
RG CHH	16 784 699	

Tableau 11 : Nombre de méthylation dans les échantillons fusionnés

Le tableau 11 présente un résumé des objets MethylBase, en fournissant pour chaque objet le nombre total de lignes et de colonnes. Pour chacun des échantillons, on observe une augmentation de la méthylation dans les échantillons traités avec l'inhibiteur de méthylation. Après la fusion des échantillons, il y a donc plus que 6 variables pour l'ensemble des échantillons. Les échantillons traités et de contrôle sont rassemblés, la seule distinction réside dans les contextes de méthylation.

Analyse en composante principale

Normaliser les échantillons par taille

Pendant une semaine, l'objectif était de normaliser les échantillons en fonction de leur taille, afin de les mettre à la même échelle. Pour ce faire, plusieurs outils ont été utilisés dans Rstudio.

Normalize

Cette bibliothèque permet d'ajuster et de mettre à l'échelle des données numériques, rendant ainsi les valeurs comparables et simplifiant leur analyse. Elle propose diverses méthodes de normalisation. Cependant, il convient de noter qu'elle n'est pas compatible avec les objets de type MethylRawlist.

PCASamples

MethylKit propose l'option PCASamples spécifiquement conçue pour effectuer des analyses en composantes principales de méthylation. Ainsi, dans un premier temps, cette option a été utilisée. Cette option utilise prcomp et comme entrée la fonction a besoin de la matrice du pourcentage de méthylation. Lors de la visualisation du résultat obtenu avec la PCA, j'avais l'impression que les données étaient impactées par le nombre de méthylation présent dans chaque échantillon, celle ayant le plus de méthylation est à gauche.

J'ai donc essayé plusieurs méthodes pour régler ce problème avec PCASamples :

- PCASamples(meth_cpg_normal_CT)
- PCASamples(meth_cpg_normal_CT, scale = *TRUE*)
Scale = TRUE, signale à prcomp que les données doivent être mise à l'échelle. Il n'y a aucune différence avec et sans ce paramètre.
- PCASamples(meth_cpg_normal_CT, scale = *TRUE*, center = *TRUE*)
Center = TRUE, Dans le processus de centrage, la moyenne de toutes les valeurs est soustraite. Ainsi, si un bourdon affiche généralement des valeurs de méthylation élevées, elle sera rapprochée vers la valeur moyenne. De la même manière, si un bourdon montre généralement des valeurs faibles, elle sera également rapprochée de cette valeur moyenne. Cette approche permet de mettre en évidence les différences par rapport à la moyenne plutôt qu'aux valeurs absolues. Aucun changement avec ce paramètre.
- PCASamples(meth_cpg_normal_CT, scale = *TRUE*, center = *TRUE*, screeplot = *TRUE*)
Le résultat sera un graphique qui illustre la répartition de la variation entre les différentes composantes principales. Cela peut vous assister dans la détermination de l'importance des différentes composantes pour vos données.

Procom

Étant donné que la fonction PCASamples utilise prcomp, j'ai décidé de me pencher sur cette fonction pour réaliser mon ACP. Elle permet de visualiser une fois la PCA réalisée chacune des composantes principales.

PCASamples choix final

La commande classique, sans argument, a été sélectionnée pour réaliser l'analyse en composante principale étant donné que les arguments n'avaient aucun impacte sur les résultats.

```
PCASamples(meth_cpg_normal_CT)
```

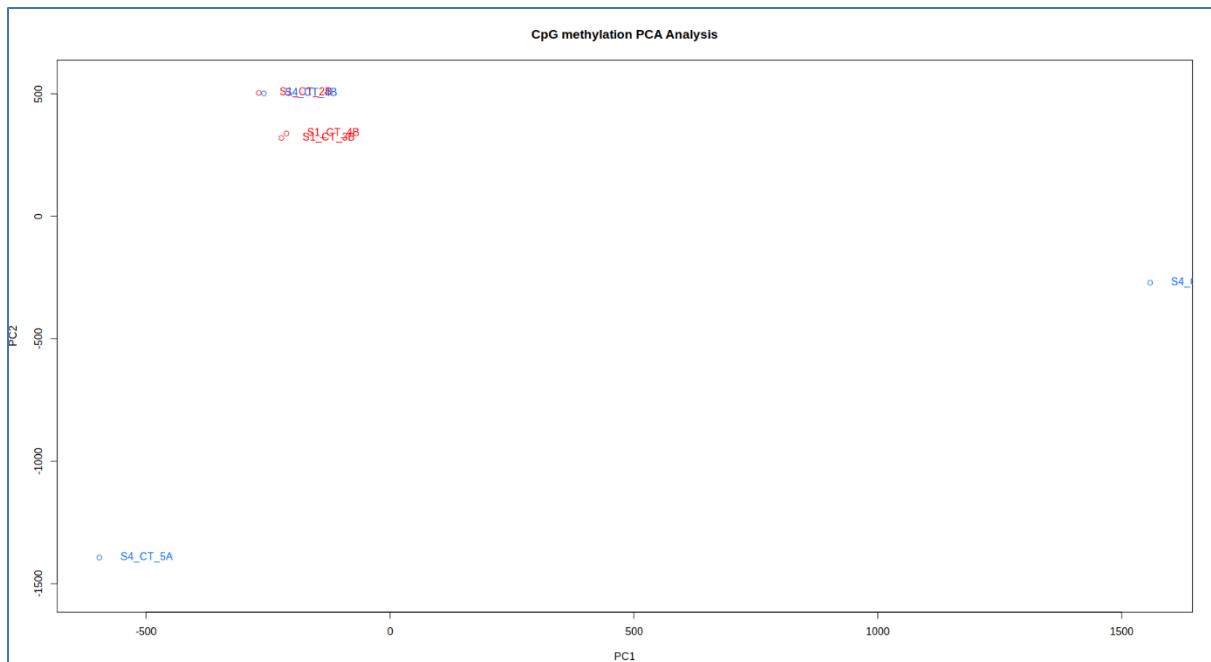


Figure 18 : analyse en composante principale pour les contrôles dans le contexte CpG

La figure 18 illustre l'analyse en composantes principales réalisée dans le contexte de méthylation CpG pour les échantillons de contrôle.

En réalité La position des échantillons sur un graphique PCA est influencée par les variations observées dans les données. Si les échantillons ayant un niveau de méthylation plus élevé se situent à droite, cela suggère que ces échantillons possèdent des profils de méthylation distincts et contribuent davantage à la variation totale dans les données. Pour consulter les analyses en composantes principales des autres contextes de méthylation des échantillons de contrôle et leur répartition en composantes principales, veuillez-vous référer à l'annexe 6.

La figure 19 représente la distribution en composantes principales de l'analyse en composante principale du contexte de méthylation CpG des échantillons de contrôle.

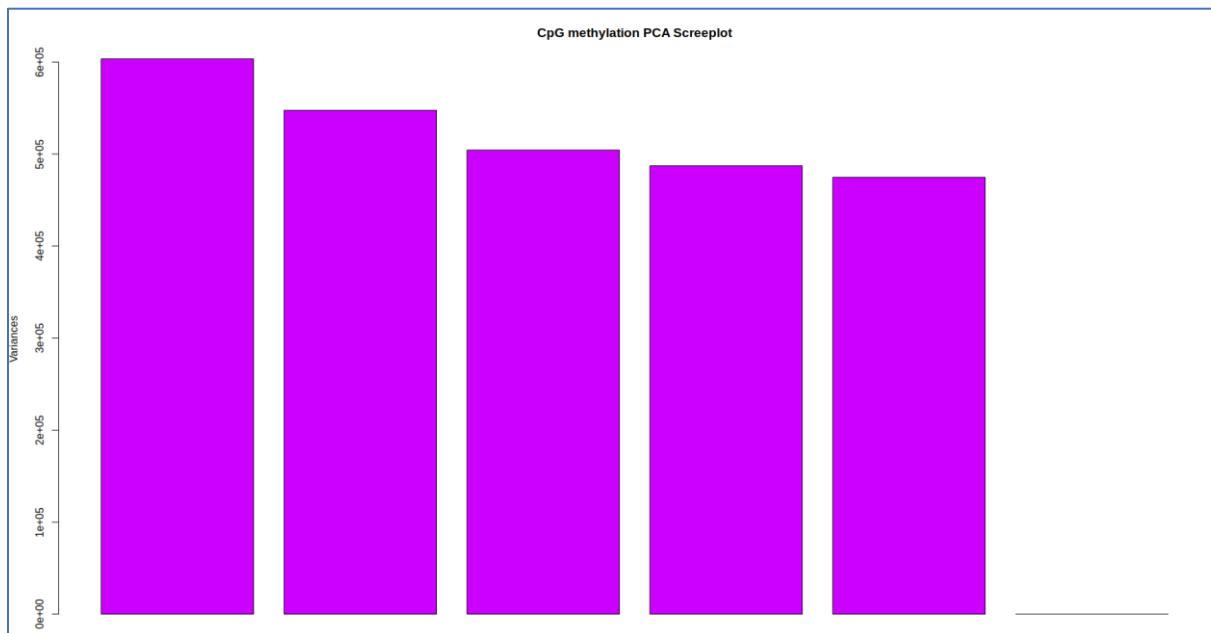


Figure 19 : distribution en composante principale pour les échantillons de contrôle du contexte CpG

La distribution montre que la plupart des informations sont distribuée en 5 composantes principales. La variation des données est principalement capturée par les cinq premières composantes principales, sans qu'une ou deux d'entre elles ne se démarquent de manière prédominante.

De la même manière, la figure 20 est l'analyse en composantes principales du contexte de méthylation CpG des échantillons traités à l'inhibiteur de méthylation.

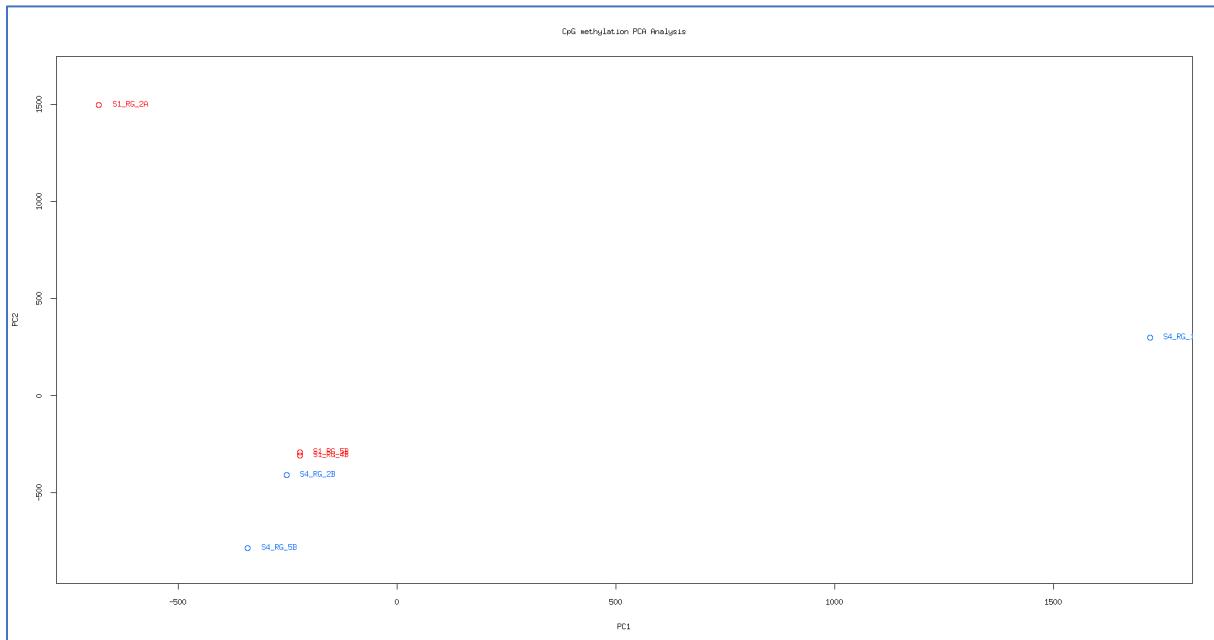


Figure 20 : analyse en composante principale pour les traités dans le contexte CpG

La figure 21 illustre la distribution des composantes principales obtenue en réalisant l'analyse en composantes principales du contexte de méthylation CpG des échantillons traités à l'inhibiteur de méthylation RG108.

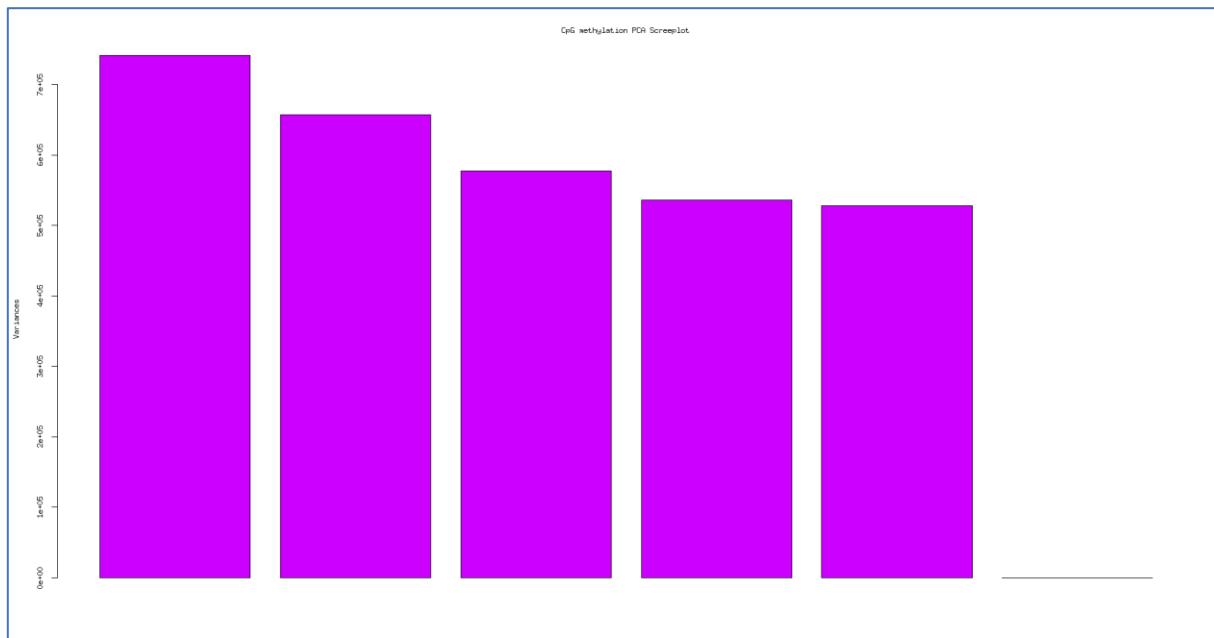


Figure 21 : distribution en composante principale pour les échantillons traités contrôle du contexte CpG

Dans cette distribution, on voit que les deux premières composantes sont toutes deux plus importantes que les trois dernières autres composantes principales. Cependant, la variation n'est pas fortement démarquée.

Étude des clusters

L'option PCASamples de la librairie methylKit ne fournit pas de détails spécifiques sur le contenu de chaque composante principale. Étant donné que la PCA n'a pas révélé de distinction nette entre deux groupes, une analyse de regroupement (cluster) a été effectuée pour visualiser les groupes formés.

```
clusterSamples(meth_cpg_normal_CT, dist="correlation", method="ward", plot=TRUE)
```

Argument distance

La distance est extrêmement importante, il définit comment les échantillons doivent être comparés pour former des groupes ou des clusters. Lorsqu'une analyse de regroupement est réalisée, l'objectif est de regrouper les échantillons similaires tout en préservant une certaine distance ou différence entre les groupes. Dans mes analyses, la valeur est correlation, ce qui signifie que la corrélation de Pearson entre les échantillons est utilisée pour mesurer la similarité entre eux.

La corrélation de Pearson évalue comment deux séries de nombres évoluent ensemble. Supposons que deux listes de nombres soient disponibles, et l'on veut déterminer s'il y a une relation entre elles. Si l'augmentation d'un nombre dans la première liste s'accompagne de l'augmentation correspondante d'un nombre dans la deuxième liste, cela indique une corrélation positive. L'absence de relation particulière entre les nombres des deux listes se traduit par une corrélation nulle.

En revanche, si l'augmentation d'un nombre dans la première liste est liée à une diminution correspondante d'un nombre dans la deuxième liste, cela constitue une corrélation négative. La corrélation de Pearson quantifie cette relation au moyen d'un coefficient compris entre -1 et 1. Un coefficient proche de 1 indique une forte corrélation positive, proche de -1 indique une forte corrélation négative, et proche de 0 indique l'absence de corrélation.

Ce coefficient se calcule en observant à quel point les nombres des deux séries diffèrent de leurs moyennes respectives, ainsi que la mesure de leur différence mutuelle. Lorsque les nombres varient conjointement davantage, la corrélation est plus forte.

Différentes méthodes de calcul de distance sont disponibles, dont "correlation", "euclidean", "maximum", "manhattan", "canberra", "binary" et "minkowski". Dans ce cas, j'ai opté pour la méthode par défaut, qui est la corrélation.

Argument méthode

L'argument method détermine la méthode de regroupement utilisée pour former les clusters. La méthode ward est spécifique et vise à organiser plusieurs échantillons en groupes en fonction de leurs similitudes. Ward cherche à former des groupes pour que les échantillons à l'intérieur de chaque groupe soient aussi proches que possible en termes de caractéristiques.

Pour ce faire, elle analyse comment la variance (les différences) entre les caractéristiques des échantillons évolue lorsqu'un nouvel échantillon est ajouté à un groupe existant. Son objectif est de minimiser cette variance afin de créer des groupes le plus homogènes possible.

Il existe plusieurs types de méthodes possibles : "ward.D", "ward.D2", "single", "complete", "average", "mcquitty", "median" ou "centroid". J'ai opté pour la méthode par défaut : "ward".

Les méthodes ward, ward.D et ward.D2 sont similaires, mais elles utilisent des formules légèrement différentes pour calculer les distances et former les clusters. ward est une version simplifiée de ward.D, tandis que ward.D2 est une version améliorée de ward.D qui tient compte des variations de la taille des groupes lors de la formation des clusters.

Ces méthodes sont souvent utilisées dans l'analyse de regroupement (clustering) pour former des groupes homogènes à partir de données similaires.

Regroupement des échantillons

La fonction ClusterSamples est utilisée pour regrouper les échantillons similaires au sein d'un ensemble de données. L'objectif est d'identifier des groupes ou des "clusters" d'échantillons qui partagent des similarités au niveau de leurs caractéristiques.

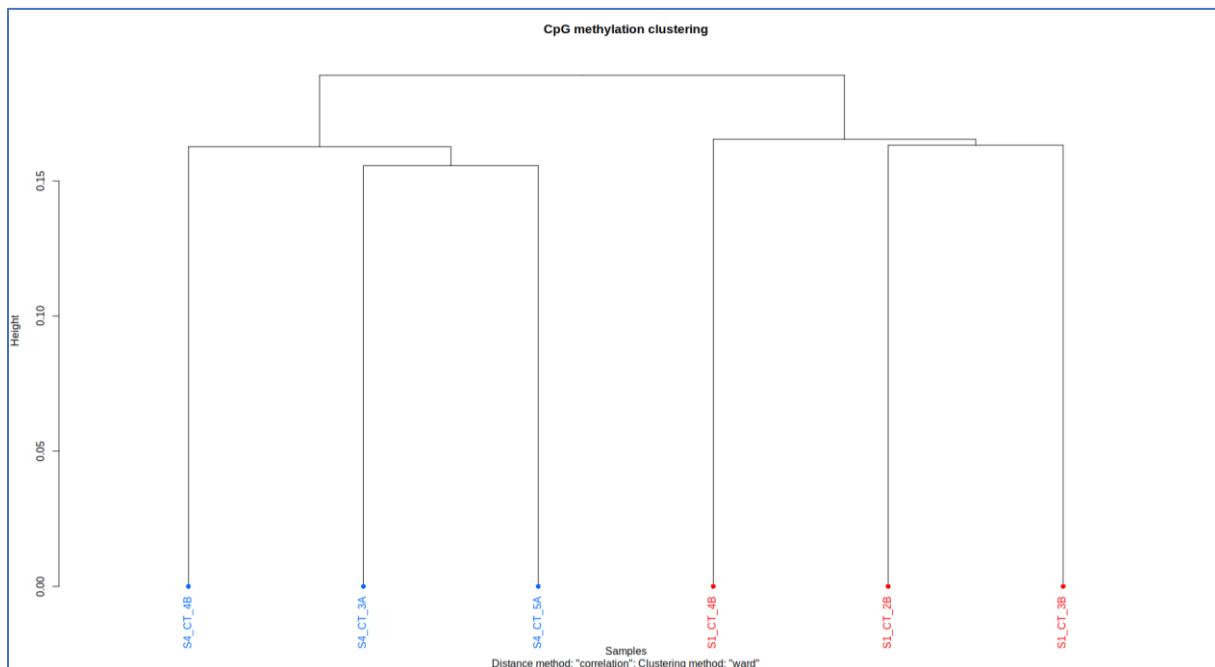


Figure 22 : répartitions en groupes pour le contexte de méthylation en CpG

La figure 22 présente le regroupement des échantillons de contrôle dans le contexte de méthylation CpG. Les résultats montrent que les échantillons de la semaine 1 et de la semaine 4 se regroupent en deux groupes distincts. Cette observation suggère que les échantillons de la même semaine présentent des profils de méthylation plus similaires entre eux que ceux des semaines différentes.

Il est important de noter que la distinction entre les groupes obtenue grâce à la fonction ClusterSamples est plus prononcée dans le contexte de méthylation CpG que dans le contexte CHG, où elle est légèrement moins évidente. Dans le contexte de méthylation CHH, cette distinction est encore moins marquée.

De plus, on observe que l'échantillon S4_CT_4B est associé aux échantillons de la semaine 1, tandis que l'échantillon S1_CT_2B est lié aux échantillons de la semaine 4.

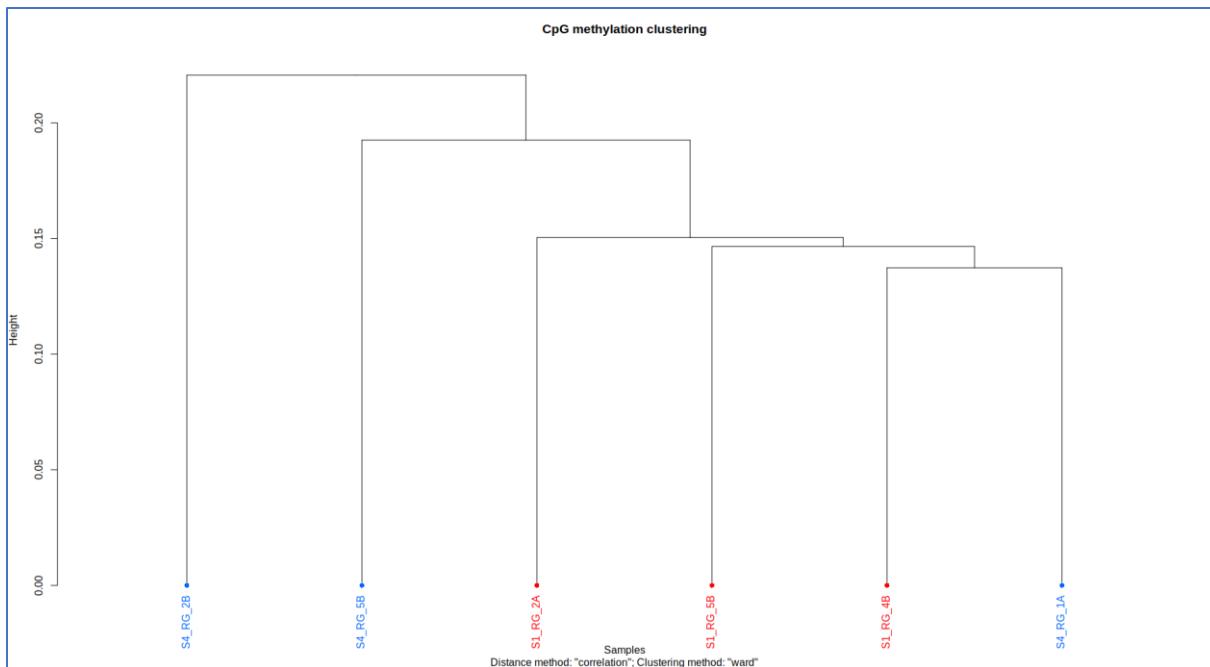


Figure 23 : répartitions en groupes échantillon traités pour le contexte de méthylation en CpG

La figure 23 montre le regroupement des échantillons traités avec l'inhibiteur de méthylation dans le contexte de méthylation CpG à l'aide de la fonction ClusterSamples.

