Mémoire

Auteur : Smacchia, Leandro
Promoteur(s) : Baurain, Denis; Hanikenne, Marc
Faculté : Faculté des Sciences
Diplôme : Master en biochimie et biologie moléculaire et cellulaire, à finalité approfondie
Année académique : 2022-2023
URI/URL : http://hdl.handle.net/2268.2/18410

Date : 28 Août 2023

Mémoire de Master en Biochimie, Biologie Moléculaire et Cellulaire, à finalité approfondie

« Analyse de la distribution de familles de transporteurs métalliques dans des rhizosphères d'Arabidopsis halleri par métagénomique shotgun »

Année académique 2022-2023

Smacchia Leandro, s214750

MAB2 BBMC

Promoteur : Denis Baurain

Co-promoteur : Marc Hanikenne

Remerciements

Tout d'abord, je tiens à remercier le professeur Denis Baurain, mon promoteur, pour son soutien et ses conseils avisés tout au long de ce mémoire, pour m'avoir appris tant de choses sur la programmation et de m'avoir donné le gout de la bioinformatique. Je le remercie tout particulièrement d'avoir toujours su trouver du temps pour répondre à mes questions et m'accompagner jusqu'au bout.

Je remercie également le professeur Marc Hanikenne, mon co-promoteur pour ses conseils éclairés et les ressources qu'il m'a fournies.

Mes remerciements vont également au docteur Valérian Lupo, anciennement doctorant de l'Unité de Phylogénomique des Eucaryotes de Denis Baurain, pour m'avoir appris des techniques de bioinformatique et répondu à plusieurs de mes questions.

Je souhaiterais également remercier mes amis mémorants du bâtiment 22 d'avoir rendu ce mémoire agréable. Merci pour tous ces moments de décompression.

Enfin, je tiens à remercier ma copine Lindsay Namur pour son soutien sans faille, surtout dans les moments difficiles. Merci d'avoir veillé avec moi lors de la rédaction de ce mémoire, merci d'avoir été là. Je remercie également ma belle-famille qui m'a toujours soutenu dans ce travail et qui était là dans les moments de panique.

Résumé

Arabidopsis halleri est espèce hyperaccumulatrice facultative, ce qui implique non seulement qu'elle tolère des concentrations en métaux lourds plus élevées que la plupart des autres espèces de plantes, mais aussi qu'elle concentre ces métaux dans ses parties aériennes. *A. halleri* peut pousser indistinctement sur des sols métallifères et non-métallifères, une propriété qui est due au moins en partie à la plante elle-même. Cependant, il a été démontré que certains micro-organismes aident les plantes à tolérer et à se développer malgré divers stress environnementaux, tels que le stress dû aux métaux lourds. Dans ce cas, ces Plant Growth Promoting Microbes (PGPM) pourraient contribuer à la tolérance au stress par le biais de transporteurs de métaux précis. L'objectif de ce travail était d'analyser les données métagénomiques précédemment collectées dans les rhizosphères d'*A. halleri* et dans le sol environnemental voisin (bulk) pour montrer les différences entre les échantillons métallifères (M) et non-métallifères (NM). Ces différences pourraient alors être liées à des groupes microbiens particuliers et/ou à des familles de transporteurs de métaux particulières.

Des alignements de référence largement échantillonnés de plus de 30 familles de transporteurs de métaux ont été assemblés et les arbres protéiques correspondants ont été inférés par maximum de vraisemblance après filtration des alignements. Les échantillons métagénomiques présentant une distribution atypique de la taille de *scaffolds* ont été écartés et les échantillons restants ont été standardisés par sous-échantillonnage aléatoire afin de contenir le même nombre total de *scaffolds*. Les alignements de référence ont été utilisés pour extraire des protéines orthologues dans les 72 échantillons métagénomiques, sur bases de recherches d'homologie sophistiquées. Les séquences environnementales ont ensuite été placées dans les arbres de référence par maximum de parcimonie et les quatre arbres enrichis (c'est-à-dire M-rhizosphère / M-bulk / NM-rhizosphère / NM-bulk) de chaque famille ont été comparés visuellement. Parallèlement, les compositions taxonomiques de tous les échantillons ont été analysées par des analyses en composantes principales et des tests PERMANOVA, afin d'identifier les familles pour lesquelles la séparation des échantillons reflétait le type d'échantillon.

Alors que la plupart des cartes ACP n'ont pas révélé de différences évidentes entre les types d'échantillon, certaines d'entre elles ont montré un partitionnement relativement clair des quatre groupes ou des deux ensembles M et NM. Il s'agit notamment des *clusters* de protéines des familles Nramp, Lyse_ILT, ATPase de type P, CDF et RND. Ces observations ont été soutenues par un test statistique PERMANOVA évaluant le partitionnement des groupes en fonction de la métallicité sur la base des composantes principales, à l'exception de la famille des ATPases de type P, bien qu'il s'agisse de la famille dont le partitionnement en fonction de la métallicité est visuellement le plus flagrant sur l'ensemble des cartes ACP. Aucune conclusion claire n'a encore été tirée sur les organismes ou les groupes microbiens responsables de ce partitionnement. Cela est dû à la faible résolution des organismes de référence (en raison d'un large échantillonnage dans les trois domaines), et peut-être aussi à un manque de données. En effet, une très grande proportion de *reads* et de *contigs* non identifiés a été rejetée lors de l'assemblage. Toutefois, ce travail met en évidence des familles de transporteurs de métaux prometteuses et jette les bases pour de futures recherches. Un échantillonnage plus dense de taxons, spécifique à chaque famille de transporteurs, devrait être utilisé pour inférer des arbres de référence à plus haute résolution et identifier plus précisément les organismes qui pourraient jouer un rôle dans le caractère d'hyperaccumulation facultative d'*A. halleri*.

Table des matières

Liste des abréviations

ROS	Reactive Oxygen Species
PGPM	Plant Growth Promoting Microbes
CEC	Cation Exchange Capacity
Ca-CEC	Calcium Cation Exchange Capacity
NGS	Next Generation Sequencing
qPCR	quantitative PCR (Polymerase Chain Reaction)
PGPB	Plant Growth Promoting Bacteria
MAGs	Metagenome-Assembled Genomes
ML	Maximum de vraisemblance (Maximum Likelihood)
bp	Base pairs
BRH	Best Reciprocal Hits
RRT	Recursive Residuals-based Test
M-Bulk1	Échantillon de sol métallifère, sous-échantillon de type sol
M-rhiz	Échantillon de sol métallifère, sous-échantillon de type rhizosphère
N-Bulk1	Échantillon de sol non-métallifère, sous-échantillon de type sol
N-rhiz	Échantillon de sol non-métallifère, sous-échantillon de type rhizosphère
MP	Maximum de parcimonie (Maximum Parsimony)
ACP	Analyse en Composantes Principales
PC	Composante Principale (Principal Component)

Table des illustrations

Table des tableaux

1. Introduction

1.1. Pollution des sols et résistance aux métaux

Notre agroécosystème est influencé par des stress biotiques et abiotiques. Ces différents facteurs peuvent influencer négativement la productivité et la survie de la flore dans chaque environnement. Les facteurs abiotiques majeurs de stress dus au changement climatique et à l'activité humaine sont la température, la sécheresse, la salinité et la pollution aux métaux lourds. Ces facteurs peuvent également influencer les facteurs biotiques de stress qui à leur tour auront un impact sur la flore et la diversité microbienne du sol (Kumar & Verma, 2018). Les principaux stress biotiques et abiotiques sont repris dans le schéma de la figure 1.

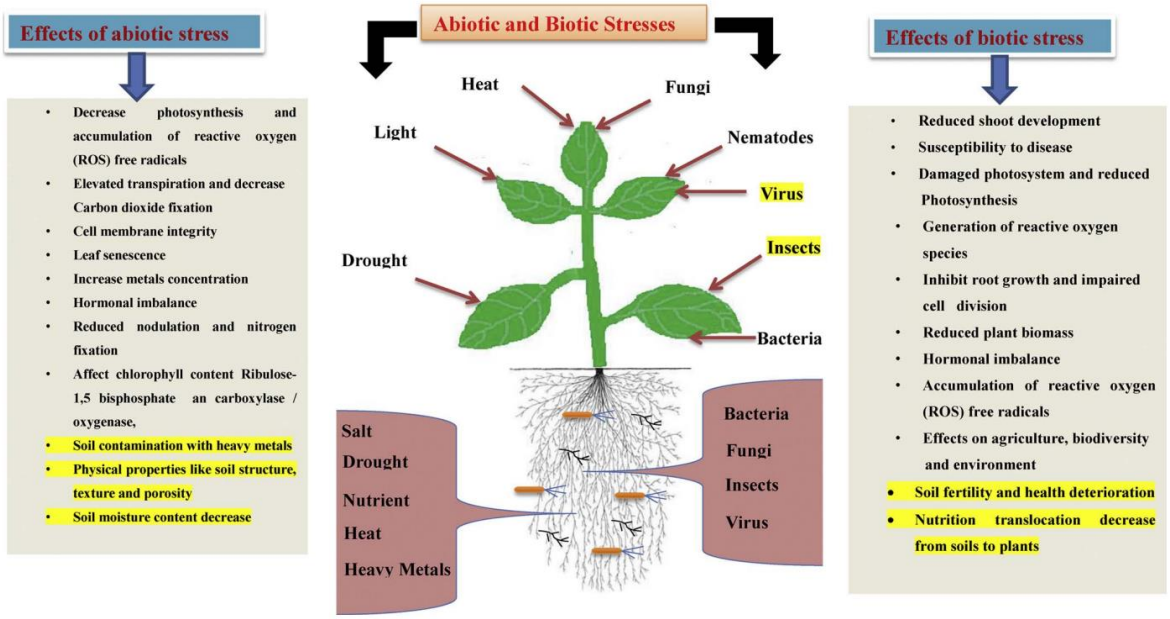


Figure 1 : Principaux stress biotiques et abiotiques chez les plantes (Kumar & Verma, 2018).

Certains sols présentent une concentration anormalement élevée en métaux lourds. Ces derniers sont caractérisés par une densité supérieure à 4g/cm^3, sont non dégradables et toxiques à faibles concentrations (Kumar & Verma, 2018). Ces environnements qualifiés de métallifères sont alors toxiques pour la plupart des organismes vivants, ce qui implique des communautés différentes, particulièrement la flore, sessile par essence (Hautekeete et al., 2018). Les activités industrielles humaines, en particulier les industries métallurgiques et les exploitations minières ont étendu la distribution de ces environnements en polluant certains sols (Hautekeete et al., 2018; Pollard et al., 2014). Il est à noter que les sols métallifères sont hétérogènes du point de vue écologique. En effet, la dichotomie « sols métallifères » et « sols non-métallifères » est une simplification grossière de la réalité, tant pour les environnements contaminés que pour les environnements « sains ». Les paramètres environnementaux, ainsi que les communautés microbiennes varient fortement d'un site métallifère à un autre. Les différents métaux et d'autres facteurs abiotiques représentent probablement une pression de sélection différente, propre à chaque combinaison de paramètres environnementaux (Hautekeete et al., 2018).

Les plantes vivant sur ces sols riches en métaux sont appelées métallophytes. Ces plantes peuvent parfois vivre uniquement dans ces environnements (eumétallophytes ou métallophytes obligatoires).

Une autre partie de ces plantes sont capables de vivre sur des sols métallifères et sur des sols non-métallifères (pseudométallophytes ou métallophytes facultatifs). Cette deuxième classe peut elle-même être subdivisée en trois catégories selon l'écotype dominant : Pseudométallophytes volontaires, indifférentes et accidentelles (Pollard et al., 2014).

1.1.1. Hyperaccumulateurs de métaux

La majorité des métallophytes survivent aux excès de métaux en les excluants par divers mécanismes de leurs parties aériennes. À l'inverse, une petite partie des plantes résistantes aux métaux ne tolèrent pas seulement leur présence mais les accumulent en concentrations exceptionnellement élevées dans leurs parties aériennes sans en subir les effets toxiques. Ces plantes sont alors qualifiées d'hyperaccumulateurs (Pollard et al., 2014). Cependant, les plantes capables d'accumuler de grandes concentrations d'éléments métalliques dans des conditions artificielles métallifères reproduites en laboratoire ne doivent pas être considérées comme hyperaccumulateurs ; pour ce faire, elles doivent présenter la caractéristique à l'état naturel. En effet, la plupart des plantes peuvent être forcées de concentrer ces éléments en laboratoire, au prix d'une croissance réduite, d'une potentielle stérilité, voire de la mort de l'organisme (Pollard et al., 2014).

La définition d'un hyperaccumulateur est un peu arbitraire mais universelle. Elle se base sur la définition arbitraire et logique de seuils de concentrations en métaux et est applicable à toutes les espèces de plantes. Une telle plante doit contenir dans ses feuilles un élément métallique dont la concentration dépasse un seuil défini, propre à chaque métal et prenant en compte la variance phénotypique environnementale (Pollard et al., 2014). Ces concentrations-seuils doivent être plus élevées (de deux ou trois ordres de grandeur) que les concentrations habituelles dans la majorité des espèces vivant sur des sols non pollués par des métaux. Ces seuils doivent également dépasser les taux des plantes vivant sur sols métallifères d'au moins un ordre de grandeur. Ils sont par exemple de 100µg/g pour le cadmium, 1000 µg/g pour le nickel et 3000 µg/g pour le zinc (en µg de métal par g de tissu foliaire séché) (Pollard et al., 2014).

Plusieurs hypothèses ont été formulées sur les avantages de l'hyperaccumulation de métaux. Cela pourrait être purement un mécanisme de tolérance à de fortes concentrations en métaux, un mécanisme d'interférence allélopathique avec d'autres plantes dans la compétition pour un environnement, un mécanisme de résistance à la sécheresse, un mécanisme de défense contre les herbivores et les pathogènes ou encore une caractéristique co-sélectionnée avec un autre avantage (Boyd & Martens, 1998; Pollard et al., 2014). L'hypothèse de la défense contre les herbivores et les pathogènes est de plus en plus soutenue. Cette défense basée sur des éléments-traces est diversifiée du point de vue composition et influencée par des facteurs spatiaux, temporels et physicochimiques. Les organismes interagissant avec la plante pourraient également contribuer à la dynamique de cette diversité chimique (Putra & Müller, 2023).

À peu près 750 espèces de plantes vasculaires ont été qualifiées d'hyperaccumulateurs, ce qui représente moins de 0.2% du nombre total de taxons de ce groupe et en fait donc une caractéristique rare. La grande majorité de ces hyperaccumulateurs (90%) accumulent du nickel, pour la plupart sur des sols ultramafiques. Ces sols issus de l'érosion de roches serpentines possèdent une composition pauvre en matière organique et nutriments essentiels et riche en métaux lourds, typiquement du fer, du cobalt et du nickel. Tout comme les autres métallophytes, les hyperaccumulateurs peuvent être obligatoires ou facultatifs. Entre 10 et 15 % des espèces d'hyperaccumulateurs sont facultatifs. La majorité est donc endémique de sols métallifères (Pollard et al., 2014).

Certaines espèces de métallophytes possèdent des populations d'hyperaccumulateurs et des populations simplement tolérantes et non-hyperaccumulatrices comme, respectivement, les populations PL22 et I16 chez *A. halleri* (Corso et al., 2018; Schvartzman et al., 2018).

1.1.2. Hyperaccumulateurs facultatifs

La plupart des études menées sur les hyperaccumulateurs facultatifs le sont sur des espèces de brassicacées, et sur l'accumulation du zinc et du cadmium. D'un côté, les brassicacées sont de petites plantes herbacées facile à cultiver et à croiser, ce qui facilite les études génétiques et physiologiques. Elles ont aussi pour avantage d'être toutes, en particulier *Arabidopsis halleri*, relativement proches d'*Arabidopsis thaliana*, un organisme modèle pour lequel existent un génome complet et de nombreuses ressources bioinformatiques (Merlot et al., 2021; Pollard et al., 2014). D'un autre côté, le cadmium est un polluant répandu et toxique auquel nous sommes exposés, principalement via la nourriture végétale (Clemens et al., 2013). Ce champ de recherche contraste avec la répartition du phénomène, plus commun chez les plantes tropicales, issues d'une multitude de familles et poussant sur des sols ultramafiques et hyperaccumulant du nickel (Hanikenne & Nouet, 2011). Le spectre de familles investiguées devra-t-être étendu dans le cadre des recherches menées sur les hyperaccumulateurs facultatifs pour inférer un modèle de compréhension du phénomène.

Contrairement à la simple tolérance aux métaux, l'hyperaccumulation est une caractéristique propre à l'espèce et non à la population. Là où la tolérance se développe sous une forte pression de sélection dans les populations touchées par la pollution aux métaux, l'hyperaccumulation apparaît présente au sein de l'espèce entière. Il semble cependant que la distribution de la capacité à accumuler les métaux soit plus complexe et puisse varier d'un métal à un autre. L'hyperaccumulation du zinc et du cadmium est considérée comme une caractéristique constitutive de l'espèce *A. halleri* (Krämer, 2010). Cette caractéristique dépend de plusieurs facteurs morphologiques et physiologiques comme le développement des racines ou le transport membranaire. Chez les hyperaccumulateurs facultatifs, les populations vivant sur sols non-métallifères semblent posséder une plus grande capacité à concentrer les métaux dans leurs parties aériennes, mais présentent une tolérance moindre, en comparaison des populations métallicoles (Pollard et al., 2014).

Pour expliquer le bénéfice adaptatif de l'hyperaccumulation comme trait propre à l'espèce, qui ne s'exprime pas pleinement chez les populations hors sols métallifères, trois hypothèses sont formulées. Une première possibilité est la simple conservation phylogénétique. Le trait facultatif serait développé à partir du trait obligatoire et les hyperaccumulateurs facultatifs descendraient des hyperaccumulateurs obligatoires par un mécanisme de colonisation de nouveaux milieux, non-métallifères. Une autre possibilité est l'avantage incrémental. Les plantes auraient un avantage à concentrer toujours un peu plus les métaux, que ce soit pour une possible défense, ou dans un but nutritif. En conséquence, l'hyperaccumulation sur sols contaminés pourrait être due à une prédisposition et ne présenter aucun avantage en soi. Enfin, comme pour les hyperaccumulateurs obligatoires, ce trait pourrait avoir été accidentellement sélectionné avec des avantages nutritifs ou liés à l'homéostasie du métal et pourrait donc ne présenter aucun avantage. De plus, ces hypothèses ne sont pas mutuellement exclusives (Boyd & Martens, 1998; Pollard et al., 2014).

1.2. *Arabidopsis halleri*

A. halleri est une espèce hyperaccumulatrice facultative de zinc et de cadmium. Cependant, toutes les populations métallicoles de l'espèce ne se comportent pas de la même façon. En effet, certaines populations se comportent en hyperaccumulateurs, d'autres en exclueurs de cadmium (Corso et al., 2018; Meyer et al., 2015; Schvartzman et al., 2018).

Le niveau d'accumulation de zinc et de cadmium est très différent entre les deux types de populations. Cette variation est accompagnée d'une variation de l'expression de gènes associés aux mécanismes de tolérance du cadmium en présence de fortes concentrations de ce métal (Corso et al., 2018; Schvartzman et al., 2018). Les gènes *IRT1* et *CAX4*, encodant respectivement le transporteur de fer divalent principalement responsable de la capture du cadmium et un transporteur membranaire exerçant une influence sur le transport du cadmium, sont plus fortement exprimés chez une population hyperaccumulatrice d'*A. halleri*, PL22 (Pologne). Le gène *OPT3*, encodant un autre transporteur membranaire, est induit par la présence de cadmium (Corso et al., 2018).

Là où une augmentation du niveau de cadmium environnant induit certaines différences dans l'expression de gènes influençant le transport et la capture du cadmium, l'expression des gènes de transport du zinc dans les parties aériennes de la plante reste pratiquement inchangée lors d'une augmentation du niveau de zinc environnant (Corso et al., 2018; Schvartzman et al., 2018). La forte expression constitutive de gènes de l'homéostasie du métal dans les parties aériennes d'*A. halleri* donne lieu à des niveaux de zinc basaux tolérés plus élevés et explique la répartition du caractère à l'échelle de l'espèce (Becher et al., 2004; Talke et al., 2006). La tolérance au zinc apparaît donc comme un trait constitutif de l'espèce (Schvartzman et al., 2018). Les changements de niveau d'accumulation du zinc apparaissent alors contrôlés par des processus racinaires (Schvartzman et al., 2018). La forte expression du gène *HEAVY METAL ATPAse 4* (*HMA4*) dans les racines est un facteur clé de l'hyperaccumulation au sein de l'espèce (Claus et al., 2013; Hanikenne et al., 2008). En effet, la duplication du gène *HMA4* concomitante à la spéciation d'*A. halleri* pourrait expliquer en partie son trait hyperaccumulateur (Pollard et al., 2014). Ce gène, alors présent en trois exemplaires, joue un rôle central dans la translocation du zinc et du cadmium des racines aux parties aériennes de la plante (Hanikenne et al., 2008). *HMA4* est pourtant davantage exprimé dans une population exclueur de cadmium (I16), plutôt que chez la population PL22, du moins dans les parties aériennes. Une explication pourrait être l'exclusion du cadmium des tissus photosynthétiques des feuilles, grâce à l'expression de *HMA4* dans le mésophylle (Corso et al., 2018).

I16, une population d'*A. halleri* excluant le zinc le fait via une capture contrôlée par les racines, en limitant la prise de zinc aux quantités nécessaires, là où PL22, une population hyperaccumulatrice adopte une stratégie de translocation contrôlée entre les racines et les parties aériennes. Bien que la famille *ZIP* contienne des transporteurs du zinc candidats à l'explication de l'augmentation de la capture de ce métal par les racines, cette différence ne provient pas d'un changement dans l'expression de gènes de cette famille. Un gène candidat, *IRT1*, dont la protéine sert à la fois au transport du zinc et du fer (et du cadmium), est plus exprimée chez les hyperaccumulateurs de la population PL22 et pourrait grandement contribuer à une capture plus importante du zinc. Cela pourrait témoigner d'une adaptation plus poussée des exclueurs de métaux de la population I16, possédant une régulation plus spécialisée de l'homéostasie du fer et du zinc (Schvartzman et al., 2018; Talke et al., 2006).

Chez PL22, certains gènes impliqués dans le métabolisme des flavonoïdes sont surexprimés et ces flavonoïdes se voient accumulés dans les parties aériennes des individus. Ce métabolisme est en lien

direct avec la détoxification des métaux et des ROS (*Reactive Oxygen Species*). En effet, les flavonoïdes, inhibent les effets des ROS et protègent donc les cellules de potentiels dommages oxydatifs dus aux métaux (Corso et al., 2018).

L'hypertolérance aux métaux chez *A. halleri* est principalement contrôlée par les parties aériennes et les transporteurs de métaux y étant exprimés. Au contraire, l'hyperaccumulation des métaux est principalement contrôlée par les racines et les transporteurs y étant exprimés (Corso et al., 2018).