Les résultats montrent que l'échantillon 1A de la semaine 4 partage davantage de similitudes dans ses profils de méthylation avec les échantillons de la semaine 1, plutôt qu'avec les autres échantillons de la semaine 4. Les groupes formés par les échantillons de la semaine 1 et de la semaine 4 ne sont pas clairement distincts entre eux. En revanche, contrairement aux échantillons de contrôle, les échantillons du contexte de méthylation CpG ne montrent pas une distinction nette entre les deux groupes.

Pour consulter les regroupements des différents contextes de méthylation, veuillez-vous référer à l'annexe 7. Il convient de souligner que pour les autres contextes de méthylation, aucun groupe distinct n'est également observé. En ce qui concerne le contexte de méthylation CHH, les deux groupes sont fortement mélangés, montrant une absence de similarité entre eux.

Analyse des régions différemment méthylées

La fonction "calculateDiffMeth" du package methylKit permet de quantifier les variations de méthylation entre divers groupes d'échantillons. Elle requiert en entrée une matrice de données de méthylation (meth_cpg_normal_CT) et effectue des calculs pour repérer les sites CpG qui affichent des différences significatives de méthylation entre les différents groupes d'échantillons.

```
diff_CT_cpg = calculateDiffMeth(meth_cpg_normal_CT)
```

La fonction "calculateDiffMeth" du package "methylKit" permet d'effectuer une analyse comparative de la méthylation entre différents groupes d'échantillons. Elle évalue les différences de méthylation au niveau des sites CpG en calculant les moyennes de méthylation pour chaque groupe et chaque site CpG. Ensuite, elle applique des tests statistiques pour déterminer si ces différences sont statistiquement significatives. Pour éviter les erreurs dues aux multiples comparaisons, les p-valeurs sont ajustées. Les sites CpG présentant des

différences de méthylation significatives sont identifiés en utilisant des seuils de p-valeur ajustée. Les résultats de cette analyse sont cruciaux pour les études génétiques et de méthylation, car ils permettent de localiser les régions du génome où des différences de méthylation entre les groupes sont particulièrement remarquables.

Les p-valeurs indiquent à quel point les résultats que nous avons observés pourraient être le résultat du hasard, en se basant sur l'idée que l'effet que nous pensons voire n'existe en réalité pas (hypothèse nulle).

```
diff_CT_cpg = calculateDiffMeth(meth_cpg_normal_CT)
```

Échantillons	Nombre de ligne	Nombre de colonne
CT CpG	7 551 874	7 colonnes
CT CHG	3 873 433	
CT CHH	14 975 951	
RG CpG	8 677 236	
RG CHG	4 436 128	
RG CHH	16 784 699	

Tableau 12 : différence de méthylation pour les échantillons

La tableau 12 montre les différences de méthylation CT_CPG, il y a la même valeur en nombre de lignes et donc en nombre de méthylation dans MethRead que dans les régions différentiellement méthylées.

Régions hyper/hypo méthylées

L'analyse des régions hyperméthylées et hypométhylées vise à identifier les sites CpG qui présentent des différences significatives de méthylation entre les groupes. Ces différences peuvent impliquer des sites CpG méthylés dans certains échantillons mais non méthylés dans d'autres, tout en manifestant des variations de méthylation remarquables entre les groupes. L'objectif principal est de repérer les changements de méthylation liés à l'âge des bourdons, en établissant une comparaison indépendante des niveaux de méthylation spécifiques dans chaque échantillon individuel.

La différence de méthylation imposée est de 10, et une q-value de 0,01 est utilisée pour identifier les régions présentant une hyperméthylation ou une hypométhylation significative. Dans les régions hyperméthylées, il y a une augmentation des groupes méthyle (-CH3) attachés aux cytosines des sites CpG. Cela peut entraîner une inhibition de l'expression des gènes situés dans ces régions. Dans les régions hypométhylées, il y a une diminution des groupes méthyle attachés aux cytosines des sites CpG. Cela peut entraîner une expression accrue des gènes situés dans ces régions.

Les commandes nécessaires pour effectuer des analyses de différences de méthylation, les régions hyperméthylées, hypométhylées et le total dans le contexte de méthylation CpG sur les échantillons de contrôle sont les suivantes :

```
diff10.hypo= getMethylDiff("diff_CT_cpg"), difference = 10, qvalue = 0.01, type = "hyper")
```

```
diff10.hypo= getMethylDiff("diff_CT_cpg"), difference = 10, qvalue = 0.01, type = "hypo")
```

```
diff10.hypo= getMethylDiff("diff_CT_cpg"), difference = 10, qvalue = 0.01)
```

Les commandes sont exécutées en un temps très court, quasiment instantanément.

Échantillons	Hyper	Hypo	Méthylation
CT CpG	1 975	1 919	3 894
CT CHG	6	8	14
CT CHH	14	17	31
RG CpG	3 131	2 431	5 562
RG CHG	5	7	12
RG CHH	19	24	43

Tableau 13 : récapitulatif des régions hyper / hypo méthylées

Le tableau 13 montre un récapitulatif des différences de méthylation pour chacun des fichiers contenant les différences de méthylation.

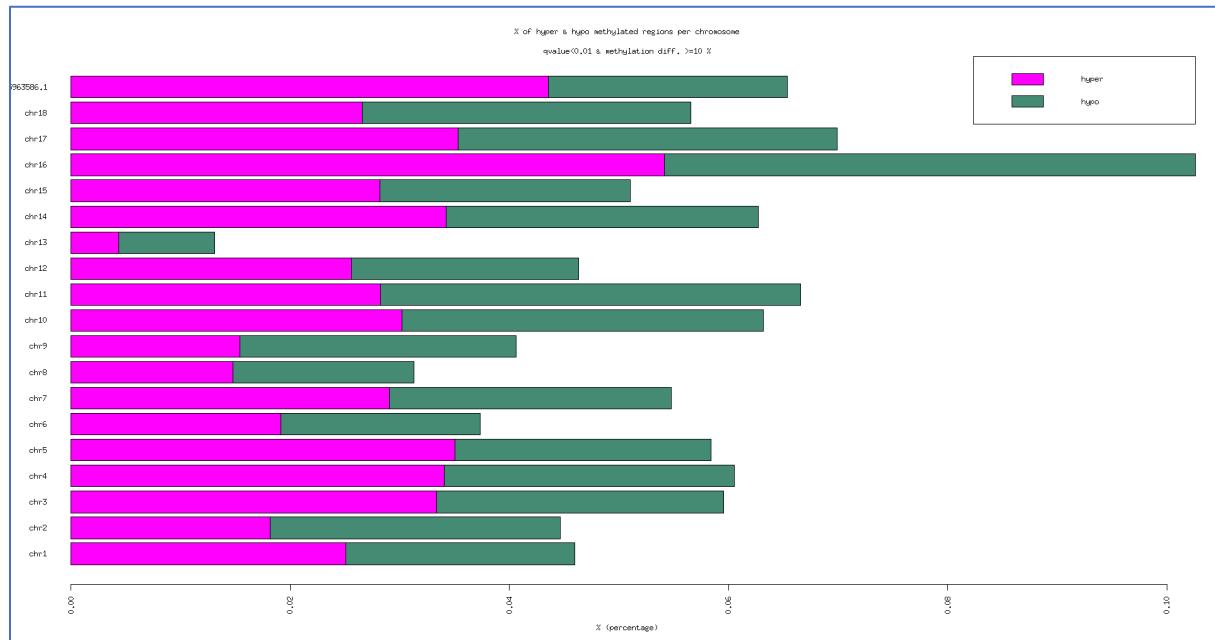


Figure 24 : répartition des différences de méthylation contexte CpG

La figure 24 illustre la répartitions des différences de méthylation sur les chromosomes correspondant pour les échantillons de contrôle dans le contexte de méthylation CpG. Vous trouverez en annexe 8 la répartition des différences de méthylation.

Annotation régions différemment méthylées

Information des informations d'annotation

Afin de réaliser cette tâche, il est nécessaire de télécharger le fichier BED à partir du site de NCBI, spécifiquement à partir du projet de séquençage du génome de référence.

<p>Sélectionnez la source du fichier</p> <ul style="list-style-type: none"> <input checked="" type="radio"/> Tout (2) <input type="radio"/> RefSeq uniquement (1) <input type="radio"/> GenBank uniquement (1) 	<p>Sélectionnez les types de fichiers</p> <ul style="list-style-type: none"> <input type="checkbox"/> Séquences du génome (FASTA) <input type="checkbox"/> Fonctionnalités d'annotation (GTF) <input checked="" type="checkbox"/> Fonctions d'annotation (GFF) <input type="checkbox"/> Séquence et annotation (GBFF) <input type="checkbox"/> Transcriptions (FASTA) <input type="checkbox"/> Séquences codantes génomiques (FASTA) <input type="checkbox"/> Protéine (FASTA) <input type="checkbox"/> Rapport de séquence (JSONL) <input checked="" type="checkbox"/> Rapport de données d'assemblage (JSONL)
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figure 25 : téléchargement des fichiers possibles

Le format BED [11] (*Browser Extensible Data*) est utilisé pour stocker des informations génétiques et des annotations sous forme de fichier texte. Ces fichiers adoptent un format tabulaire où chaque ligne représente une région spécifique du génome. Chacune d'elle comporte au moins trois informations essentielles : le chromosome, la position de début et la position de fin de la région d'autres informations supplémentaires peuvent être ajoutée.

Le fichier 26 montre le contenu du fichier GFF3 [12] contient plusieurs informations supplémentaires pour chaque entrée, notamment le nom du chromosome, la source des informations, le type de séquence, les positions de début et de fin, l'indication du brin (*forward* ou *reverse*), la phase (0, 1 ou 2, indiquant la position dans un codon), et l'identifiant unique attribué à chaque élément génomique, on peut également mettre dans cette colonne des informations supplémentaires.

Figure 26 : contenu du fichier GFF3

La différence majeure entre un GFF classique et un GFF réside dans la clarté, il vise à faciliter l'échange d'annotation génétique. Le format GFF3 est recommandé lors de l'annotation génétique.

Conversion fichier GFF en format BED

Après avoir téléchargé le fichier GFF3, il faut le décompresser avec unzip à la ligne de commande. Une fois dézippé, un dossier nommé "NCBI" est créé contenant le fichier GFF3 décompressé et donc utilisable. Il existe plusieurs programmes disponibles pour effectuer la conversion vers le format BED.

Une fois le fichier transformé, il faut extraire les informations souhaitées du fichier BED avec l'option `readTranscriptFeature` de la librairie Genomation [13]. Ces informations sont utiles dans la suite des analyses.

La Figure 25 illustre les divers types de données disponibles en téléchargement à partir du projet d'assemblage du génome du *Bombus terrestris*.

Il n'existe pas de fichier BED directement téléchargeable depuis le site de NCBI. Cependant, il est possible de réaliser une conversion à partir d'un fichier GFF au format BED. Pour cette raison, le fichier GFF correspondant a été téléchargé.

Gffread

```
gffread -T input.gff3 -o output.bed
```

L'option -T permet de convertir le fichier GFF3 en fichier BED, La conversion s'est réalisée sans aucun problème cependant, lorsque j'utilise la fonction readTranscriptFeature avec le fichier au format BED, il y a une erreur.

Gff2bed

Cette option fait partie du programme bedops, pour l'installer cette option, la commande apt-get install est utilisée. Cette commande prend en entrée un fichier GFF3 et sort un fichier BED.

```
Gff2bed < genomic.gff > genomic.bed
```

Lorsque la commande ReadTranscriptFeature est exécutée avec le fichier genomic.bed, une erreur survient lors du calcul de la table. Pour résoudre ce problème, j'ai tenté d'ajouter un argument à la commande : remove.unusual = FALSE. Cependant, cela n'a pas résolu l'erreur contrôle la suppression des entités inhabituelles ou inattendues lors de la conversion.

GenomeTools

Cet outil en ligne sert à vérifier la conformité d'un fichier GFF3 avec les spécifications du format GFF3. Il prend en charge des fichiers dont la taille est limitée à 50 Mo et qui peuvent être compressés en formats gz ou bz2. L'objectif est de s'assurer que le fichier GFF3 respecte les règles et les normes du format GFF3, ce qui peut être la cause de la mauvaise conversion en format tabulaire. Étant donné que le fichier GFF3 excède la taille de 160 Mo, les 1 000 premières lignes du fichier ont été sélectionnées.

The screenshot shows the "Validateur en ligne GFF3" page. On the left, there's a sidebar with buttons for "Aperçu", "Télécharger", "Parcourir la source", "Traqueur d'incidents", "Documentation", "AnnotationEsquise", "Validateur GFF3" (which is highlighted in yellow), and "Licence". The main area has a title "Validateur en ligne GFF3" and a sub-instruction: "Utilisez ce formulaire pour télécharger un fichier d'annotation GFF3 (jusqu'à 50 Mo, peut être .gz ou .bz2 compressé) qui est ensuite validé par rapport à la [spécification GFF3](#) (éventuellement en utilisant également le fichier OBO Sequence Ontology actuel)." Below this is a "Fichier d'annotation:" input field with a note "Choisir un fichier" and a message "Aucun fichier choisi". There are three checked checkboxes: "activer le mode "nettoyage"" (tries to correct errors), "utiliser Sequence Ontology pour valider les types et les relations parent-enfant" (use Sequence Ontology to validate types and parent-child relations), and "signaler également le contenu des lignes GFF concernées" (also report the content of affected GFF lines). A "Validez ce fichier !" button is present. At the bottom, there's a note: "Ce validateur GFF3 fait partie de la distribution GenomeTools que vous pouvez [télécharger](#) sur votre ordinateur. Utilisez l'outil gff3validator pour valider vos propres fichiers GFF3 – éventuellement plus volumineux – (avec l'option -typecheck pour utiliser l'ontologie de séquence). Utilisez l'outil gff3 avec l'option -tidy pour les ranger (-help affiche d'autres options)." and an error message: "Erreur GenomeTools : tapez "Inc_RNA" à la ligne 230 dans le fichier "/var/www/servers/genometools.org/htdocs/cgi-bin/gff3/genomic_1K.gff" n'est pas valide". A copyright notice at the bottom right reads "Copyright © 2012 Sescha Steinbeis. Dernière mise à jour : 2015-01-25".

Figure 27 : résultat du validateur de fichier GFF3

La figure 27 présente la validation infructueuse du fichier GFF3, montrant une anomalie telle que la présence d'un inconnu : Inc_RNA. Normalement, seuls les termes "mRNA", "ncRNA", "rRNA" et "tRNA" devraient apparaître. Cependant, cela ne devrait pas causer de souci lors de la conversion du GFF3 au format BED.

Awk

Pour obtenir un format tabulaire, j'ai décidé d'utiliser awk afin de voir si le problème venant de la conversion ou de l'utilisation du fichier BED.

```
Awk ' !/^##|^# ! {print $1,$4,$5}' genomic.gff > genomic_AWK.bed
```

La première partie de la commande, '!/^##|^#', exclut les lignes qui débutent par "##" ou "#" (commentaires). Cela permet d'ignorer les en-têtes et les commentaires du fichier GFF. Si une ligne ne débute pas de cette manière, les valeurs des colonnes 1, 4 et 5 sont extraites, puis les résultats sont enregistrés dans le fichier genomic_AWC.bed.

```
NC_063269.1 1 18372659  
NC_063269.1 192994 199951  
NC_063269.1 192994 199951  
NC_063269.1 199714 199951  
NC_063269.1 195655 195777  
NC_063269.1 195471 195588  
NC_063269.1 195136 195397  
NC_063269.1 192994 195073
```

La figure 28 montre les premières lignes du fichier genomic_AWC.bed, ce fichier contient uniquement trois colonnes l'identifiant chromosomique et la position de début et de fin.

Lors de l'utilisation du fichier converti avec awk, j'obtiens une nouvelle erreur : impossible de trouver une méthode héritée pour la fonction 'convertbed2introns' pour la signature de caractère. Cette erreur correspond à de mauvaises données fournies.

Figure 28 : résultat du awk

Utilisation du GFF3 directement

Base de données de transcrits (TxDB)

Dans un premier temps, le fichier genomic.gff est importé dans l'environnement de travail de RStudio.

```
gff3_file <- "genomic.gff"
```

La commande a pour objectif de générer une base de données de transcrits (TxDB) en utilisant le fichier GFF. Une base de données de transcrits est un système de stockage et d'organisation des informations concernant les ARN produits à partir de l'ADN d'un génome. Dans ce contexte, les transcrits font référence aux molécules d'ARN qui sont générées à partir des séquences d'ADN dans le génome.

```
txdb <- makeTxDbFromGFF(gff3_file, format = "gff3")
```

Maintenant que la base de données est construite, elle peut être utilisée dans la suite des analyses.

Récupération des informations dans TxDB

Pour récupérer spécifiquement les détails liés aux gènes à partir de la base de données des transcrits, vous avez la possibilité d'utiliser cette commande. De plus, il est aussi envisageable d'opter pour l'extraction d'autres informations, comme les exons, les introns, les promoteurs et les régions codantes (cds).

```
Genes <- GenomicFeatures::genes(txdb)
```

Objet GRanges sur les informations génomiques

En utilisant cette commande, les données relatives aux gènes sont transformées en un objet GRanges. L'objet GRanges représente les informations des gènes sous la forme d'intervalles génomiques, tout en incluant des attributs comme les identifiants des gènes et les annotations fonctionnelles.

```
genes_GRanges <- as(genes, "GRanges")
```

Les deux variables genes et genes_GRanges contiennent les mêmes données, mais elles sont présentées de manière différente.

Objet GRanges sur les régions différemment méthylées

```
Diff10_GRanges <- as(diff10, "GRanges")
```

Pour utiliser la variable diff10 qui contient les régions différemment méthylées dans les analyses ultérieures et effectuer une annotation, il est nécessaire de convertir cette variable en un objet GRanges. Pour la suite des analyses, l'objet MethylDiff doit être converti en objet GRanges.

Pourcentage des régions différemment méthylées annotée

L'objectif de la fonction genomatation::annotateWithFeature est d'évaluer le pourcentage de recouvrement entre les régions différemment méthylées et les informations génomiques des gènes. Cette fonction permet de quantifier le degré de superposition entre les régions différemment méthylées et les différentes parties des gènes, comme les exons, les promoteurs, les introns, cds.

```
genomatation::annotateWithFeature(diff10_GRanges, genes_GRanges)
```

```
> genomatation::annotateWithFeature(diff10_GRanges, genes_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 3894
-----
percentage of target elements overlapping with features:
genes_GRanges      other
    97.3          2.7
percentage of feature elements overlapping with target:
[1] 15.06
```

La figure 29 illustre le pourcentage des régions présentant des différences de méthylation par rapport aux informations relatives aux différentes parties des gènes, spécifiquement dans le contexte de méthylation CpG CT.

Figure 29 : Pourcentage d'annotation entre diff10 et les infos

Le pourcentage de recouvrement des régions méthylées par rapport aux autres informations génomiques pour les différents contextes de méthylation et les échantillons traités et de contrôle est présenté dans l'annexe 9.

Annotation des régions différemment méthylées

La commande annotateWithGeneParts fait partie du package Genomatation et sert à annoter les régions génomiques différemment méthylées, présentes dans l'objet diff10_GRanges. En y ajoutant les informations concernant les différentes parties des gènes, comme les exons, les introns, les promoteurs et les régions codantes.

Commande classique d'annotation

```
annotateWithGeneParts(diff10_GRanges, genes_GRanges)
```

L'objet diff10_GRanges ainsi que l'objet genes_GRanges sont tous les deux du type GRanges. Néanmoins, la fonction ne peut pas prendre deux objets GRanges en tant qu'arguments simultanément. Cela engendre une erreur indiquant qu'aucune méthode héritée de la fonction n'a été trouvée pour la signature GRanges, GRanges.

Initialement, j'ai tenté de fournir l'objet contenant les différences de méthylation, soit l'objet methyldiff, en tant qu'argument. Cependant, la fonction ne prend pas en charge les objets methyldiff, et ce n'était pas la bonne modification à apporter. L'argument requis doit être un objet du type GRanges.

La conversion de l'objet genes_GRanges en un objet de type GRangesList était une option envisageable pour résoudre ce problème. Cependant, malgré plusieurs tentatives, aucune des méthodes testées n'a abouti. Utiliser la librairie GenomicRanges aurait pu permettre de réaliser cette conversion en utilisant la commande :

```
genes_GRangesList <- GRangesList(genes_GRanges)
genes_GRangesList <- GRangesList(list(genes_GRanges))
genes_GRangesList <- as(genes_GRanges, "GRangesList")
```

Cependant, il faut noter que l'objet genes_GRangesList résultant de cette conversion est en réalité un objet compressedGRangesList, ce qui le rend inutilisable dans la commande annotateWithGeneParts.

Programme d'annotation

Après avoir consacré plusieurs jours à tenter de convertir l'objet de type GRanges en un objet de type GRangesList et à essayer de l'utiliser avec la commande annotateWithGeneParts, j'ai finalement opté pour une approche différente. J'ai décidé de créer mon propre programme en utilisant le langage Python, le script se trouve en annexe 10.

Le code permet d'obtenir des statistiques sur le nombre de SNP annotés et non annotés dans la sortie et fournit aussi un ratio d'annotation pour évaluer la proposition de SNP annoté par rapport à tous les SNP. Il permet d'annoter les positions SNP différemment méthylées en fonction d'un fichier d'annotation génétique.

Préparation des données

Pour faciliter la manipulation des données dans le programme Python, l'option choisie a été de les convertir en format texte. Plutôt que d'opter pour un tableau Excel pour une utilisation plus pratique, la compatibilité entre les systèmes d'exploitation, tels que Linux et LibreOffice, a été un problème. Cependant, le format texte reste uniforme quel que soit le système d'exploitation. Les deux objets GRanges ont donc été convertis en format texte.

La librairie data.table est utilisée pour réorganiser les données sous forme de tableau de base. Cette structure de données est optimisée et facilite la manipulation de grandes quantités de données. Pour convertir les objets en format data.table il faut faire cette commande

```
Gene_table <- as.data.table(gene_Granges)
```

Ensuite, il est nécessaire d'enregistrer ce tableau, mais il y a des considérations à prendre en compte lors de l'enregistrement.

```
Write.table(gene_table, file = 'gene_annotation.txt', quote = FALSE, rox.names = FALSE)
```

L'argument quote = FALSE permet de supprimer les éventuels guillemets autour des valeurs, tandis que l'argument row.names = FALSE permet de ne pas inclure les numéros de lignes. Il faut également faire la même chose pour le fichier diff10_GRanges.

Code python : Vérification des arguments

Dans la première partie du code, le programme annotation.py permet la gestion des arguments donnés au script via ligne de commande. Dans le cas où il manque des informations où s'il y a un argument non pris en charge, le programme le signal.

```
try:
    opts, args = getopt.getopt(sys.argv[1:], "hd:g:o:")
except getopt.GetoptError:
    print ("\n", "### Utilisation non valide ####", "\n")
    print ("Utilisation = DMS_gene.py <options> -d <DMS_pos.txt> -g <genes_pos.txt> -o <output>")
    print ("Pour avoir des informations sur le code, faire: annotation.py -h")
    sys.exit(99)
```

Figure 30 : Vérification des arguments

La Figure 30 présente la première partie du code, qui est utilisée pour vérifier les arguments fournis à partir de la ligne de commande.