Beaucoup de gènes présupposés jouer un rôle dans l'hypertolérance et l'hyperaccumulation du zinc et du cadmium chez *A. halleri*, dont *HMA4*, *MTP1*, *NAS2* et d'autres gènes de la famille *ZIP* ne sont pas exprimés différentiellement entre les deux populations, ce qui suggère un rôle dans la tolérance et l'hyperaccumulation constitutive de l'espèce. L'adaptation récente plus poussée aux sites métallifères serait alors due à d'autres procédés (Schvartzman et al., 2018). Cette adaptation pourrait également être purement convergente entre les différentes unités génétiques de l'espèce. L'expression multicopie du gène *MTP1* augmentant le stockage vacuolaire du zinc dans les parties aériennes pourrait avoir évolué uniquement chez les populations métallicoles d'*A. halleri* (Meyer et al., 2016; Schvartzman et al., 2018). Ces gènes restent associés à l'hyperaccumulation et leurs fonctions dans l'absorption cellulaire, le chargement et déchargement dans le xylème et le stockage vacuolaire dont partie du modèle explicatif du caractère d'hyperaccumulation (Merlot et al., 2021).

1.3. Communautés microbiennes du sol

Les Plant Growth Promoting Microbes (PGPM) apparaissent comme une approche viable pour surpasser l'impact négatif des changements environnementaux et contrer les stress biotiques et abiotiques subis par les plantes. Des consortia microbiens adaptés aux conditions de chaque environnement, de chaque sol pourraient servir d'ingénieurs écologiques et permettre à la flore de se développer dans les diverses situations de stress biotiques et abiotiques (Kumar & Verma, 2018).

Les PGPM peuvent être bénéfiques à la croissance des plantes par plusieurs mécanismes. Ils peuvent produire certains régulateurs de croissance et certaines hormones ou réguler la nutrition de la plante. Certains métabolites produits induisent une résistance aux phytopathogènes, comme des sidérophores privant les pathogènes de fer et neutralisant les métaux lourds. D'autres PGPM sont capables de fixer l'azote atmosphérique, de solubiliser les phosphates, de mobiliser des nutriments, ou encore de produire des exopolysaccharides ou des rhizobitoxines (molécule inhibant la production d'éthylène, un facteur de sénescence). Les PGPM peuvent également produire des enzymes clés en cas de stress, comme la glucanase ou la chitinase. Les mycorhizes fongiques représentent également des contributions importantes à la croissance des plantes (Kumar & Verma, 2018).

1.3.1. Résistance aux métaux lourds assistée par les microbes

Là où les techniques de remédiation classiques des sols sont des approches lourdes et peu rentables et efficaces, certaines plantes tolèrent ces stress métalliques et contribuent à la dépollution des sols, ce qui porte le nom de phytoremédiation. Les communautés bactériennes des rhizosphères d'espèces hyperaccumulatrices de métaux ont été relativement peu investiguées jusqu'à ces cinq dernières années. Les bactéries rhizosphériques peuvent pourtant être d'une importance capitale dans la phytoremédiation et l'accumulation chez ces espèces (Benizri & Kidd, 2018; Kumar & Verma, 2018; Lopez et al., 2019). En effet, les plantes accumulatrices de métaux atteignent leur capacité

d'accumulation maximum uniquement en présence des communautés microbiennes indigènes de leurs rhizosphères (de Souza et al., 1999). Les communautés microbiennes associées à ces plantes contribuent aux mécanismes de tolérance, d'accumulation et de phytoremédiation. Ces PGPM sont par exemple des rhizobactéries ou des firmicutes. Les mycorhizes jouent également un rôle dans cette résistance (Kumar & Verma, 2018).

Les PGPM participent à la décontamination des sols contaminés aux métaux lourds par des mécanismes chimiques et physiques. Les mécanismes majeurs sont l'accumulation, qu'elle soit intracellulaire ou extracellulaire, la séquestration et la biotransformation. Cette dernière consiste en la conversion biochimique de métaux hautement toxiques en d'autres formes moins toxiques, en changeant par exemple le niveau d'oxydation de l'atome. Certains microbes possèdent même la capacité d'éliminer complètement certains métaux lourds (Kumar & Verma, 2018). En effet, *Pseudomonas alcaliphila* est capable de dégrader les complexes nickel-citrate et d'éliminer le nickel par co-précipitation avec du Fe^{3+}, bioaccumulation et précipitation (Qian et al., 2012). D'autres espèces de *Pseudomonas* sont capables de neutraliser le cadmium en le séparant de ses complexes organiques grâce à la production d'acides organiques ou à accumuler par biosorption grâce à des peptides et sidérophores (Kushwaha et al., 2022).

L'aide apportée par les PGPM à la phytoremédiation et la résistance au stress métallique peut prendre une forme directe ou indirecte. Les formes directes impliquent des mécanismes influant sur la biodisponibilité des métaux lourds. La solubilisation, la volatilisation, l'efflux, l'imperméabilité aux métaux, l'accumulation et la séquestration par les exopolysaccharides, la complexation des métaux et la détoxification enzymatique en sont des exemples. Ces processus se font notamment via la synthèse de sidérophores, un composé organique léger complexant les métaux (principalement le fer) et permettant l'internalisation dans le cytosol. Les bactéries associées aux plantes peuvent appartenir à la rhizosphère, qui sont alors plus à même d'agir sur l'environnement, par exemple en synthétisant des sidérophores. D'autres bactéries comme les rhizobactéries sont endophytiques et ont alors plus d'influence sur la physiologie de la plante. Ces bactéries peuvent produire des nitrogénases dans le but de fixer l'azote atmosphérique, ou alors des phytohormones comme les auxines pour promouvoir la croissance des racines et améliorer la capture de nutriments. Ces PGPM peuvent également produire des petites molécules telles que l'acide citrique ou l'acide gluconique, et modifier l'état d'oxydation des métaux lourds pour améliorer leur solubilité, mobilisation et disponibilité (Kumar & Verma, 2018; Kushwaha et al., 2022).

Un résumé visuel des mécanismes d'aide directe apportée par les PGPM à la résistance au stress métallique est repris à la figure 2. Dans ce travail, ce sont plus les transporteurs de métaux des procaryotes qui nous intéressent. Ces derniers sont très diversifiés. Il existe en effet de nombreuses familles de transporteurs de métaux (CDF, ATPases, Nramp, ZIP, …). Ces transporteurs sont nécessaires à la capture de minéraux essentiels, mais aussi à la régulation de ces éléments (Hall & Williams, 2003).

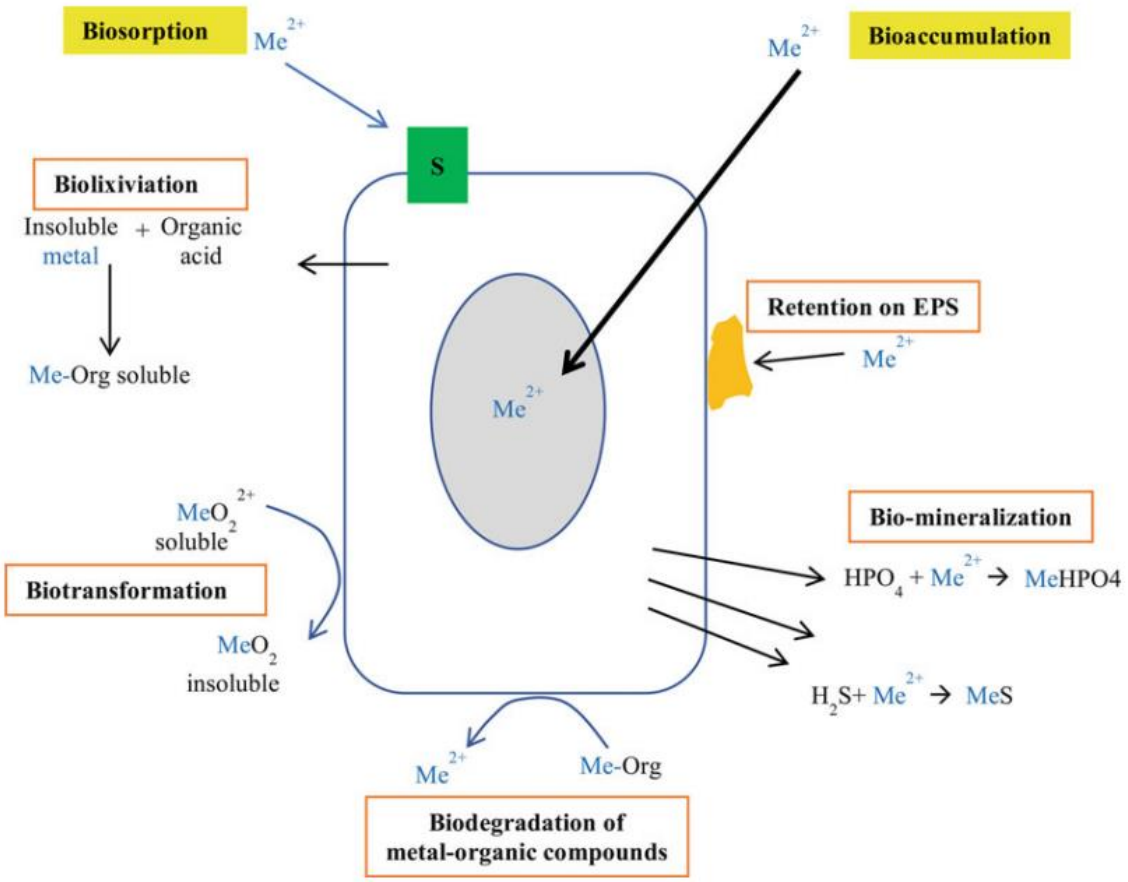


Figure 2 : Mécanismes d'aide directe apportée par les PGPM à la résistance des plantes au stress métallique (Benizri & Kidd, 2018)

Les mécanismes d'aide indirects consistent à améliorer la croissance des plantes, dans un environnement ou cette dernière est limitée par la difficulté de capture des nutriments. Les PGPM promeuvent la croissance et le développement des plantes auxquelles ils sont associés en produisant des régulateurs de croissance tels que l'auxine IAA ou l'ACC désaminase, enzyme inhibant la production d'éthylène. Ils améliorent également la nutrition des plantes en fixant l'atmosphérique ou en mobilisant des nutriments, comme expliqué dans la section précédente. L'inhibition de l'infection par des pathogènes constitue également un apport indirect des PGPM à cette résistance au stress métallique (Kumar & Verma, 2018).

Les applications potentielles de ces communautés de PGPM sont diverses. Ils peuvent à la fois servir de bio-fertiliseurs et de bio-pesticides et donc remplacer les équivalents inorganiques. Des applications industrielles sont également envisageables parmi lesquelles une technologie de remédiation des sols pollués, en particulier une version améliorée de la phytoremédiation des sols pollués aux métaux (Kumar & Verma, 2018).

1.3.2. Composition des communautés microbiennes

La capacité de stockage de cations considérés comme des nutriments, aussi appelée Cation Exchange Capacity (CEC) est un facteur environnemental fortement corrélé à l'abondance relative des différents phyla bactériens représentés dans les rhizosphères d'espèces de plantes hyperaccumulatrices de métaux (Lopez et al., 2019). Cette caractéristique des sols est une des caractéristiques les plus liées à

la diversité dans les communautés bactériennes du sol (Docherty et al., 2015; Liddicoat et al., 2018). La Ca-CEC en particulier semble avoir une influence non négligeable sur la composition des communautés bactériennes (Lopez et al., 2019). D'autres facteurs abiotiques exercent une forte influence sur la diversité des communautés microbiennes. Le pH des sols en est un parfait exemple. Ce facteur édaphique est déterminant dans la richesse écologique de ces communautés, particulièrement en ce qui concerne les bactéries (Mishra et al., 2023). La variabilité de la composition des communautés microbienne dépend de tellement de facteurs qu'il en est compliqué de retenir un facteur ayant une influence claire et systématique sur cette composition (Islam et al., 2020).

La présence de métaux lourds dans les sols peut également impliquer des différences de compositions dans les communautés bactériennes. Certains groupes capables de tolérer la contamination à ces métaux lourds dominent logiquement les communautés bactériennes des sols métallifères. Ces groupes peuvent alors servir d'indicateurs microbiens de pollution aux métaux lourd (Lazzaro et al., 2008). Plusieurs études ont montré que les phyla bactériens *Actinobacteria* et *Proteobacteria* sont dominants dans les sols contaminés aux métaux (Lazzaro et al., 2008; Lopez et al., 2019; Tipayno et al., 2012). Les facteurs abiotiques et biotiques exerçant une influence sur la composition microbienne des sols sont repris à la figure 3.

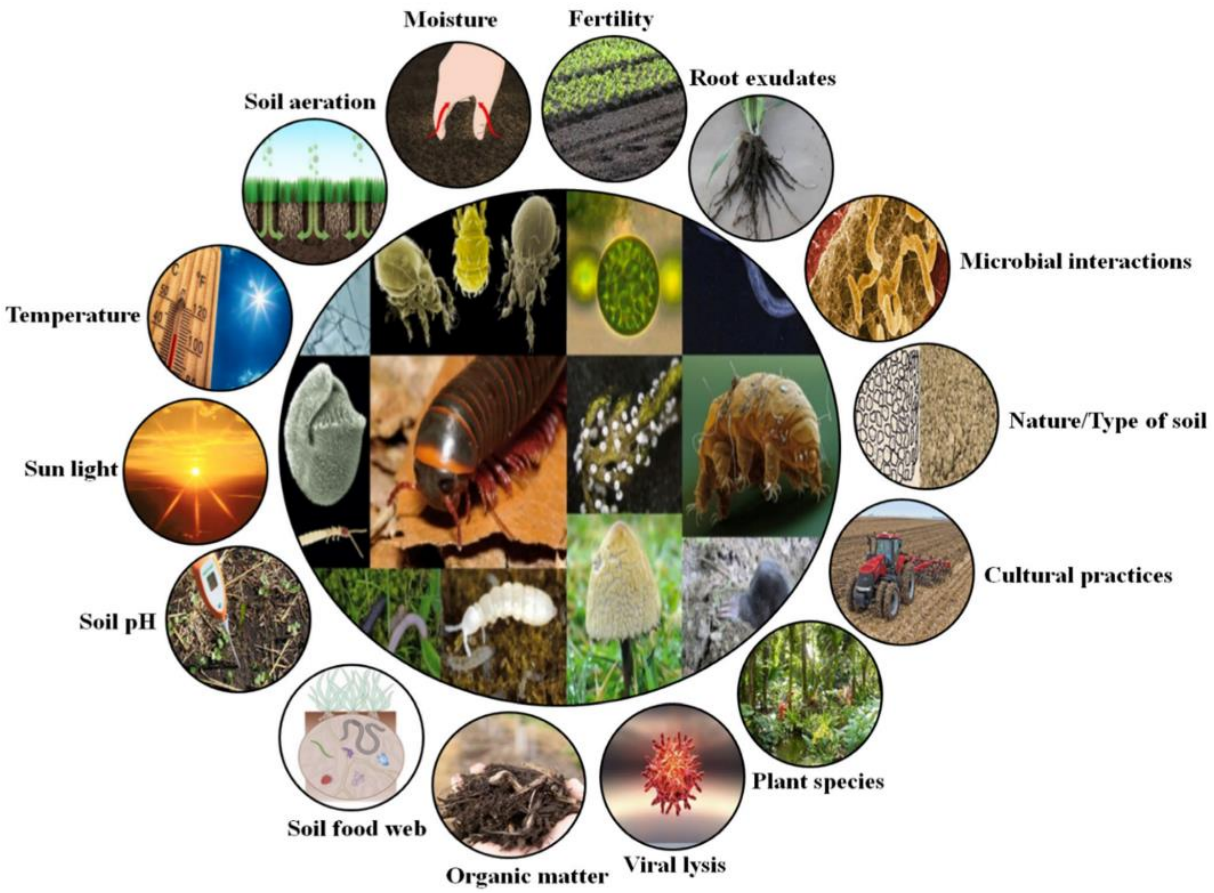


Figure 3 : Facteurs abiotiques et biotiques influant la composition du microbiome des sols (Islam et al., 2020).

Outre les variations dans les différents facteurs édaphiques, la micro-localisation dans le sol est un paramètre important ayant un impact sur la composition microbienne. Deux micro-environnements sont à distinguer : la rhizosphère et la masse du sol ou *bulk soil*. Dans le premier, les différents microbes se retrouvent à proximité immédiate de l'organisme végétal et plutôt en surface. Le deuxième micro-environnement comprend la terre plus profonde, au moins à quelques centimètres des racines de la plante (Islam et al., 2020).

Les groupes microbiens les plus abondants dans les sols sont les bactéries et les champignons. En effet, ces derniers comptent 10^2 à 10^4 fois plus de biomasse que les autres taxons microbiens (Islam et al., 2020). La majorité de la diversité des espèces bactériennes et archéennes correspond à une minorité de phyla de ces domaines. De plus, la plupart de ces espèces sont associées à des lignées non décrites, rares ou mal étudiées et par conséquent assez différentes des espèces référencées dans les bases de données. Des conclusions similaires peuvent être inférées de la diversité fongique et protiste (Islam et al., 2020).

1.4. Métagénomique

La grande majorité des procaryotes connus ne peuvent pas être cultivés en laboratoire, ce qui rend obligatoires d'autres moyens d'études de ces lignées. Une de ces autres méthodes consiste à les étudier directement dans leur habitat naturel. Les stratégies d'études en culture classiques sous-estiment grandement la diversité microbienne (Islam et al., 2020). La génétique, la génomique, la transcriptomique ou la protéomique de ces microbes peuvent être étudiées grâce aux technologies de la méta-omique. Ainsi, nous pouvons étudier la génomique de ces nombreuses bactéries et archées en caractérisant un échantillon de l'environnement par séquençage shotgun et métagénomique. Le principal défi des méta-omiques est la quantité et l'incomplétude des données récoltées (Myrold & Nannipieri, 2013).

La métagénomique et les autres méta-omiques permettent par exemple d'étudier le métabolisme de communautés microbiennes. Elles permettent également d'investiguer les dynamiques et variations structurelles génomiques. Outre la caractérisation fonctionnelle et taxonomique des microbes, les méta-omiques permettent l'analyse des interactions inter-espèces telles que le mutualisme et le parasitisme (Hiraoka et al., 2016). L'étude intégrative de plusieurs jeux de données métagénomiques fait ressortir des modèles généraux, ainsi que des lois déterminant les interactions entre les microbes et leur environnement et l'évolution de leurs génomes (Hiraoka et al., 2016).

1.4.1. Métagénomique des sols

Le domaine d'étude des microbiomes du sol s'est beaucoup développé durant ces dix-huit dernières années, avec l'introduction de la technologie de séquençage par synthèse et l'avènement des technologies de séquençage *next-generation* (NGS). De nombreux articles scientifiques portant sur l'étude des microbiomes du sol, par des méthodes moléculaires telles que la qPCR, le séquençage NGS basé sur des amplicons ou encore la métagénomique shotgun ont été publiés. Ces avancées et techniques, particulièrement les technologies NGS et la métagénomique ont largement contribué à une meilleure compréhension des communautés microbiennes, y compris leur rôle dans les sols. Elles ont cependant encore beaucoup à offrir à ce domaine d'étude (Mishra et al., 2023).

Par essence, les méta-omiques visent à cataloguer tous les gènes, transcrits, protéines, … présents dans un environnement particulier, ce qui permet par la suite une meilleure compréhension des fonctions et relations entre les différents acteurs du sol. Les gènes, protéines, et organismes méconnus peuvent également être étudiés grâce à ces méta-omiques, ils nécessitent précisément ce type d'approche. Nous pouvons en dire de même pour les communautés dont on ne connait pas la composition (Myrold & Nannipieri, 2013).

La métagénomique et les technologies associées ont permis plusieurs avancées scientifiques. L'étude et la découverte de nouveaux antibiotiques ont été permises en sondant et catégorisant des bactéries

non-cultivables grâce à la métagénomique (Islam et al., 2020; Myrold & Nannipieri, 2013). La composition taxonomique des rhizosphères, ainsi que les associations fonctionnelles de communautés microbiennes associées aux racines de plantes ont également pu être investiguées grâce aux méta-omiques (Islam et al., 2020). Le domaine des archées, représentant environ 0.5 à 10 % des procaryotes du sol est difficile à étudier. En effet, peu d'espèces d'archées peuvent être cultivées, d'où l'importance d'études métagénomiques d'échantillons de sols. Ces dernières permettent d'en mesurer l'abondance et la diversité. Les archées jouent un rôle majeur dans l'assimilation et la minéralisation de matières organiques et du méthane. Elles jouent également un rôle important dans le cycle de l'azote et ont un impact direct sur la qualité des sols (Islam et al., 2020).

1.4.2. Métagénomique des sols métalliques

L'hyperaccumulation facultative d'*A. halleri* est en partie due à la plante, mais une composante microbienne existe. Les études métagénomiques des communautés microbiennes des sols métallifères peuvent alors renseigner sur la diversité fonctionnelle potentiellement associée à ce phénomène. Après les gènes relatifs aux métabolismes des acides aminés, lipides et glucides, les gènes de transport membranaire sont parmi les plus abondants lors d'une prédiction de fonction de gènes d'une communauté bactérienne associée à la rhizosphères d'hyperaccumulateurs de nickel. Ces gènes pourraient avoir une importance dans les interactions entre les bactéries et l'environnement métallifère (Lopez et al., 2019).

Les différents écotypes d'*A. halleri* exercent une influence différente sur les biotes du sol. En conséquence, certains écotypes sont plus adaptés que d'autres à la phytoremédiation. Il a été montré que la diversité fonctionnelle des communautés bactériennes du sol est plus élevée lorsque ces communautés vivent dans un sol affecté par un des écotypes métallicoles d'*A. halleri* (Klimek et al., 2023). Les rhizosphères d'*A. halleri* accueillent des microorganismes tolérants à certains métaux spécifiques, mais également des PGPM. Parmi ces derniers, des bactéries des familles *Burkholderiaceae* et *Sphingomonadaceae* ont été identifiés (Kushwaha et al., 2022). Les PGPB des groupes *Rhodoplanes*, *Crossiella* et *Solirubrobacteraceae*, dont les deux derniers sont des actinobactéries, ont été associés à l'hyperaccumulation du zinc et du cadmium chez *A. halleri* (Kushwaha et al., 2022). A leurs tours, des populations spécifiques de bactéries associées à la rhizosphère d'hyperaccumulateurs de métaux influencent la physiologie de la plante (Merlot et al., 2021). Il a été montré que le microbiote spécifique des graines affectait la germination chez *A. halleri* dans des environnements métallifères, contribuant potentiellement à l'hyperaccumulation de cette espèce (Murawska-Wlodarczyk et al., 2022). Elles participent à la croissance et à la protection contre les stress abiotiques. Ces organismes sont, pour la plupart, non cultivables et sont donc étudiés par des méthodes métagénomiques (Merlot et al., 2021).

La contamination aux métaux lourds affecte la structure des communautés microbiennes. C'est en particulier vrai pour la composante procaryote des sols. La localisation précise dans le sol est également corrélée à la diversité microbienne. Ainsi, les communautés rhizosphériques et du sol sont différentes (Kushwaha et al., 2022). Cette dernière différence est influencée par la métallicité de l'environnement. En effet, la différence est accentuée dans les sols métallifères. *A. halleri* recrute une communauté microbienne rhizosphérique particulièrement unique (comparée au sol plus profond) dans des environnements pollués aux métaux (Kushwaha et al., 2022). Ces propriétés ne sont pas propres à *A. halleri*. Des conclusions identiques ont été établies pour l'espèce végétale *Atriplex lentiformis* (Honeker et al., 2019; Valentín-Vargas et al., 2018).

2. But du travail

Ce mémoire s'inscrit dans la continuité du projet entrepris par Amandine Bertrand lors de son doctorat. Il s'agit d'un projet d'étude comparative métagénomique des communautés microbiennes du sol, et plus particulièrement associées à la rhizosphère d'*Arabidopsis halleri* vivant sur deux types de sol, métallifères et non-métallifères. Le but du projet est d'investiguer la composante microbienne, surtout procaryote, qui pourrait contribuer au trait d'hyperaccumulation facultative de l'espèce.

Ce mémoire reprend le projet (avorté suite au départ de l'étudiante) dans sa partie analyse des données de séquençage du sol. Cette partie est construite autour d'une technique innovante d'analyse de gènes et d'espèces présents dans une communauté. Le principe est de choisir des familles de gènes d'intérêt, impliqués dans le transport des métaux, de construire une représentation de la phylogénie de ces familles et d'y insérer les données de séquençage du sol par recherche d'orthologie afin d'étudier la distribution de séquences procaryotes au travers des sites d'échantillonnage.

L'objectif principal du doctorat d'Amandine Bertrand et de ce mémoire est l'analyse fonctionnelle des communautés microbiennes du sol et de la rhizosphère d'*Arabidopsis halleri*. Cette analyse et ces données ont pour but de mettre en évidence certains organismes et/ou certains gènes ayant une potentielle contribution à l'adaptation d'*A. halleri* à son écotype vivant sur sol pollués aux métaux.

L'objectif technique de ce travail est de tester la méthode imaginée pour le projet et, en quelque sorte, de concevoir un protocole applicable à d'autres familles de protéines. Un autre objectif est d'orienter le choix vers certaines familles de transporteurs de métaux à analyser plus finement.

3. Matériel et méthodes

3.1. Environnement

3.1.1. Hardware

Lors de ce mémoire, le cluster de calcul *durandal* a été utilisé afin de mener à bien les calculs lourds requis par certains programmes, mais aussi de stocker les données. Il s'agit d'un système IBM/Lenovo Flex composé d'un gros nœud de calcul x440 et de onze nœuds plus petits x240. Ce cluster de calcul appartenant à InBioS-PhytoSYSTEMS et géré par l'Unité de Phylogénomique des Eucaryotes possède 228 cœurs physiques répartis dans les 12 nœuds de calcul donnant accès à 456 cœurs logiques en Hyper-Threading. Il totalise également 2880 cœurs CUDA (GPU), 2.9 TB de RAM et 162 TB d'espace de stockage. Le système d'exploitation de *durandal* est le système Linux CentOS 6.6.

Mon ordinateur portable a également été utilisé. Il s'agit d'un Asus VivoBook F512D comprenant un processeur AMD Ryzen™ 5 3500U à 4 cœurs physiques (8 cœurs logiques), 12 GB de RAM et 237 GB d'espace de stockage.

3.1.2. Unix

UNIX est un type de système d'exploitation incluant un interpréteur en ligne de commande (ou shell) et dont Linux est une implémentation très répandue. Lors de ce mémoire, de nombreuses manipulations ont été effectuées dans un shell UNIX, fourni par le programme MobaXterm. Ce programme permet également la connexion à distance entre mon ordinateur portable et *durandal* via SSH (Secure SHell), protocole de connexion sécurisée à distance.

3.1.3. iTOL

iTOL ou interactive Tree Of Life est un outil en ligne permettant de visualiser, manipuler et annoter des arbres phylogénétiques. Cet outil a été utilisé pour produire le visuel de chacun des arbres de ce mémoire.

3.1.4. Perl

Les différents scripts utilisés lors de ce mémoire ont été rédigés en Perl. Ce langage de programmation inspiré des langages de scripts sed, awk et bash, mais aussi de structures du langage C, permet de traiter de l'information textuelle de manière pratique, notamment par un support natif des expressions régulières.

3.1.5. R

Le langage de programmation R, conçu pour une manipulation et un stockage efficaces des données, ainsi que des analyses diverses, incluant des analyses statistiques et graphiques, a été utilisé lors de ce mémoire, principalement pour des tests statistiques et des représentations graphiques de certaines données.

3.2. Données disponibles

Cette section retrace les éléments et traitements antérieurs à ce mémoire, qui ont été réalisés par Amandine Bertrand, Valérian Lupo et Charlotte Balent. Elle s'appuie donc sur les cahiers de labo de ces personnes et sur mon exploration de leurs arborescences de fichiers. Les différentes tâches réalisées par Amandine Bertrand ont été difficiles à reconstruire, et plusieurs fichiers ont dû être nettoyés avant utilisation. Aussi, cette section reprend une explication globale du projet dans l'état dans lequel il a été repris. Toutes les productions précédentes n'ont pas été utilisées lors de ce mémoire. Une représentation schématique de ce qui a été utile à la réalisation du mémoire se trouve en début de section « Résultats », à la figure 6.

3.2.1. Collecte des données

Les données de ce travail proviennent du projet *MetaRhizoMet+* entrepris par Amandine Bertrand lors de son doctorat. Dix sites à *Arabidopsis halleri* se trouvant en Allemagne ont été échantillonnés. Parmi ces dix sites, cinq sont pollués aux métaux et cinq ne le sont pas. Dans chacun de ces sites, un ou plusieurs carrés d'échantillonnage (au total quarante-trois) ont été choisis et cinq à dix plantes ont été échantillonnées au niveau de leurs organes aériens, de leur rhizosphère, d'un bloc de terre autour de la rhizosphère et d'un bloc de terre plus profond. Au sein d'un même carré, les échantillons de chaque type de sous-échantillon (rhizosphère, bloc de terre 1, …) sont analysés en une fois en tant qu'un « pool » d'échantillons. Ainsi, chaque carré comporte 4 sous-échantillons portant les noms *bulk1*, *bulk2*, *rhiz* ou *root* et shoot. Une liste détaillée des sites, de leur localisation, ainsi qu'une carte d'échantillonnage se trouvent en annexe 1. La stratégie d'échantillonnage est schématisée à la figure 4.

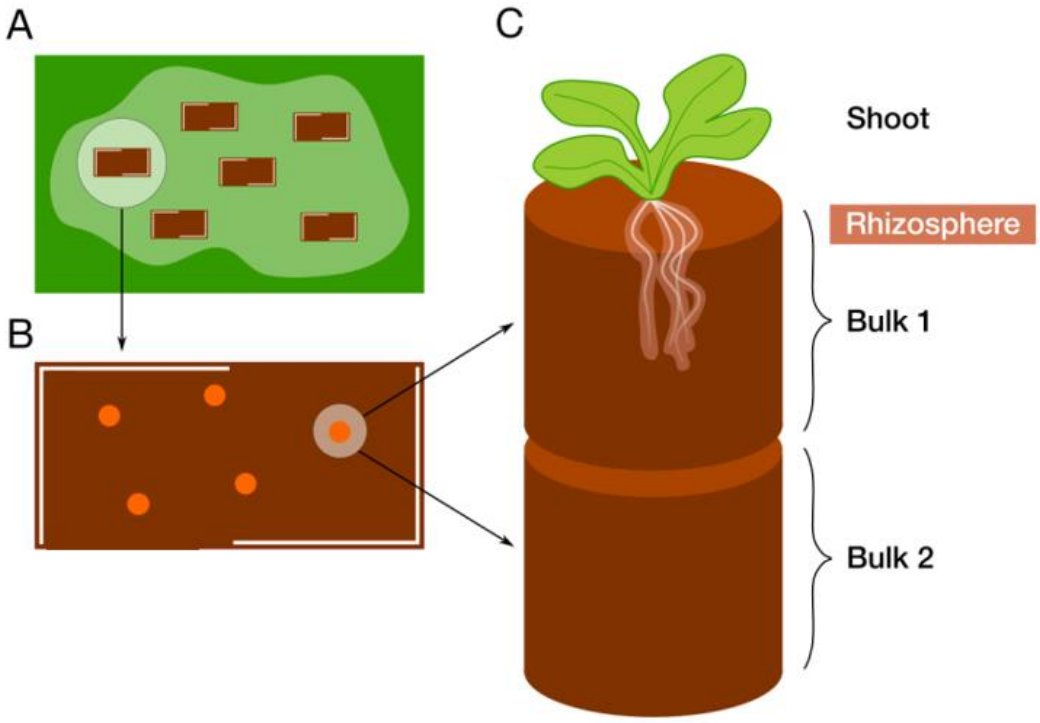


Figure 4 : Stratégie d'échantillonnage. A : Dans chaque site, un ou plusieurs rectangles de 1x2m, appelés carrés d'échantillonnage ont été tracés. B : Cinq à dix plantes sont échantillonnés par carré. C : Chaque plante est échantillonnée à quatre endroits différents. Tous les sous-échantillons provenant d'un même carré sont regroupés en un pool par sous-échantillon (Shoot / Rhizosphere / Bulk1 / Bulk2).

D'autres carrés ont été choisis dans ces mêmes sites afin d'en échantillonner le sol en tant que contrôle négatif dans l'analyse des paramètres physico-chimiques. Cette analyse de paramètres physico-chimiques comprend une mesure du pH, un dosage du phosphore, de l'azote et de la matière organique, ainsi qu'une analyse par ICP-AES (Inductively Coupled Plasma – Atomic Emission Spectroscopy) du contenu (total, échangeable et extractible) en différents métaux de chacun des échantillons. La texture du sol, la minéralisation de l'azote, la respiration microbienne et la biomasse microbienne ont également été mesurés.

Une analyse métagénétique basée sur la sous-unité 16S de l'ARN ribosomique a également été conduite dans le but d'analyser la taxonomie des communautés microbiennes et de corréler ces données aux métadonnées physico-chimiques des sites et des carrés échantillonnés.

Étant donné l'objectif du projet, soit étudier la distribution de différentes familles de transporteurs de métaux chez les communautés microbiennes des rhizosphères d'*A. halleri*, une extraction d'ADN a été réalisée sur les différents échantillons, suivie d'un séquençage shotgun par la technologie de séquençage Illumina paired-end. Le séquençage a été réalisé à l'aide d'un séquenceur NextSeq500 du GIGA à l'ULiège pour donner des *reads* de 75 paires de bases. Ces données sont ainsi contenues dans 172 fichiers .fastq différents, un fichier dans chaque sens de lecture pour les sous-échantillons *rhiz* (rhizosphère) et *bulk1* (bloc de terre autour de la rhizosphère) pour chacun des 43 carrés échantillonnés. Ces données de séquençage brutes ont ensuite été prétraitées par Valérian Lupo, alors doctorant au sein du même laboratoire, l'unité de Phylogénomique des Eucaryotes de Denis Baurain.

3.2.2. Traitement des données brutes de séquençage

La première étape du traitement des données brutes consiste à contrôler la qualité des *reads* et les filtrer, mais également à les raccourcir pour enlever les adaptateurs et les bases de mauvaises qualité. Le nettoyage des *reads* a été réalisé à l'aide de l'outil fastp (v0.19.6). Brièvement, les extrémités des *reads* comprenant les adaptateurs ayant servi au séquençage Illumina, ainsi que quelques bases de mauvaise qualité de séquençage ont été retirées. Le contrôle de la qualité des *reads* a été réalisé avec l'outil FastQC (v0.11.5). Cet outil d'analyse produit des rapports renseignant par exemple la qualité de séquençage moyenne à chaque position (base) des *reads*, la proportion de bases non-déterminées à chaque position ou encore le niveau de duplication des séquences (dans quelle mesure chaque *read* est présent en multiples copies).

Des métagénomes correspondants à chaque échantillon (site x rhiz/bulk1) ont alors été assemblés à partir des *reads* nettoyés par le programme SPAdes (v3.10.1). Ce programme est basé sur la construction de graphes de De Bruijn et l'utilisation de k-bimer, c'est-à-dire des k-mers extraits de paires de *reads* et dont la distance est estimée sur base de leurs chemins respectifs dans les graphes de De Bruijn. À l'aide de ces deux jeux d'informations, SPAdes reconstitue des morceaux de génomes et assemble donc les *reads* en *contigs*. Ces *contigs* sont alors reliés entre eux pour former des *scaffolds*. Ce sont ces *scaffolds*, sous forme de fichiers .fasta que j'ai utilisés pour réaliser la manipulation principale de ce mémoire, à savoir l'enrichissement d'arbres de références représentant des familles de transporteurs de métaux par le programme Forty-Two à partir de ces données de séquençage. Ces assemblages ont néanmoins été manipulés et analysés davantage au préalable.

Une tentative de partitionnement des *scaffolds* en bins, ou binning a ensuite été réalisée, d'une part avec le programme MetaBAT2 (v2.12.1), qui se concentre sur l'assemblage de génomes d'organismes procaryotes, et d'autre part avec le programme CONCOCT (v1.1.0), qui assemble des génomes

procaryotes et eucaryotes. Ces deux programmes ont été utilisés en parallèles sur les mêmes données et non en série. Pour mener à bien ce binning, les deux programmes nécessitent des *scaffolds* triés et organisés selon une carte.' Le programme BaMM (v1.7.3) organise et place les *reads* sur les *scaffolds* pour connaître leur couverture respective et permettre à CONCOCT et MetaBAT2 d'utiliser ces informations pour déduire les multiples organismes en jeu, à des abondances différentes. Ces programmes réalisent un assemblage des *scaffolds* disponibles et organisés en organismes ou bins (ceux-ci ne correspondent pas forcément exactement à des génomes complets d'organismes réels).

Divers outils ont alors été utilisés pour analyser la qualité de ces assemblages finaux (MAGs pour Metagenome-Assembled Genomes). Des données statistiques ont été générés pour chaque MAG par le programme QUAST (v5.0.2) ou Quality Assessment Tool for Genome Assemblies. QUAST nous donne notamment la taille de chaque MAG, la longueur médiane des *scaffolds* composant chaque MAG, mais également le pourcentage de GC de ces MAGs. L'information de couverture de séquençage médiane de chaque base des bins constitue une donnée intéressante, liée à l'abondance de l'organisme concerné. Cette donnée nous est fournie dans des fichiers annexes produits par CONCOCT et MetaBat2, bien qu'il faille utiliser le programme perl get_cov.pl pour récupérer cette donnée des fichiers de sortie de MetaBat2. La complétude des génomes ainsi formés et leur niveau de contamination est ensuite évaluée à l'aide des outils EukCC (v0.3) et CheckM (v1.0.7). Là où le second est adapté à l'analyse de génomes des trois domaines du vivant, le premier est destiné au contrôle qualité de génomes eucaryotes, dont il peut également donner la lignée. Enfin, la taxonomie des lignées bactériennes a été déterminée à l'aide de la base de données GTDB ou Genome Taxonomy Database. Il s'agit d'une base de données de taxonomie bactérienne de référence basée sur une approche phylogénomique et utilisée depuis 2019 par le Bergey's Manual Trust. Ces données caractérisant la qualité des assemblages sont rassemblées dans des tableaux correspondant à chaque échantillon séquencé.

Lors des différentes étapes d'assemblage, des pertes sont observées. Les bins assemblés ne comportent qu'une petite partie des données de départs. Certains *reads* sont écartés en raison de leur faible qualité, d'autres sont perdus à l'assemblage de *contigs* qui eux-mêmes sont parfois inutilisés lors de la constitution des MAGs. Pour comprendre l'étendue de ce qui a été écarté à cette dernière étape et a atterri dans ce qu'on peut appeler la « poubelle », Charlotte Balent a mené une analyse taxonomique quantitative de cette poubelle dans le but de choisir les échantillons les plus intéressants à refaire séquencer par la technologie Nanopore et ainsi obtenir des génomes complets de microorganismes inconnus.

Premièrement, la proportion de *contigs* utilisés pour la formation de bins ainsi que la proportion de *reads* formant des *contigs* jetés à la poubelle et des *contigs* assemblés en bins ont été calculées grâce aux scripts perl get_cov_all.pl et bin_get_cov_all.pl conçus par Valérian Lupo.

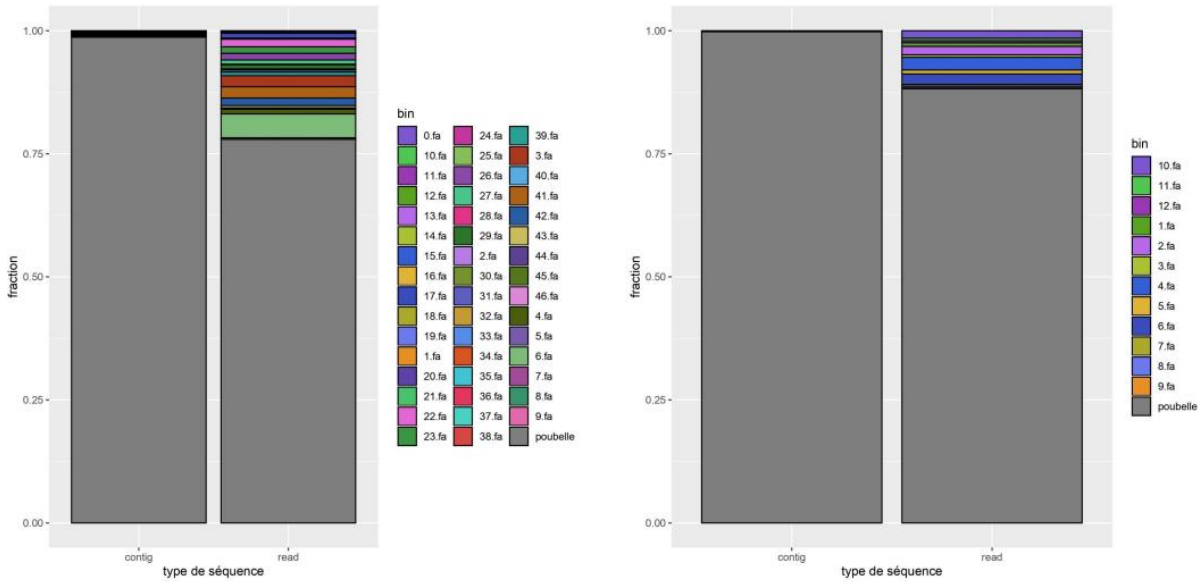


Figure 5 : Proportions de reads et contigs -poubelles de l'échantillon MGA18_rhiz. Bins assemblés à l'aide de CONCOCT à gauche et de MetaBAT2 à droite

Ces graphiques générés sous R par Charlotte Balent montrent une proportion écrasante de *contigs* écartés de la formation de bins et « jetés à la poubelle » (figure 5). Cependant, les *contigs* repris dans ces bins représentent une fraction de *reads* minoritaire, mais non négligeable. Il est alors probable que de nombreux *contigs* repris dans la poubelle soient issus de l'assemblage de peu de *reads*.

Ensuite, les programmes diamond (v0.9.30.131) et physeter.pl (v0.213470) ont permis d'analyser grossièrement la taxonomie des *contigs* jetés lors du regroupement en bins. Cette étape a servi à évaluer la proportion de *contigs*-poubelles affiliés à chacun des trois domaines du vivant. Environ 62% des *contigs*-poubelles sont restés non identifiés alors qu'environ 23% ont été identifiés comme bactériens, 11% eucaryotiques et 3% archéens. Enfin, les *reads* ayant passé les contrôle qualité initiaux (fastp) mais ayant fini par être écartés de la formation de *contigs* ont été analysés à l'aide d'une base de données construite par Kraken2. Ce programme se base sur la proportion en k-mer des différents *reads* pour les associer à un organisme.

3.2.3. Création d'alignements multiples de référence

Dans une autre tâche de son projet, Amandine Bertrand s'est également attelée à la sélection de familles de transporteurs de métaux, qui pourraient être impliqués dans la composante microbienne de l'adaptation d'*Arabidopsis halleri* aux sols pollués aux métaux. La littérature a été analysée et 21 familles ont été retenues. Une liste des familles initialement retenues est reprise en annexe 2. Dans la littérature liée à ces familles, plusieurs protéines de référence, issues d'organismes modèles, citées et décrites dans certains articles ont été choisies dans le but de constituer des ensembles de protéines bien décrites. La liste des protéines retenues pour chaque famille figure en annexe 3.

Ces familles ont servi à la création d'alignements multiples, enrichis par plusieurs méthodes. Mais avant cela, une révision des 21 groupes était nécessaire pour deux raisons. Premièrement, certaines familles de gènes ou de protéines décrites dans la littérature ne sont pas forcément des familles au sens phylogénétique, mais plutôt au sens fonctionnel. Les familles fonctionnelles de protéines peuvent alors être constituées de protéines paralogues ou même analogues, or nous cherchons à construire

des alignements multiples de référence de protéines orthologues. Deuxièmement, certaines familles multigéniques comprennent plusieurs groupes de gènes distincts, qu'il serait préférable d'utiliser et d'étudier séparément, aussi bien pour des raisons de temps de calcul et de lisibilité lors de l'analyse que pour la pertinence des interprétations. 135 protéines réparties dans 21 familles constituent le point de départ de cette étape. Les 135 séquences ont été réunies dans un seul fichier, puis séparées en clusters à l'aide de l'outil CD-HIT (v4.6), paramétrée à 40% de seuil d'identité global. Les 99 clusters ainsi formés sont ensuite regroupés en plus gros clusters grâce à une comparaison des sorties réciproques de hmmscan pour les clusters comprenant plusieurs séquences et de phmmer pour les clusters représentés par un singleton (HMMER v3.3). Les séquences récoltées par les différents profils ou singleton ont été comparées et les 99 clusters ont été fusionnés en 34 clusters, reprenant les 135 séquences initiales. La liste des clusters figure en annexe 4. Un exemple simple est celui de la famille nramp. Les 10 séquences nramp initiales sont d'abord réparties en un cluster de 2 séquences, un de 3 séquences et un de 5 séquences par CD-HIT. Des profils HMM sont construits pour ces trois clusters qui chacun retrouvent toutes les séquences nramp parmi les 135 séquences initiales avec hmmscan. C'est pourquoi un cluster unique reprenant ces 10 séquences est finalement constitué.

Chacun des 34 clusters a ensuite été enrichi. Cet enrichissement permet de passer de maximum une dizaine de séquences de référence issues d'organismes modèles à un alignement multiple de quelques centaines de séquences provenant potentiellement d'un ensemble d'organismes représentatifs de la diversité du Vivant. Cet enrichissement s'est fait en deux étapes.