En important la bibliothèque getopt en Python, vous pouvez analyser les arguments de la ligne de commande et séparer les options de leurs valeurs. sys.argv[] est utilisé pour récupérer les éléments d'une ligne de commande à partir du terminal.

```
Par exemple : python script.py -d données.csv -g génome.fa -o output
```

sys.argv[] récupère cette liste : [script.py, -d, données.csv, -g, génome.fa, -o, output] et donc en mettant sys.argv[1] on exclut le premier élément de la liste, ce qui signifie qu'il ne reste que [-d, données.csv, -g, génome.fa, -o, output]. En d'autres termes, cela permet d'ignorer le nom du script dans la liste. L'expression "hd : g : o" est utilisée pour spécifier les options attendues et les options nécessitant un argument dans la ligne de commande. Chaque lettre représente une option, et lorsque la lettre est suivie de ":", cela indique que cette option nécessite un argument.

Il est crucial de noter qu'il y a la création de deux listes distinctes : opts et args. La liste opts regroupe les options analysées par la fonction getopt.getopt. En parallèle, la liste "args" rassemble les arguments restants une fois que la fonction getopt.getopt a été exécutée. Dans l'**exemple précédent**, la liste opts contient deux tuples : (-d, données.csv) et (-g, génome.fa), tandis que la liste "args" contient les options optionnelles (output).

La seconde partie de la première commande except getopt.getopt permet de signaler si les arguments donné ne sont pas valide. Alors, il y a un message d'erreur qui s'affiche simplement avec des prints indiquant une mauvaise utilisation du script. Sys.exit(99) permet de mettre un code de sortie, 0 si tout s'est bien passé et 99 s'il y a eu une erreur.

```
camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Elements/bourdons/annotation/Rstudio$ python gene_annotation.py -d CT/cpg/diff10_ct_cpg.txt -i genes_annotation.txt -o test
### Utilisation non valide ####
Utilisation = DMS_gene.py <options> -d <DMS_pos.txt> -g <genes_pos.txt> -o <output>
Pour avoir des informations sur le code, faire: annotation.py -h
```

Figure 31 : erreur lorsqu'un argument non répertorié est inscrit

La Figure 31 illustre la sortie du programme lorsqu'un argument est spécifié mais n'est pas répertorié parmi les arguments possibles.

Code python : analyses des arguments de la commande

La seconde partie du code, illustrée dans l'annexe 10 A dans la partie de la boucle *for*, permet d'analyser les différents arguments fournis, c'est-à-dire les arguments présents dans les listes args et opts.

La variable opts contient les options de la commande récupérées par la fonction getopt.getopt. Chaque option est composée d'une étiquette (opt) et d'un argument éventuel (arg). Dans ce cas-ci, si l'option -h est présente dans la ligne de commande, le script affiche simplement une aide pour exécuter correctement le programme, puis il se termine avec la fonction sys.exit(). C'est ce que montre la figure 32.

```
camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Elements/bourdons/annotation/Rstudio$ python gene_annotation.py -h
Obtenir l'annotation génétique des positions DMS à partir d'un fichier d'annotation génétique.

Utilisation = DMS_gene.py <options> -d <DMS_pos.txt> -g <genes_pos.txt> -o <output_name>

DMS_pos.txt = Une liste délimitée par des espaces de tous les SNP différemment méthylés dans laquelle la première colonne est le chrm, et la seconde la position du SNP. Toutes les autres colonnes sont copiées dans la sortie.

genes_pos.txt = Un fichier délimité par des espaces contenant tous les gènes (un par ligne). Les colonnes ont le format suivant : chrm, start, end, width, strand, gene_id.

output_name = Le nom pour enregistrer le fichier de sortie. Le fichier de sortie contiendra toutes les informations du fichier d'entrée DMS_pos.txt et une colonne supplémentaire contenant l'identifiant du gène.

attention, avant de lancer le programme, il faut avoir formaté les deux fichiers au format texte!
```

Figure 32 : Affichage de l'aide du programme

Ensuite, dans la suite de cette partie, nous traitons les options de la commande provenant du terminal. Lorsqu'au moins deux options sont spécifiées (parmi -d, -g et -o), le traitement s'effectue comme suit :

- Si l'option -d est spécifiée, l'étiquette opt reçoit la valeur -d et arg prend la valeur du nom du fichier. Ensuite, le fichier est ouvert avec la fonction open() et son contenu est attribué à la variable in_DMS.
- Si l'option -g est spécifiée, de manière similaire, opt prend la valeur -g et "arg" prend la valeur du nom du fichier. Ce fichier est également ouvert avec open() et son contenu est stocké dans la variable in_gene.
- Si l'option -o est présente, encore une fois, opt prend la valeur -o et arg reçoit le nom du fichier. Ce fichier est ouvert avec open() et son contenu est sauvegardé dans la variable out.

Dans le scénario où la commande du terminal contient moins de 2 options, une erreur est générée pour signaler le manque d'arguments. Cela indique que la commande est incorrecte et qu'il est nécessaire de fournir plus d'arguments.

Et enfin, le else lorsqu'aucun argument n'est spécifié à la ligne de commande, le programme signal qu'il faut davantage d'argument et propose d'ajouter -h pour obtenir de l'aide.

```
camille@camille-Lenovo-ideapad-330-15ARR:/media/camille/Elements/bourdons/annotation/Rstudio$ python gene_annotation.py -o test
Traceback (most recent call last):
  File "/media/camille/Elements/bourdons/annotation/Rstudio/gene_annotation.py", line 31, in <module>
    assert False, "Ce programme attend deux fichiers textes, contenant les différences de méthylation et les informations génétiques. faire h pour plus d'information"
AssertionError: Ce programme attend deux fichiers textes, contenant les différences de méthylation et les informations génétiques. faire h pour plus d'information
```

Figure 33 : erreur quand il n'y a pas assez d'argument

Code python : création de variables

```
header = True
total_annotated = 0
total_NotAnnotated = 0
gene_dict = {}
chrm_dict = {}
gene_list = []
```

La figure X montre la partie est simplement une assignation de variable, dictionnaire et liste. Cependant, header = TRUE, signifie que dans les fichiers importés la première ligne est celle représentant les entêtes de colonne.

Figure 34 : déclaration de variables

Code python : stocker les informations des gènes

Dans cette partie du code, des dictionnaires sont créés pour stocker les informations des gènes provenant du fichier d'annotation au format texte gene_annotation.txt. L'utilisation de dictionnaires facilite la manipulation des informations. Le dictionnaire gene_dic est conçu pour stocker les détails des gènes, et les informations essentielles sont regroupées sous une même clé.

L'annexe 10 B montre la partie du code permettant de réaliser le stockage des informations des gènes. La boucle *for* est utilisée pour parcourir le contenu du fichier gene_annotation.txt ligne par ligne, en ignorant la première ligne si elle est considérée comme un en-tête. Lorsqu'il y a un en-tête, la boucle passe à la ligne suivante. Si la ligne n'est pas un en-tête, des actions sont entreprises.

Dans un premier temps, la ligne est divisée en une liste appelée entry, en utilisant les espaces comme délimiteurs. Cela permet de supprimer les espaces en fin de ligne. Des variables sont créées pour stocker les informations de chaque ligne du fichier : Le nom du chromosome, la position de début et de fin du gène, l'identifiant du gène.

Une vérification est effectuée pour déterminer si le nom du chromosome existe déjà en tant que clé dans le dictionnaire chrm_dict. Si c'est le cas, la valeur associée à cette clé est récupérée. Dans ce dictionnaire, les identifiants des gènes et les chromosomes sont enregistrés. L'identifiant du gène actuel est ajouté à la liste d'identifiants des gènes associée au chromosome en question.

Les identifiants des gènes sont regroupés en une chaîne de caractères séparée par des virgules, formant ainsi une liste. Si le nom du chromosome n'existe pas encore en tant que clé, cela signifie que c'est le premier gène pour ce chromosome. Dans ce cas, une nouvelle clé est créée dans le dictionnaire chrm_dict avec le nom du chromosome, et l'identifiant du gène actuel est enregistré comme première valeur de la liste associée à cette clé.

Code python : obtenir l'annotation

Le fichier contenant les différences de méthylation (DMS) est lu, et on commence à annoter les positions des SNP en les comparant aux positions de début et de fin des gènes. Les SNP sont vérifiés pour voir s'ils se trouvent à l'intérieur des positions de début et de fin des gènes. SNPs = *Single Nucleotide Polymorphism*. Un type de mutation qui va se passer dans un seul nucléotide.

La suite de l'annexe 10 B montre même manière que pour le fichier précédent, la boucle for lit le fichier ligne par ligne. La méthode r.strip() est utilisée pour supprimer les caractères de nouvelle ligne ou d'espacement à droite de la ligne. Si une en-tête est détectée, l'option *false* permet de l'ignorer. La fonction out.write permet d'écrire le contenu des annotations dans le fichier de sortie tout en y insérant l'en-tête. Continue la boucle ensuite.

La variable entry crée en séparant les valeurs de la liste avec des espaces avec split, contenant les valeurs de la ligne. La variable pos_SNP extraction des deux éléments de la liste entry c'est-à-dire la position en un int. Ensuite, récupération la liste des identifiants des gènes qui sont sur le même chromosome extrait via DMS et on les stocks dans search_gene.

Le deuxième for parcourt les éléments de search_gene, gene_annot est initialisée False avant de débuter la comparaison pour chaque identifiant de gène. La variable gene_info récupère les informations des gènes correspondant dans le dictionnaire gene_dict. le if fait la comparaison entre la position du SNP et les positions de début et de fin du gène.

Les informations des SNP sont enregistrées en écrivant la ligne complète du SNP dans le fichier de sortie. total.annotated compte le nombre total de SNP annotés, et gene_annot indique si le SNP a été annoté avec un gène. La commande break permet de sortir de la boucle une fois que l'annotation d'un SNP a été réalisée.

Le dernier bloc if dans le code intervient lorsque le chevauchement n'existe pas entre le SNP et tous les gènes associés à son chromosome (recherche_gene). Cela signifie que le SNP ne peut pas être annoté avec un gène. Lorsque gene_annot est faux, cela indique que le SNP n'a pas été annoté avec un gène. La commande out.write ajoute "NA" pour indiquer qu'il n'y a pas d'annotation pour ce SNP total_not.annotated compte le nombre total de SNP non annotés avec un gène.

Affichage des résultats

Ensuite, le programme affiche le nombre de SNP annotés et non annotés. Pour calculer le taux d'annotation en pourcentage, j'ai divisé le nombre de SNP annotés par le nombre total de SNP (à la fois annotés et non annotés), puis multiplié le résultat par 100. Le résultat est ensuite affiché avec le symbole de pourcentage.

$$\left(\frac{\text{Nombre de SNP annotés}}{\text{Nombre total de SNP(annoté et non annotés)}} \right) * 100$$

Voici la commande à utiliser pour exécuter le programme dans le terminal :

```
Python annotation.py -d diff10_ct_chg.txt -g gene_annotation.txt -o annotation_python
```

```
-----
Total SNPs annotés: 3789
Total SNPs non annotés: 105
Annotation ratio: 97.30354391371341%
```

Lorsque le programme s'est terminé, il y a un affichage des résultats. La figure X représente le résultat obtenu avec le fichier des régions différemment méthylée du contexte de méthylation CpG pour les échantillons de contrôle avec les informations des gènes.

Figure 35 : résultat d'annotation

Le tableau 14 récapitulation des résultats obtenus à la suite du programme d'annotation génétique.

Échantillons	SNP annotés	SNP non annotés	Ration d'annotation
CT CpG	3 789	105	97,30 %
CT CHG	12	0	100,0 %
CT CHH	31	0	100,0 %

RG CpG	5 439	123	97,79 %
RG CHG	14	0	100,0 %
RG CHH	41	2	95,35 %

Tableau 14 : récapitulatif des gènes annotés

Après l'exécution du programme, voici un aperçu du contenu du fichier de sortie au format, à titre illustratif, en commençant par la section CT_CpG.

seqnames start end width strand pvalue qvalue meth.diff gene_id
NC_063269.1 252980 252980 1 - 1.67461547043454e-07 0.000607155464242229 42.1052631578947 LOC100650265
NC_063269.1 254096 254096 1 + 2.8602672491034e-14 1.09646588180485e-09 59.6153846153846 LOC100650265
NC_063269.1 261879 261879 1 + 1.94277626377245e-07 0.000685909376072947 -36.7816091954023 LOC105666912
NC_063269.1 263564 263564 1 - 1.59126337431588e-08 8.6089869593616e-05 37.8621378621379 LOC100650629
NC_063269.1 269692 269692 1 - 6.44829787935653e-07 0.00184947714012031 -37.9689754689755 LOC100650754
NC_063269.1 287965 287965 1 - 5.50715149655469e-07 0.00162458258597236 20.5128205128205 LOC100650872
NC_063269.1 288060 288060 1 - 1.56997838258464e-19 2.32476057411823e-14 51.9607843137255 LOC100650872
NC_063269.1 295221 295221 1 + 5.41518663961184e-10 5.18969634376041e-06 52.4193548387097 LOC100650872
NC_063269.1 295228 295228 1 + 3.55763898735694e-09 2.50163464688101e-05 50.5218216318786 LOC100650872
NC_063269.1 639617 639617 1 - 6.53336130462535e-10 6.12908339987656e-06 34.9206349206349 LOC100652027
NC_063269.1 667589 667589 1 - 5.53346837162177e-08 0.000244804076892049 -36.516290726817 LOC100642551
NC_063269.1 670296 670296 1 - 1.00897323129839e-10 1.24582884489691e-06 -42.1370967741936 LOC100652148
NC_063269.1 670923 670923 1 + 1.68701836111055e-08 9.05483304818291e-05 26.008064516129 LOC100652148
NC_063269.1 671352 671352 1 - 2.77651837803951e-13 7.73723872680397e-09 -47.5609756097561 LOC100652148
NC_063269.1 672521 672521 1 - 4.46555170520155e-11 6.21054950610815e-07 -40.2777777777777 LOC100652148
NC_063269.1 690194 690194 1 - 6.52580167443679e-07 0.00186815890804911 -26.865671641791 LOC100642312
NC_063269.1 704630 704630 1 - 1.03441136546786e-06 0.00273329051650847 -42.4968474148802 LOC100642790
NC_063269.1 704885 704885 1 - 5.03650759416732e-08 0.000226130028247294 -28.8135593220339 LOC100642790
NC_063269.1 708823 708823 1 - 6.85523490957632e-09 4.37615133368739e-05 -44.5626477541371 LOC100642910
NC_063269.1 727045 727045 1 + 1.55938171683191e-07 0.000573611994321396 30.4347826086956 LOC100643027
NC_063269.1 727286 727286 1 + 7.24052427177594e-08 0.000305472217845775 38.2056074766355 LOC100643027
NC_063269.1 728898 728898 1 + 2.1248511205304e-08 0.000109757920184709 25.609756097561 LOC100643027
NC_063269.1 731777 731777 1 - 3.23284736463357e-11 4.74981633442505e-07 -55.266887104393 LOC100643270
NC_063269.1 738311 738311 1 - 5.83070618069217e-08 0.0002560004409346561 28.8135593220339 LOC100643837
NC_063269.1 739613 739613 1 - 2.68399174297084e-07 0.00089686581681222 26.5625 LOC100643837
NC_063269.1 786578 786578 1 - 3.18428721347061e-06 0.00676749305322272 -38.138761944329 LOC100645550
NC_063269.1 791140 791140 1 + 1.03605811281733e-07 0.000413954881707541 -45.9733893557423 NA

La Figure présente le contenu du fichier CT_CpG après l'exécution du programme Python d'annotation. Les listes de gènes illustrent les variations existantes entre les différents âges, révélant des changements liés à l'âge pour ces gènes.

Conclusion

La question de départ était d'évaluer les changements de méthylation induits par l'âge, indépendamment du traitement. Lorsque l'on regroupe les échantillons traités et les échantillons témoins, on observe non seulement les changements de méthylation liés à l'âge, mais également ceux liés au traitement.

Pour répondre à la question initiale, il s'est avéré qu'il existe une hyperméthylation à l'échelle du génome chez les ouvrières traitées par rapport aux témoins, à deux moments distincts. Cette augmentation des niveaux d'hyperméthylation induite par l'utilisation de l'agent hypométhylant RG108 pourrait être attribuée à l'influence de cet agent sur les gènes ayant un impact direct ou indirect sur la méthylation de l'ADN.

Le projet de recherche n'a malheureusement pas pu aboutir dans son intégralité. Les résultats obtenus ont mis en évidence des variations de méthylation entre les différents groupes d'âge. Cependant, en raison de contraintes temporelles, il n'a pas été possible de mener une analyse statistique approfondie pour déterminer la validité de ces résultats. Une autre piste que j'aurais souhaité explorer était l'attribution de fonctions biologiques aux gènes annotés, afin d'évaluer leur possible impact sur le processus de vieillissement.

Dans le contexte de la méthylation CpG, où plusieurs milliers de gènes sont impliqués, l'analyse individuelle de chacun d'eux est une tâche ardue. Par conséquent, il est nécessaire de trouver une approche pour regrouper ces gènes afin de les analyser plus simplement. Malheureusement, il n'a pas été possible d'examiner l'ontologie des gènes faute de temps.

Pour établir de manière concrète le rôle des gènes dans le processus de vieillissement, il devient nécessaire d'explorer les gènes candidats au sein du laboratoire. Cette démarche implique le traitement spécifique de ces gènes, que ce soit en inhibant ou en amplifiant leur fonction. Une fois ces manipulations effectuées, il devient possible d'établir une corrélation directe entre le gène étudié et le phénomène de vieillissement.

Perspectives

Fiabilité des résultats

Pour accroître la fiabilité des résultats, il est envisageable d'incorporer des tests statistiques aux procédures, ce qui permettra de déterminer la crédibilité des résultats obtenus. Dans le but de déterminer si la distinction entre les deux groupes en fonction de l'âge était réellement significative.

L'approche statistique consiste à effectuer des permutations aléatoires des traitements, puis à identifier les cytosines présentant des méthylations différentielles de manière similaire. Si le même nombre de cytosines est observé en modifiant les traitements, cela indiquerait que les résultats pourraient être obtenus par hasard plutôt qu'en lien avec l'âge. L'objectif est de démontrer que le mélange des groupes ne conduit pas aux mêmes niveaux de méthylation différentielle.

Au sein de methylKit, il existe une fonction appelée `reorganize` qui permet de réorganiser de manière aléatoire les groupes de traitement. Mon plan était de répéter cette opération plusieurs fois, puis de calculer les régions présentant une méthylation différentielle pour chacun de ces nouveaux groupes. Les résultats de ces calculs auraient été stockés dans une matrice. Par la suite, l'évaluation aurait été effectuée pour déterminer combien de fois le même nombre de régions présentant une méthylation différentielle a été obtenu par hasard. Cette approche statistique aurait alors permis de déterminer le degré de confiance à accorder aux données.

Par exemple, si l'analyse initiale identifie 100 cytosines présentant une méthylation différentielle entre les deux âges, une fois les groupes de traitement mélangés de manière aléatoire, on peut constater que dans 90% des cas, seulement 5 régions présentent une méthylation différentielle. Cette cohérence renforce la fiabilité de nos résultats et confirme que l'âge joue un rôle distinctif dans nos données.

Ontologie des gènes

Les GO sont classées en trois catégories principales : les fonctions biologiques, les fonctions moléculaires et les composants cellulaires. L'objectif consiste à assigner une fonction particulière à chaque gène annoté par le programme python. Les GO sont souvent utilisées pour définir la fonction d'un gène spécifique.

Un gène peut présenter plusieurs GO différents. Si 100 gènes possèdent plusieurs GO, une comparaison est effectuée avec tous les autres gènes du génome afin de déterminer si ces 100 gènes partagent plusieurs GO liées à une fonction spécifique. En examinant le pourcentage dans le génome, il est possible de constater si les résultats correspondent aux attentes.

Annexe 1 A : rapport d'alignement Bismark S1_CT_2B

```
Final Alignment report
=====
Sequence pairs analysed in total:      53079323
Number of paired-end alignments with a unique best hit: 32623049
Mapping efficiency:      61.5%
Sequence pairs with no alignments under any condition: 11911456
Sequence pairs did not map uniquely:      8544818
Sequence pairs which were discarded because genomic sequence could not be extracted:      14

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      16370778      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      16252257      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed:      1127935486

Total methylated C's in CpG context:      2675239
Total methylated C's in CHG context:      931944
Total methylated C's in CHH context:      4342190
Total methylated C's in Unknown context:      1826

Total unmethylated C's in CpG context:      320136448
Total unmethylated C's in CHG context:      163136879
Total unmethylated C's in CHH context:      636712786
Total unmethylated C's in Unknown context:      283766

C methylated in CpG context:      0.8%
C methylated in CHG context:      0.6%
C methylated in CHH context:      0.7%
C methylated in unknown context (CN or CHN):      0.6%
```

Annexe 1B : rapport d'alignement Bismark S1_CT_3B

```
Final Alignment report
=====
Sequence pairs analysed in total:      65940094
Number of paired-end alignments with a unique best hit: 38827188
Mapping efficiency:      58.9%
Sequence pairs with no alignments under any condition: 15732893
Sequence pairs did not map uniquely:    11380013
Sequence pairs which were discarded because genomic sequence could not be extracted:    17

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      19460877      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      19366294      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed:   1348561630

Total methylated C's in CpG context:  3084413
Total methylated C's in CHG context:  986938
Total methylated C's in CHH context:  4131427
Total methylated C's in Unknown context:      1770

Total unmethylated C's in CpG context: 380447444
Total unmethylated C's in CHG context: 193947135
Total unmethylated C's in CHH context: 765964273
Total unmethylated C's in Unknown context:      351279

C methylated in CpG context:    0.8%
C methylated in CHG context:    0.5%
C methylated in CHH context:    0.5%
C methylated in unknown context (CN or CHN):    0.5%

Bismark completed in 0d 7h 9m 26s
```

Annexe 1C : rapport d'alignement Bismark S1_CT_4B

```
Final Alignment report
=====
Sequence pairs analysed in total:      61917622
Number of paired-end alignments with a unique best hit: 38915379
Mapping efficiency:      62.9%
Sequence pairs with no alignments under any condition: 14238230
Sequence pairs did not map uniquely:      8764013
Sequence pairs which were discarded because genomic sequence could not be extracted:      8

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      19516193      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      19399178      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed:      1363636507

Total methylated C's in CpG context:      2823484
Total methylated C's in CHG context:      983284
Total methylated C's in CHH context:      4007539
Total methylated C's in Unknown context:      1708

Total unmethylated C's in CpG context:      393297374
Total unmethylated C's in CHG context:      198670733
Total unmethylated C's in CHH context:      763854093
Total unmethylated C's in Unknown context:      348956

C methylated in CpG context:      0.7%
C methylated in CHG context:      0.5%
C methylated in CHH context:      0.5%
C methylated in unknown context (CN or CHN):      0.5%

Bismark completed in 0d 10h 55m 19s
```

Annexe 1D : rapport d'alignement Bismark S1_RG_2A

```
Final Alignment report
=====
Sequence pairs analysed in total:      77864681
Number of paired-end alignments with a unique best hit: 51689441
Mapping efficiency:      66.4%
Sequence pairs with no alignments under any condition: 12994844
Sequence pairs did not map uniquely: 13180396
Sequence pairs which were discarded because genomic sequence could not be extracted:      21

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      25908433      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      25780987      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed: 1846185040

Total methylated C's in CpG context: 4047086
Total methylated C's in CHG context: 1403416
Total methylated C's in CHH context: 5470219
Total methylated C's in Unknown context: 2081

Total unmethylated C's in CpG context: 549918278
Total unmethylated C's in CHG context: 274183434
Total unmethylated C's in CHH context: 1011162607
Total unmethylated C's in Unknown context: 472712

C methylated in CpG context: 0.7%
C methylated in CHG context: 0.5%
C methylated in CHH context: 0.5%
C methylated in unknown context (CN or CHN): 0.4%

Bismark completed in 0d 8h 30m 51s
```