La première étape d'enrichissement a été réalisée avec le programme Forty-Two.pl. Forty-Two est un programme qui permet de rajouter à un alignement multiple préexistant des séquences orthologues depuis une base de données. Une liste de *queries* comprenant des organismes fortement représentés dans les alignements multiples est donnée. De cette liste découle une liste de *query_seqs* pour chaque alignement multiple. Le principe fonctionnel central de Forty-Two est de chercher, par BLAST, des séquences homologues à cette liste de *query_seqs* dans les banques de données candidates. Une liste de séquences homologues est alors produite. Lorsque le contrôle d'orthologie par une banque de données de référence est activé, Forty-Two cherche le « *Best Hit* » (séquence ressemblant le plus) de chaque *query_seq* dans la base de données de référence. Une liste de *Best hits* est alors générée. Les séquences homologues sont également « blastées » contre les organismes de référence et génèrent une autre liste de *Best hits*. Le contrôle par triangulation peut alors s'opérer : les *Best Hits* générés par les séquences homologues dans les organismes de référence doivent se retrouver dans les *Best hits* générés par les queries. Dans ce cas, Forty-Two considère ces séquences comme orthologues et l'ajoute à l'alignement multiple. Les 34 clusters ont été enrichis par des séquences d'organismes provenant de la banque de données « life-tqmd-of73 » assemblées par l'unité de Phylogénomique des Eucaryotes. Cette banque de données comprend 310 protéomes, dont 151 protéomes bactériens (Léonard et al., 2021), 86 protéomes archéens (Léonard et al., 2021) et 73 protéomes eucaryotes (Van Vlierberghe et al., 2021). Elle a été construite de sorte à représenter au mieux toute la diversité du vivant par environ 300 organismes. Chaque organisme est alors présent pour représenter un taxon plus large que la simple espèce. Pour cet enrichissement, le contrôle d'orthologie par des protéomes de références a été activé. Une liste des organismes uniques dont proviennent les 135 séquences de départ a été récupérée à partir des Ids de ces protéines et grâce à la commande perl efetch. Les protéomes de ces organismes ont ensuite été récupéré à partir du NCBI grâce à la commande bash wget. Le nombre d'organismes de référence utilisés pour satisfaire la condition de *Best reciprocal hit* (BRH) est fixé à 1. Ainsi, lorsqu'une protéine homologue est trouvée dans la banque de protéines candidates par Forty-Two, la protéine doit trouver le même *Best hit* que la séquence query dont elle est issue, dans l'organisme de référence correspondant au meilleur *Best hit* de la séquence query. La

liste des organismes sélectionnés pour les queries et pour les protéomes de référence est reprise en annexe 5.

La deuxième étape d'enrichissement a été réalisée à l'aide de la suite HMMER. Les clusters enrichis par Forty-Two ont été alignés par le programme MAFFT (v7.453) puis un profil HMM a été construit grâce à l'outil hmmbuild. Ces profils ont été utilisés par hmmsearch pour sonder la banque de données « life-tqmd-of73 », qui a donc servi comme source de séquences lors des deux étapes d'enrichissement des clusters. L'objectif d'un deuxième enrichissement à partir de la même base de données était de récupérer les éventuels candidats rejetés à tort par le contrôle d'orthologie de Forty-Two. Une partie des séquences trouvées par la suite HMMER a été sélectionnée par le programme ompa-pa.pl (v0.211430) sur base de la longueur des hits, de la couverture du profil, du niveau de paralogie et de la e-value des hits. Cette série de commandes et programmes donne alors un deuxième set de séquences pour chaque cluster.

L'étape d'après a été de réunir, pour chaque cluster, les deux alignements résultant d'une part des séquences de départ et des rajouts de Forty-Two et d'autre part de la collecte grâce à un profil HMM. Pour réaliser cette tâche, le programme two-scalp.pl (v0.231010) a été la solution. Ce programme est conçu pour aligner entre eux deux jeux de séquences déjà alignés ou non. Cette étape permet donc d'aligner le deuxième set de séquences sur le premier, déjà aligné, ce qui est plus efficace et conservatif que de fusionner deux jeux de séquences et de les réaligner de A à Z.

Amandine Bertrand a par la suite construit des arbres RAxML sur base de ces alignements, en prenant d'abord soin de retirer les colonnes présentes que chez un nombre réduit de séquences, puis les séquences trop courtes. Cette filtration est réalisée par le programme ali2phylip.pl (v0.212670). Cet outil réduit séquentiellement le nombre de colonnes puis de séquences d'un alignement multiple sur base de deux valeurs-seuils. D'abord, les colonnes contenant trop de *gaps* sont éliminées, ce qui correspond aux positions manquantes dans la plupart des séquences. Ensuite, les séquences trop courtes sont éliminées. Plus précisément, en appliquant une valeur appelée max (ici 0.3), le programme retire en premier lieu toutes les colonnes représentées par moins qu'un certain nombre de séquences (ici 30%). En spécifiant ensuite une valeur appelée min (ici 0.3), le programme élimine les séquences d'une taille inférieure à une certaine fraction de la plus longue séquence (non-alignée) présente dans l'alignement, après retrait des colonnes superflues (ici 30%). Le programme RAxML (v8.1.17) a ensuite été utilisé pour construire le produit final des familles de référence, des arbres phylogénétiques de protéines. Ces arbres ont été construits par maximum de vraisemblance (ML), selon le modèle PROTGAMMALGF et avec un *bootstrap* donnant la robustesse des groupes sur 100 itérations de la construction de l'arbre. Ces arbres ont néanmoins été reconstruits lors de ce mémoire.

Lorsque j'ai repris le projet, je me suis principalement servi des alignements de référence. Les arbres produits par Amandine Bertrand ont uniquement servi à l'exploration des données et une partie du prototypage.

3.3. Standardisation des données

La première partie du travail d'analyse des données de séquençage du sol consiste à homogénéiser ces données, reprises au stade d'assemblage en *scaffolds*, avant *binning* donc, puisque cette étape place une majorité de *scaffolds* dans la « poubelle ». En effet, l'analyse a une visée comparative, il est alors important de rendre les données comparables, ce qui pourrait se faire soit en filtrant et sous-échantillonnant les échantillons, soit en normalisant les résultats d'une façon ou d'une autre. Ici, pour

permettre une interprétation potentiellement visuelle des analyses envisagées, c'est la première approche qui a été retenue. Pour ce faire, les échantillons les plus différents parmi les 86 échantillons, que l'on peut qualifier d' « *outliers* », ont été écartés et le volume des données par catégorie d'échantillon (Sol métallifère, sol ; Sol métallifère, rhizosphère ; Sol non-métallifère, sol ; Sol non-métallifère, rhizosphère) a été homogénéisé tout en prêtant une attention particulière à la conservation des différents sites géographiques, ainsi qu'au nombre de réplicas par site géographique, dans le but d'éviter un effet de zone, l'influence d'une population particulière sur les interprétations.

3.3.1. Caractérisation des échantillons

Premièrement, les données concernant la taille des *scaffolds* ont été récupérées des différents fichiers de séquençage des échantillons. Dans ces fichiers, chaque scaffold possède une première ligne descriptive, typique des fichiers FASTA et contenant les informations de la séquence. Ici, cette ligne comprend un nom sous la forme « NODE_X », X représentant le numéro du nœud, de la séquence. Cette ligne contient également la taille du *scaffold* en paires de bases (bp) et la couverture de séquençage du scaffold. Les tailles de chaque *scaffold*, pour chaque échantillon, ont été récupérées et stockées dans un fichier de deux colonnes contenant le numéro de l'échantillon de 1 à 86 et la taille des *scaffolds* en bp grâce au job suivant.

```
#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q *q

#$ -N grep_scaffolds_size

grep \> /media/vol2/scratch/vlupo/metagenomic/meta/*/scaffolds.fasta | \
  perl -nle 's/.+\/meta\/([0-9]+)_.+length_([0-9]+)_.+/$1\t$2/ ; print' \
  > scaffolds_size_all.tsv
```

Le volume de données imposant rend tous les calculs sous R assez lents, ce qui complique l'analyse. Un sous-échantillonnage aléatoire est alors effectué dans le but de réduire d'un facteur 100 la taille des données et de fluidifier les analyses, grâce à la ligne de commande suivante.

```
Perl -nle 'print if int(rand(100)) < 1' scaffolds_size_all.tsv \
  > scaffolds_size_all_r100.tsv
```

Une analyse comparative des échantillons entre eux, sur base de leurs distributions de la taille des *scaffolds* est ensuite conduite sous R afin de déterminer les échantillons les plus différents. Ces analyses ont également été menées sur les tailles des *contigs* ayant servi à assembler les *scaffolds*. Les résultats de ces dernières diffèrent peu de ce qui est obtenu avec les *scaffolds* et se trouvent dès lors en annexe 6. En effet, la standardisation des données a été poursuivie avec les fichiers contenant les *scaffolds* et ce sont ces derniers qui ont été utilisés par la suite dans ce mémoire.

3.3.1.1. Distribution des tailles des scaffolds

Les distributions ont d'abord été visualisées à l'aide d'un graphique de type *boxplot* réalisé avec le code R suivant. Pour chaque échantillon, une boite est générée, les quartiles sont indiqués par les

extrémités de la boite et une bande incluse dans la boite (médiane). L'axe des ordonnées représentant la taille de chaque *scaffold* est représentée en échelle logarithmique pour plus de clarté.

```
Library(ggplot2)

# Import data
scaffolds <- read.csv("scaffolds_size_all_r100.tsv",
  header = FALSE, sep = "\t", col.names = c("Sample", "Scaffold_Size"))
scaffolds$Scaffold_Size <- as.numeric(scaffolds$Scaffold_Size)
scaffolds$Sample <- as.factor(scaffolds$Sample)

# Distribution of scaffolds size : boxplots
output_file <- "scaffolds_boxplots.png"
my_plot <- ggplot(scaffolds, aes(group=Sample, x = Sample, y = Scaffold_Size)) +
  geom_boxplot() +
  scale_y_log10() +
  xlab("Sample") +
  ylab("Scaffold Size") +
  ggtitle("Scaffold Size Distribution") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
ggsave(output_file, my_plot, width = 10, height = 6, dpi = 300)
```

3.3.1.2. Comparaisons multiples

Les différents échantillons sont alors comparés entre eux à l'aide de tests statistiques de Wilcoxon. Le test de Wilcoxon-Mann-Whitney est un test statistique non paramétrique permettant de comparer deux populations par le rang de chacun des individus statistiques. Ce test est adapté à la comparaison des échantillons sur base de la taille des *scaffolds*, dont la distribution n'est pas normale (les distributions se trouvent en annexe 7).

Au vu de la quantité des données, le moindre écart est considéré comme significatif. Cela a pour conséquence de rendre pratiquement toutes les comparaisons deux à deux statistiquement significatives. Pour éviter ces résultats ininterprétables et réduire les effets du volume des données, plusieurs stratégies ont été envisagées et employées. Premièrement, la correction de Bonferroni a été appliquée. Celle-ci consiste à diviser le seuil alpha de significativité par le degré de liberté, soit le nombre de comparaisons effectuées, ici 3655. Malgré cela, aucun *outlier* ne pouvait être discriminé sur base du nombre de tests statistiquement significatifs. Un test de Kolmogorov-Smirnov a été envisagé pour remplacer le test de Wilcoxon. Ce test non paramétrique compare les distributions au lieu de comparer les médianes des deux échantillons testés. Le changement de méthode statistique n'ayant pas été probant, une autre méthode a alors été envisagée. Un sous-échantillonnage aléatoire moins conservateur a alors été appliqué pour réduire la taille des données d'un facteur 1000 au lieu d'un facteur 100. Cette réduction des données et la correction de Bonferroni ont été employés dans la version finale de la mise en évidence des échantillons les plus différents. Les autres résultats obtenus avec les autres méthodes sont repris en annexe 8. Les comparaisons multiples ont été réalisées à l'aide du code R suivant.

```
Library(stats)

# Import data
scaffolds_1000 <- read.csv("scaffolds_size_all_r1000.tsv",
  header = FALSE, sep = "\t", col.names = c("Sample", "Scaffold_Size"))
scaffolds_1000$Scaffold_Size <- as.numeric(scaffolds_1000$Scaffold_Size)
scaffolds_1000$Sample <- as.factor(scaffolds_1000$Sample)
```

```
# Wilcoxon pairwise tests
pairwise_tests_1000 <- combn(levels(scaffolds_1000$Sample), 2, function(x) {
  group1 <- scaffolds_1000$Scaffold_Size[scaffolds_1000$Sample == x[1]]
  group2 <- scaffolds_1000$Scaffold_Size[scaffolds_1000$Sample == x[2]]
  p_value <- wilcox.test(group1, group2)$p.value * (86 * 85) / 2
  c(x[1], x[2], p_value)
}, simplify = TRUE)
pairwise_data_1000 <- data.frame(t(pairwise_tests_1000), stringsAsFactors = FALSE)
```

Ensuite, un rapport textuel au format markdown des résultats des comparaisons multiples a été produit grâce au code R suivant, dans le but de consigner les résultats sous une forme traitable dans le shell.

```
Library(knitr)

# Markdown report
report_1000 <- c(
  "# Statistical Analyses of Scaffolds (1000)",
  "",
  "## Pairwise Wilcoxon Rank-Sum Tests:",
  paste("Pairwise Wilcoxon rank-sum tests were performed to compare the scaffold s
izes between all pairs of samples."),
  "",
  "```{r, results='asis'}",
  kable(pairwise_data_1000,
  caption = "P-values for pairwise Wilcoxon rank-sum tests"),
  "```"
)
writeLines(report_1000, "scaffolds_1000_statistical_analyses.md")
```

Enfin, une représentation visuelle des 3655 comparaisons a été générée sous R, à l'aide du code suivant. Il s'agit d'un graphique représentant chaque échantillon contre chaque échantillon et affichant un carré rouge lorsque le test est statistiquement significatif, selon les p-values corrigées, ou un point noir lorsqu'il ne l'est pas.

```
Library(ggplot2)

# Pairwise comparisons dot matrix
pairwise_comp_df_1000 <- data.frame(pairwise_data_1000)
colnames(pairwise_comp_df_1000) <- c("Sample_1", "Sample_2", "p_value")

pairwise_comp_df_1000$p_value <- as.numeric(
  pairwise_comp_df_1000$p_value)
pairwise_comp_df_1000$Sample_1 <- factor(
  pairwise_comp_df_1000$Sample_1, levels = 1:86)
pairwise_comp_df_1000$Sample_2 <- factor(
  pairwise_comp_df_1000$Sample_2, levels = 1:86)

scatter_plot_1000 <- ggplot(
  pairwise_comp_df_1000, aes(x = Sample_1, y = Sample_2)) +
  geom_tile(
       data = pairwise_comp_df_1000[pairwise_comp_df_1000$p_value < 0.05, ],
       fill = "red",
       alpha = 0.5,
       width = 0.9,
       height = 0.9) +
  geom_point(
```

```
        data = pairwise_comp_df_1000[pairwise_comp_df_1000$p_value >= 0.05, ],
        color = "black",
        size = 2,
        shape = 16) +
  scale_x_discrete(expand = c(0, 0.5),
    limits = unique(pairwise_comp_df_1000$Sample_1)) +
  scale_y_discrete(expand = c(0, 0.5),
    limits = rev(unique(pairwise_comp_df_1000$Sample_2))) +
  annotate("text",
           x = 0.5,
           y = 0.5,
           label = « Significant Comparisons »,
           color = "red",
           fontface = "bold",
           size = 4,
           hjust = 0.5,
           vjust = 0.5) +
  labs(x = "Sample 1", y = "Sample 2", title = "Sample Comparisons (1000)") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  coord_cartesian(xlim = c(1, 86), ylim = c(1, 86))
ggsave("sample_scaffolds_1000_comparisons_plot.png",
  plot = scatter_plot_1000, width = 8, height = 6, dpi = 300)
```

3.3.2. Retrait des outliers et sélection des échantillons

3.3.2.1. Identification et retrait des outliers

Afin de pouvoir écarter les échantillons les plus différents, les tests de Wilcoxon statistiquement significatifs calculés lors des comparaisons multiples ont été dénombrés, par échantillon puis associés aux informations de site de chaque échantillon.

```
# Compter les différences statistiquement significatives pour chaque sample
grep -P « \|[0-9]+\. » Scaffolds_1000_statistical_analyses.md | \
  perl -nle 's/\|([0-9]+)\s+\|([0-9]+)\s+\|([0-9\.e\-]+)\s+\|/$1\t$2\t$3/ ;
             print if $3 < 0.05' | \
  cut -f1,2 > significant_differences_list

for f in $(cat significant_differences_list); do \
  echo $f; done | sort | uniq -c | \
  perl -nle 's/\s+([0-9]+)\s+([0-9]+)/$2\t$1/; print' > SD_count.tsv

# Fusionner avec les informations de site de chaque sample
sort -n /media/vol2/home/lsmacchia/cp_6-twoscalp/prototype_42/nramp_42/scaffolds/i
nfo_samples_novaseq_germany_formated.tab | grep -v \# \
  > info_samples_novaseq_germany_formated.tab
sort -n SD_count.tsv > SD_count_sorted.tsv
paste SD_count_sorted.tsv info_samples_novaseq_germany_formated.tab | \
  perl -nle 's/([0-9]+)\s+([0-9]+)\s+[0-9]+\s+(.+)/$1\t$2\t$3/; print' | \
  sort -n -k2 > samples_counts_info
```

Dans l'optique de retirer les échantillons les plus différents, une distribution cumulative du nombre de différences statistiquement significatives a ensuite été calculée. Sur base de cette distribution, des *breakpoints* ont été déterminés à l'aide de la fonction du même nom provenant du package R strucchange. Cette fonction utilise l'algorithme RRT (Recursive Residuals-based Test) qui détermine

par des tests statistiques où couper une distribution pour qu'aucun changement structurel ne soit observé au sein de chaque segment. Grâce à ces coupures, un choix peut être fait sur le nombre d'*outliers* à retirer. Le code R suivant a permis de produire une représentation visuelle de ces *breakpoints*. La distribution non-cumulative du nombre de différences significatives se trouve en annexe 9.

```
Library(ggplot2)
library(strucchange)

# Import data
sd_counts <- read.csv("SD_count.tsv",
  header = FALSE, sep = "\t", col.names = c("Sample", "Significant_Tests"))

# Sort the data frame by the number of significant differences in ascending order
sd_counts_sorted_increasing <- sd_counts[order(
  sd_counts$Significant_Tests, decreasing = FALSE), ]

# Convert Sample column to factor with desired order
sd_counts_sorted_increasing$Sample <- factor(
  sd_counts_sorted_increasing$Sample, levels = sd_counts_sorted_increasing$Sample)

# Create a cumulative count column
sd_counts_sorted_increasing$Cumulative_Counts <- cumsum(
  sd_counts_sorted_increasing$Significant_Tests)

# Perform change point detection using the CUSUM test with a smaller h value
cpt <- breakpoints(
  Significant_Tests ~ 1, data = sd_counts_sorted_increasing, h = 0.1)

# Create the cumulative distribution plot with the estimated breakpoint
plot <- ggplot(
  sd_counts_sorted_increasing, aes(x = Sample, y = Cumulative_Counts)) +
  geom_point(color = "blue", size = 3) +
  geom_line(aes(group = 1), color = "blue") +
  geom_vline(xintercept = cpt$breakpoints, color = "red") +
  labs(x = "Sample ID", y = "Cumulative Counts of Significant Differences") +
  ggtitle("Cumulative Distribution of Significant Differences") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))

# Save the plot as an image
ggsave("cumulative_distribution_breakpoint.png",
  plot = plot, width = 8, height = 6, dpi = 300)
```

Une nouvelle liste des échantillons est alors générée par un simple « egrep -v » précisant les échantillons outliers à ne pas reprendre.

3.3.2.2. Sélection des échantillons

Les quatre catégories d'échantillons (M-Bulk1, M-rhiz, N-Bulk1, N-rhiz) ne comportent pas le même nombre d'échantillons au départ (23 pour M-Bulk1 et M-rhiz ; 20 pour N-Bulk1 et N-rhiz). Le retrait des *outliers* a également contribué à créer des différences dans le nombre d'échantillons par catégorie. Il a donc fallu uniformiser le nombre d'échantillons par catégorie, en sélectionnant des échantillons à retirer pour les catégories surreprésentées, ce qui a été fait en plusieurs étapes. Premièrement, les échantillons contenant le moins de *scaffolds* ont été retirés, jusqu'à rencontrer un échantillon à ne pas

retirer, parce qu'il n'appartient pas à une catégorie surreprésentée, ou parce qu'il réduit le nombre de réplicas d'un site en particulier à 1, ou à 0. En effet, lors de l'étape suivante, les *scaffolds* ont été sous-échantillonnés aléatoirement au sein de chaque échantillon dans le but d'avoir un nombre égal de *scaffolds* par échantillon. Retirer les échantillons comportant le moins grand nombre de *scaffolds* prend alors tout son sens : éviter de perdre trop de données. Ensuite, ce sont les échantillons les plus différents, selon les mêmes critères que pour déterminer et écarter les *outliers*, dans les catégories surreprésentées, excepté ceux qui réduiraient le nombre de réplicas d'un site en particulier à 1, ou à 0, qui sont retirés.

Ces manipulations, bien que conceptuellement un peu subtiles, se réalisent avec de simples lignes de commandes utilisant les commandes bash « sort », « uniq -c » et « egrep -v ». Le nombre de *scaffolds* est compté pour chaque échantillon grâce aux deux premières commandes, les échantillons les plus différents ont déjà été déterminée et la sélection se fait en écartant les échantillons grâce à la troisième commande. Le code bash plus détaillé de cette étape se trouve en annexe 10.