Annexe 1E : rapport d'alignement Bismark S1_RG_4B

```
Final Alignment report
=====
Sequence pairs analysed in total:      72272758
Number of paired-end alignments with a unique best hit: 47693268
Mapping efficiency:      66.0%
Sequence pairs with no alignments under any condition: 12375561
Sequence pairs did not map uniquely: 12203929
Sequence pairs which were discarded because genomic sequence could not be extracted:      22

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      23892047      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      23801199      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed: 1680148430

Total methylated C's in CpG context: 3418438
Total methylated C's in CHG context: 1051814
Total methylated C's in CHH context: 4227474
Total methylated C's in Unknown context: 1812

Total unmethylated C's in CpG context: 486940294
Total unmethylated C's in CHG context: 245528613
Total unmethylated C's in CHH context: 938981797
Total unmethylated C's in Unknown context: 451765

C methylated in CpG context: 0.7%
C methylated in CHG context: 0.4%
C methylated in CHH context: 0.4%
C methylated in unknown context (CN or CHN): 0.4%

Bismark completed in 0d 14h 57m 37s
```

Annexe 1F : rapport d'alignement Bismark S1_RG_5B

```
Final Alignment report
=====
Sequence pairs analysed in total:      68851435
Number of paired-end alignments with a unique best hit: 45738413
Mapping efficiency:      66.4%
Sequence pairs with no alignments under any condition: 11306771
Sequence pairs did not map uniquely: 11806251
Sequence pairs which were discarded because genomic sequence could not be extracted:      23

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      22921727      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      22816663      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed: 1597870209

Total methylated C's in CpG context: 3160262
Total methylated C's in CHG context: 1021732
Total methylated C's in CHH context: 4146603
Total methylated C's in Unknown context: 1587

Total unmethylated C's in CpG context: 461248655
Total unmethylated C's in CHG context: 232316080
Total unmethylated C's in CHH context: 895976877
Total unmethylated C's in Unknown context: 413954

C methylated in CpG context: 0.7%
C methylated in CHG context: 0.4%
C methylated in CHH context: 0.5%
C methylated in unknown context (CN or CHN): 0.4%
```

Bismark completed in 0d 15h 26m 39s

Annexe 1G : rapport d'alignement Bismark S4_CT_3A

```
Final Alignment report
=====
Sequence pairs analysed in total:      77722842
Number of paired-end alignments with a unique best hit: 52251701
Mapping efficiency:      67.2%
Sequence pairs with no alignments under any condition: 14094140
Sequence pairs did not map uniquely: 11377001
Sequence pairs which were discarded because genomic sequence could not be extracted:      17

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      26226412      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      26025272      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed: 1839098222

Total methylated C's in CpG context: 4002508
Total methylated C's in CHG context: 1406929
Total methylated C's in CHH context: 5628478
Total methylated C's in Unknown context: 2164

Total unmethylated C's in CpG context: 538537128
Total unmethylated C's in CHG context: 271092394
Total unmethylated C's in CHH context: 1018430785
Total unmethylated C's in Unknown context: 450481

C methylated in CpG context: 0.7%
C methylated in CHG context: 0.5%
C methylated in CHH context: 0.5%
C methylated in unknown context (CN or CHN): 0.5%

Bismark completed in 0d 7h 15m 8s
```

Annexe 1H : rapport d'alignement Bismark S4_CT_4B

```
Final Alignment report
=====
Sequence pairs analysed in total:      61533724
Number of paired-end alignments with a unique best hit: 40078008
Mapping efficiency:      65.1%
Sequence pairs with no alignments under any condition: 12599675
Sequence pairs did not map uniquely: 8856041
Sequence pairs which were discarded because genomic sequence could not be extracted:      12

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      20113979      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      19964017      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed: 1396373212

Total methylated C's in CpG context: 3035474
Total methylated C's in CHG context: 1043547
Total methylated C's in CHH context: 4299428
Total methylated C's in Unknown context: 1872

Total unmethylated C's in CpG context: 399287298
Total unmethylated C's in CHG context: 202966032
Total unmethylated C's in CHH context: 785741433
Total unmethylated C's in Unknown context: 353611

C methylated in CpG context: 0.8%
C methylated in CHG context: 0.5%
C methylated in CHH context: 0.5%
C methylated in unknown context (CN or CHN): 0.5%


Bismark completed in 0d 11h 10m 55s
```

Annexe 1 i : rapport d'alignement Bismark S4_CT_5A

```
Final Alignment report
=====
Sequence pairs analysed in total:      77674533
Number of paired-end alignments with a unique best hit: 49840112
Mapping efficiency:      64.2%
Sequence pairs with no alignments under any condition: 14403130
Sequence pairs did not map uniquely: 13431291
Sequence pairs which were discarded because genomic sequence could not be extracted:      15

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      24965429      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      24874668      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed: 1765028704

Total methylated C's in CpG context: 3847881
Total methylated C's in CHG context: 1338068
Total methylated C's in CHH context: 5288927
Total methylated C's in Unknown context: 2182

Total unmethylated C's in CpG context: 522212757
Total unmethylated C's in CHG context: 261333421
Total unmethylated C's in CHH context: 971007650
Total unmethylated C's in Unknown context: 423228

C methylated in CpG context: 0.7%
C methylated in CHG context: 0.5%
C methylated in CHH context: 0.5%
C methylated in unknown context (CN or CHN): 0.5%

Bismark completed in 0d 8h 9m 8s
```

Annexe 1J : rapport d'alignement Bismark S4_RG_1A

```
Final Alignment report
=====
Sequence pairs analysed in total:      97832201
Number of paired-end alignments with a unique best hit: 60160651
Mapping efficiency:      61.5%
Sequence pairs with no alignments under any condition:  22395237
Sequence pairs did not map uniquely:    15276313
Sequence pairs which were discarded because genomic sequence could not be extracted:    24
```

```
Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      30135937      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      30024690      ((converted) bottom strand)
```

```
Number of alignments to (merely theoretical) complementary strands being rejected in total:      0
```

Final Cytosine Methylation Report

```
=====
Total number of C's analysed:  2144040313
```

```
Total methylated C's in CpG context:  4728299
Total methylated C's in CHG context:  1689191
Total methylated C's in CHH context:  6557379
Total methylated C's in Unknown context:      2561
```

```
Total unmethylated C's in CpG context:  641291038
Total unmethylated C's in CHG context:  319977157
Total unmethylated C's in CHH context:  1169797249
Total unmethylated C's in Unknown context:      507520
```

```
C methylated in CpG context:      0.7%
C methylated in CHG context:      0.5%
C methylated in CHH context:      0.6%
C methylated in unknown context (CN or CHN):      0.5%
```

Annexe 1K : rapport d'alignement Bismark S4_RG_2B

```
Final Alignment report
=====
Sequence pairs analysed in total:      60108871
Number of paired-end alignments with a unique best hit: 30373501
Mapping efficiency:      50.5%
Sequence pairs with no alignments under any condition: 21741205
Sequence pairs did not map uniquely:    7994165
Sequence pairs which were discarded because genomic sequence could not be extracted:      15

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      15209937      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      15163549      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed:   1060476763

Total methylated C's in CpG context:  2412100
Total methylated C's in CHG context:  771038
Total methylated C's in CHH context:  3160193
Total methylated C's in Unknown context:      1432

Total unmethylated C's in CpG context: 304133629
Total unmethylated C's in CHG context: 154358046
Total unmethylated C's in CHH context: 595641757
Total unmethylated C's in Unknown context:      265272

C methylated in CpG context:      0.8%
C methylated in CHG context:      0.5%
C methylated in CHH context:      0.5%
C methylated in unknown context (CN or CHN):      0.5%


Bismark completed in 0d 10h 27m 2s
```

Annexe 1L : rapport d'alignement Bismark S4_RG_5B

```
Final Alignment report
=====
Sequence pairs analysed in total:      59789122
Number of paired-end alignments with a unique best hit: 39253356
Mapping efficiency:      65.7%
Sequence pairs with no alignments under any condition: 10848322
Sequence pairs did not map uniquely: 9687444
Sequence pairs which were discarded because genomic sequence could not be extracted:      11

Number of sequence pairs with unique best (first) alignment came from the bowtie output:
CT/GA/CT:      19693430      ((converted) top strand)
GA/CT/CT:      0      (complementary to (converted) top strand)
GA/CT/GA:      0      (complementary to (converted) bottom strand)
CT/GA/GA:      19559915      ((converted) bottom strand)

Number of alignments to (merely theoretical) complementary strands being rejected in total:      0

Final Cytosine Methylation Report
=====
Total number of C's analysed: 1359223473

Total methylated C's in CpG context: 3156067
Total methylated C's in CHG context: 998214
Total methylated C's in CHH context: 4124485
Total methylated C's in Unknown context: 1591

Total unmethylated C's in CpG context: 384578918
Total unmethylated C's in CHG context: 197341876
Total unmethylated C's in CHH context: 769023913
Total unmethylated C's in Unknown context: 331567

C methylated in CpG context: 0.8%
C methylated in CHG context: 0.5%
C methylated in CHH context: 0.5%
C methylated in unknown context (CN or CHN): 0.5%


Bismark completed in 0d 11h 54m 43s
```

Annexe 2 A : Statistiques pourcentage de méthylation Semaine 1 et 4 CT

```
> getMethylationStats(normal_cov_CT_cpg[[1]], plot = FALSE, both.strands = FALSE)
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.8003 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905 14.285714 44.444444 83.333333 100.000000

> getMethylationStats(normal_cov_CT_cpg[[2]], plot = FALSE, both.strands = FALSE)
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.7757 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 4.347826 13.333333 42.857143 84.375000 100.000000

> getMethylationStats(normal_cov_CT_cpg[[3]], plot = FALSE, both.strands = FALSE)
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.657 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 4.166667 7.692308 29.629630 76.470588 100.000000

> getMethylationStats(normal_cov_CT_cpg[[4]], plot = FALSE, both.strands = FALSE)
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.7378 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905 10.000000 28.000000 78.260870 100.000000

> getMethylationStats(normal_cov_CT_cpg[[5]], plot = FALSE, both.strands = FALSE)
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.6079 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 4.347826 8.333333 33.333333 77.777778 100.000000

> getMethylationStats(normal_cov_CT_cpg[[6]], plot = FALSE, both.strands = FALSE)
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.7137 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905 9.090909 27.272727 82.352941 100.000000
```

Annexe 2 B : Statistiques pourcentage de méthylation RG S1 et S4

```
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.000 0.000 0.000 0.681 0.000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 4.166667 8.333333 28.571429 81.250000 100.000000

methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.6306 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 3.448276 7.692308 31.034483 82.142857 100.000000

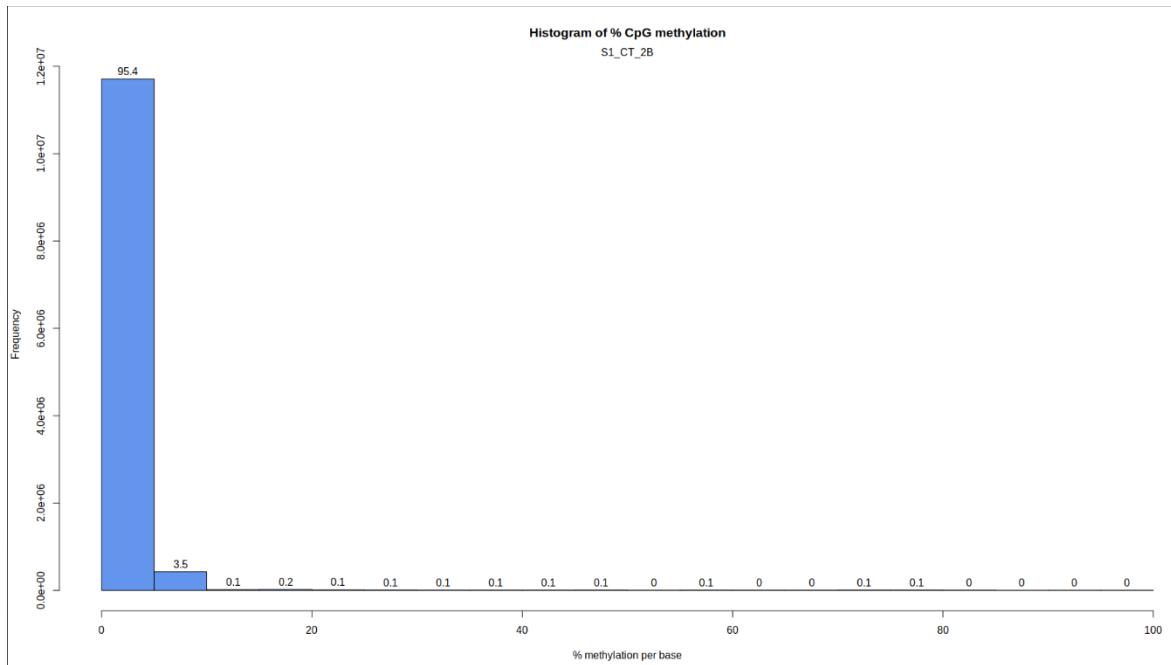
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.6148 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 3.448276 7.692308 29.166667 81.250000 100.000000

methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.7375 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 3.448276 7.692308 26.666667 80.000000 100.000000

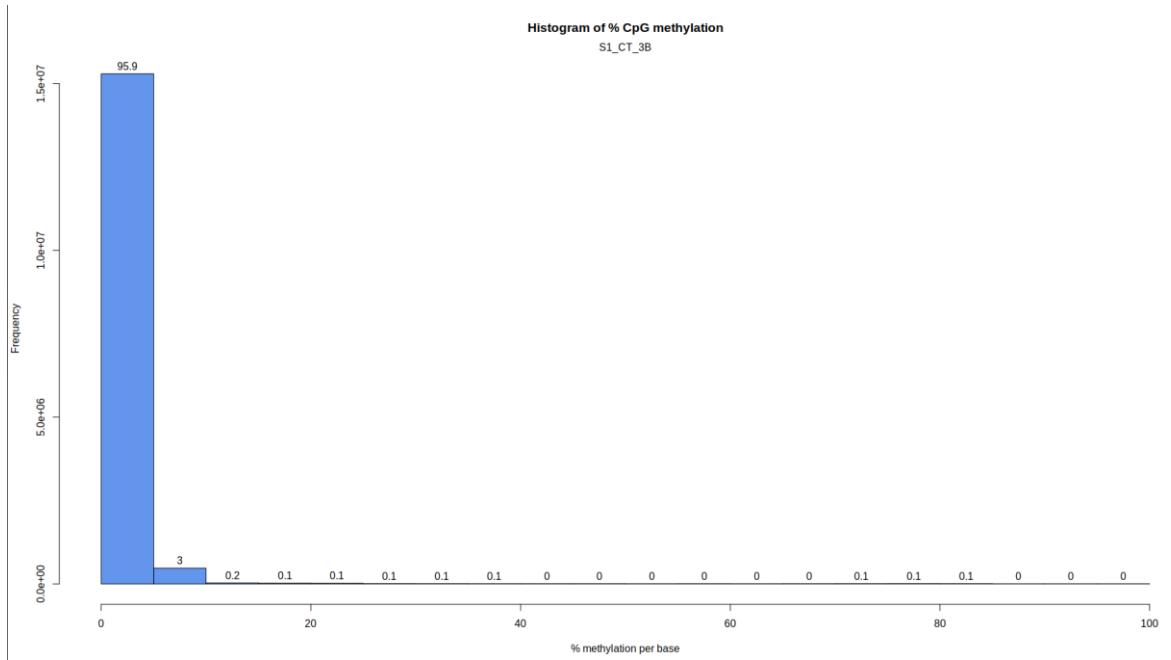
methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.9629 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 7.692308 12.500000 45.833333 86.956522 100.000000

methylation statistics per base
summary:
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.0000 0.0000 0.0000 0.9029 0.0000 100.0000
percentiles:
  % 10% 20% 30% 40% 50% 60% 70% 80% 90% 95% 99% 99.5% 99.9% 100%
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 6.250000 13.333333 42.857142 85.71429 100.000000
```

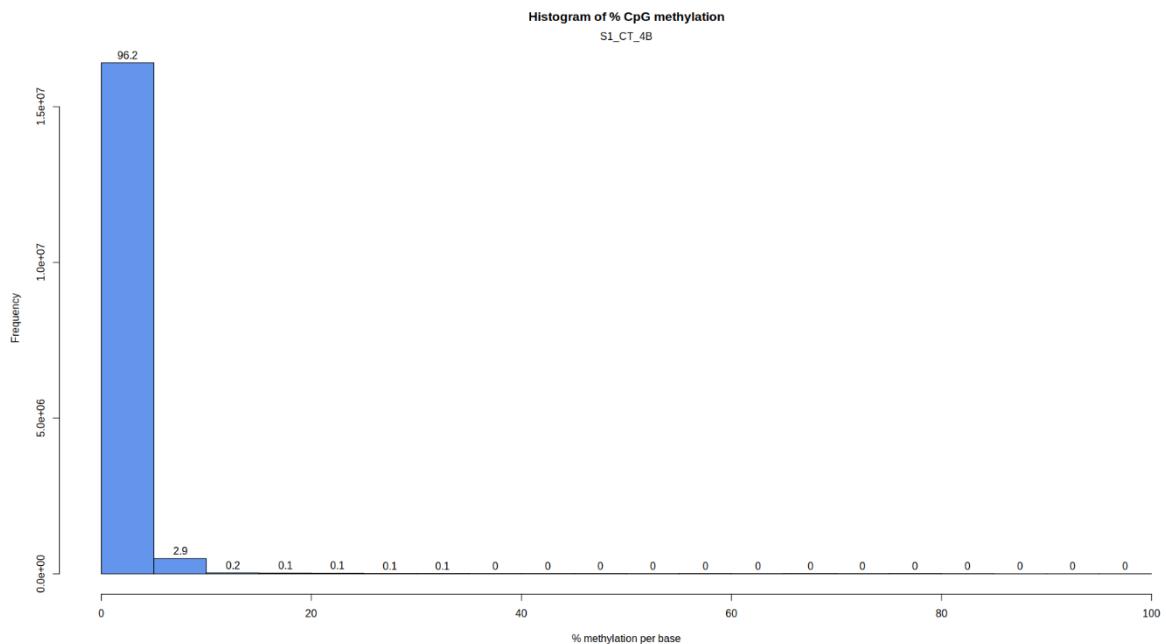
Annexe 3 A : histogramme de l'échantillon S1_CT_2B CpG



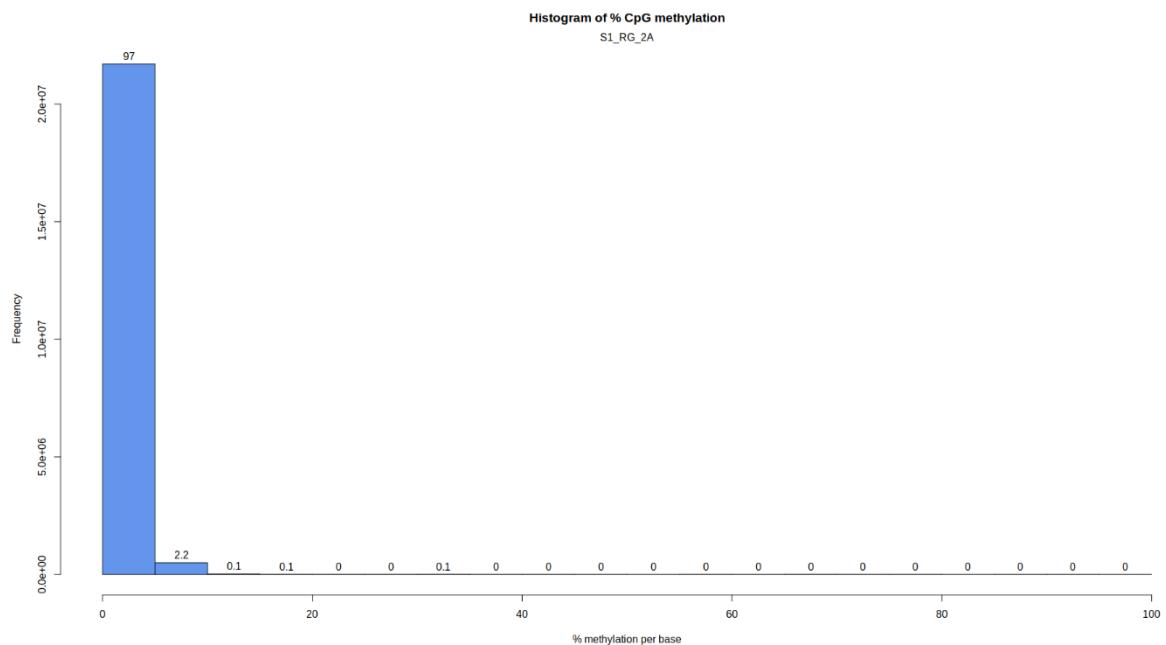
Annexe 2 B : histogramme de l'échantillon S1_CT_3B CpG



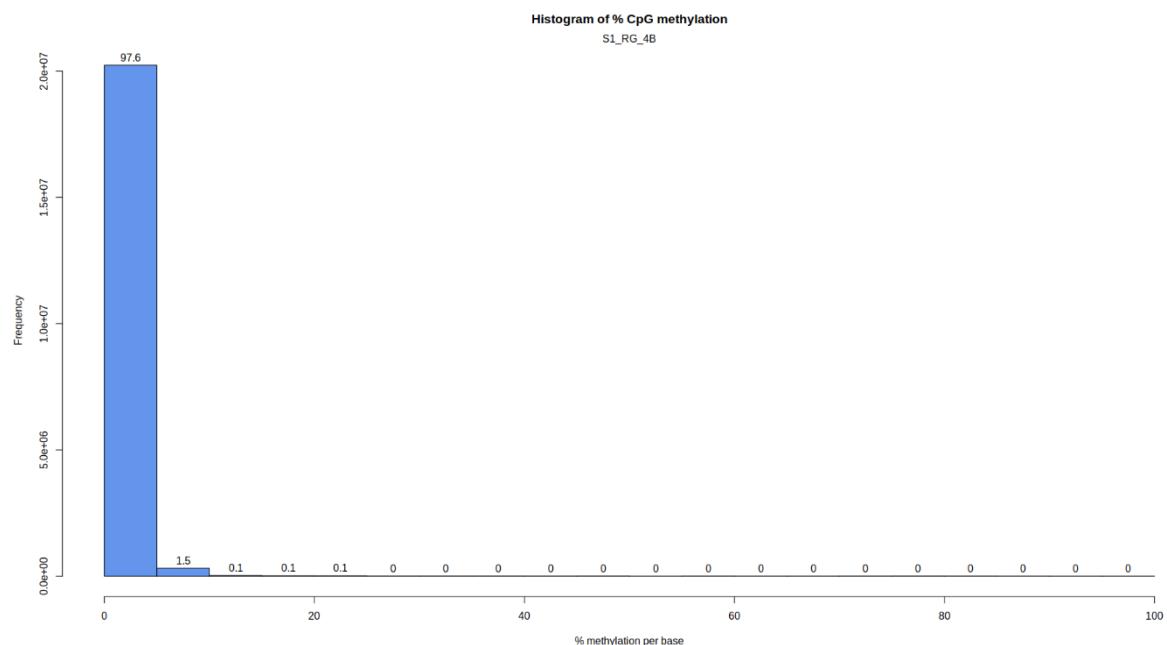
Annexe 2 C : histogramme de l'échantillon S1_CT_4B CpG