3.3.3. Volume des données

Dans cette dernière étape de standardisation des données, chaque échantillon a été sous-échantillonné de manière aléatoire sur base du rapport entre le nombre de *scaffolds* qu'il contient et le nombre de *scaffolds* du plus petit échantillon. Cette fois-ci, le concept est direct, mais les commandes utilisées sont un peu plus subtiles. Pour commencer, les fichiers FASTA contenant les *scaffolds* de tous les échantillons ont été récupérés par des liens symboliques, renommés (ils ont tous le même nom au départ) et les échantillons précédemment écartés sont rejetés. Par exemple, un fichier scaffolds.fasta est alors renommé Environmental_sample_MGA11_Bulk1.fasta. Ceci se fait en ligne de commande dans le shell bash.

```
For i in {1..58..1}; do ln -s /media/vol2/scratch/vlupo/metagenomic/meta/${i}_NGS2
0-Q0$(($i+41))_BHVC5VDSXY_S$(($i+877))_L004/scaffolds.fasta \
  scaffolds_$i.fasta; done
for i in {59..86..1}; do ln -s /media/vol2/scratch/vlupo/metagenomic/meta/${i}_NGS
20-Q$(($i+41))_BHVC5VDSXY_S$(($i+877))_L004/scaffolds.fasta \
  scaffolds_$i.fasta; done

for i in {1..86..1}; do mv scaffolds_$i.fasta \
  Environmental_sample_$(cat info_samples_novaseq_germany_formated.tab | \
  grep "^$i " | cut -d' ' -f2); done

for f in $(cat selection_list); do mv Environmental_sample_$f temp_$f; done
rm -f Environmental_sample_*
for f in $(cat selection_list); do mv temp_$f Environmental_sample_$f.fasta; done
```

Ensuite vient la partie de sous-échantillonnage à proprement parler. Les fichiers FASTA sont convertis au format ALI grâce au programme fasta2ali.pl. Ce format permet de stocker chaque séquence en une ligne descriptive et une ligne contenant la séquence. Trois manipulations sont ensuite séquentiellement appliquées à chaque fichier grâce à un script généré par une boucle for : 1) les deux lignes de chaque séquence sont rassemblées en une ligne, séparées via une tabulation ; 2) un nombre compris entre 0 et le nombre de *scaffolds* de l'échantillon est généré aléatoirement pour chaque séquence, s'il est plus petit que le nombre de *scaffolds* du plus petit échantillon, la séquence est gardée ; 3) les deux lignes du format ALI sont de nouveau séparées.

```
# scaffolds counts
for f in $(cat selection_list); do \
  grep $f samples_scaffolds-counts_info.tsv; done | \
  sort -k2 -rn > selected-samples_scaffolds-counts_info.tsv

# subsampling
for f in $(cat selection_list); do \
  echo "grep -v \# Environmental_sample_$f.ali | \
        perl -ne 's/(>.*)\n/\$1\t/; print' | \
        perl -nle 'print if
                   int(rand(
                   $(grep $f selected-samples_scaffolds-counts_info.tsv | \
                     cut -f2)))
                   < $(tail -n1 selected-samples_scaffolds-counts_info.tsv | \
                       cut -f2)' | \
        perl -nle 's/(>.*)\t/\$1\n/; print' \
        > Environmental_sample_${f}_subsampled.ali"; \
done > subsampling.sh
cat subsampling.sh | sh
```

Les fichiers ALI sont alors reconvertis au format FASTA grâce au programme ali2fasta.pl (v0.212670).

3.4. Criblage des clusters prometteurs

Cette étape de criblage est l'une des plus importante de ce mémoire. Elle consiste en l'application de la méthode de rajout des données métagénomiques du sol sur des alignements de référence et de l'analyse de ces rajouts mise au point pendant la majeure partie du mémoire. Brièvement, des alignements de référence sont produits sur base de la sélection et de l'enrichissement opérés par Amandine Bertrand. Des arbres de référence sont ensuite produits par ML. Les données de séquençage du sol sont rajoutées séparément par catégorie d'échantillon par recherche d'orthologie au sein des alignements de référence. Ces rajouts sont ensuite placés dans les arbres de référence. Finalement ces placements sont analysés par analyse en composantes principales et des observations quant aux différences entre rajouts depuis les quatre catégories d'échantillons dans chaque set de protéines de références sont réalisées pour déterminer les familles les plus intéressantes.

3.4.1. Production d'alignements de référence

Premièrement, les alignements de référence d'Amandine Bertrand ont été nettoyés à l'aide du programme cdhit-tax-filter.pl. Cet outil permet de rassembler en un cluster plusieurs séquences suffisamment similaires. Ici, le seuil d'identité a été défini à 100%, ce qui permet d'éliminer tous les doublons, soient les protéines provenant simultanément des deux méthodes d'enrichissement, et de repartir d'une liste de séquences uniques.

```
#!/bin/bash

#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q fatnodes.q
```

```
#$ -t 1-34
#$ -tc 34

#$ -N cdhit-tax-filter

parameters=`sed -n "${SGE_TASK_ID} p" file_list`
parameterArray=($parameters)
x=${parameterArray[0]}

perl -I /media/vol2/home/vlupo/develop/bio-must-drivers/lib/ \
/media/vol2/home/vlupo/develop/bio-must-drivers/bin/cdhit-tax-filter.pl \
--taxdir=/media/vol1/databases/taxdump-20210216/ \
--filter=filter.idl \
--identity=1 \
--out=-uniq ${x}
```

Ensuite, ces alignements multiples ont été réalignés à l'aide du programme MAFFT. L'algorithme utilisé est L-INS-I. Cette méthode d'alignement est plus gourmande en calcul, mais bien plus précise que l'heuristique par défaut de MAFFT, ce qui permet de générer des alignements de référence de bonne qualité, au prix de quelques jours de calcul.

```
#!/bin/bash

#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q fatnodes.q

#$ -t 1-34
#$ -tc 34

#$ -N mafft

parameters=`sed -n "${SGE_TASK_ID} p" file_list`
parameterArray=($parameters)
x=${parameterArray[0]}

linsi ${x} > $(echo ${x} | cut -d'.' -f1)-linsi.fasta
```

Ces alignements de référence ont par la suite été utilisés deux fois. Ils ont servi à la production d'arbres de référence, mais également de base pour le rajout des données de séquençage du sol.

3.4.2. Inférence d'arbres de référence

À partir des alignements de référence, des arbres de référence ont été produits. Les alignements multiples ont d'abord été filtrés à l'aide du programme ali2phylip.pl. Lors de la construction d'un arbre phylogénétique, une séquence trop petite sera probablement mal placée. Celle-ci est trop peu décrite par son nombre de bp ou d'acides aminés. La réduction de ces alignements multiples est réalisée à l'aide du job suivant.

```
#!/bin/bash

#$ -S /bin/bash
#$ -V
```

```
#$ -cwd

#$ -q fatnodes.q

#$ -t 1-34
#$ -tc 34

#$ -N a2p

parameters=`sed -n "${SGE_TASK_ID} p" file_list`
parameterArray=($parameters)
x=${parameterArray[0]}

ali2phylip.pl --max=0.3 --min=0.3 --p80 ${x}
```

Les arbres de maximum de vraisemblance ont alors pu être construits grâce à la suite RAxML. Le programme phy2raxml.pl a permis de générer un job personnalisé, ainsi qu'une commande de soumission de job pour chaque alignement réduit grâce à la ligne de commande suivante.

```
phy2raxml.pl \
--model=PROTGAMMALG4X *.p80 \
--nomemspec \
--pthreads=auto \
--queue=*q \
> qsub_list.sh

cat qsub_list.sh | sh
```

Un exemple de job de construction d'un arbre de maximum de vraisemblance, ainsi que de la ligne de commande servant à la soumission du job sont repris ci-dessous.

```
## ML-tree job

#!/bin/bash

#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q *q

#$ -N nramp-P38925-42-mafft-aligned-long-uniq-linsi-RAXML-PROTGAMMALG4X-100xRAPIDB
P

# -x -f a -N 100 will do a 100x rapid bootstrap analysis
raxmlHPC-PTHREADS-AVX -T 9 -s nramp-P38925-42-mafft-aligned-long-uniq-linsi.p80 \
-n nramp-P38925-42-mafft-aligned-long-uniq-linsi-RAXML-PROTGAMMALG4X-100xRAPIDBP \
-m PROTGAMMALG4X -N 100 -f a -x 1975021703574 -p 1975021703574


## Job submission
qsub -pe snode 9 \
  nramp-P38925-42-mafft-aligned-long-uniq-linsi-RAXML-PROTGAMMALG4X-100xRAPIDBP.sh
```

3.4.3. Insertion des données de séquençage par recherche d'orthologie

Lors de cette étape cruciale, les données métagénomiques issues des rhizosphères d'*A. halleri* échantillonnées dans 72 sites en Allemagne ont été interrogées par recherche d'orthologie grâce au programme Forty-Two, sur base des alignements multiples de référence précédemment construits. Brièvement, l'utilisation du programme se fait en quatre étapes :

1) La mise en place des banques de séquences candidates.
2) La sélection des *queries*.
3) La mise en place des protéomes de référence pour le contrôle d'orthologie.
4) La création de fichiers de configuration et l'exécution du programme.

Une caractéristique des différents alignements multiples est à prendre en compte au préalable. Ceux-ci peuvent être constitués de protéines issues d'organismes de n'importe laquelle des sept combinaisons possibles des trois domaines du vivant. Cette information est importante, à la fois pour la sélection des *queries* et des organismes de référence. En effet, si les *queries* sont toutes eucaryotes et que l'alignement de comprend que des bactéries, aucune séquence ne sera prise comme *query* et le programme utilisera la séquence la plus longue par défaut. L'effet sur le contrôle d'orthologie est encore plus désastreux : si les protéomes de référence sont eucaryotes, mais qu'une famille de protéines est uniquement bactérienne, les séquences de la famille en question ne trouveront pas de *Best Hit* dans les protéomes de référence et, par conséquent, aucune protéine candidate ne sera considérée comme orthologue. L'alignement multiple associé ne sera alors pas enrichi par Forty-Two.

La composition des alignements multiples a alors été étudiée afin de les répartir dans plusieurs groupes en fonction des domaines représentés et de créer un set de *queries* et de protéomes de référence adapté à chaque combinaison de domaines.

3.4.3.1. Composition des alignements de référence

Premièrement, la taxonomie des différentes séquences est récupérée pour chaque alignement multiple, au niveau du domaine, grâce au programme fetch-tax.pl. Ce programme utilise une liste d'identifiants de séquence (format taxid comprenant l'ID du taxon, format baseid contenant en plus le nom du taxon, format mustid comprenant en plus l'ID de la séquence après un caractère @, …) pour aller rechercher la lignée, ou seulement les niveaux précisés de la taxonomie de l'organisme issue du NCBI. Le code bash suivant prépare les listes d'identifiants.

```
for f in $(ls * | cut -d'@' -f1); do grep \> $f | cut -c2- > ${f}_mustids; done
```

Le job suivant permet quant à lui l'exécution du programme fetch-tax.pl.

```
#!/bin/bash

#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q fatnodes.q

#$ -t 1-33
#$ -tc 33
```

```
#$ -N fetch-tax

parameters=`sed -n "${SGE_TASK_ID} p" file_list`
parameterArray=($parameters)
x=${parameterArray[0]}

fetch-tax.pl \
--taxdir=/media/vol1/databases/taxdump-20210216/ \
--item-type=mustid ${x} \
--levels=superkingdom
```

En vue d'une analyse sous R, un seul tableau de données en deux colonnes contenant le nom du cluster et la taxonomie de chaque organisme unique (domaine) a été généré à partir de tous les fichiers de taxonomie produits par fetch-tax.pl.

```
for f in $(ls *tax | cut -d'-' -f1,2); do \
  cut -f2,4 $f-42-mafft-aligned-long-uniq-linsi.tax | sort | uniq | \
  env file=$f perl -nle 'print "$ENV{file}"."\t"."$_"' | cut -f1,3; \
done > all_clusters_uniqorgs_tax.tsv
```

Un graphique sous la forme d'un *stacked chart* a ensuite été généré sous R. Celui-ci montre pour chaque cluster le partitionnement relatif du nombre d'organismes uniques parmi les trois domaines du vivant. Si un cluster possède au moins 5% d'organismes uniques appartenant à un domaine, il est considéré comme représenté dans ce domaine. Ainsi, si un cluster possède 80% d'eucaryotes, 18% de bactéries et 2% d'archées, il sera considéré comme appartenant au groupe « Bactéries et Eucaryotes ».

```
library(ggplot2)
library(dplyr)
library(forcats)

# Import data
all_clusters_uniqorgs_tax <- read.csv("all_clusters_uniqorgs_tax.tsv",
  sep = "\t", header = FALSE, col.names = c("cluster", "domain"))
all_clusters_uniqorgs_tax$cluster <- as.factor(all_clusters_uniqorgs_tax$cluster)

# Calculate counts of superkingdoms per cluster
all_cluster_uniqorgs_tax_counts <- as.data.frame(
  table(all_clusters_uniqorgs_tax$cluster, all_clusters_uniqorgs_tax$domain))
colnames(all_cluster_uniqorgs_tax_counts) <- c("cluster", "domain", "count")

# Calculate relative proportions and classify clusters based on the key
order_clusters <- all_cluster_uniqorgs_tax_counts %>%
  group_by(cluster) %>%
  summarize(
    rel_proportion_prokaryotes = sum(
      count[domain %in% c("Bacteria", "Archaea")]) / sum(count),
    rel_proportion_bacteria = sum(count[domain == "Bacteria"]) / sum(count),
    rel_proportion_archaea = sum(count[domain == "Archaea"]) / sum(count),
    rel_proportion_eukaryotes = sum(count[domain == "Eukaryota"]) / sum(count)
  ) %>%
  mutate(
    key = as.numeric(case_when(
      rel_proportion_bacteria < 0.05 & rel_proportion_archaea < 0.05 &
      rel_proportion_eukaryotes >= 0.05 ~ 7,
      rel_proportion_bacteria < 0.05 & rel_proportion_archaea >= 0.05 &
      rel_proportion_eukaryotes < 0.05 ~ 6,
      rel_proportion_bacteria >= 0.05 & rel_proportion_archaea < 0.05 &
```

```
      rel_proportion_eukaryotes < 0.05 ~ 5,
      rel_proportion_bacteria >= 0.05 & rel_proportion_archaea >= 0.05 &
      rel_proportion_eukaryotes < 0.05 ~ 4,
      rel_proportion_bacteria < 0.05 & rel_proportion_archaea >= 0.05 &
      rel_proportion_eukaryotes >= 0.05 ~ 3,
      rel_proportion_bacteria >= 0.05 & rel_proportion_archaea < 0.05 &
      rel_proportion_eukaryotes >= 0.05 ~ 2,
      rel_proportion_bacteria >= 0.05 & rel_proportion_archaea >= 0.05 &
      rel_proportion_eukaryotes >= 0.05 ~ 1,
      TRUE ~ 0
    ))
  )

# Arrange clusters by decreasing relative abundance of prokaryotes
order_clusters <- order_clusters %>%
  arrange(key, desc(rel_proportion_prokaryotes)) %>%
  pull(cluster)

# Set the order of domains within each bar
all_cluster_uniqorgs_tax_counts$domain <- factor(
  all_cluster_uniqorgs_tax_counts$domain,
  levels = c("Eukaryota", "Archaea", "Bacteria"))

# Reorder clusters based on key and relative abundance
all_cluster_uniqorgs_tax_counts$cluster <- factor(
  all_cluster_uniqorgs_tax_counts$cluster, levels = order_clusters)

# Calculate relative proportions within each cluster again
all_cluster_uniqorgs_tax_counts <- all_cluster_uniqorgs_tax_counts %>%
  group_by(cluster) %>%
  mutate(rel_proportion = count / sum(count))

# Create the bar chart using ggplot2
my_plot <- ggplot(data = all_cluster_uniqorgs_tax_counts,
  aes(x = cluster, y = rel_proportion, fill = domain)) +
  geom_bar(stat = "identity", position = "stack") +
  geom_vline(xintercept = c(12.5, 16.5, 26.5, 30.5), color = "black") +
  labs(x = "Cluster", y = "Relative Proportion",
  title = "Superkingdom Distribution per Cluster (organisms represented once)") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))

# Save the plot as a PNG file
ggsave(filename = "superkingdom_uniqorgs_key_distribution.png", plot = my_plot)
```

Dans une seconde partie de ce code R, un tableau reprenant chaque cluster ainsi que le groupe auquel ils appartiennent respectivement en fonction de leur composition a été produit.

```
# Create a data frame to store the cluster-key mapping with verbose names
cluster_key_mapping <- order_clusters %>%
  arrange(key, desc(rel_proportion_prokaryotes)) %>%
  select(cluster, key) %>%
  mutate(verbose_key = case_when(
    key == 1 ~ "Three domains",
    key == 2 ~ "Bacteria & Eukaryota",
    key == 3 ~ "Archaea & Eukaryota",
    key == 4 ~ "Prokaryota",
    key == 5 ~ "Bacteria",
    key == 6 ~ "Archaea",
    key == 7 ~ "Eukaryota",
```

```
    TRUE ~ "Unknown"
  ))

# Save the table with verbose keys
write.table(cluster_key_mapping,
  "cluster_key_mapping.tsv", sep = "\t", quote = FALSE, row.names = FALSE)
```

3.4.3.2. Mise en place de banques de séquences candidates

La sélection des échantillons et le sous-échantillonnage ont été préalablement effectués. Il ne restait alors qu'à générer des banques de données utilisables par l'algorithme BLAST à partir de ces échantillons et à constituer les fichiers *bank-mapper* nécessaires à Forty-Two. Ces derniers précisent quelles banques utiliser dans un répertoire donné, à quel organisme correspond la banque et éventuellement un filtre taxonomique. Dans notre cas, le nom des « organismes » est « environmental sample_code » et « code » représente l'identifiant de chaque échantillon (ex : MGA11_rhiz). Le filtre taxonomique est un contrôle de la taxonomie des séquences ajoutées. Si la séquence ajoutée est proche d'une protéine appartenant à un organisme différent du filtre taxonomique, celle-ci sera considérée comme une contamination et contiendra le nom binomial de l'organisme auquel il aura été affilié. Ce filtre est ici utilisé de manière détournée puisqu'il indique un genre de gastéropode marin (en principe absent des échantillons de sol). Le but poursuivi était d'ajouter un nom d'organisme à toutes les séquences environnementales ajoutées. Le job suivant invoque le programme makeblastdb pour générer les banques de données.

```
#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q *q

#$ -t 1-72
#$ -tc 72

#$ -N makeblastdb

parameters=`sed -n "${SGE_TASK_ID} p" file_list`
parameterArray=($parameters)
x=${parameterArray[0]}

makeblastdb -in ${x} -dbtype nucl -out `basename ${x} .fasta` -parse_seqids
```

Quatre fichiers *bank-mapper*, un par catégorie d'échantillon (M-Bulk1, M-rhiz, N-Bulk1, N-rhiz) ont alors été constitués par les commandes bash suivantes.

```
ls Environmental_sample_*nsq | grep "NGA.._rhiz" | cut -f1 -d'.' | \
  perl -nle '$bank = $_; s/_/ /;
             print join "\t", "\l$_", $bank, "+Lottia"."\t"."fake filter"' \
  > bank-mapper_NGA_rhiz.idm

ls Environmental_sample_*nsq | grep "NGA.._Bulk1" | cut -f1 -d'.' | \
  perl -nle '$bank = $_; s/_/ /;
             print join "\t", "\l$_", $bank, "+Lottia"."\t"."fake filter"' \
  > bank-mapper_NGA_Bulk1.idm

ls Environmental_sample_*nsq | grep "MGA.._rhiz" | cut -f1 -d'.' | \
  perl -nle '$bank = $_; s/_/ /;
```

```
            print join "\t", "\l$_", $bank, "+Lottia"."\t"."fake filter"' \
  > bank-mapper_MGA_rhiz.idm

ls Environmental_sample_*nsq | grep "MGA.._Bulk1" | cut -f1 -d'.' | \
  perl -nle '$bank = $_; s/_/ /;
            print join "\t", "\l$_", $bank, "+Lottia"."\t"."fake filter"' \
  > bank-mapper_MGA_Bulk1.idm
```

3.4.3.3. Sélection des *queries*

Un ensemble d'organismes représentatifs des grands groupes du Vivant et les plus représentés dans les différents alignements multiples (pour augmenter les chances de trouver des *query_seqs* dans chaque alignement multiple) a été assemblé. Le nombre d'alignements multiples dans lesquels sont présents chaque organisme a été compté, puis la taxonomie de ces organismes a été récupérée. Un représentant de chaque grand groupe, celui présent dans le plus d'alignements multiples différents, a été sélectionné de sorte à assembler une liste de *queries* diversifiée pour les trois domaines. Cet ensemble a ensuite été fractionné en trois listes, une pour chaque domaine, et réassemblé pour proposer une liste de *queries* adaptée à chaque groupe de clusters, selon les combinaisons de domaines représentés au sein des alignements associés à ces derniers.

```
grep \> ../*-uniq-linsi.fasta | \
  perl -nle 's/([^>]+>)([^ _]+)[ |_]([^@]+)@.+/$1$2_$3/; print' | \
  sort | uniq | cut -d':' -f2 | sort | uniq -c | sort -n | \
  perl -nle 's/\s+([0-9]+) >(.+)/$1\t$2/; print' \
  > organism_reccurence.tsv

fetch-tax.pl \
--taxdir=/media/vol1/databases/taxdump-20210216/ \
--item-type=baseid organism_reccurence.tsv \
--col=2
cut -f1,4 organism_reccurence.tax > organism_taxonomy.tsv
join -t$'\t' -12 -21 organism_reccurence.tsv organism_taxonomy.tsv \
  > organism_reccurence_taxonomy.tsv

nano organism_reccurence_taxonomy.tsv
# Sélection d'organismes les plus représentés pour représenter les grands groupes
grep \+ organism_reccurence_taxonomy.tsv | cut -c2- | cut -d';' -f1-4 | \
  perl -nle 's/cellular organisms; //; print' \
  > query_selection

for d in Archaea Eukaryota Bacteria; do grep $d query_selection | cut -f1 | \
  perl -nle 's/_/ /; print' > sel_${d}_queries.txt; done

cat sel_*_queries.txt > 3dom_queries.txt
cat sel_Archaea_queries.txt sel_Bacteria_queries.txt > Prok_queries.txt
cat sel_Bacteria_queries.txt sel_Eukaryota_queries.txt > Bact_Euk_queries.txt
cp sel_Bacteria_queries.txt Bact_queries.txt
cp sel_Eukaryota_queries.txt Euk_queries.txt
```

3.4.3.4. Mise en place de protéomes de référence

Dans la même logique de représentativité, plus un organisme de référence est présent dans différents alignements multiples, plus celui-ci aura une probabilité importante de posséder une protéine des différentes familles et de pouvoir servir de contrôle d'orthologie. Pour cette raison, l'ensemble des

organismes de référence choisis est le même que l'ensemble des *queries*. Les protéomes de ces organismes ont été récupérés depuis la banque de données life-tqmd-of73, excepté pour *E. coli* qui n'y figure pas. Son protéome a été téléchargé au préalable par Amandine Bertrand depuis le NCBI.

```
perl -nle 's/ /_/; print' sel_Eukaryota_queries.txt \
  > link_list_euk
perl -nle 's/ /_/; print' sel_Archaea_queries.txt | cut -d'_' -f3- \
  > link_list_arch
perl -nle 's/ /_/; print' sel_Bacteria_queries.txt | cut -d'_' -f3- \
  > link_list_bact
# removing E.coli from list for manual retrieving

ln -s /media/vol2/home/abertrand/MetaRhizoMet/t33a/1-forty-two/ref_org/GCF_000800765.1.faa
for l in $(cat ref_org_selection/link_list_*); do \
  ln -s /media/vol1/databases/life-tqmd-of73/*/$l* ; done
```

Des banques de données utilisables par BLAST ont ensuite été générées, d'une manière équivalente à la génération de banques de séquences candidates, à une seule différence : la nature protéique des séquences. Pour chaque groupe de clusters, un fichier *ref_org-mapper* a été constitué. Ce dernier précise au programme Forty-Two quelles banques utiliser comme protéomes de référence et c'est sur ces fichiers qu'une différence a été introduite pour traiter différemment les groupes de clusters (en fonction de la taxonomie). Pour produire ces fichiers, la taxonomie des procaryotes, dont les banques sont représentées par des GCF, a été récupérée.

```
ls *psq | grep GCF | \
  perl -nle '$bank = $_; s/([^_]+)_([^_]+)_\S+/$1_$2/;
             print join "\t", "$_", $bank' \
  > procaryotes_bank_GCF.list

ls *psq | grep GCF | cut -f-2 -d'_' | cut -d'.' -f-2 > procaryotes.idl
fetch-tax.pl \
--taxdir=/media/vol1/databases/taxdump-20210216/ \
--item-type=taxid \
--org-mapper procaryotes.idl

paste procaryotes_bank_GCF.list procaryotes.org-idm | \
  perl -anle 'print "$F[2] $F[3]\t$F[1]"' | \
  perl -nle 's/\.psq//; print' > ref_org_Prok-mapper.idm

perl -nle '$bank = $_; s/_/ /; print join "\t", "$_", $bank' link_list_euk \
  > ref_org_Euk-mapper.idm

cat ref_org_Euk-mapper.idm ref_org_Prok-mapper.idm > ref_org_3dom-mapper.idm
grep -v -F -f link_list_arch ref_org_Prok-mapper.idm > ref_org_Bact-mapper.idm
cat ref_org_Bact-mapper.idm ref_org_Euk-mapper.idm > ref_org_Bact_Euk-mapper.idm
```

3.4.3.5. Création de fichiers de configuration et exécution du programme

Forty-Two s'exécute avec une simple ligne de commande, la complexité de sa configuration réside dans la préparation des fichiers dont dépend le programme. Tous les paramètres et les fichiers nécessaires à Forty-Two sont spécifiés dans un fichier de configuration, lequel est mentionné dans la ligne de commande invoquant le programme. Ces fichiers de configuration sont difficiles à écrire à la main. Un programme a alors été conçu pour aider à les construire : yaml-generator-42.pl (v 0.213470).

Celui-ci génère un fichier de configuration et un fichier exécutable (build-yaml.sh) capable de reproduire le fichier de configuration. Dans notre cas, quatre séries de configurations sont nécessaires à l'exécution du programme pour chacune des quatre catégories d'échantillons de données métagénomiques (M-Bulk1 ; M-rhiz ; N-Bulk1 ; Nrhiz). En parallèle, les différents groupes de clusters déterminés nécessitent également plusieurs paramétrages différents. Des sept groupes possibles, les clusters sont finalement répartis parmi cinq groupes. Ceci nous amène à vingt combinaisons différentes de paramètres, chacune adaptée à certains cas (au total, les différents job-arrays lancent 132 instances du programme en simultané). Pour gagner du temps et éviter les erreurs, un *template* contenant des variables de la forme [% nom_de_la_variable %] à la place des paramètres différant dans les vingt combinaisons est créé à la fois pour l'exécutable capable de produire les fichiers de configuration et le job à soumettre à *durandal*.

```
yaml-generator-42.pl --run_mode=phylogenomic --out_suffix=[% suffix %] \
--queries \
/media/vol2/home/lsmacchia/screening/5-forty-two/query_selection/[% query %] \
--evalue=1e-05 --homologues_seg=yes --max_target_seqs=10000 --templates_seg=no \
--bank_dir /media/vol2/home/lsmacchia/screening/5-forty-two/scaffolds \
--bank_suffix=.nsq \
--bank_mapper \
/media/vol2/home/lsmacchia/screening/5-forty-two/scaffolds/[% bank %] \
--code=1 \
--ref_brh=on --ref_bank_dir \
/media/vol2/home/lsmacchia/screening/5-forty-two/ref_banks \
--ref_bank_suffix=.psq \
--ref_bank_mapper /media/vol2/home/lsmacchia/screening/5-forty-two/ref_banks/ref_o
rg_selection/[% ref %] \
--ref_org_mul=[% mul %] --ref_score_mul=0.99 \
--trim_homologues=on --trim_max_shift=20000 --trim_extra_margin=15 \
--merge_orthologues=on --merge_min_ident=0.9 --merge_min_len=40 \
--aligner_mode=blast --ali_skip_self=off --ali_cover_mul=1.1 \
--ali_keep_old_new_tags=off --ali_keep_lengthened_seqs=keep \
--tax_reports=on --tax_dir /media/vol1/databases/taxdump-20210216 \
--best_hit \
--tol_check=off
```

Ce fichier indigeste comporte nos cinq variables, à savoir le suffixe ajouté aux fichiers de sortie ([% suffix %]), la liste de *queries* ([% query %]), la liste des échantillons contenant les séquences candidates ([% bank %]), la liste des organismes de référence ([% ref %]) et la fraction d'organismes de référence nécessaires pour satisfaire la condition de *Best reciprocal hits*, responsable du contrôle d'orthologie. Ce dernier paramètre a une valeur différente en fonction du groupe de clusters considéré. En effet, un nombre fixe d'organismes de référence, à savoir 8, a été choisi, afin d'homogénéiser le contrôle d'orthologie pour chaque groupe de clusters et de le rendre indépendant du nombre d'organismes de référence. Les vingt versions de ce fichier ont été générées puis exécutées grâce aux commandes suivantes.

```
ls *queries.txt | grep -v sel\_ | cut -d'/' -f2 | rev | cut -d'_' -f2- | rev \
  > query-ref-mul.list
nano query-ref-mul.list # Ajout du paramètre mul (fraction de ref_orgs)
# 3dom-0.25  /  Bact_Euk-0.33  /  Bact-0.61  /  Euk-0.72  /  Prok-0.38

for b in $(cat sample_categories.txt); do \
  env bank=$b perl -nle 'print "$ENV{bank}"."-"."$_"' query-ref-mul.list; done \
  > parameters.list

perl -F'\-' -anle '($bank,$query_ref,$mul) = @F; print "tpage
```

```
  --define bank=bank-mapper_${bank}.idm
  --define ref=ref_org_${query_ref}-mapper.idm
  --define query=${query_ref}_queries.txt
  --define mul=$mul
  --define suffix=-42-$bank-$query_ref
  build-42-template-yaml.sh > build-42-yaml-$bank-$query_ref-$mul.sh"' \
  parameters.list > batch.sh

cat batch.sh | sh
cat build-42-yaml-* | sh
```

Le contenu du fichier parameters.list, soit la liste des associations de paramètres, est reprise en annexe 11. Le *template* des fichiers de jobs comporte trois paramètres.

```
#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q smallnodes.q

#$ -t 1-[% cluster_nb %]
#$ -tc [% cluster_nb %]

#$ -N forty-two

parameters=`sed -n "${SGE_TASK_ID} p" [% cluster_list %]`
parameterArray=($parameters)
x=${parameterArray[0]}

export BMD_BLAST_BINDIR=/media/vol1/apps/ncbi-blast-2.7.1+/bin/

forty-two.pl \
--config=config-42-[% config %].yaml \
--verbosity=3 \
--outdir=42-[% config %]  \
  ${x}-42-mafft-aligned-long-uniq-linsi.fasta 2> 42_[% config %]_${x}.log
```

Les vingt versions de ce job ont été générées puis soumises grâce aux commandes suivantes. Le contenu du fichier parameters_jobs.list, soit la liste des associations de paramètres, est reprise en annexe 12.

```
ln -s ../2bis-data_information/cluster_key_mapping.tsv
grep Three cluster_key_mapping.tsv | cut -f1 > 3dom_clusters
grep "Bacteria & Eukaryota" cluster_key_mapping.tsv | cut -f1 > Bact_Euk_clusters
grep Prokaryota cluster_key_mapping.tsv | cut -f1 > Prok_clusters
grep Eukaryota cluster_key_mapping.tsv | grep -v Bacteria | cut -f1 > Euk_clusters
grep Bacteria cluster_key_mapping.tsv | grep -v Eukaryota | cut -f1 \
  > Bact_clusters

ls *clusters > cluster_list_nb
nano cluster_list_nb # Ajout du nombre de clusters associés à chaque groupe
# 3dom_clusters 12  /  Bact_clusters 4  /  Bact_Euk_clusters 4
# Euk_clusters 3  /  Prok_clusters 10

for i in {1..4..1}; do cat cluster_list_nb; done > cluster_list_nb_4times
ls config-42-* | cut -d'-' -f3- | cut -d'.' -f1 | sort > config_list
paste config_list cluster_list_nb_4times > parameters_jobs.list

perl -F'\t' -anle '($config,$cluster_list,$cluster_nb) = @F; print "tpage
```

```
  --define config=$config
  --define cluster_list=$cluster_list
  --define cluster_nb=$cluster_nb
  42_array_template.sh > 42-${config}_array.sh"' \
  parameters_jobs.list > batch_jobs.sh

cat batch_jobs.sh | sh
for c in $(cat config_list); do qsub 42-${c}_array.sh; done
```

Forty-Two produit plusieurs fichiers : un alignement multiple enrichi en séquences orthologues provenant des banques de séquences candidates ; un fichier .tax-report qui est un rapport de la taxonomie des séquences ajoutées (déterminée sur base de l'alignement de référence) ; un fichier .log contenant des informations sur le déroulement de l'exécution du programme. Ces deux derniers fichiers sont utiles pour le débogage et jeter un premier coup d'œil aux résultats. Les clusters pour lesquels Forty-Two n'a pu trouver des séquences orthologues pour aucune des quatre catégories d'échantillons ont été écartés de la suite des analyses.

3.4.4. Placements des rajouts dans les arbres de référence

Dans cette étape, les séquences orthologues rajoutées dans les alignements multiples ont été placées par maximum de parcimonie (MP) dans les arbres de référence associés. Brièvement, le nom des séquences a d'abord été formaté à travers tous les alignements multiples, lesquels ont été filtrés par ali2phylip.pl avant d'être insérés dans les arbres de référence par RAxML.

Pour prendre un exemple, voici le nom d'une séquence rajoutée par Forty-Two depuis les banques de données métagénomiques issues des échantillons de blocs de sol de sites métallifères dans l'alignement multiple du cluster nramp-P38925 :

>c#environmental_sample_MGA11_Bulk1_subsampled@NODE_721_length_7313_cov_6.47451...Deinococcus_radiodurans_R1#NEW#

Ce nom comporte le nom de l'échantillon dont provient la séquence, l'identifiant du *scaffold*, ainsi que sa longueur et sa couverture de séquençage, le nom de l'organisme affilié dans l'alignement multiple de référence et le tag #NEW#. Ce nom a alors été formaté pour retirer les éléments inutiles, réordonner les informations restantes et comporter un préfixe non chiffré (non supporté par format-tree.pl) correspondant au site dont provient la séquence :

>maab#Deinococcus_radiodurans_R1@NODE_721_length_7313_cov_6.47451#NEW#

Cette opération se fait à l'aide de la commande suivante. Le fichier sample_codes_correspondance contient les paires nom d'échantillon – code en quatre lettres associé et figure en annexe 13.

```
for c in $(cat cluster_list); do \
    for s in $(cat sample_categories.txt); do \
        perl -nle '
        BEGIN{ open $in, "<", shift; while (<$in>)
        { chomp; ($old, $new) = split "\t"; $hash{$old} = $new } }
        if (m/^>/ && m/#NEW#/)
        { s/c#environmental_sample_([MN]GA[0-9]+_[^_]+)[^@]+(@.+)\.\.\.([^#]+)#NEW
#/$hash{$1}#$3$2#NEW#/; s/\+/p/; s/environmental_sample_([MN]GA[0-9]+_[^_]+)[^@]+(
@.+)/$hash{$1}#Lottia$2/ } print' \
        sample_codes_correspondance \
        $c-42-mafft-aligned-long-uniq-linsi-42-$s*.fasta \
```

```
        > $c-42-mafft-aligned-long-uniq-linsi-42-$s-names.fasta; \
    done; \
done;
```

La filtration des alignements multiples enrichis a été réalisée avec une valeur « max » de 0.3, pour retirer les colonnes peu informatives (positions non-homologues contenant un signal bruité, voir biaisé). La valeur « min » était plus délicate à choisir. Les séquences orthologues rajoutées peuvent parfois être incorrectement considérées comme trop courtes à cause d'un domaine homologue trop petit, par exemple. Ainsi, le programme ali2phylip.pl a été utilisé avec l'option « --test-out », afin de tester divers paramètres de min. Des paramètres relatifs (0.3 et 0.2) et absolus en acides aminés (75 et 50) ont été testés, et le nombre de séquences retenues a été examiné. S'il est supérieur à 2/3 des séquences initiales (seuil arbitraire), le paramètre est jugé comme bon. Ce diagnostic, ainsi que le code ayant servi à le produire, sont repris en annexe 14. Après ce diagnostic, les alignements multiples ont été filtrés avec une valeur de min de 0.3 si elle satisfaisait la condition du test diagnostic, ou avec une valeur de min de 0.2 dans le cas contraire.

Enfin, les séquences rajoutées par Forty-Two ont été placées dans les arbres de référence par MP. Pour ce faire, un fichier job a été généré pour chaque alignement multiple enrichi grâce au programme phy2raxml.pl. Cependant, l'algorithme utilisé est différent de ce que le programme spécifie par défaut. De plus, il faut inclure l'arbre de référence dans la ligne de commande invoquant le programme RAxML. Les commandes suivantes ont permis de générer des jobs adaptés et de les soumettre.

```
phy2raxml.pl \
--model=PROTGAMMALG4X *.p80 \
--nomemspec \
--queue=smallnodes.q \
  > qsub_list.sh

for c in $(cat cluster_list); do \
  env cluster=$c perl -i -nle '
  s/-N 100 -f a -x 1975021703574/\-f y -t $ENV{cluster}.tre/; print' \
  $c-42-mafft-aligned-long-uniq-linsi-42-*sh; done

cat qsub_list.sh | sh
```

Un exemple des jobs ainsi générés est repris ci-dessous.

```
#!/bin/bash

#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q smallnodes.q

#$ -N nramp-P38925-42-mafft-aligned-long-uniq-linsi-42-MGA_Bulk1-names-RAXML-PROTG
AMMALG4X-100xRAPIDBP

raxmlHPC-AVX \
-s nramp-P38925-42-mafft-aligned-long-uniq-linsi-42-MGA_Bulk1-names.p80 \
-n nramp-P38925-42-mafft-aligned-long-uniq-linsi-42-MGA_Bulk1-names-RAXML-PROTGAMM
ALG4X-100xRAPIDBP \
-m PROTGAMMALG4X -f y -t nramp-P38925.tre -p 1975021703574
```

Dans le cas de ces jobs, RAxML produit plusieurs fichiers intéressants. Premièrement, des arbres, conservant la topologie des arbres de référence et contenant les nouvelles séquences sont générés. Ceux-ci ont été chargés sur iTOL après avoir été mis en forme par les commandes suivantes.

```
perl -i -nle 's/QUERY___//g; print' RAxML_labelledTree.*
```

Cette ligne de commande permet de retirer le tag ajouté par l'algorithme de placement de RAxML.

```
#!/bin/bash

#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q *.q

#$ -t 1-84
#$ -tc 84

#$ -N format-tree

parameters=`sed -n "${SGE_TASK_ID} p" file_list`
parameterArray=($parameters)
x=${parameterArray[0]}

format-tree.pl \
/media/vol2/home/lsmacchia/screening/9-analysis/trees_visual_analysis/${x} \
--in=rAxML_labelledTree. \
--in=-42-mafft-aligned-long-uniq-linsi \
--in=-names-RAXML-PROTGAMMALG4X-100xRAPIDBP \
--annotate \
--colorize=/media/vol2/home/lsmacchia/3domains-palette.cls \
--itol \
--taxdir=/media/vol1/databases/taxdump-20210216/
```

Ce job permet d'attribuer une taxonomie aux différentes séquences et de les colorer.

```
#!/bin/bash

#$ -S /bin/bash
#$ -V
#$ -cwd

#$ -q *.q

#$ -t 1-84
#$ -tc 84

#$ -N import-itol

parameters=`sed -n "${SGE_TASK_ID} p" tree_list`
parameterArray=($parameters)
x=${parameterArray[0]}

import-itol.pl ${x} \
--api-key=7yBpUxtUmLiCNmYLjQRoiA \
--project="Mémoire - Screening trees with differential orthologous insertions" \
--description="${x}"
```

Ce job permet d'importer les arbres mis en forme dans iTOL. Les différents enrichissements d'un même cluster à partir de catégories d'échantillons différents ont alors pu être comparés visuellement via des arbres dont les branches sont colorées en fonction de leur taxonomie.

Deuxièmement, RAxML produit des fichiers listant toutes les séquences placées, et sur quel nœud ou quelle feuille de l'arbre chacune a été placée. Dans les deux types de fichiers (arbres et placements), si une séquence peut être placée à deux endroits différents (même score de parcimonie), alors elle l'est. C'est pourquoi nous associons un poids fractionnaire à chaque placement de séquence. Ainsi, si une séquence a été placée quatre fois, chacun de ses placements se voit attribuer un poids de 0.25. Une version non-filtrée (par ali2phylip.pl) des alignements multiples a également été testée pour l'étape de placement dans les arbres de référence (car le retrait de colonnes ou de séquences pouvait mener au retrait de toutes les séquences ajoutées par Forty-Two). La « décisivité » des placements a été comparée entre alignements filtrés et non-filtrés. Une distribution globale de la fréquence de chaque poids de placement a été générée pour les alignements filtrés et non-filtrés. Les codes correspondant au calcul des poids et à la création des distributions figurent en annexe 16, accompagnés de ces derniers.

3.4.5. Analyse en composantes principales

La méthode d'analyse comparative des rajouts de séquences depuis les données métagénomiques des quatre catégories d'échantillons choisie est l'analyse en composantes principales ou ACP. Les rajouts effectués par Forty-Two peuvent être décrits et comparés de plusieurs manières. D'une part, le nombre de séquences rajoutées peut varier d'une catégorie d'échantillons à l'autre. D'autre part, la taxonomie de ces séquences, soit l'endroit dans lequel chaque séquence est placée dans l'arbre (nœud), peut aussi être différente d'une catégorie à l'autre. Les données décrivant chaque échantillon sont donc l'appartenance à l'une des quatre catégories, et les nœuds sur lesquels ont été placées les séquences de issues l'échantillon. Les nœuds représentent un nombre élevé de variables décrivant chaque échantillon, c'est pourquoi une technique de réduction des dimensions a été utilisée.

L'ACP est une méthode d'ordination organisant les différents individus statistiques sur une carte en les regroupant par similarité. Cette méthode d'analyse réduit les dimensions, potentiellement très nombreuses d'un jeu de données multivarié. Elle résume ou explique la plus grande part possible de variance entre les individus en quelques variables artificielles appelées composantes principales. Les individus statistiques sont alors organisés sur un graphique dont les axes sont des paires de composantes principales. Les variables du jeu de données sont plus ou moins colinéaires avec les composantes principales. Celles-ci sont représentées par des flèches sur les graphiques, plus les flèches sont longues, plus celles-ci se retrouvent dans les composantes principales du graphique. Leur orientation sur la carte a également une signification : une corrélation positive ou négative avec les axes.

L'analyse en composantes principales a été menée sur chaque cluster grâce à un script R. Un script *template* a d'abord été conçu puis décliné pour chaque cluster.

```
# Load necessary packages
library(reshape2)
library(vegan)  # For PCA
library(ggplot2)  # For plotting

# Read the data
[% cluster %]_placement <- read.csv("[% file %]-placement-regrouped.list",
```

```
  header = FALSE, sep = "\t", col.names = c("Sample", "Node", "Weight"))

# Create the weighted contingency table
[% cluster %]_placement_contingency <- dcast([% cluster %]_placement,
  Sample ~ Node, fun.aggregate = sum, value.var = "Weight")

# Extract sample names
[% cluster %]_sample_names <- [% cluster %]_placement_contingency$Sample

# Remove sample column for PCA
[% cluster %]_data_for_pca <- [% cluster %]_placement_contingency[, -1]

# Log-transform the data
[% cluster %]_data_for_pca_log <- log10([% cluster %]_data_for_pca + 0.1)

# Perform PCA
[% cluster %]_pca_result <- rda([% cluster %]_data_for_pca_log)

# Extract the relevant information for coloring
[% cluster %]_sample_info <- data.frame(
  Sample = [% cluster %]_sample_names,
  Metallicity = substr(as.character([% cluster %]_sample_names), 1, 1),
  SampleType = substr(as.character([% cluster %]_sample_names),
  7, nchar(as.character([% cluster %]_sample_names)))
)

# Assign colors based on Metallicity and SampleType
[% cluster %]_sample_info$Color <- ifelse(
    [% cluster %]_sample_info$Metallicity == "M" &
    [% cluster %]_sample_info$SampleType == "rhiz", "dodgerblue",
                                    ifelse(
      [% cluster %]_sample_info$Metallicity == "M" &
      [% cluster %]_sample_info$SampleType == "Bulk1", "skyblue",
                                        ifelse(
        [% cluster %]_sample_info$Metallicity == "N" &
        [% cluster %]_sample_info$SampleType == "rhiz", "green4", "lightgreen")))

# Create PCA biplot with colored samples
outfile<-paste("PCA_biplot", "[% file %]", 'pdf', sep='.')
pdf(outfile, height = 10, width = 10)
biplot([% cluster %]_pca_result)
text([% cluster %]_pca_result,
  display = "sites", labels = [% cluster %]_sample_names,
  col = [% cluster %]_sample_info$Color, cex = 0.7)
text([% cluster %]_pca_result, display = "species", col = "red")

# Add legend with the custom title
legend("topright",
  legend = unique(paste([% cluster %]_sample_info$Metallicity,
                        [% cluster %]_sample_info$SampleType, sep = "_")),
  fill = unique([% cluster %]_sample_info$Color))

# Create ellipses for the four groups
[% cluster %]_groups <- paste([% cluster %]_sample_info$Metallicity,
                              [% cluster %]_sample_info$SampleType, sep = "_")

ordiellipse([% cluster %]_pca_result,
  groups = [% cluster %]_groups, kind = "ehull", conf = 0.95,
```

```
  col = c("skyblue", "dodgerblue", "lightgreen", "green4"))

# Save the PCA biplot as an pdf file
dev.off()

# PCA summary
[% cluster %]_summary <- summary([% cluster %]_pca_result)
output_file <- "pca_[% file %]_summary.txt"
file_conn <- file(output_file, "w")
writeLines(capture.output([% cluster %]_summary), con = file_conn)
close(file_conn)

# Prepare dataframe for adonis test with informative PCs
[% cluster %]_eigenvalues <- [% cluster %]_pca_result$CA$eig
[% cluster %]_num_informative_pcs <- sum([% cluster %]_eigenvalues > 1)
[% cluster %]_pca_scores <- scores([% cluster %]_pca_result,
                           choices = 1:sum([% cluster %]_eigenvalues > 0))

[% cluster %]_data_for_adonis <- data.frame(
  [% cluster %]_pca_scores$sites,
  Metallicity = [% cluster %]_sample_info$Metallicity,
  SampleType = [% cluster %]_sample_info$SampleType
)

# Perform adonis analysis
[% cluster %]_formula_kaiser <- as.formula(paste("cbind(",
  paste0("[% cluster %]_data_for_adonis$PC", 1:[% cluster %]_num_informative_pcs,
  collapse = "+"), ") ~ Metallicity + SampleType"))
[% cluster %]_formula_2 <- as.formula(paste("cbind(",
  paste0("[% cluster %]_data_for_adonis$PC", 1:2,
  collapse = "+"), ") ~ Metallicity + SampleType"))

[% cluster %]_adonis_results <- list()
tryCatch({
  [% cluster %]_adonis_result_ke <- adonis2([% cluster %]_formula_kaiser,
  data = [% cluster %]_data_for_adonis, permutations = 999, method = "euclidean")
  [% cluster %]_adonis_results$ke <- [% cluster %]_adonis_result_ke
}, error = function(err) {
  cat("Error in ke:", conditionMessage(err), "\n")
})
tryCatch({
  [% cluster %]_adonis_result_2e <- adonis2([% cluster %]_formula_2,
  data = [% cluster %]_data_for_adonis, permutations = 999, method = "euclidean")
  [% cluster %]_adonis_results$e2 <- [% cluster %]_adonis_result_2e
}, error = function(err) {
  cat("Error in e2:", conditionMessage(err), "\n")
})
tryCatch({
  [% cluster %]_adonis_result_km <- adonis2([% cluster %]_formula_kaiser,
  data = [% cluster %]_data_for_adonis, permutations = 999, method = "manhattan")
  [% cluster %]_adonis_results$km <- [% cluster %]_adonis_result_km
}, error = function(err) {
  cat("Error in km:", conditionMessage(err), "\n")
})
tryCatch({
  [% cluster %]_adonis_result_2m <- adonis2([% cluster %]_formula_2,
  data = [% cluster %]_data_for_adonis, permutations = 999, method = "manhattan")
  [% cluster %]_adonis_results$m2 <- [% cluster %]_adonis_result_2m
}, error = function(err) {
```

```
  cat("Error in m2:", conditionMessage(err), "\n")
})

# Saving PERMANOVA results
output_file <- paste("pca_[% file %]_permanova.txt", sep = "")
file_conn <- file(output_file, "w")

for (i in seq_along([% cluster %]_adonis_results)) {
  [% cluster %]_result <- [% cluster %]_adonis_results[[i]]
  if (!is.null([% cluster %]_result)) {
    output <- capture.output([% cluster %]_result)
    writeLines(output, con = file_conn)
    cat("\n\n", file = file_conn)
  }
}
close(file_conn)
```

Ce script génère d'abord des graphiques ACP sur les données de placement transformées par la fonction logarithmique. Cette transformation a pour but de donner plus de poids aux éventuels organismes rares pouvant faire une différence entre les catégories d'échantillons. Les cartes générées organisent les différents échantillons en fonction de leur similarité. Ceux-ci sont alors colorés en fonction de leur catégorie et une ellipse de la même couleur les entoure. Ces graphiques comportent également les variables nœuds sous forme de flèches. Le script permet aussi de sauvegarder les proportions de variance expliquées par les composantes principales.