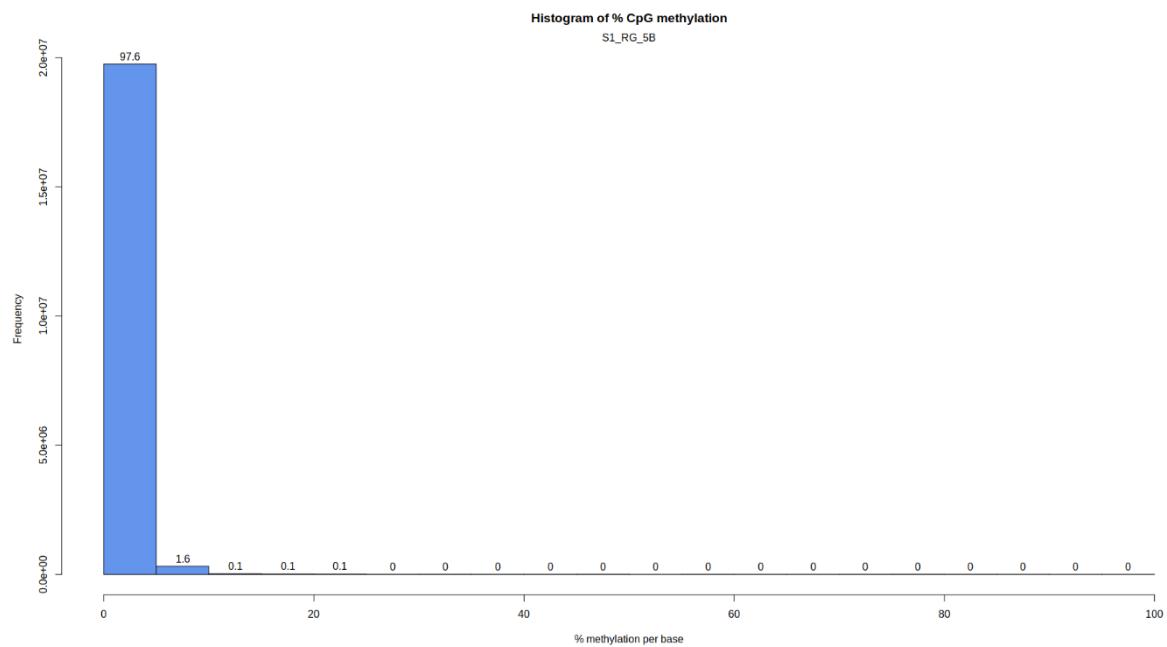
Annexe 2 D : histogramme de l'échantillon S1_RG_2A CpG



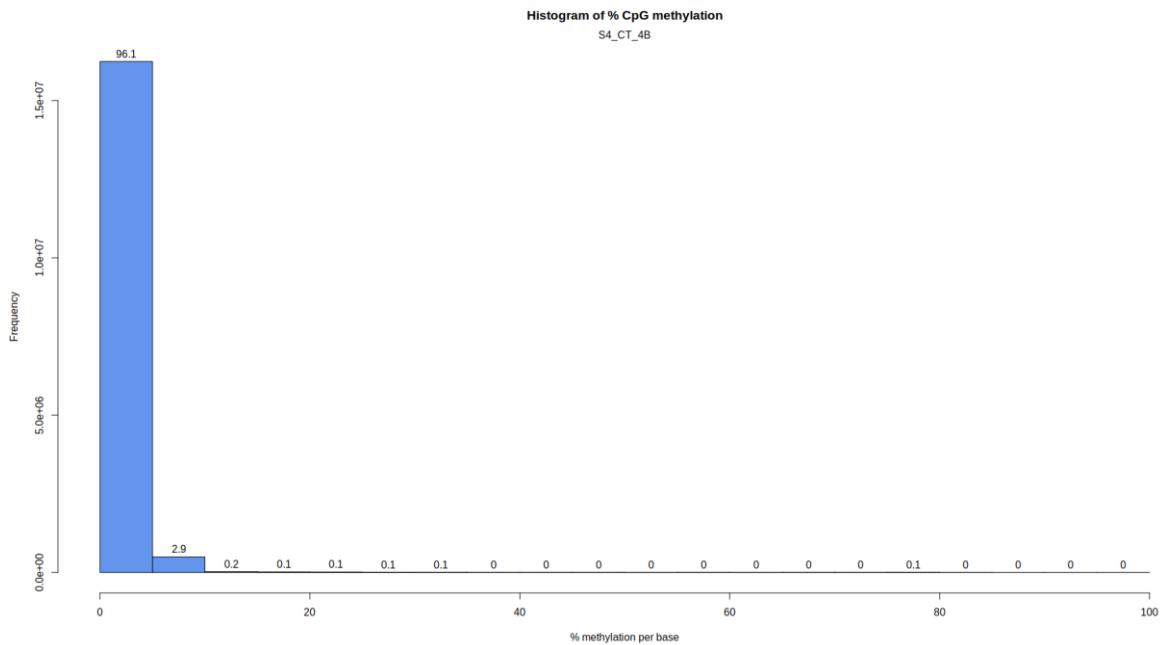
Annexe 2 E : histogramme de l'échantillon S1_RG_4B CpG



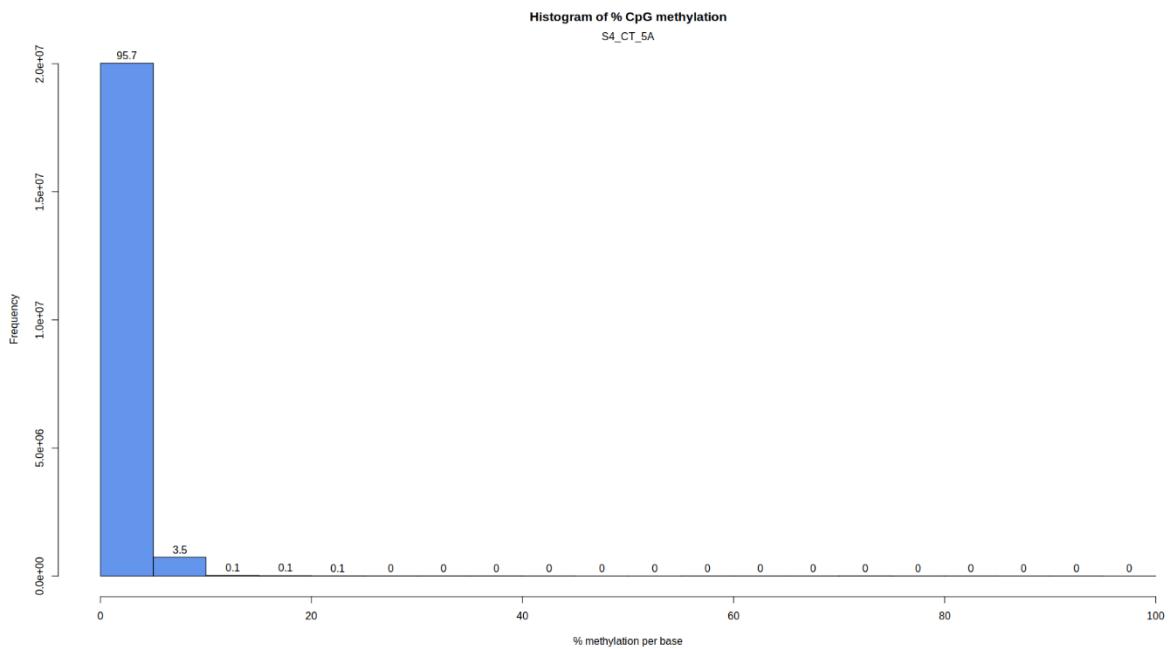
Annexe 2 F : histogramme de l'échantillon S1_RG_5B CpG



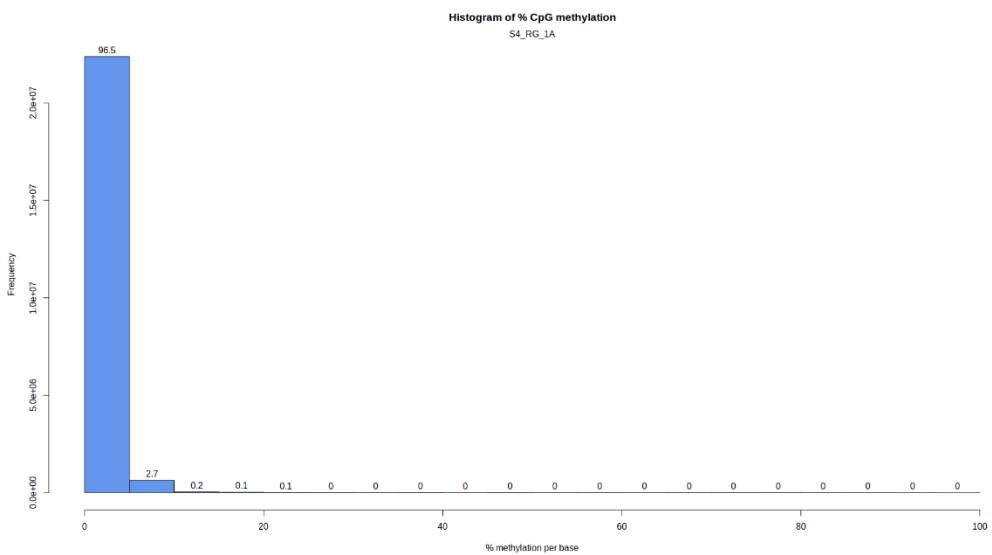
Annexe 2 G : histogramme de l'échantillon S4_CT_4B CpG



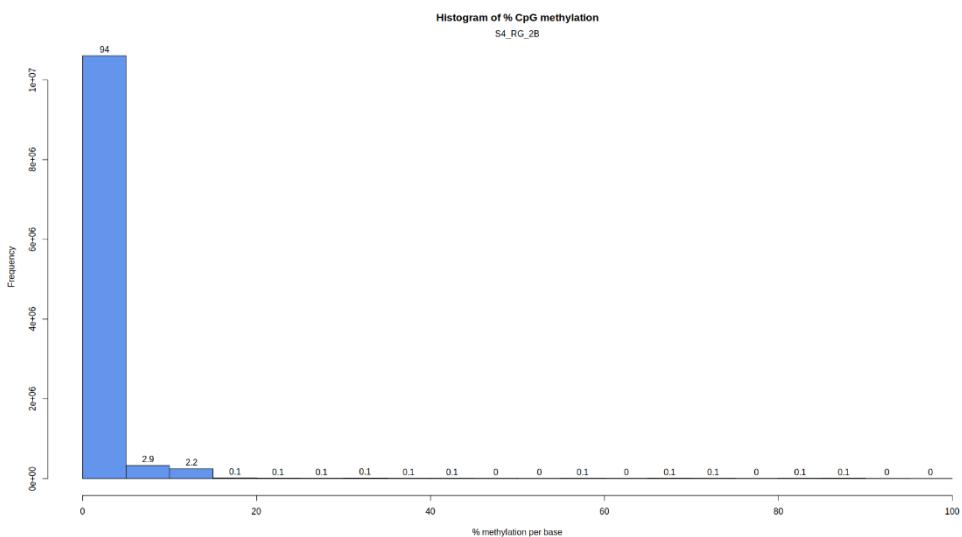
Annexe 2 H : histogramme de l'échantillon S4_CT_5A CpG



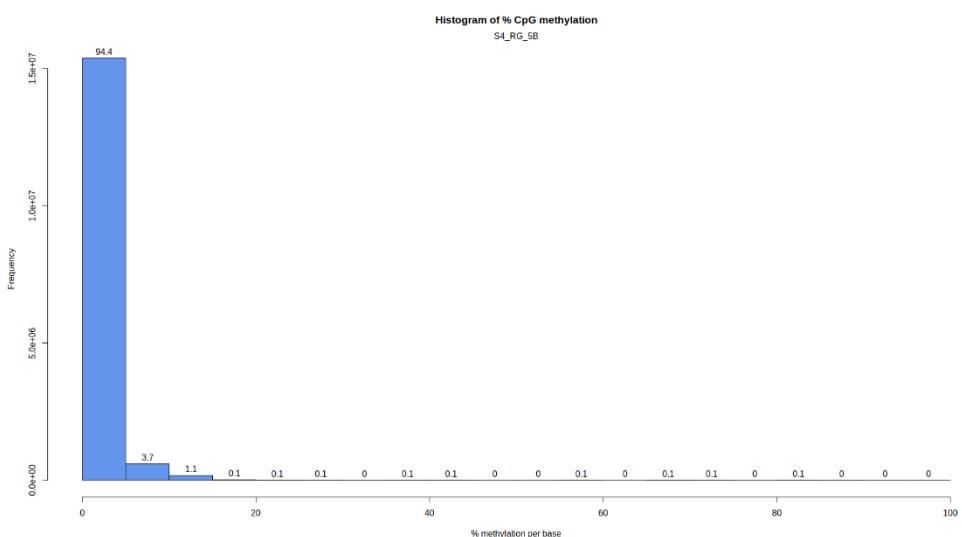
Annexe 2 i : histogramme de l'échantillon S4_RG_1A CpG



Annexe 2 J : histogramme de l'échantillon S4_RG_2B CpG



Annexe 2 K : histogramme de l'échantillon S4_RG_5B CpG



Annexe 3 A : Statistique de la couverture S1 et S4 contrôle et CpG

```
> getCoverageStats(normal_cov_CT_cpg[[1]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
  Min. 1st Qu. Median   Mean 3rd Qu.   Max.
  14.0    15.0   18.0    18.9    21.0   127.0
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%  99.5%  99.9%  100%
  14     14    15    15    17    18    19    21    22    25    28    35    37    73    127

> getCoverageStats(normal_cov_CT_cpg[[2]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
  Min. 1st Qu. Median   Mean 3rd Qu.   Max.
  13.00   15.00   18.00   18.98   22.00  123.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%  99.5%  99.9%  100%
  13     13    14    15    17    18    19    21    23    26    30    36    39    69    123

> getCoverageStats(normal_cov_CT_cpg[[3]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
  Min. 1st Qu. Median   Mean 3rd Qu.   Max.
  13.00   15.00   18.00   19.28   22.00   94.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%  99.5%  99.9%  100%
  13     13    14    15    17    18    19    22    23    27    30    36    39    53    94

> getCoverageStats(normal_cov_CT_cpg[[4]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
  Min. 1st Qu. Median   Mean 3rd Qu.   Max.
  10.00   14.00   18.00   18.55   22.00   80.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%  99.5%  99.9%  100%
  10     12    13    15    16    18    19    21    23    26    29    35    38    47    80

> getCoverageStats(normal_cov_CT_cpg[[5]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
  Min. 1st Qu. Median   Mean 3rd Qu.   Max.
  13.0    15.0    18.0    19.6    23.0   100.0
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%  99.5%  99.9%  100%
  13     13    14    15    17    18    19    22    24    27    30    37    40    55    100

> getCoverageStats(normal_cov_CT_cpg[[6]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
  Min. 1st Qu. Median   Mean 3rd Qu.   Max.
  11.00   14.00   18.00   18.45   21.00  106.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%  99.5%  99.9%  100%
  11     12    14    15    16    18    19    21    22    25    29    35    37    56    106
```

Annexe 3 B : Statistique de la couverture S1 et S4 traités et CpG

methylation statistics per base	
summary:	
Min. 1st Qu. Median Mean 3rd Qu. Max.	
0.0000 0.0000 0.0000 0.681 0.0000 100.000	
percentiles:	
0%	10%
0.000000	0.000000
20%	30%
0.000000	0.000000
40%	50%
0.000000	0.000000
60%	70%
0.000000	0.000000
80%	90%
0.000000	0.000000
90%	95%
0.000000	4.166667
95%	99%
0.000000	8.333333
99%	99.5%
0.000000	28.571429
99.5%	99.9%
0.000000	81.250000
99.9%	100%
methylation statistics per base	
summary:	
Min. 1st Qu. Median Mean 3rd Qu. Max.	
0.0000 0.0000 0.0000 0.6306 0.0000 100.0000	
percentiles:	
0%	10%
0.000000	0.000000
20%	30%
0.000000	0.000000
40%	50%
0.000000	0.000000
60%	70%
0.000000	0.000000
80%	90%
0.000000	0.000000
90%	95%
0.000000	3.448276
95%	99%
0.000000	7.692308
99%	99.5%
0.000000	31.034483
99.5%	99.9%
0.000000	82.142857
99.9%	100%
methylation statistics per base	
summary:	
Min. 1st Qu. Median Mean 3rd Qu. Max.	
0.0000 0.0000 0.0000 0.6148 0.0000 100.0000	
percentiles:	
0%	10%
0.000000	0.000000
20%	30%
0.000000	0.000000
40%	50%
0.000000	0.000000
60%	70%
0.000000	0.000000
80%	90%
0.000000	0.000000
90%	95%
0.000000	3.448276
95%	99%
0.000000	7.692308
99%	99.5%
0.000000	29.166667
99.5%	99.9%
0.000000	81.250000
99.9%	100%
methylation statistics per base	
summary:	
Min. 1st Qu. Median Mean 3rd Qu. Max.	
0.0000 0.0000 0.0000 0.7375 0.0000 100.0000	
percentiles:	
0%	10%
0.000000	0.000000
20%	30%
0.000000	0.000000
40%	50%
0.000000	0.000000
60%	70%
0.000000	0.000000
80%	90%
0.000000	0.000000
90%	95%
0.000000	4.347826
95%	99%
0.000000	9.090909
99%	99.5%
0.000000	26.666667
99.5%	99.9%
0.000000	80.000000
99.9%	100%
methylation statistics per base	
summary:	
Min. 1st Qu. Median Mean 3rd Qu. Max.	
0.0000 0.0000 0.0000 0.9629 0.0000 100.0000	
percentiles:	
0%	10%
0.000000	0.000000
20%	30%
0.000000	0.000000
40%	50%
0.000000	0.000000
60%	70%
0.000000	0.000000
80%	90%
0.000000	0.000000
90%	95%
0.000000	7.692308
95%	99%
0.000000	12.500000
99%	99.5%
0.000000	45.833333
99.5%	99.9%
0.000000	86.956522
99.9%	100%
methylation statistics per base	
summary:	
Min. 1st Qu. Median Mean 3rd Qu. Max.	
0.0000 0.0000 0.0000 0.9829 0.0000 100.0000	
percentiles:	
0%	10%
0.000000	0.000000
20%	30%
0.000000	0.000000
40%	50%
0.000000	0.000000
60%	70%
0.000000	0.000000
80%	90%
0.000000	0.000000
90%	95%
0.000000	6.250000
95%	99%
0.000000	13.333333
99%	99.5%
0.000000	42.857142
99.5%	99.9%
0.000000	85.714290
99.9%	100%

Annexe 4 A : Statistiques couverture des contrôles CpG semaine 1 et 4

```
> getCoverageStats(normal_cov_CT_cpg[[1]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
    Min. 1st Qu. Median      Mean 3rd Qu.      Max.
    14.0    15.0   18.0     18.9    21.0    127.0
percentiles:
    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%   99.5%   99.9%   100%
    14     14    15    15    17    18    19    21    22    25    28    35    37    73    127

> getCoverageStats(normal_cov_CT_cpg[[2]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
    Min. 1st Qu. Median      Mean 3rd Qu.      Max.
    13.00   15.00   18.00    18.98   22.00   123.00
percentiles:
    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%   99.5%   99.9%   100%
    13     13    14    15    17    18    19    21    23    26    30    36    39    69    123

> getCoverageStats(normal_cov_CT_cpg[[3]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
    Min. 1st Qu. Median      Mean 3rd Qu.      Max.
    13.00   15.00   18.00    19.28   22.00   94.00
percentiles:
    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%   99.5%   99.9%   100%
    13     13    14    15    17    18    19    22    23    27    30    36    39    53    94

> getCoverageStats(normal_cov_CT_cpg[[4]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
    Min. 1st Qu. Median      Mean 3rd Qu.      Max.
    10.00   14.00   18.00    18.55   22.00   80.00
percentiles:
    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%   99.5%   99.9%   100%
    10     12    13    15    16    18    19    21    23    26    29    35    38    47    80

> getCoverageStats(normal_cov_CT_cpg[[5]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
    Min. 1st Qu. Median      Mean 3rd Qu.      Max.
    13.0    15.0   18.0     19.6    23.0   100.0
percentiles:
    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%   99.5%   99.9%   100%
    13     13    14    15    17    18    19    22    24    27    30    37    40    55    100

> getCoverageStats(normal_cov_CT_cpg[[6]], plot = FALSE, both.strands = FALSE)
read coverage statistics per base
summary:
    Min. 1st Qu. Median      Mean 3rd Qu.      Max.
    11.00   14.00   18.00    18.45   21.00  106.00
percentiles:
    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99%   99.5%   99.9%   100%
    11     12    14    15    16    18    19    21    22    25    29    35    37    56    106
```

Annexe 4 B : Statistiques couverture des traités CpG semaine 1 et 4

```
read coverage statistics per base
summary:
  Min. 1st Qu. Median      Mean 3rd Qu.      Max.
  12.00   18.00  21.00    22.13   26.00  110.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99% 99.5% 99.9% 100%
  12    14    16    18    20    21    23    24    28    32    34    41    44    61    110

read coverage statistics per base
summary:
  Min. 1st Qu. Median      Mean 3rd Qu.      Max.
  13.00   17.00  21.00    22.52   26.00  122.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99% 99.5% 99.9% 100%
  13    14    16    18    20    21    24    25    28    32    35    43    46    67    122

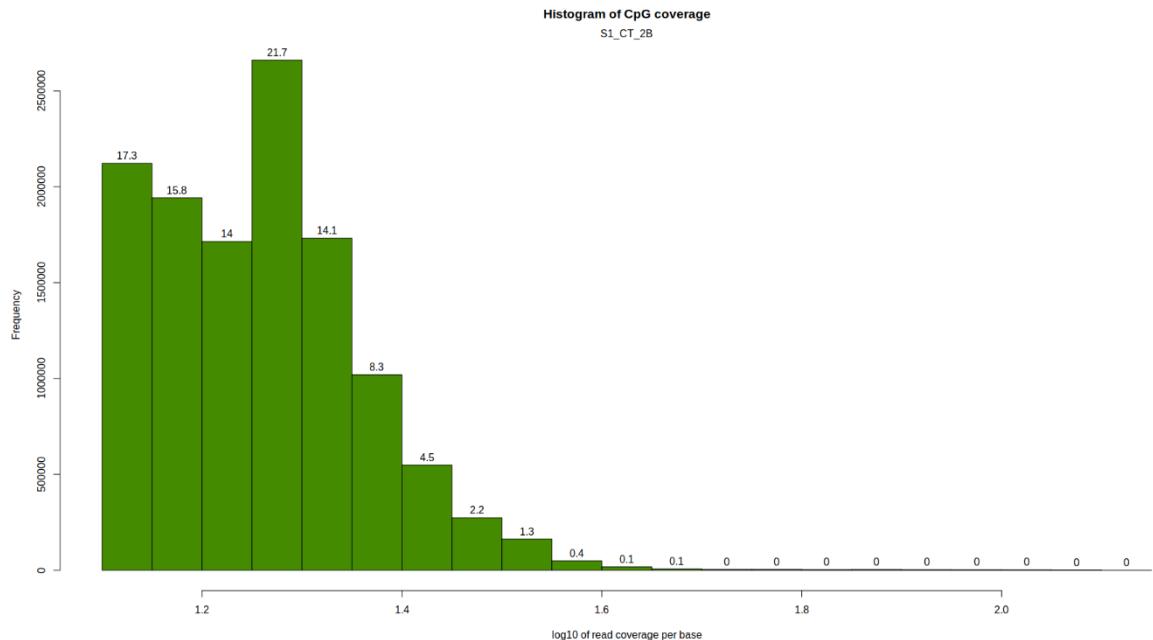
read coverage statistics per base
summary:
  Min. 1st Qu. Median      Mean 3rd Qu.      Max.
  13.00   17.00  21.00    22.39   26.00  121.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99% 99.5% 99.9% 100%
  13    14    16    18    20    21    24    25    28    32    35    43    46    67    121

read coverage statistics per base
summary:
  Min. 1st Qu. Median      Mean 3rd Qu.      Max.
  10.00   16.00  21.00    21.62   26.00  108.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99% 99.5% 99.9% 100%
  10    13    15    17    19    21    23    25    27    31    34    40    43    57    108

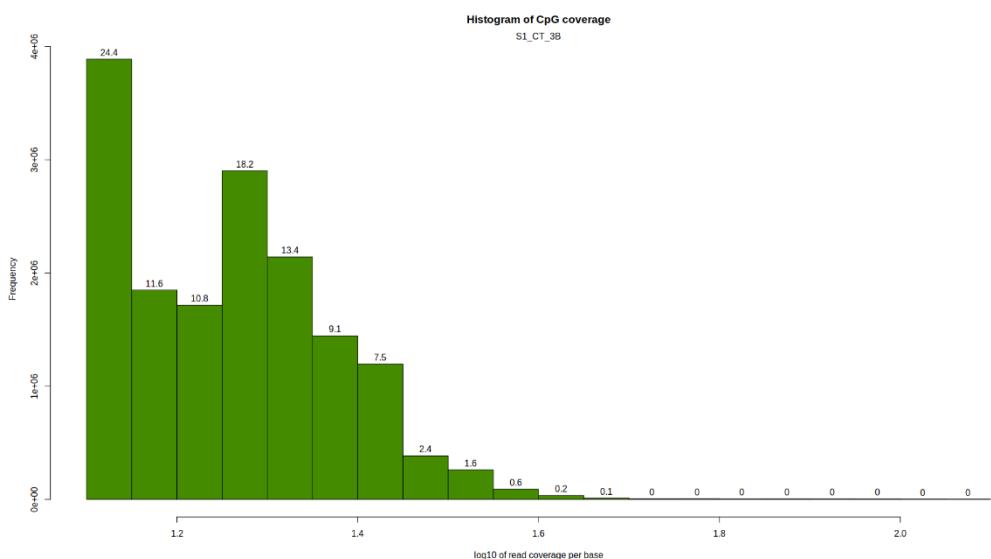
read coverage statistics per base
summary:
  Min. 1st Qu. Median      Mean 3rd Qu.      Max.
  16.00   18.00  21.00    21.71   24.00  165.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99% 99.5% 99.9% 100%
  16    16    18    18    19    21    21    23    26    29    32    39    44    94    165

read coverage statistics per base
summary:
  Min. 1st Qu. Median      Mean 3rd Qu.      Max.
  15.00   18.00  21.00    23.02   27.00  134.00
percentiles:
  0%   10%   20%   30%   40%   50%   60%   70%   80%   90%   95%   99% 99.5% 99.9% 100%
  15    15    16    18    20    21    24    26    28    32    36    44    48    72    134
```