Enfin, il reprend quatre variantes du même test statistique, la PERMANOVA. Cette dernière est similaire à l'ANOVA et se base sur la permutation aléatoire des échantillons. Les tests statistiques ont été réalisés en prenant en compte les deux premières composantes principales, ce qui correspond à l'éventuel partitionnement observé sur les graphiques en deux dimensions, ou toutes les composantes principales jugées informatives par la règle de Kaiser, c'est-à-dire dont la valeur propre est supérieure à 1 (autant de variance expliquée qu'une variable seule). D'autre part, le test de permutations nécessite le calcul des distances entre les échantillons, pour lequel deux méthodes ont été testées, la distance euclidienne et la distance de Manhattan. Les deux ont été choisis par rapport à la nature des données, des valeurs continues et parfois négatives (coordonnées selon les composantes principales). La distance euclidienne mesure la longueur directe entre deux points dans l'espace des composantes principales, tandis que la distance de Manhattan mesure la distance totale en parcourant les axes. Dans le cas d'un test de permutation, la distance euclidienne privilégie les différences globales, tandis que la distance de Manhattan met l'accent sur les différences de chaque axe indépendamment. Ce script a été décliné et exécuté avec les commandes suivantes.

```
ls *regrouped.list | cut -d'-' -f-2 | \
  perl -nle '$file = $_; s/-/_/; print join "\t", $_, $file' > parameters_list

perl -anle '($cluster,$file) = @F; print "tpage
  --define cluster=$cluster
  --define file=$file  PCA_template.R
  > PCA_$file.R"' \
  parameters_list > batch.sh

cat batch.sh | sh

for c in $(cut -f2 parameters_list); do \
  Rscript PCA_$c.R; done 1> PCA.log 2>> PCA.log
```

4. Résultats

La figure 6 reprend toutes les étapes réalisées en amont de ce mémoire qui sont utiles à ce dernier.

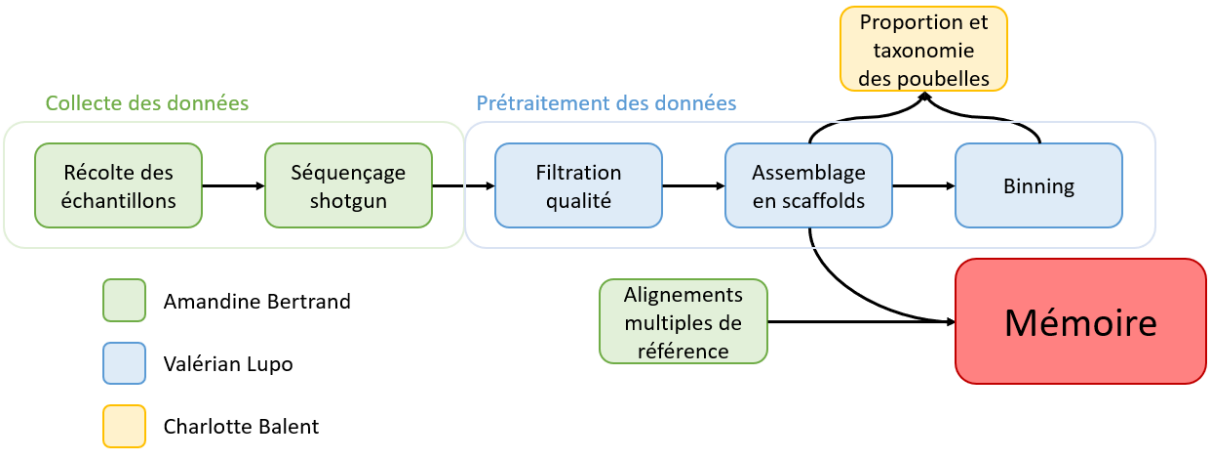


Figure 6 : Cheminement des différentes tâches antérieures à ce mémoire

4.1. Standardisation des données

Lors du processus de standardisation des données métagénomiques, la distribution des tailles des *scaffolds* a été étudiée. La première étape a donné lieu à une visualisation de cette distribution par un graphique de type *boxplot* (figure 7).

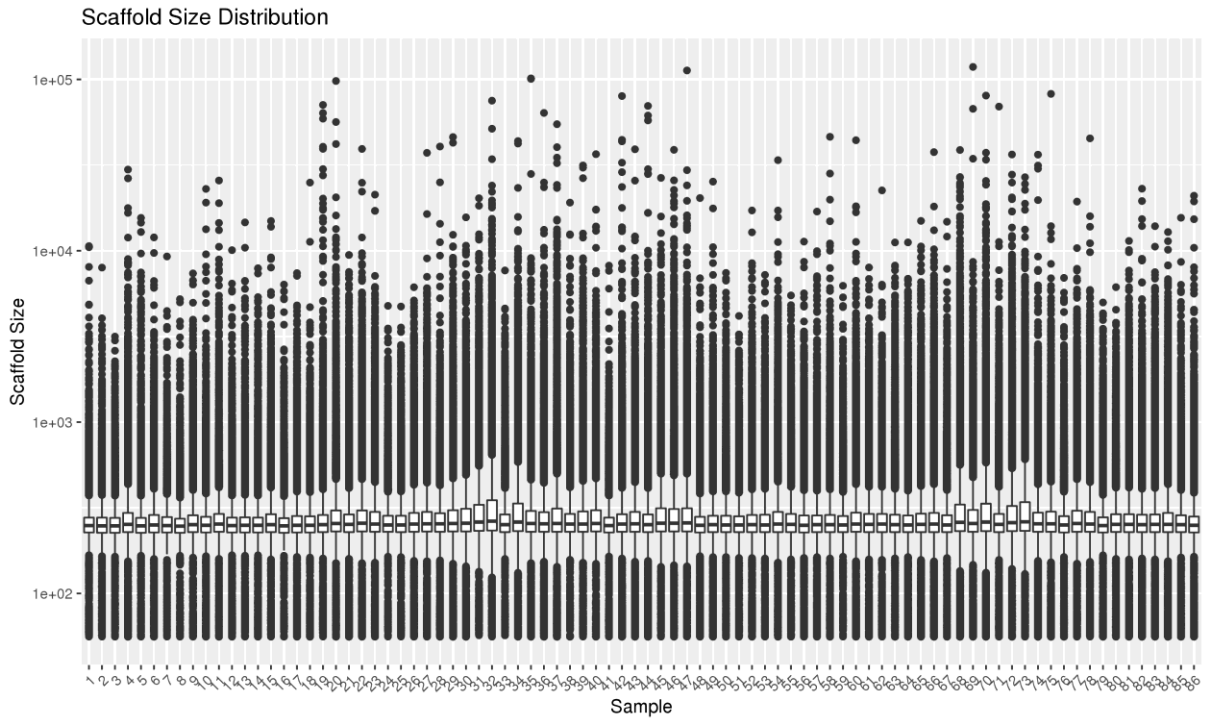


Figure 7 : Distribution de la taille des scaffolds de chaque échantillon de données métagénomiques, échelle logarithmique.

Il est difficile d'observer des différences claires entre les échantillons sur base de ce graphique. La médiane ne semble pas varier beaucoup d'un échantillon à l'autre. De plus, les espaces interquartiles de chaque distribution sont similaires.

Pour pouvoir déterminer les échantillons dont la distribution des tailles de *scaffolds* s'écarte de celle des autres, il a fallu faire appel à des tests statistiques. Ces derniers sont à la fois plus rigoureux, et capable de déceler des différences réelles, mais invisibles à l'œil nu. Voici, à la figure 8, une matrice reprenant la significativité des 3655 tests (par paire) de Wilcoxon-Mann-Whitney entre les 86 échantillons.

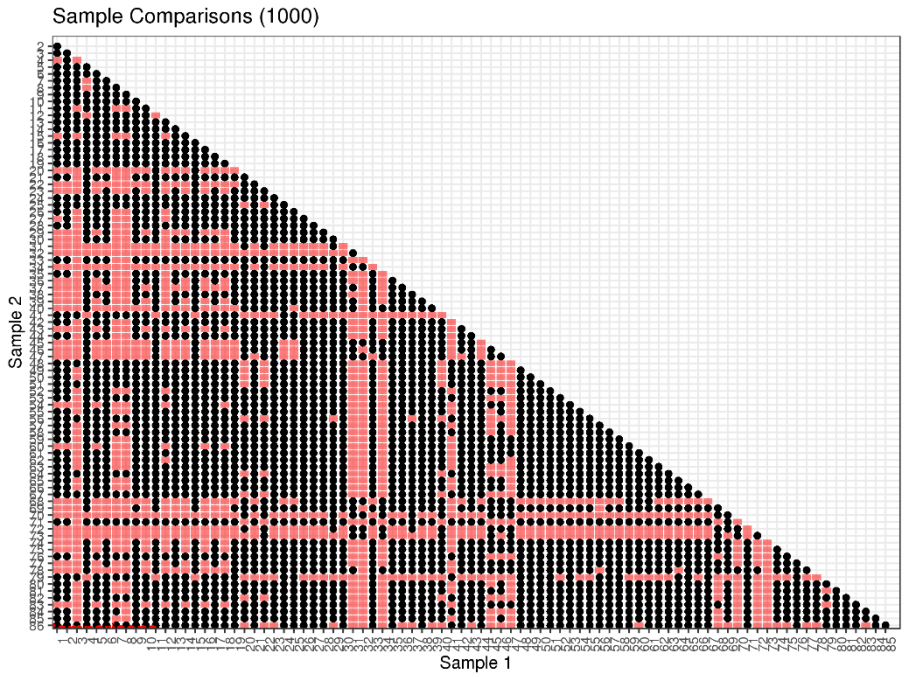


Figure 8 : Matrice de significativité des tests (par paire) de Wilcoxon-Mann-Whithney. Les intersections rouges indiquent une différence statistiquement significative entre les deux échantillons. Les intersections noires indiquent l'absence de différence significative.

Beaucoup de données figurent sur ce graphique (3655). Pour comprendre son intérêt, il faut regarder les lignes dans leur ensemble. En effet, si nous observons les résultats de l'échantillon numéro 31 par exemple, elle apparaît très majoritairement rouge. Ceci implique une majorité de différences significatives détectées entre l'échantillon 31 et les autres échantillons. Lorsque nous observons l'échantillon 53, elle apparaît à contrario dominée par le noir. Cela implique alors une minorité de différences significatives entre cet échantillon et les autres. Ainsi, les échantillons les plus différents, ou *outliers* sont mis en évidence par les lignes (et colonnes) plus rouges. Comme pour l'échantillon 31, certains sont flagrants. Cependant, pour une meilleure précision, une distribution cumulative du nombre de différences statistiquement significatives par échantillon a été générée.

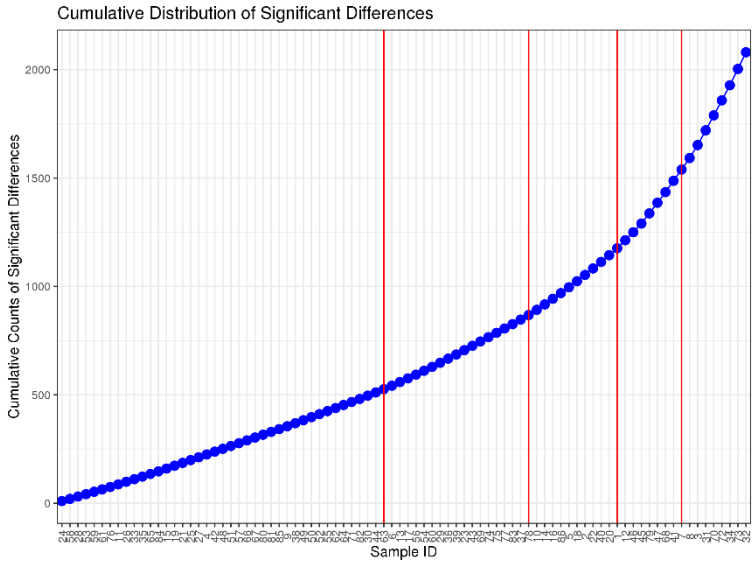


Figure 9 : Distribution cumulative du nombre de différences statistiquement significatives par échantillon. En rouge les points de rupture potentiels aussi appelés breakpoints.

Sur ce graphique, les échantillons sont ordonnés du « moindre *outlier* » au « plus *outlier* », basé sur le nombre de différences significatives avec d'autres échantillons. Des changements de structure potentiels ont été calculés et sont représentés par les lignes rouges verticales. Ces derniers indiquent alors où la distribution peut être coupée en segments homogènes (figure 9). En conséquence une série plus ou moins importante d'échantillons peuvent être considérés comme des *outliers*. Une approche conservative a été choisie ici, dans le but de minimiser la perte de données. Ainsi, seuls les sept échantillons les plus différents (NGA13_rhiz ; MGA14_Bulk1 ; MGA36_rhiz ; MGA34_Bulk1 ; MGA35_Bulk1 ; MGA36_Bulk1 ; MGA15_Bulk1) ont été écartés des analyses ultérieures.

Après homogénéisation du nombre d'échantillons par catégorie, chacune de celles-ci en comportait 18, soit 72 au total. La liste des échantillons repris dans la suite des analyses, ainsi que le nombre de *scaffolds* qu'ils contiennent après sous-échantillonnage aléatoire (certains *scaffolds* sont retirés aléatoirement pour uniformiser le nombre de *scaffolds* dans chaque échantillon) sont repris dans le tableau 1.

Échantillons de sols métallifères		**Échantillons de rhizosphères des sols métallifères**		**Échantillons de sols non-métallifères**		**Échantillons de rhizosphères des sols non-métallifères**	
MGA11_Bulk1	1397373	MGA11_rhiz	1397484	NGA11_Bulk1	1398258	NGA11_rhiz	1397269
MGA12_Bulk1	1397682	MGA13_rhiz	1397688	NGA12_Bulk1	1396765	NGA12_rhiz	1397894
MGA13_Bulk1	1397126	MGA14_rhiz	1398245	NGA21_Bulk1	1397858	NGA22_rhiz	1396954
MGA16_Bulk1	1397161	MGA15_rhiz	1398145	NGA22_Bulk1	1396593	NGA23_rhiz	1397483
MGA17_Bulk1	1397326	MGA16_rhiz	1397954	NGA23_Bulk1	1396878	NGA31_rhiz	1397819
MGA18_Bulk1	1398496	MGA17_rhiz	1397278	NGA32_Bulk1	1397287	NGA32_rhiz	1397956
MGA21_Bulk1	1397732	MGA18_rhiz	1397584	NGA33_Bulk1	1396889	NGA33_rhiz	1396873
MGA22_Bulk1	1396800	MGA21_rhiz	1397506	NGA34_Bulk1	1398901	NGA34_rhiz	1396905
MGA23_Bulk1	1397683	MGA22_rhiz	1398613	NGA35_Bulk1	1396101	NGA35_rhiz	1396809
MGA24_Bulk1	1397898	MGA24_rhiz	1398347	NGA41_Bulk1	1398728	NGA41_rhiz	1398773
MGA25_Bulk1	1398506	MGA25_rhiz	1397874	NGA42_Bulk1	1396441	NGA42_rhiz	1397112
MGA26_Bulk1	1396294	MGA26_rhiz	1397152	NGA43_Bulk1	1397791	NGA43_rhiz	1397481

MGA31_Bulk1	1397625	MGA32_rhiz	1398243	NGA44_Bulk1	1397136	NGA44_rhiz	1398303
MGA32_Bulk1	1398859	MGA34_rhiz	1396923	NGA51_Bulk1	1396461	NGA51_rhiz	1397634
MGA33_Bulk1	1397046	MGA35_rhiz	1398071	NGA52_Bulk1	1397062	NGA52_rhiz	1398874
MGA41_Bulk1	1397071	MGA41_rhiz	1398084	NGA53_Bulk1	1397400	NGA53_rhiz	1397403
MGA42_Bulk1	1397285	MGA42_rhiz	1398889	NGA54_Bulk1	1397894	NGA54_rhiz	1396459
MGA51_Bulk1	1397732	MGA51_rhiz	1398433	NGA55_Bulk1	1397656	NGA55_rhiz	1399240

Tableau 1 : Liste des échantillons sélectionnés et nombre de scaffolds associés pour chaque catégorie. En bleu, les échantillons issus de sols métallifères. En vert, les échantillons issus de sols non-métallifères. Les échantillons de rhizosphères sont représentés dans une version plus foncée de la couleur.

4.2. Rajouts de séquences environnementales orthologues

Le cluster lyse_nico-P76425 (liste en annexe 4) a été écarté des analyses en raison du nombre trop faible de séquences présentes au sein de ce dernier. Il n'en comporte que trois, ce qui n'est pas suffisant pour inférer un arbre phylogénétique informatif, puisqu'une seule combinaison est possible.

Les trente-trois clusters ont été répartis en cinq groupes (hors de sept combinaisons théoriques) en fonction de leur taxonomie. Ainsi, il existe dans nos alignements de référence des clusters à trois domaines, des clusters bactériens et eucaryotes, des clusters procaryotes, des clusters purement eucaryotes et des clusters purement bactériens. L'appartenance des clusters à ces groupes est affichée sur le graphique en barres de la figure 10. Celui-ci affiche également le partitionnement relatif de chaque cluster entre les trois domaines du Vivant.

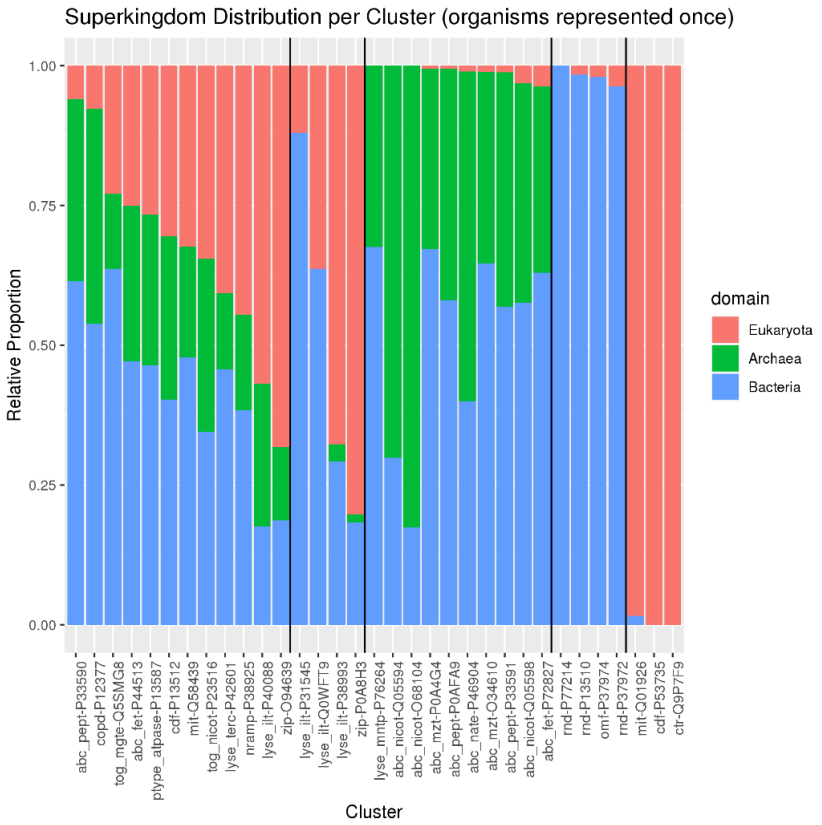


Figure 10 : Partitionnement relatif des clusters entre les trois domaines. Les clusters sont organisés par groupe et ordonnés selon la proportion de procaryotes (décroissant). Dans l'ordre : les clusters à trois domaines ; les clusters bactériens et eucaryotes ; les clusters procaryotes, les clusters bactériens ; les clusters eucaryotes.