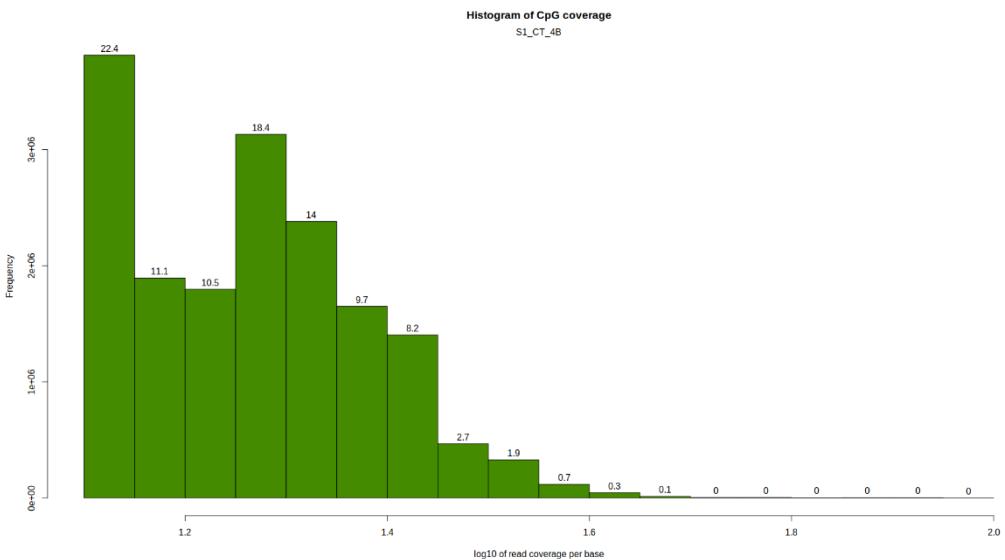
Annexe 5 A : Histogramme de couverture CpG S1_CT_2B



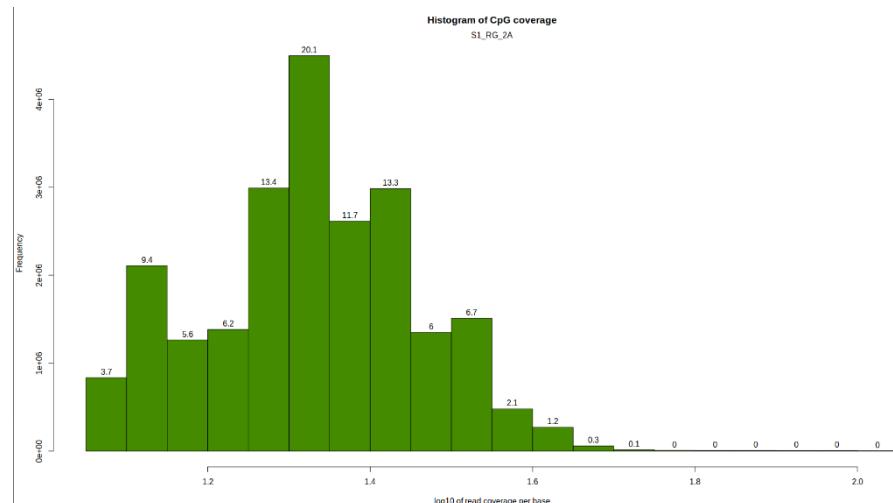
Annexe 5 B : Histogramme de couverture CpG S1_CT_3B



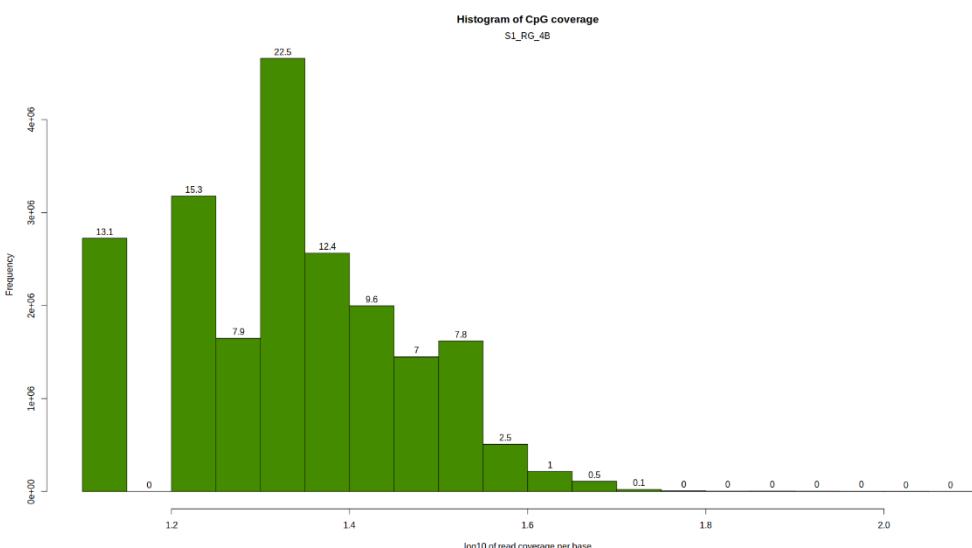
Annexe 5 C : Histogramme de couverture CpG S1_CT_4B



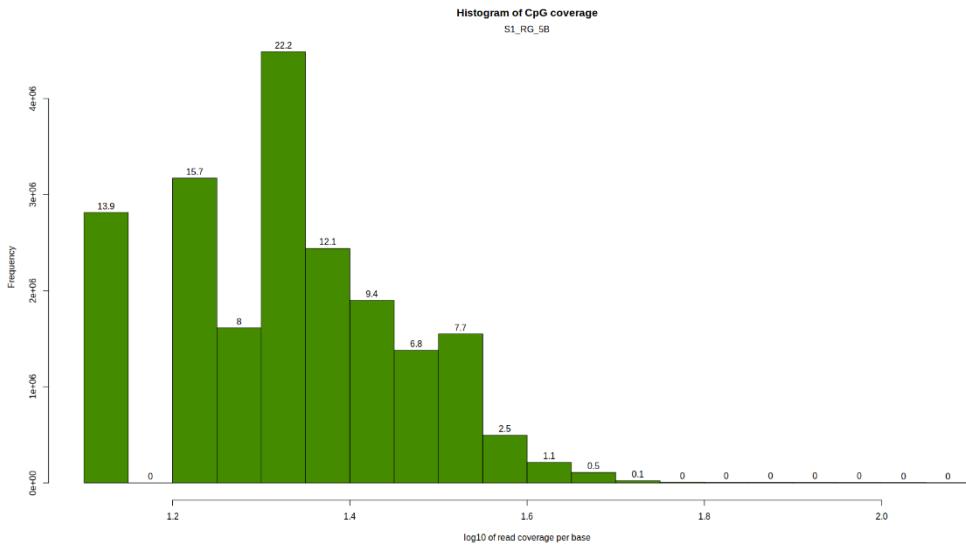
Annexe 5 D : Histogramme de couverture CpG S1_RG_2A



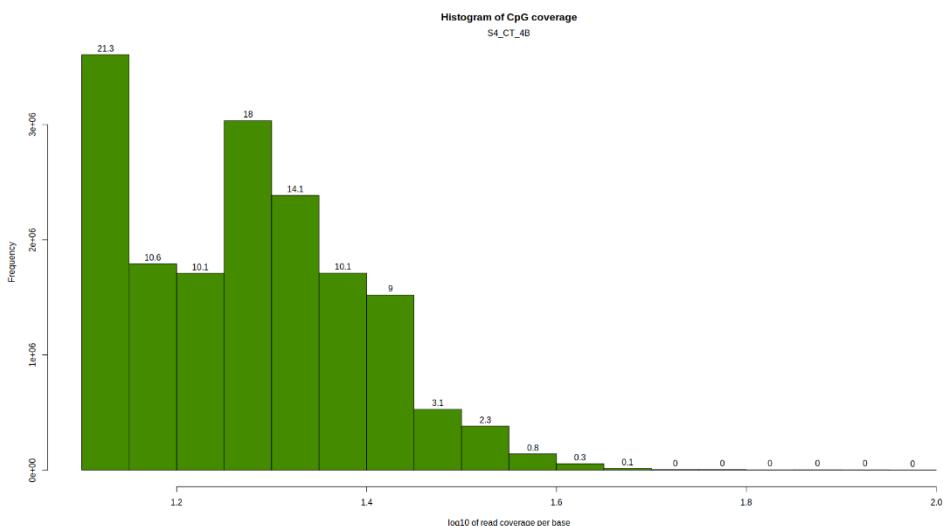
Annexe 5 E : Histogramme de couverture CpG S1_RG_4B



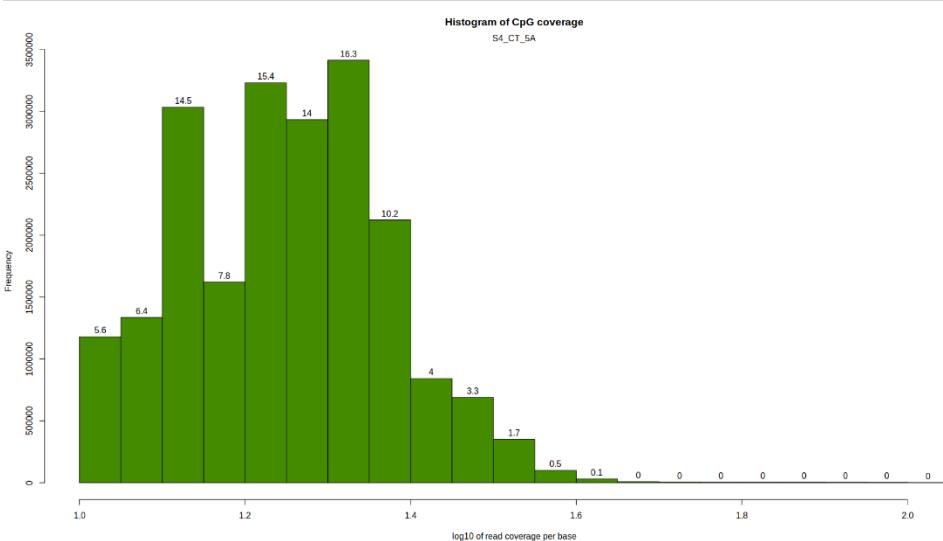
Annexe 5 F : Histogramme de couverture CpG S1_RG_5B



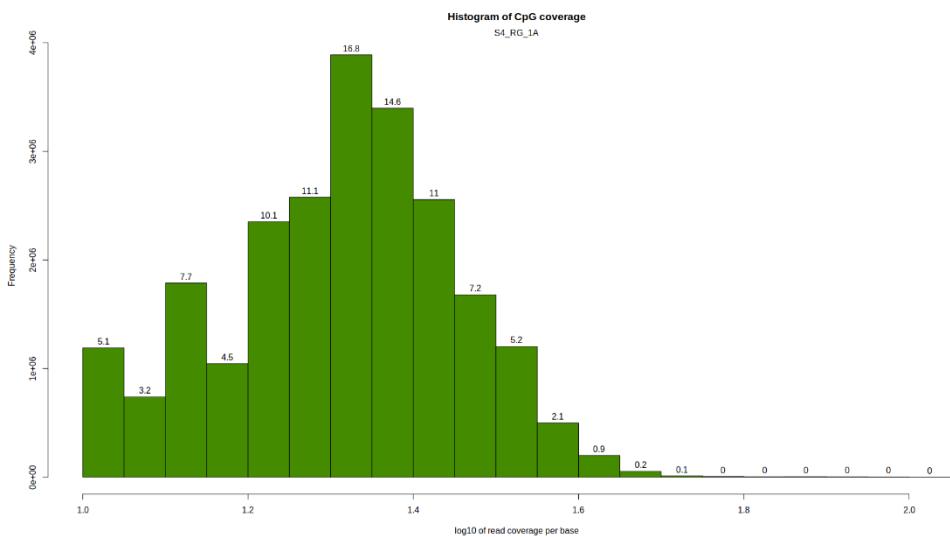
Annexe 5 H : Histogramme de couverture CpG S4_CT_4B



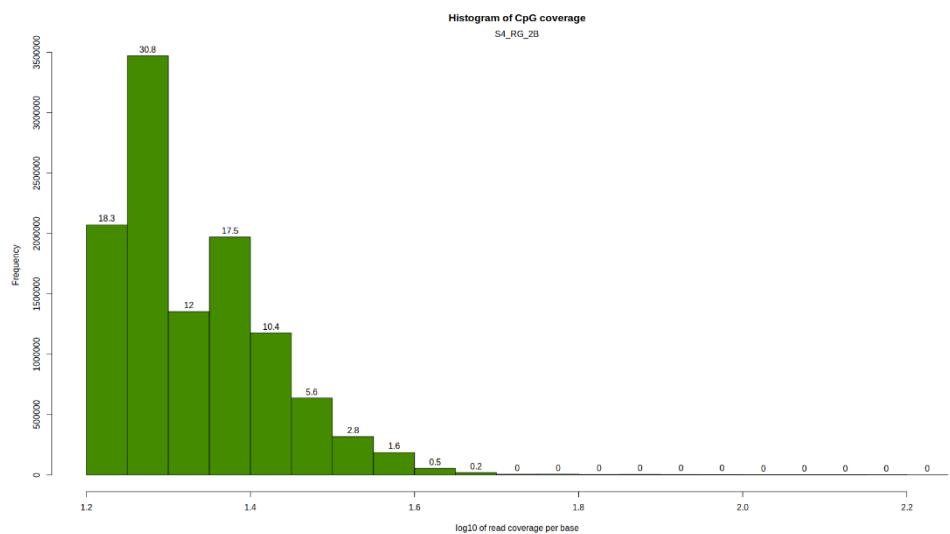
Annexe 5 I : Histogramme de couverture CpG S4_CT_5A



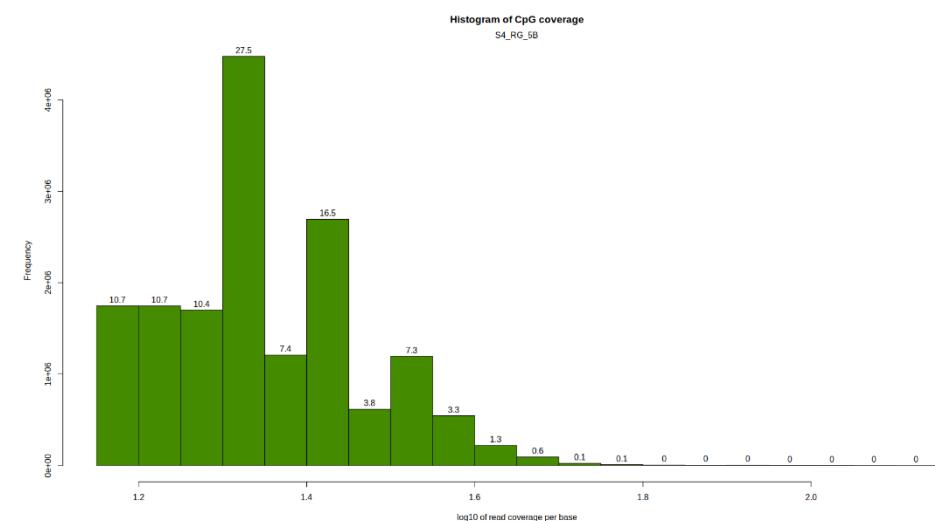
Annexe 5 J : Histogramme de couverture CpG S4_RG_1A



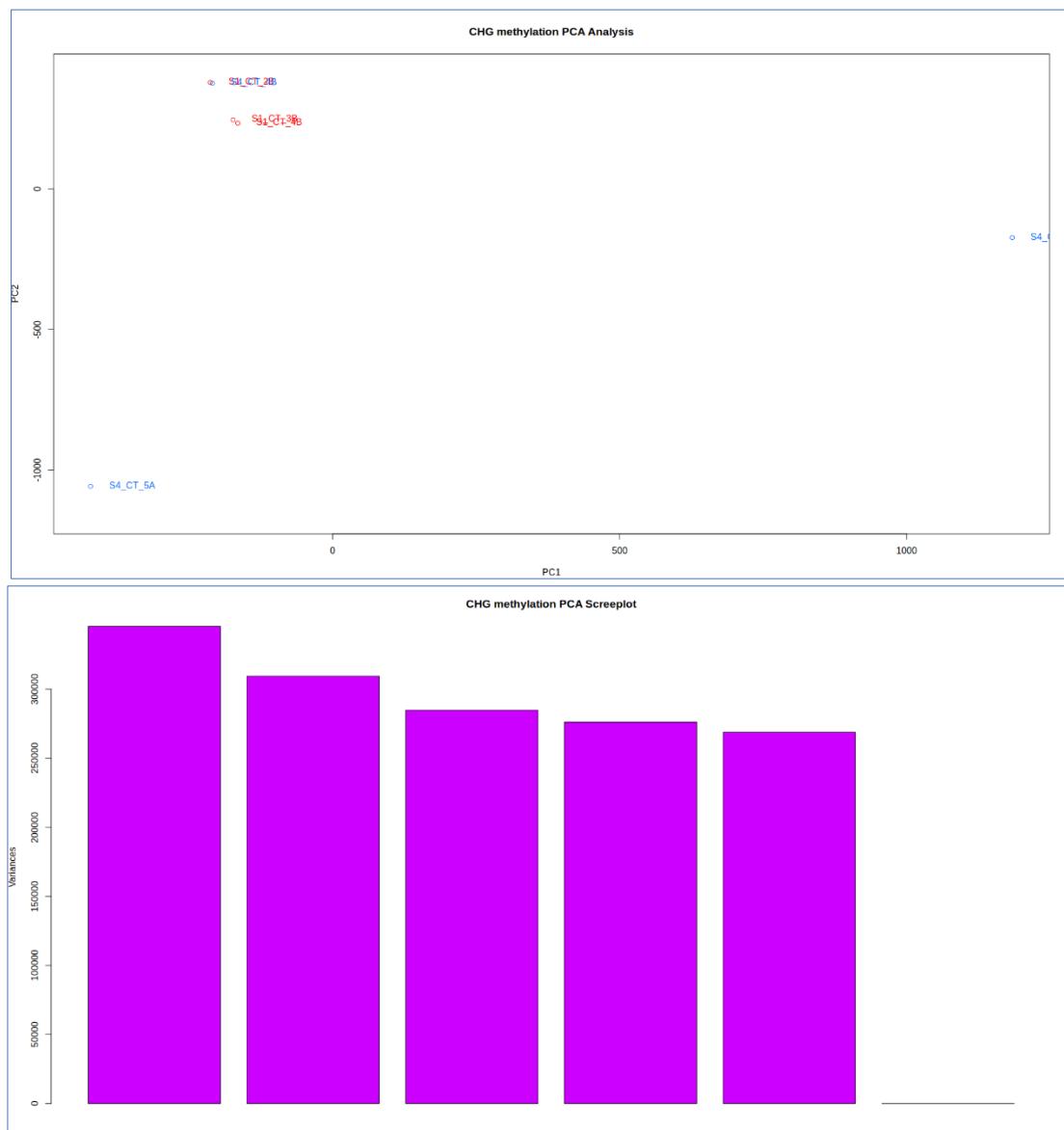
Annexe 5 K : Histogramme de couverture CpG S4_RG_2B



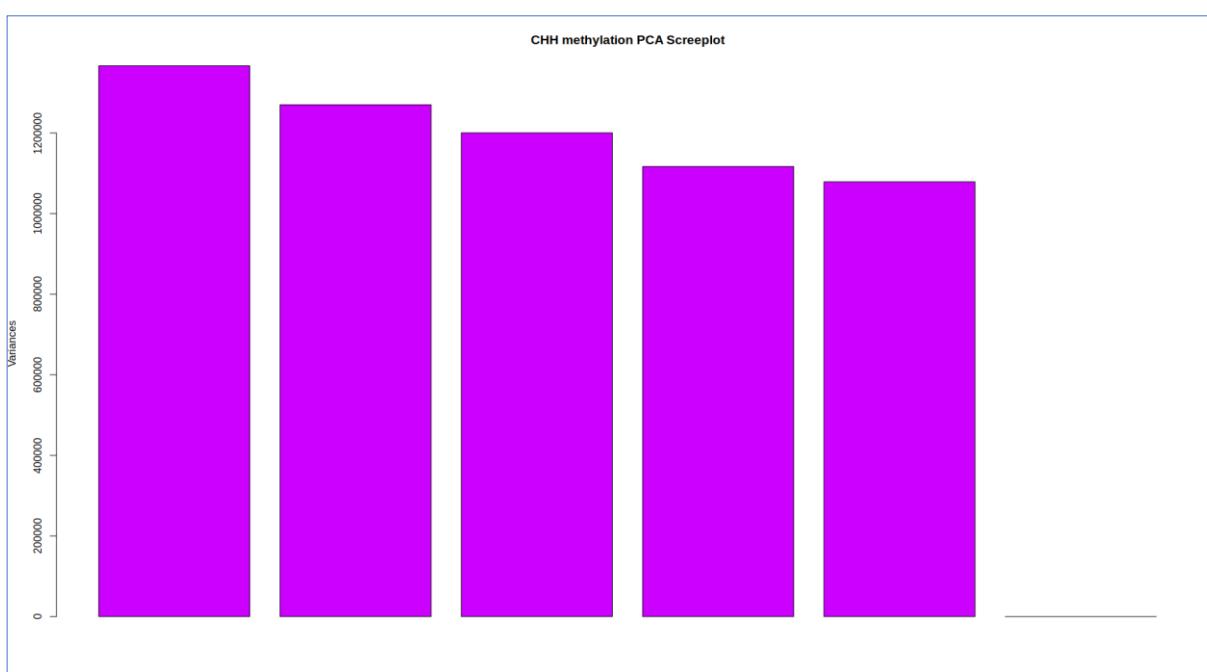
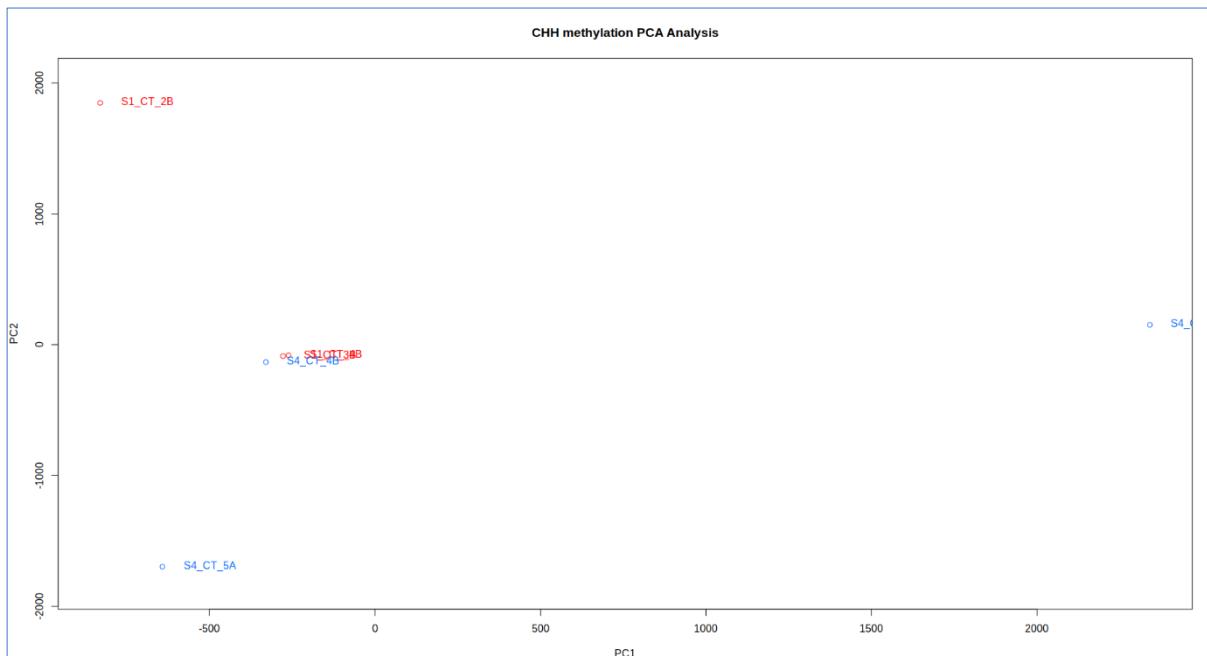
Annexe 5 L : Histogramme de couverture CpG S4_RG_5B



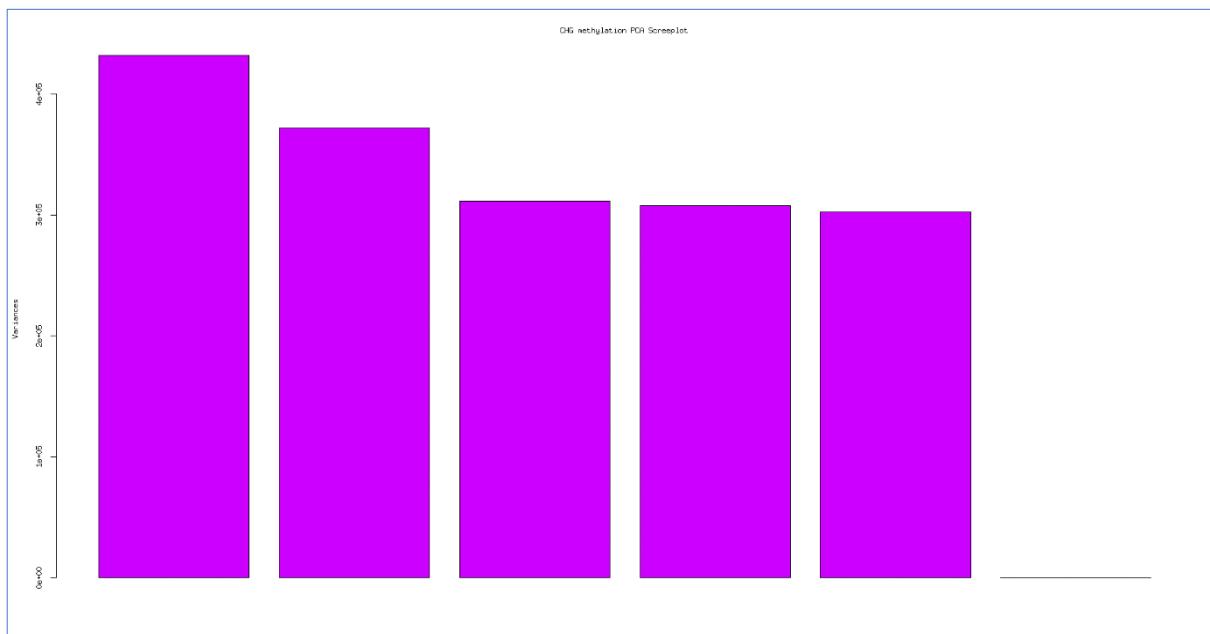
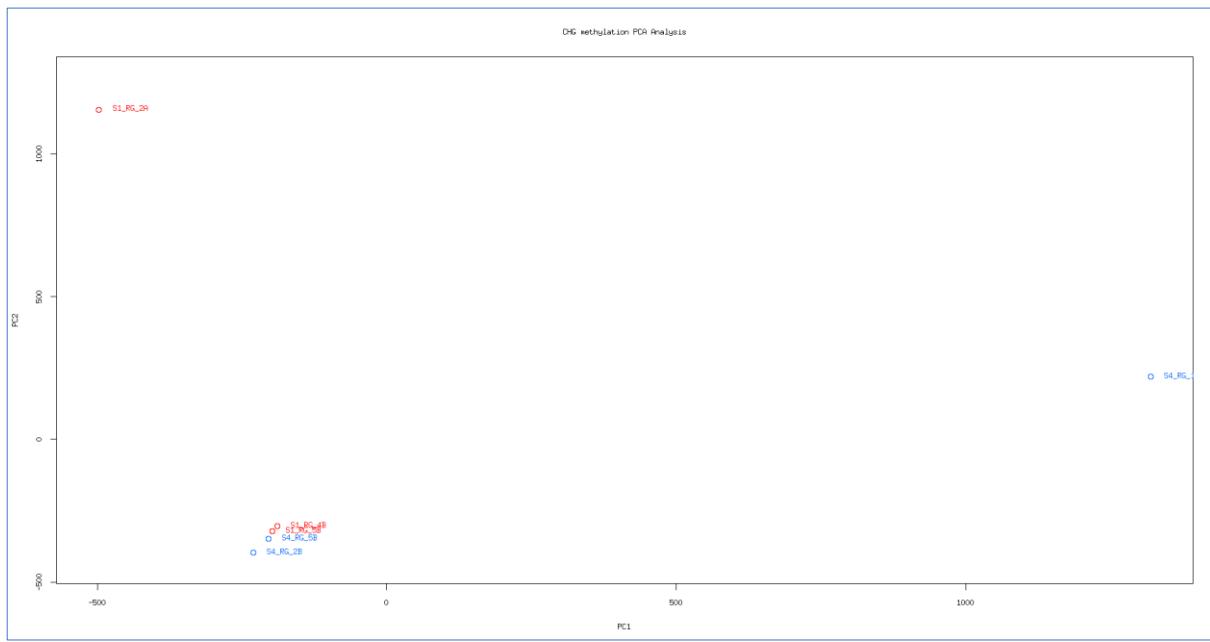
Annexe 6 A : PCA contexte CHG échantillon contrôle et leur CP



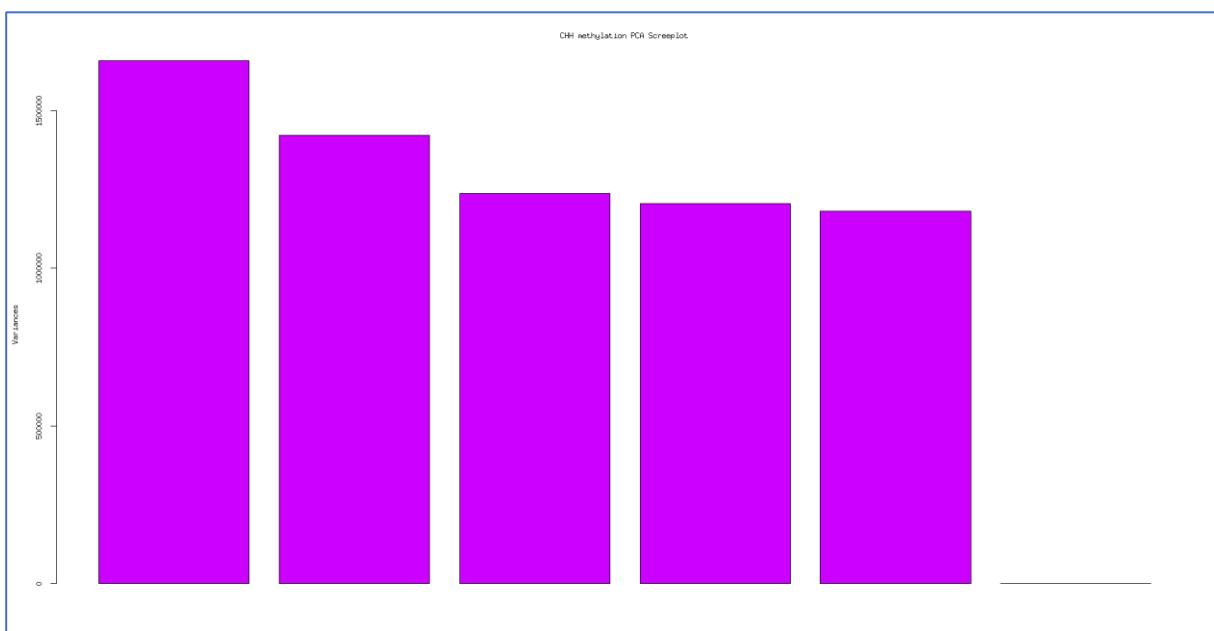
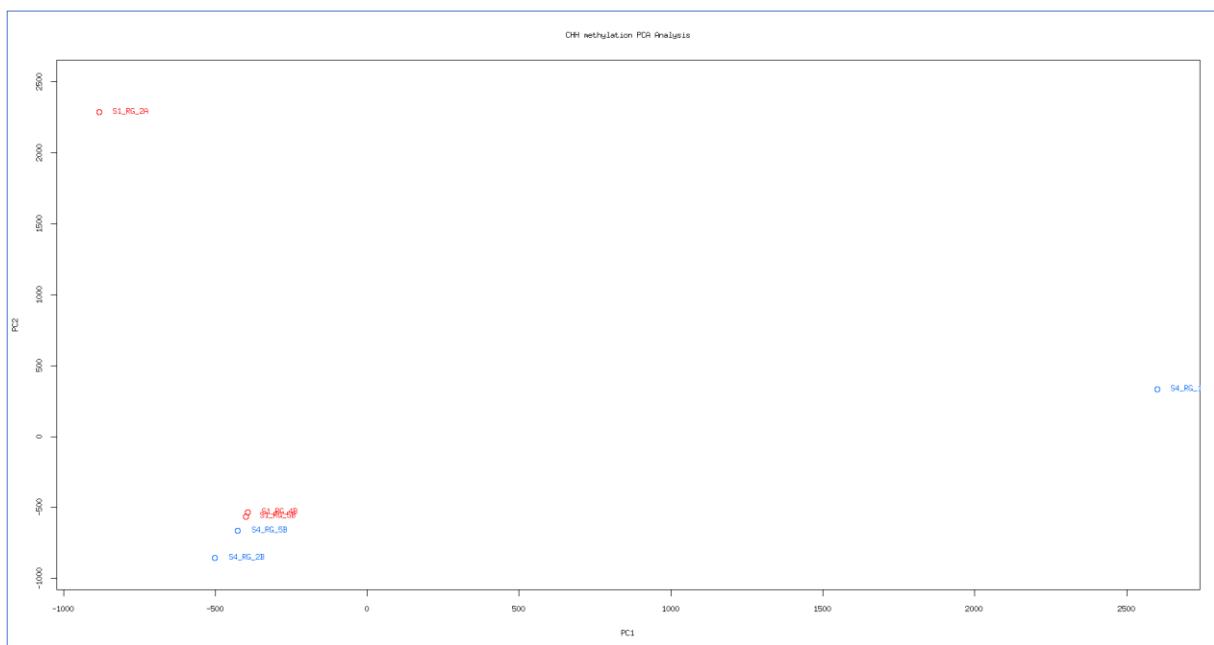
Annexe 6 B : PCA contexte CHH échantillon contrôle et leur CP



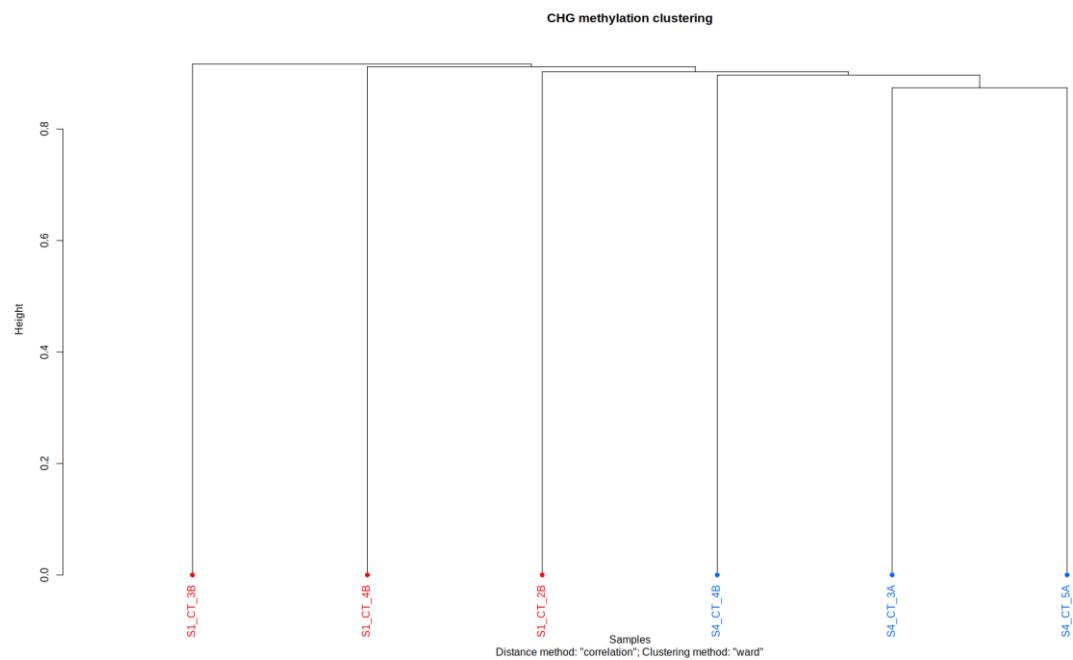
Annexe 6 C : PCA contexte CHG échantillons traités et leur CP



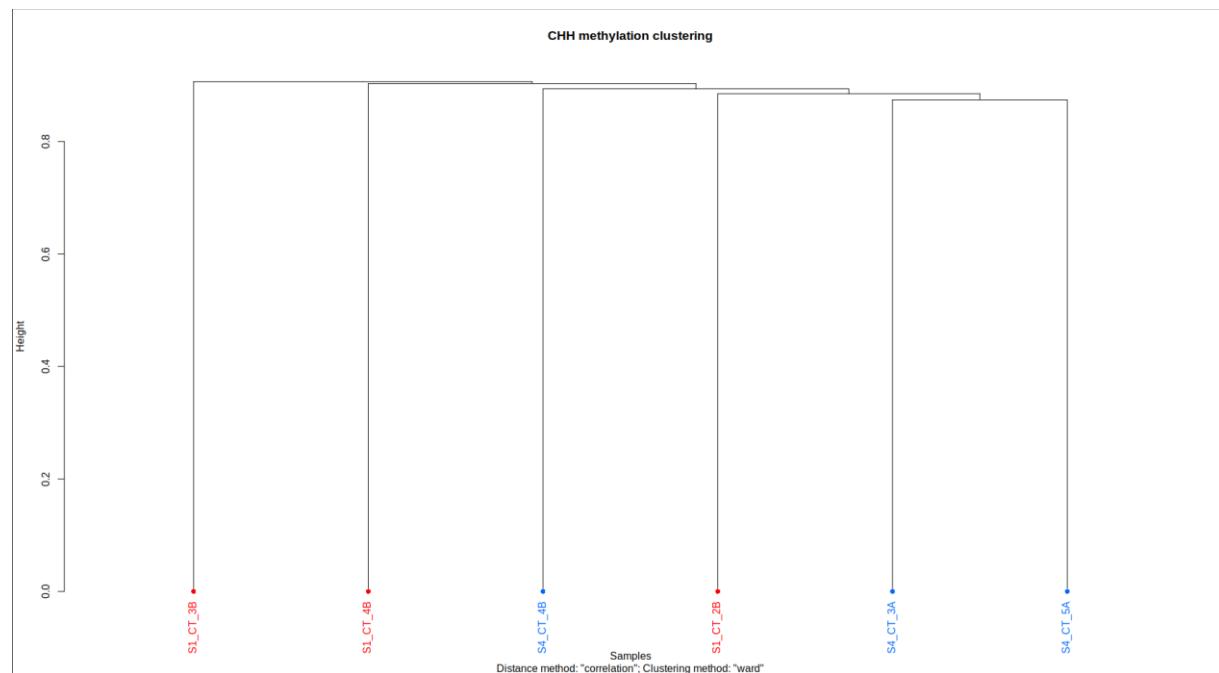
Annexe 6 D : PCA contexte CHH échantillons traités et leur CP



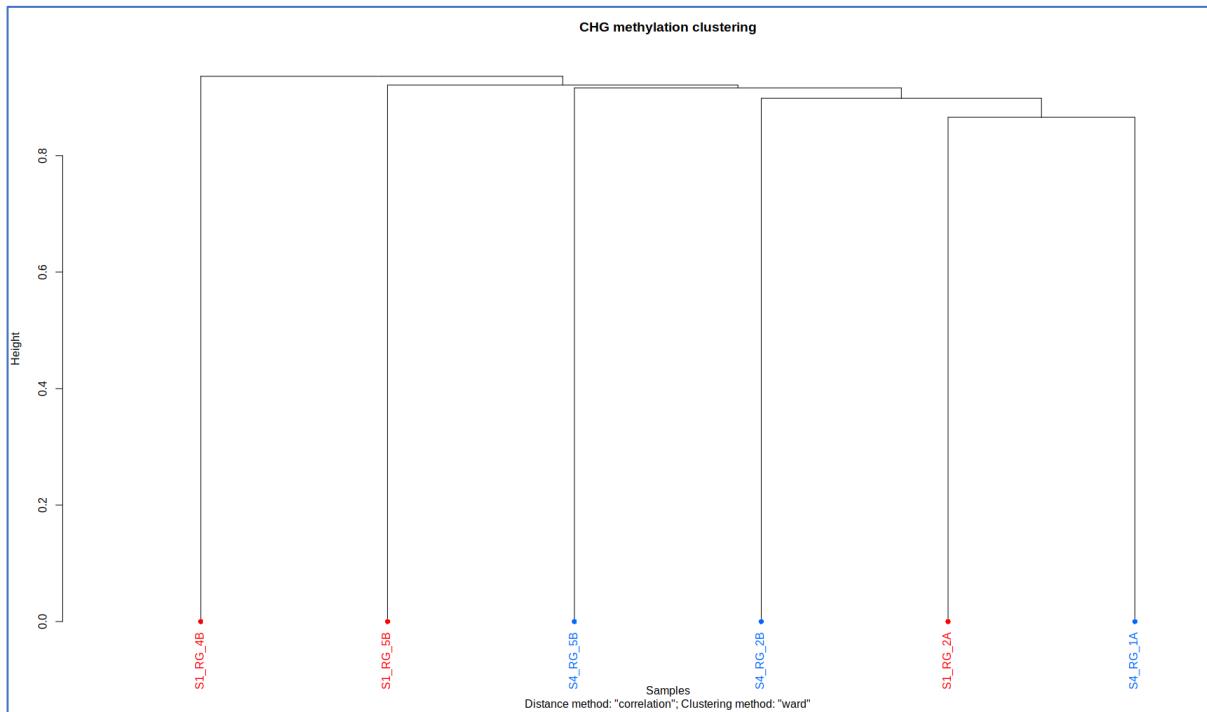
Annexe 7 A : échantillon de contrôle groupé contexte CHG



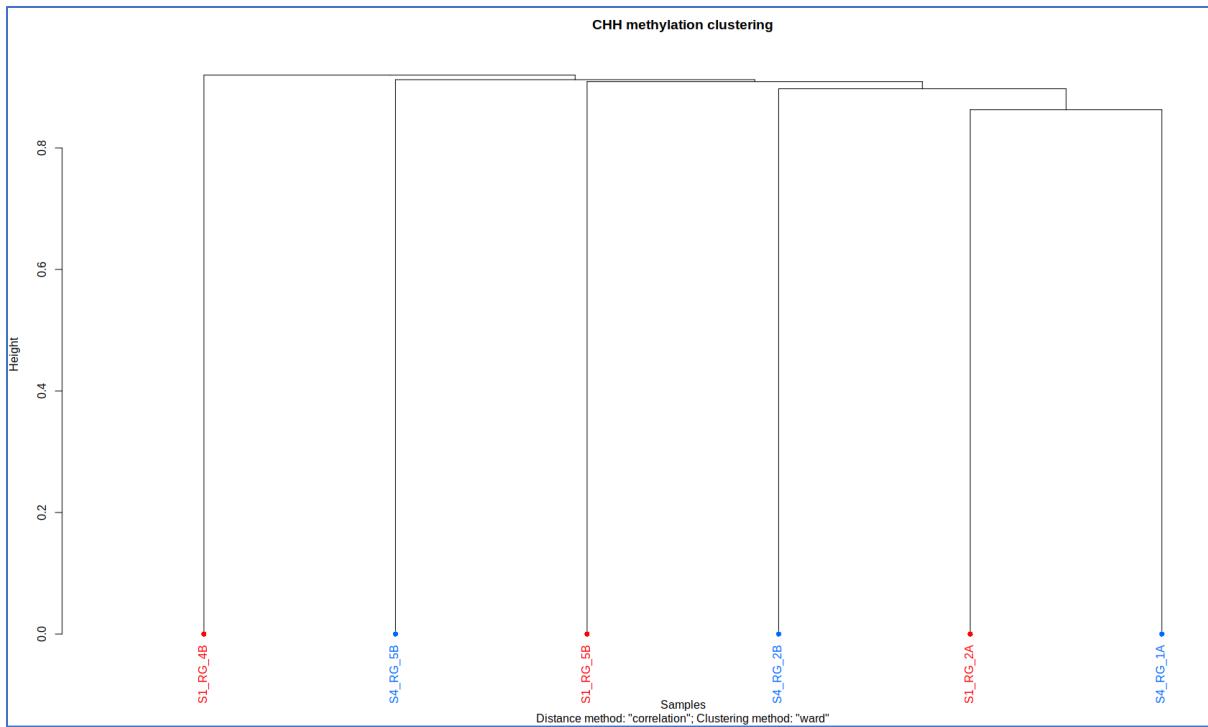
Annexe 7 B : échantillon de contrôle groupé contexte CHH



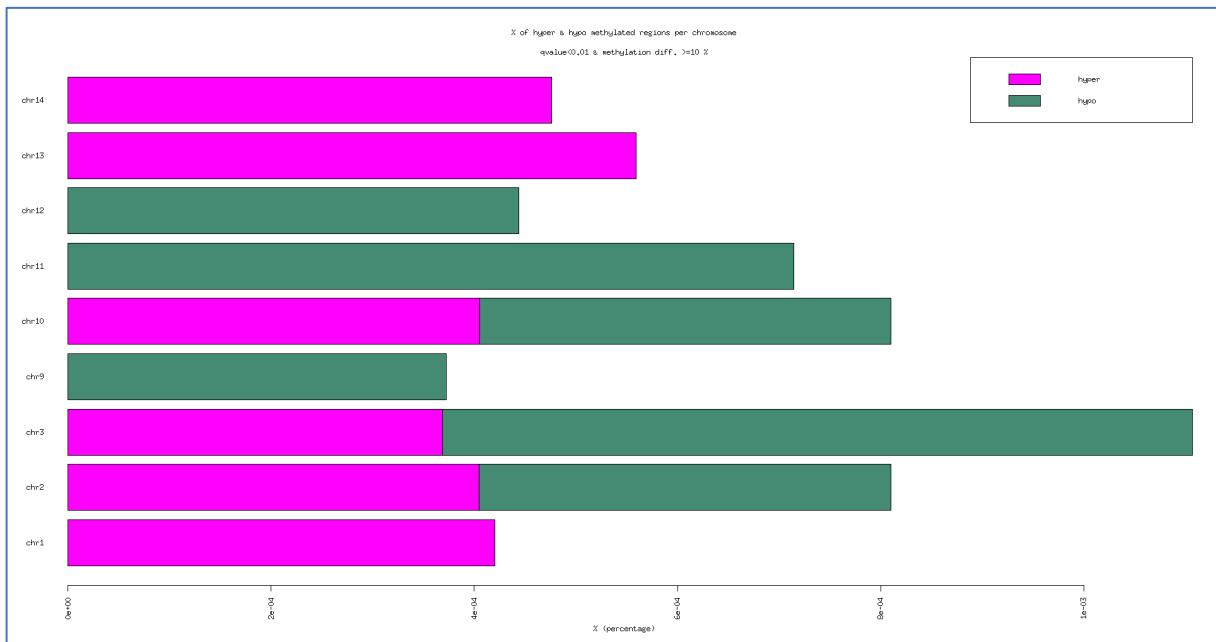
Annexe 7 C : échantillon traité groupé contexte CHG



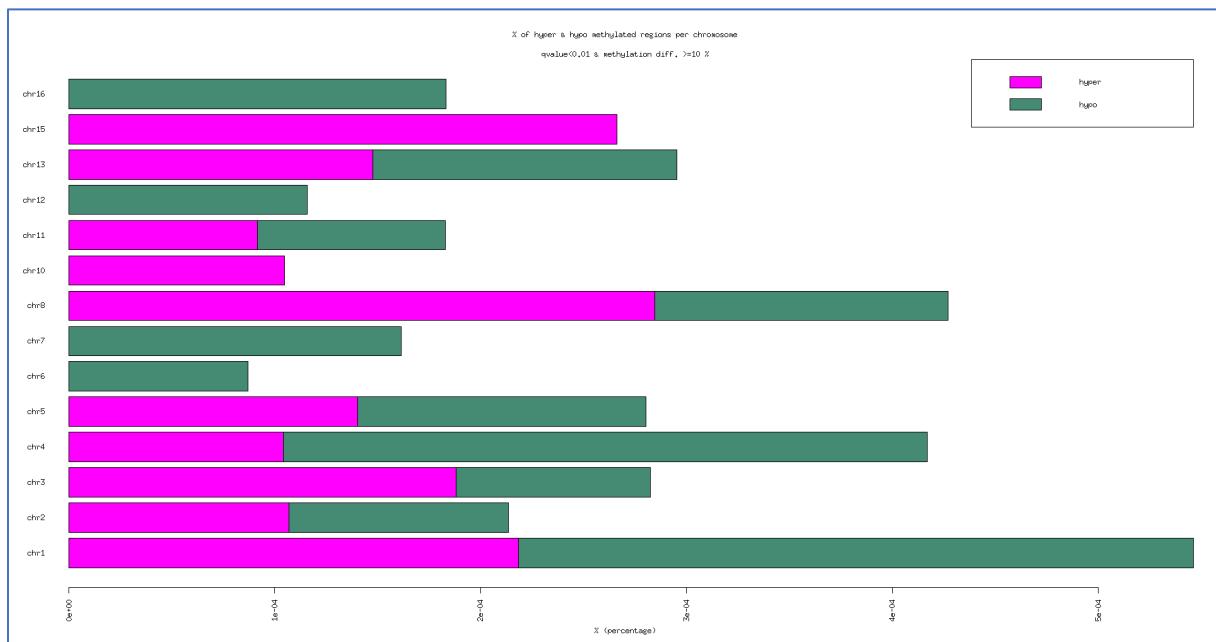
Annexe 7 D : échantillon traité groupé contexte CHH



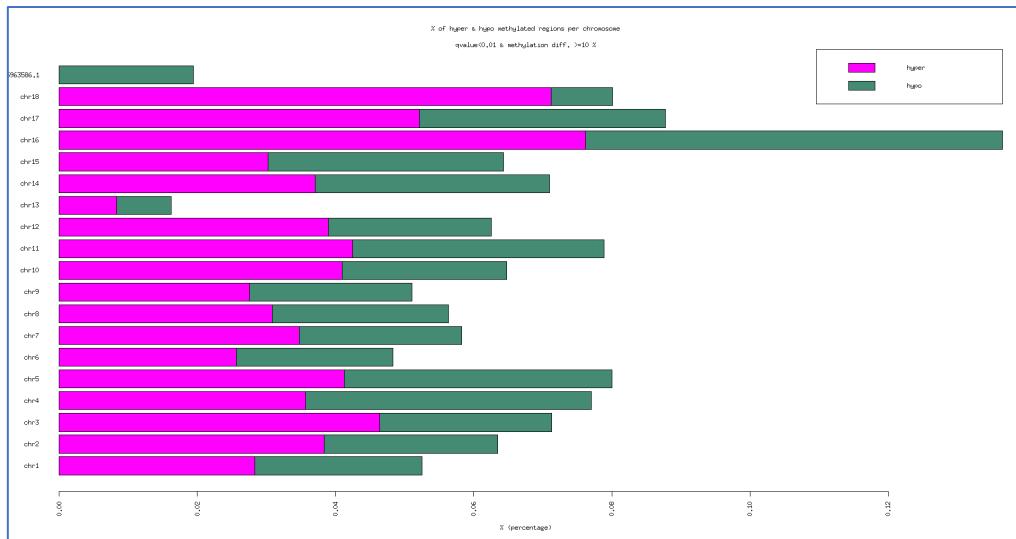
Annexe 8 A : Répartition des DMS échantillons contrôles contexte CHG



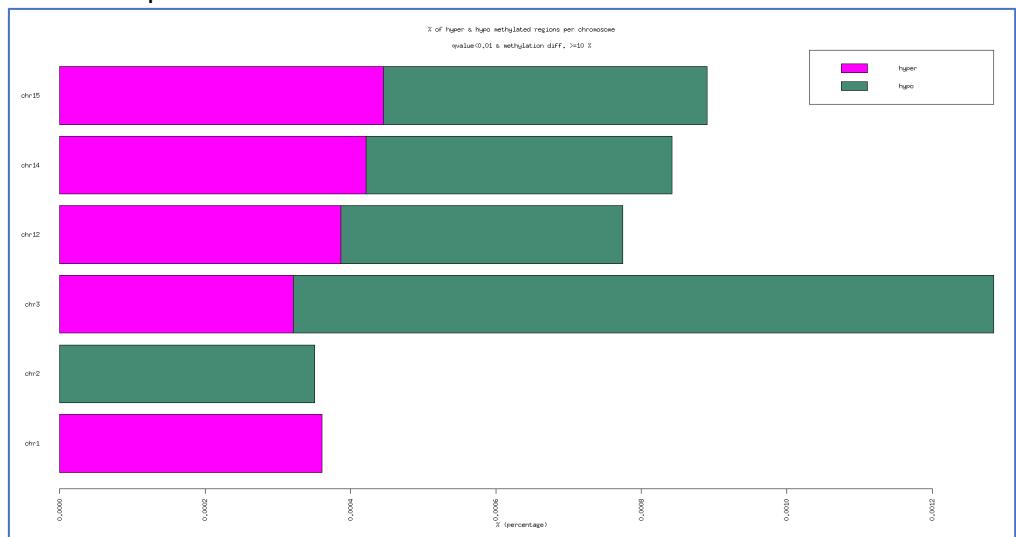
Annexe 8 B : Répartition des DMS échantillons contrôles contexte CHH



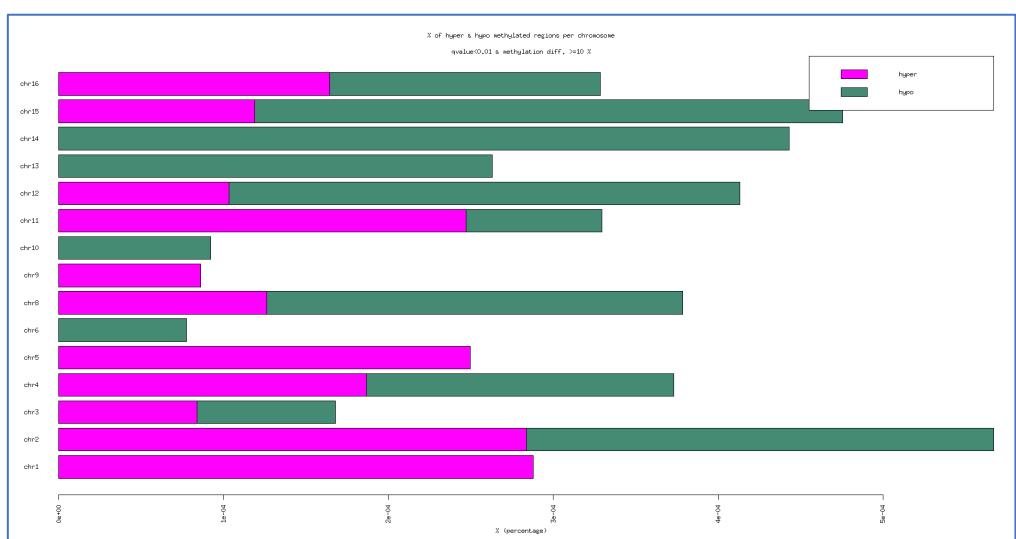
Annexe 8 C : Répartition des DMS échantillons contrôles contexte CpG



Annexe 8 D : Répartition des DMS échantillons contrôles contexte CHG



Annexe 8 E : Répartition des DMS échantillons contrôles contexte CHH



Annexe 9 A : pourcentage de présence des différences informations CT CpG

```
> diff10_GRanges <- as(diff10, "GRanges")
> save(diff10_GRanges, file = "diff10_GRanges.Rdata")
> genomatome::annotateWithFeature(diff10_GRanges, genes_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 3894
-----
percentage of target elements overlapping with features:
genes_GRanges      other
    97.3          2.7

percentage of feature elements overlapping with target:
[1] 15.06

> genomatome::annotateWithFeature(diff10_GRanges, promoters_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 3894
-----
percentage of target elements overlapping with features:
promoters_GRanges      other
    21.11         78.89

percentage of feature elements overlapping with target:
[1] 3.78

> genomatome::annotateWithFeature(diff10_GRanges, cds_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 3894
-----
percentage of target elements overlapping with features:
cds_GRanges      other
    69.18         30.82

percentage of feature elements overlapping with target:
[1] 2.4

> genomatome::annotateWithFeature(diff10_GRanges, exonic_parts_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 3894
-----
percentage of target elements overlapping with features:
exonic_parts_GRanges      other
    85.57         14.43

percentage of feature elements overlapping with target:
[1] 2.27

> genomatome::annotateWithFeature(diff10_GRanges, introns_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 3894
-----
percentage of target elements overlapping with features:
introns_GRanges      other
    18.82         81.18

percentage of feature elements overlapping with target:
[1] 0.61
```

Annexe 9 B : pourcentage de présence des différences informations CT CHG

```
> genomatation::annotateWithFeature(diff10_GRanges, genes_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 14
-----
percentage of target elements overlapping with features:
genes_GRanges      other
100                0

percentage of feature elements overlapping with target:
[1] 0.11

> genomatation::annotateWithFeature(diff10_GRanges, promoters_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 14
-----
percentage of target elements overlapping with features:
promoters_GRanges      other
42.86                57.14

percentage of feature elements overlapping with target:
[1] 0.04

> genomatation::annotateWithFeature(diff10_GRanges, cds_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 14
-----
percentage of target elements overlapping with features:
cds_GRanges      other
78.57                21.43

percentage of feature elements overlapping with target:
[1] 0.01

> genomatation::annotateWithFeature(diff10_GRanges, exonic_parts_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 14
-----
percentage of target elements overlapping with features:
exonic_parts_GRanges      other
92.86                7.14

percentage of feature elements overlapping with target:
[1] 0.01

> genomatation::annotateWithFeature(diff10_GRanges, introns_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 14
-----
percentage of target elements overlapping with features:
introns_GRanges      other
14.29                85.71

percentage of feature elements overlapping with target:
[1] 0
```

Annexe 9 C : pourcentage de présence des différences informations CT CHH

```
> genomatation::annotateWithFeature(diff10_GRanges, genes_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 31
-----
percentage of target elements overlapping with features:
genes_GRanges      other
    100            0

percentage of feature elements overlapping with target:
[1] 0.25

> genomatation::annotateWithFeature(diff10_GRanges, promoters_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 31
-----
percentage of target elements overlapping with features:
promoters_GRanges      other
    38.71          61.29

percentage of feature elements overlapping with target:
[1] 0.08

> genomatation::annotateWithFeature(diff10_GRanges, cds_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 31
-----
percentage of target elements overlapping with features:
cds_GRanges      other
    64.52          35.48

percentage of feature elements overlapping with target:
[1] 0.02

> genomatation::annotateWithFeature(diff10_GRanges, exonic_parts_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 31
-----
percentage of target elements overlapping with features:
exonic_parts_GRanges      other
    64.52          35.48

percentage of feature elements overlapping with target:
[1] 0.02

> genomatation::annotateWithFeature(diff10_GRanges, introns_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 31
-----
percentage of target elements overlapping with features:
introns_GRanges      other
    38.71          61.29

percentage of feature elements overlapping with target:
[1] 0.01
```

Annexe 9 D : pourcentage de présence des différences informations RG CpG

```
> genomatation::annotateWithFeature(diff10_GRanges, genes_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 5562
-----
percentage of target elements overlapping with features:
genes_GRanges      other
    97.79          2.21

percentage of feature elements overlapping with target:
[1] 19.4

> genomatation::annotateWithFeature(diff10_GRanges, promoters_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 5562
-----
percentage of target elements overlapping with features:
promoters_GRanges      other
    20.59          79.41

percentage of feature elements overlapping with target:
[1] 5.34

> genomatation::annotateWithFeature(diff10_GRanges, cds_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 5562
-----
percentage of target elements overlapping with features:
cds_GRanges      other
    70.53          29.47

percentage of feature elements overlapping with target:
[1] 3.4

> genomatation::annotateWithFeature(diff10_GRanges, exonic_parts_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 5562
-----
percentage of target elements overlapping with features:
exonic_parts_GRanges      other
    86.35          13.65

percentage of feature elements overlapping with target:
[1] 3.12

> genomatation::annotateWithFeature(diff10_GRanges, introns_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 5562
-----
percentage of target elements overlapping with features:
introns_GRanges      other
    17.87          82.13

percentage of feature elements overlapping with target:
[1] 0.8
```

Annexe 9 E : pourcentage de présence des différences informations RG CHG

```
> genomatation::annotateWithFeature(diff10_GRanges, genes_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 12
-----
percentage of target elements overlapping with features:
genes_GRanges      other
    100            0

percentage of feature elements overlapping with target:
[1] 0.09

> genomatation::annotateWithFeature(diff10_GRanges, promoters_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 12
-----
percentage of target elements overlapping with features:
promoters_GRanges      other
    50             50

percentage of feature elements overlapping with target:
[1] 0.03

> genomatation::annotateWithFeature(diff10_GRanges, cds_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 12
-----
percentage of target elements overlapping with features:
cds_GRanges      other
    100            0

percentage of feature elements overlapping with target:
[1] 0.02

> genomatation::annotateWithFeature(diff10_GRanges, exonic_parts_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 12
-----
percentage of target elements overlapping with features:
exonic_parts_GRanges      other
    100            0

percentage of feature elements overlapping with target:
[1] 0.01

> genomatation::annotateWithFeature(diff10_GRanges, introns_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 12
-----
percentage of target elements overlapping with features:
introns_GRanges      other
    0              100

percentage of feature elements overlapping with target:
[1] 0
```

Annexe 9 F : pourcentage de présence des différences informations RG CHH

```
> genomatation::annotateWithFeature(diff10_GRanges, genes_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 43
-----
percentage of target elements overlapping with features:
genes_GRanges      other
    95.35          4.65

percentage of feature elements overlapping with target:
[1] 0.34

> genomatation::annotateWithFeature(diff10_GRanges, promoters_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 43
-----
percentage of target elements overlapping with features:
promoters_GRanges      other
    48.84          51.16

percentage of feature elements overlapping with target:
[1] 0.16

> genomatation::annotateWithFeature(diff10_GRanges, cds_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 43
-----
percentage of target elements overlapping with features:
cds_GRanges      other
    60.47          39.53

percentage of feature elements overlapping with target:
[1] 0.02

> genomatation::annotateWithFeature(diff10_GRanges, exonic_parts_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 43
-----
percentage of target elements overlapping with features:
exonic_parts_GRanges      other
    69.77          30.23

percentage of feature elements overlapping with target:
[1] 0.03

> genomatation::annotateWithFeature(diff10_GRanges, introns_GRanges)
summary of target set annotation with feature annotation:
Rows in target set: 43
-----
percentage of target elements overlapping with features:
introns_GRanges      other
    30.23          69.77

percentage of feature elements overlapping with target:
[1] 0.01
```

Annexe 10 A : programme d'annotation partie 1

```

import sys, getopt

try:
    opts, args = getopt.getopt(sys.argv[1:], "hd:g:o:")
except getopt.GetoptError:
    print ("\n", "## Utilisation non valide ###", "\n")
    print ("Utilisation = DMS_gene.py <options> -d <DMS_pos.txt> -g <genes_pos.txt> -o <output>")
    print ("Pour avoir des informations sur le code, faire: annotation.py -h")
    sys.exit(99)

for opt, arg in opts:
    if opt == '-h':
        print ("\\n" + " Obtenir l annotation génétique des positions DMS à partir d un fichier d annotation génétique." + "\\n")
        print ("Utilisation = DMS_gene.py <options> -d <DMS_pos.txt> -g <genes_pos.txt> -o <output name>")
        print ("\\nDMS_pos.txt = Une liste délimitée par des espaces de tous les SNP différemment méthylés dans laquelle la première colonne = Un fichier délimité par des espaces contenant tous les gènes (un par ligne). Les colonnes ont le format suivant: nom du gène, position, état de méthylation")
        print ("\\nattention, avant de lancer le programme, il faut avoir formaté les deux fichiers au format texte! \\n")
        sys.exit()
    elif len(opts) >= 2:
        if opt in ("-d"):
            in_DMS = open(arg)
        if opt in ("-g"):
            in_gene_annotation = open(arg)
        if opt in ("-o"):
            out = open(arg, "w")
    elif len(opts) < 1:
        print ('\\n', '## Argument manquant ###'+ '\\n'+ '\\n' 'utilisez -h pour avoir des informations sur le programme \\n')
        sys.exit(1)
    else:
        assert False, "Ce programme attend deux fichiers textes, contenant les différence de méthylation et les informations génétiques. \\n"

```

Annexe 10 B : programme d'annotation partie

```

for line in in_gene_annotation:
    if header is True:
        header = False
        continue
    else:
        line = line.rstrip()
        entry = line.split(" ")
        chrm_gene = str(entry[0])
        start_gene = int(entry[1])
        end_gene = int(entry[2])
        gene_id = str(entry[-1])
        gene_dict[gene_id] = [chrm_gene, start_gene, end_gene]
        if chrm_gene in chrm_dict.keys():
            gene_list = chrm_dict[chrm_gene]
            chrm_dict[chrm_gene] = gene_list + "," + gene_id
        else:
            chrm_dict[chrm_gene] = gene_id

header = True
for line in in_DMS:
    line = line.rstrip()
    if header is True:
        header = False
        out.write(line + " gene_id\\n")
        continue
    else:
        entry = line.split(" ")
        chrm_SNP = str(entry[0])
        pos_SNP = int(entry[1])

        search_genes = chrm_dict[chrm_SNP].split(",")
        for gene in search_genes:
            gene_annot = False
            gene_info = gene_dict[gene]
            if (pos_SNP >= gene_info[1]) and (pos_SNP <= gene_info[2]):
                out.write(line + " " + gene + " " + "\\n")
                gene_annot = True
                total_annotated += 1
                break
            if gene_annot is False:
                out.write(line + " NA " + "\\n")
                total_NotAnnotated += 1

print("\\n-----\\n")
print("Total SNPs annotés: " + str(total_annotated) + "\\n" + "Total SNPs non annotés: " + str(total_NotAnnotated))
print("Annotation ratio: " + str((total_annotated*100)/(total_annotated + total_NotAnnotated)) + "%")
print("\\n-----\\n")

out.close()
sys.exit(0)

```

Références

1. Inventaire National du Patrimoine Naturel, « Bombus terrestris (Linnaeus, 1758) - Bourdon Terrestre (LE) », Inventaire National du Patrimoine Naturel. https://inpn.mnhn.fr/espece/cd_nom/53104/tabc/fiche
2. X. Liu et al., « UHRF1 targets DNMT1 for DNA methylation through cooperative binding of hemi-methylated DNA and methylated H3K9 », Nature Communications, vol. 4, no 1, mars 2013, doi : 10.1038/ncomms2562.
3. « Le rôle de la méthylation de l'ADN révélé à l'échelle du génome dans l'embryon de souris | INSB », INSB. <https://www.insb.cnrs.fr/fr/cnrsinfo/le-role-de-la-methylation-de-ladn-revele-lechelle-du-genome-dans-lembryon-de-souris>
4. X. Liu et al., « UHRF1 targets DNMT1 for DNA methylation through cooperative binding of hemi-methylated DNA and methylated H3K9 », Nature Communications, vol. 4, no 1, mars 2013, doi: 10.1038/ncomms2562.
5. « Épigénétique », StudySmarter FR. <https://www.studysmarter.fr/resumes/biologie/corps-humain/epigenetique/>
6. L. Yang, B. Yu, Y. Liang, Y. Lu, et W. Li, « Time-varying and non-linear associations between metro ridership and the built environment », Tunnelling and Underground Space Technology, vol. 132, p. 104931, févr. 2023, doi : 10.1016/j.tust.2022.104931.
7. FelixKrueger, « TrimGalore/Docs/Trim_Galore_User_Guide.md at Master · FelixKrueger/TrimGalore », GitHub. https://github.com/FelixKrueger/TrimGalore/blob/master/Docs/Trim_Galore_User_Guide.md
8. Felix Krueger « Bismark Bisulfite Mapper – User Guide - v0.15.0 », https://rawgit.com/FelixKrueger/Bismark/master/Docs/Bismark_User_Guide.html
9. Dpryan, « GitHub - dpryan79/MethylDackel : a (mostly) universal methylation extractor for BS-SEQ experiments » , GitHub. <https://github.com/dpryan79/MethylDackel/tree/master>
10. « MethylKit.knit » <https://bioconductor.org/packages/devel/bioc/vignettes/methylKit/inst/doc/methylKit.html>
11. « Ouvrir des fichiers BED », [fileext.com](https://fileext.com/fr/extension-de-fichier/BED), [En ligne]. Disponible sur : <https://fileext.com/fr/extension-de-fichier/BED>
12. « GFF3 File Format ». <http://www.ensembl.org/info/website/upload/gff3.html>
13. « Genomation », Bioconductor. <https://bioconductor.org/packages/release/bioc/html/genomation.html>