Les cinq groupes de clusters ont alors pu être enrichis par Forty-Two, qui a été configuré pour rechercher des séquences orthologues parmi les échantillons métagénomiques standardisés selon des paramètres adaptés à chaque groupe. Un résumé des résultats bruts produits par Forty-Two est repris dans le tableau 2, aux côtés d'informations diagnostiques. Ce tableau vise à donner une première idée de la qualité des résultats.

Cluster	Nombre de rajouts M-Bulk1	Nombre de rajouts M-rhiz	Nombre de rajouts N-Bulk1	Nombre de rajouts N-rhiz	Nombre d'organismes de référence effectifs	Nombre d'organismes de référence problématiques	Nombre de séquences initiales
abc_fet-P44513	53420	55129	53626	53879	8	8	1940
ptype_atpase-P13587	23868	20428	16331	15757	8	7	1828
rnd-P37972	16182	10424	10893	8344	8	0	103
abc_pept-P33591	9455	8992	9635	9034	8	7	355
abc_pept-P0AFA9	9041	8456	9485	8654	8	8	362
cdf-P13512	2566	2197	1045	1062	8	7	470
nramp-P38925	1962	1735	1448	1538	8	1	259
abc_pept-P33590	1510	823	1140	684	8	8	350
lyse_terc-P42601	1060	1402	1172	1479	8	6	130
rnd-P13510	1173	492	719	459	8	5	132
lyse_ilt-Q0WFT9	437	976	519	949	1	0	14
lyse_ilt-P38993	472	198	95	106	8	7	241
copd-P12377	381	50	267	121	4	2	14
tog_mgte-Q5SMG8	327	237	316	203	8	3	176
rnd-P77214	228	205	74	86	2	0	5
abc_mzt-P0A4G4	108	121	135	133	8	1	253
mit-Q58439	78	76	111	82	8	7	217
lyse_mntp-P76264	10	73	25	63	6	0	39
abc_mzt-O34610	44	51	60	62	8	6	309
abc_nicot-Q05594	26	38	22	22	8	8	91
omf-P37974	29	24	13	23	8	8	103
lyse_ilt-P31545	25	18	29	25	4	4	28
zip-P0A8H3	3	26	4	22	8	3	151
abc_nicot-Q05598	2	17	5	9	8	0	211
zip-O94639	0	13	6	11	8	8	242
abc_nicot-O68104	1	8	3	3	8	6	23
abc_nate-P46904	4	0	1	0	8	8	155
tog_nicot-P23516	0	0	0	0	8	7	35
mit-Q01926	0	0	0	0	8	5	148
lyse_ilt-P40088	0	0	0	0	8	5	76
ctr-Q9P7F9	0	0	0	0	8	8	132
cdf-P53735	0	0	0	0	0	0	6
abc_fet-P72827	0	0	0	0	8	8	221

Tableau 2 : Informations diagnostiques sur le déroulé de la recherche de séquences par orthologie de Forty-Two. Nombre de séquences rajoutées en fonction de la catégorie d'échantillon ; nombre d'organismes de référence utilisés par le programme (liste en annexe 15) ; nombre d'organismes de référence problématiques ; nombre de séquences de l'alignement de référence.

Certains alignements n'ont pas du tout été enrichis par Forty-Two. La raison de cet échec n'apparait pas évidente au vu du tableau. Des organismes de référence sont utilisés, certains sont problématiques (leurs *Best hits* ne sont pas en BRH avec les *Best hits* des autres organismes de référence), mais c'est aussi le cas d'alignements ayant été enrichis. Le nombre de séquences initiales ne semble pas non plus pouvoir expliquer l'échec de certains enrichissements. Cependant, une tendance est observable : un alignement riche attire plus de séquences environnementales. Il existe cependant des contre-exemples tels rnd-P37972, un alignement de 100 séquences figurant parmi ceux ayant attiré le plus de nouvelles séquences ; abc_fet-P72827 n'attire quant à lui aucune nouvelle séquence malgré les 221 protéines composant l'alignement. Les six clusters pour lesquels **aucune des quatre catégories**

d'échantillons n'a su fournir une séquence orthologue (tog_nicot-P23516 ; mit-Q01926 ; lyse_ilt-P40088 ; ctr-Q9P7F9 ; cdf-P53735 ; abc_fet-P72827) ont été écartés de la suite des analyses, ce qui nous amène à vingt-sept clusters.

4.3. Placement des séquences environnementales rajoutées

L'étape de placement des nouvelles séquences dans les arbres de référence n'a pas non plus fonctionné pour tous les alignements dans leur version filtrée, c'est pourquoi une version non-filtrée des alignements, plus conservative par rapport aux séquences rajoutées par Forty-Two, a été testée en parallèle. En effet, les vingt-sept clusters, enrichis par quatre jeux de données métagénomiques, ont donné cent huit alignements multiples. Parmi ceux-ci, quatre-vingt-cinq ont passé l'étape sans encombre. Parmi les 23 alignements en échec, 11 ont rencontré une erreur logique, l'absence de séquences à placer : trois alignements n'avaient pas été enrichis, et huit autres ont perdu leurs nouvelles séquences lors de la filtration par ali2phylip.pl. Les 12 autres erreurs étaient plus complexes : au moins une protéine de l'arbre de référence était absente de l'alignement enrichi. En effet, l'ajout de nouvelles séquences a mené à l'insertion de nouvelles colonnes, ce qui a modifié l'effet du programme ali2phylip.pl qui a alors pu raccourcir certaines séquences de référence en-dessous du seuil minimum et par conséquent les éjecter de l'alignement. Ces douze alignements enrichis appartenaient aux trois mêmes clusters (abc_fet-P44513 ; copd-P12377 ; rnd-P37972), qui ont donc été écartés de la suite des analyses. Ces 23 erreurs n'ont pas été rencontrées dans la version alternative du *pipeline*, sans filtration.

Après le placement des nouvelles séquences dans les arbres de référence, les nouveaux arbres ont pu être observés. L'avantage de la méthode de placement est que la topologie de l'arbre ne change pas entre les quatre enrichissements par Forty-Two. Cela rend les arbres ainsi générés en principe directement comparables. Néanmoins, l'analyse à l'œil nu est limitée par le volume des données, parfois très conséquent. Les plus grands arbres ne sont d'ailleurs pas visualisables dans iTOL, qui n'a pas su les gérer. Voici cependant l'exemple du cluster nramp-P38925.

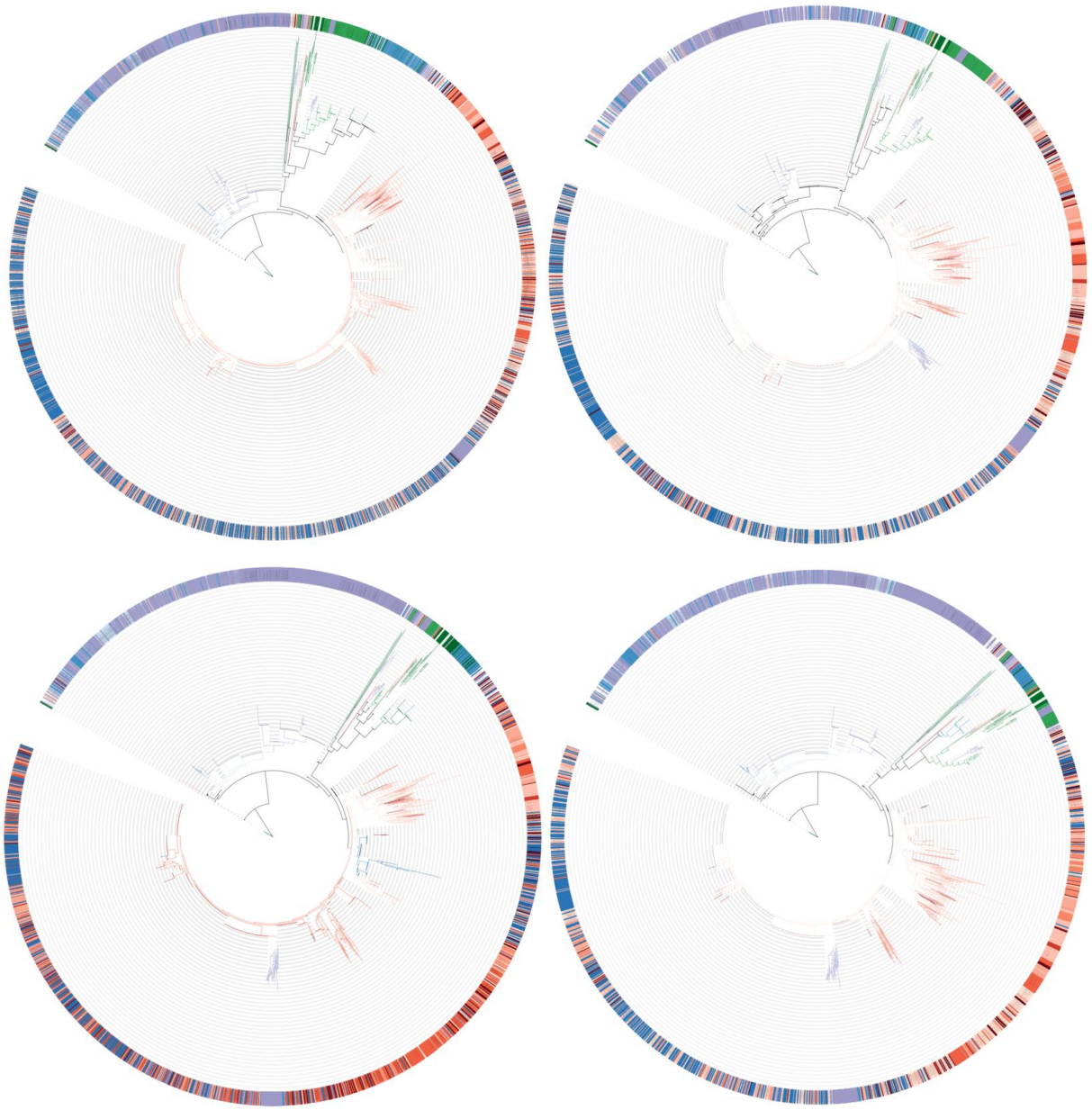
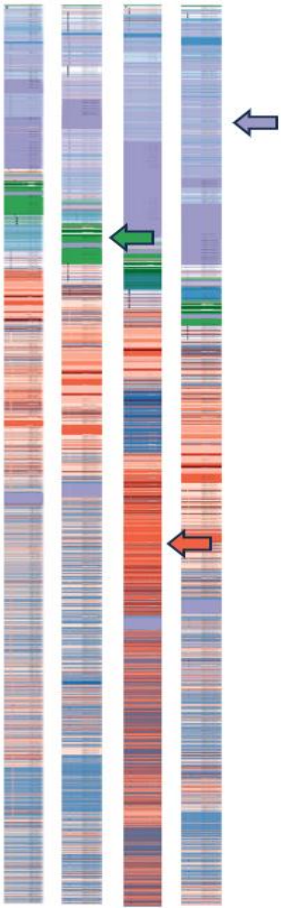


Figure 11 : Arbres colorés issus du placement des nouvelles séquences rajoutées par Forty-Two dans l'alignement multiple du cluster Nramp visualisés dans iTOL. En haut à gauche, données métagénomiques de sols métallifères ; en haut à droite, sols non-métallifères ; en bas à gauche, rhizosphères d'un sol métallifère ; en bas à droite, rhizosphères d'un sol non-métallifère. Les couleurs rouges et oranges correspondent à des séquences eucaryotes, les couleurs bleutées à des séquences bactériennes et les couleurs vertes à des séquences archéennes.

Ces arbres proposent un aperçu visuel grossier des différences entre les quatre jeux de données métagénomiques. Cependant, grâce à la topologie conservée et aux couleurs, il est possible de déceler des variations (figure 11). Les variations les plus faciles à observer sont celles qui semblent différencier les échantillons rhizosphériques des échantillons de sols. En effet, le sous-arbre majoritairement bactérien, visible en bleu à gauche du sous-arbre impliquant des archées est plus important dans les échantillons rhizosphériques. Ce sous-arbre procaryote semble à son tour légèrement plus important dans les arbres enrichis par des données du sol (Bulk). D'autres structures diffèrent également si l'on observe attentivement ces images, de même que la coloration de certaines parties des arbres (moitié basse).



Ces arbres enrichis peuvent également être linéarisés et observés côte à côte (figure 12). De cette manière, les bandes colorées correspondant aux différents grands groupes taxonomiques sont encore plus faciles à comparer, à la manière de bandes de migration d'un gel d'électrophorèse. Cependant, la topologie de l'arbre n'est plus visible et les sous-arbres ne sont plus distinguables.

De nouveau, nous pouvons observer une partie bactérienne plus importante dans les rhizosphères (flèche bleue), ainsi qu'une partie procaryote mixte plus large dans les échantillons de sols (flèche verte). L'arbre constitué à partir des données métagénomiques de rhizosphères de sols métallifères semble se distinguer des trois autres dans la moitié basse de l'image (flèche orange).

Toutefois, les observations et comparaisons visuelles des arbres d'un même cluster restent limitées, principalement par la quantité de données à analyser d'un simple coup d'œil.

Figure 12 : Arbres colorés, linéarisés issus du placement des nouvelles séquences rajoutées par Forty-Two dans l'alignement multiple du cluster Nramp visualisés dans iTOL. Respectivement, données métagénomiques de : sols métallifères ; sols non-métallifères ; rhizosphères d'un sol métallifère ; rhizosphères d'un sol non-métallifère. Les couleurs rouges et oranges correspondent à des séquences eucaryotes, les couleurs bleutées à des séquences bactériennes et les couleurs vertes à des séquences archéennes.

4.4. Analyses en composantes principales

Le placement des nouvelles séquences dans les arbres de référence a donné lieu à un nombre élevé de variables « nœud » correspondant à un endroit précis de l'arbre sur lequel une séquence environnementale a été placée. Ce sont ces variables nœuds décrivant les placements qui constituent les différences entre les 72 échantillons. La réduction des données opérée par l'analyse en composantes principales a permis de représenter en deux dimensions une partie de cette variance. Lors de cette étape, un nouveau cluster a été écarté des analyses : abc_nicot-O68104. Celui-ci comportait des données identiques pour six échantillons, tous de la même catégorie, et une légère variation pour un septième échantillon, toujours de la même catégorie. L'analyse a par conséquent rencontré des erreurs et n'a pas produit de résultat. Le tableau 3 reprend la proportion de variance expliquée par les deux premières composantes principales de chaque cluster. Ces fractions de variance sont alors celles affichées sur toutes les cartes ACP, construites avec, pour axes, les deux premières composantes principales.

Cluster	Variance expliquée par les PC1 et PC2 (+ Filtration)	Variance expliquée par les PC1 et PC2 (- Filtration)
abc_fet-P44513	NA	0.06579
abc_mzt-O34610	0.2429	0.2377
abc_mzt-P0A4G4	0.19823	0.19395
abc_nate-P46904	1.0000	1.0000
abc_nicot-Q05594	0.4461	0.4294
abc_nicot-Q05598	0.9712	0.4691
abc_pept-P0AFA9	0.08954	0.09292
abc_pept-P33590	0.19322	0.15377
abc_pept-P33591	0.10412	0.09948
cdf-P13512	0.24167	0.1970
copd-P12377	NA	0.6610
lyse_ilt-P31545	0.9108	0.87016
lyse_ilt-P38993	0.49116	0.36126
lyse_ilt-Q0WFT9	0.5011	0.5480
lyse_mntp-P76264	0.3217	0.3222
lyse_terc-P42601	0.19243	0.18795
mit-Q58439	0.2325	0.18797
nramp-P38925	0.21046	0.18853
omf-P37974	0.4313	0.4852
ptype_atpase-P13587	0.19902	0.17588
rnd-P13510	0.2314	0.21618
rnd-P37972	NA	0.2659
rnd-P77214	1.0000	0.9820
tog_mgte-Q5SMG8	0.22900	0.20742
zip-O94639	0.7606	0.4819
zip-P0A8H3	0.6028	0.4625

Tableau 3 : Fractions de variance entre les échantillons reprises par les deux premières composantes principales, et par conséquent affichées sur les cartes ACP. Valeurs pour les alignements multiples enrichis et filtrés ou non par ali2phylip.pl.

La fraction de variance expliquée par les deux premières composantes principales est presque systématiquement plus élevée sur les alignements traités par ali2phylip.pl. Les alignements non-filtrés semblent parasités par un bruit inhérent aux séquences trop courtes, c'est d'ailleurs ce que l'on peut constater lorsque l'on regarde la distribution relative des poids associés aux placements, reprise en annexe 16. Dans seulement deux cas, moins de 10% de la variance est reprise dans les deux premières composantes principales.

Lorsque l'on compare les cartes ACP produites à partir d'alignements filtrés ou non, elles diffèrent très peu et montrent une organisation des échantillons similaires dans les deux cas. La plupart du temps, les cartes produites à partir d'alignements non-filtrés semblent simplement un peu moins nettes, ce qui n'impacte pas ou peu les observations. Voici l'exemple du cluster abc_pept-P33591.

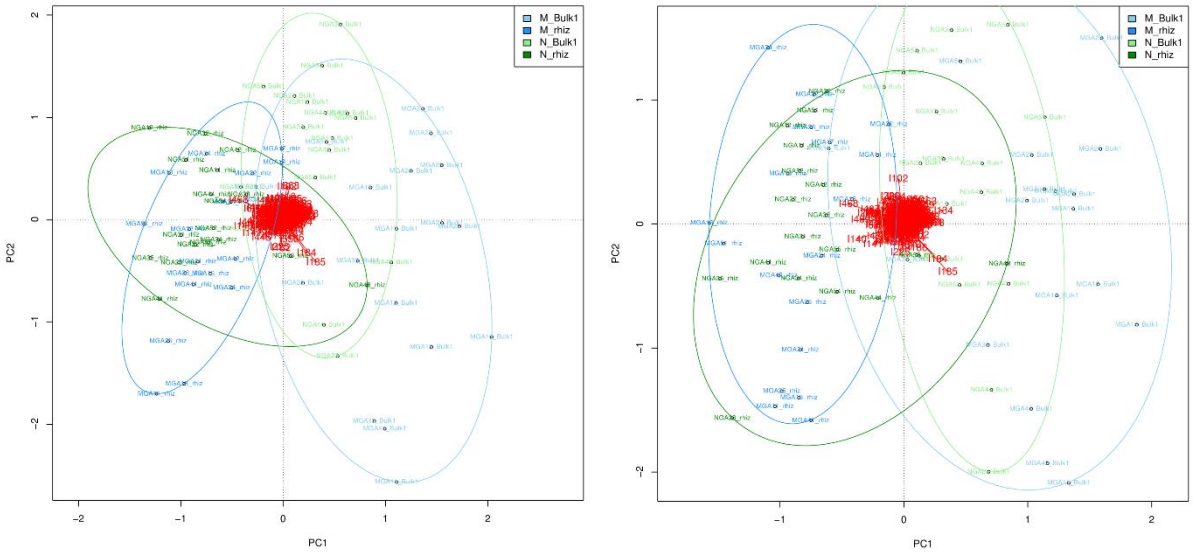


Figure 13 : Cartes ACP du cluster abc_pept-P33591, dont les alignements multiples ont été filtrés (gauche) ou non (droite). En vert, les échantillons issus de sols non-métallifères. En bleu, les échantillons issus de sols métallifères. Les échantillons rhizosphériques sont représentés dans une couleur plus foncée.

Nous pouvons observer sur la figure 13 quelques différences mineures entre les deux graphiques. Les ellipses entourant les échantillons d'une catégorie (intervalle de confiance de 95%) sont plus larges dans la version non-filtrée. De plus, les différents groupes semblent se confondre davantage dans cette même version, d'apparence moins nette.

Pour des raisons de proportion de variance exprimée par les axes, et de cohérence vis-à-vis de la raison d'être de l'étape de filtration par ali2phylip.pl, la version filtrée est privilégiée lorsque c'est possible. Toutes les cartes ACP produites à partir d'alignements enrichis et filtrés se trouvent en annexe 17. Toutes les cartes ACP produites à partir d'alignements enrichis non-filtrés se trouvent en annexe 18. Neuf clusters ont retenu mon attention de par leur carte ACP apparaissant partitionner, ou du moins ne pas superposer, les groupes d'échantillons métalliques et non-métalliques. Il s'agit des clusters cdf-P13512, lyse_ilt-P31545, lyse_ilt-38933, nramp-P38925, ptype_atpase-P13587, rnd-P13510 et zip-P0A8H3 (méthode d'alignements enrichis filtrés) et des clusters abc_fet-P44513 et rnd-P37972 (pour lesquels seule la méthode sans filtration a fonctionné).

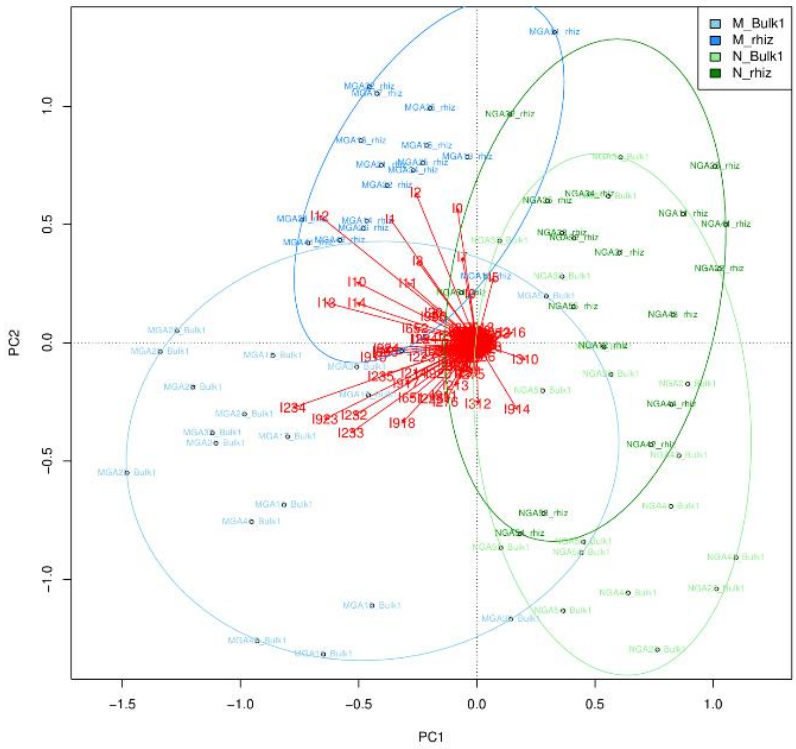


Figure 14 : Carte ACP du cluster cdf-P13512 (filtré)

Nous pouvons observer sur la figure 14 une nette séparation entre les échantillons issus de sols métallifères et ceux issus de sols non-métallifères pour ce cluster à trois domaines. La répartition selon le type de sous-échantillon (rhizosphère ou sol) ne semble également pas aléatoire, bien que ces deux groupes se recouvrent partiellement.

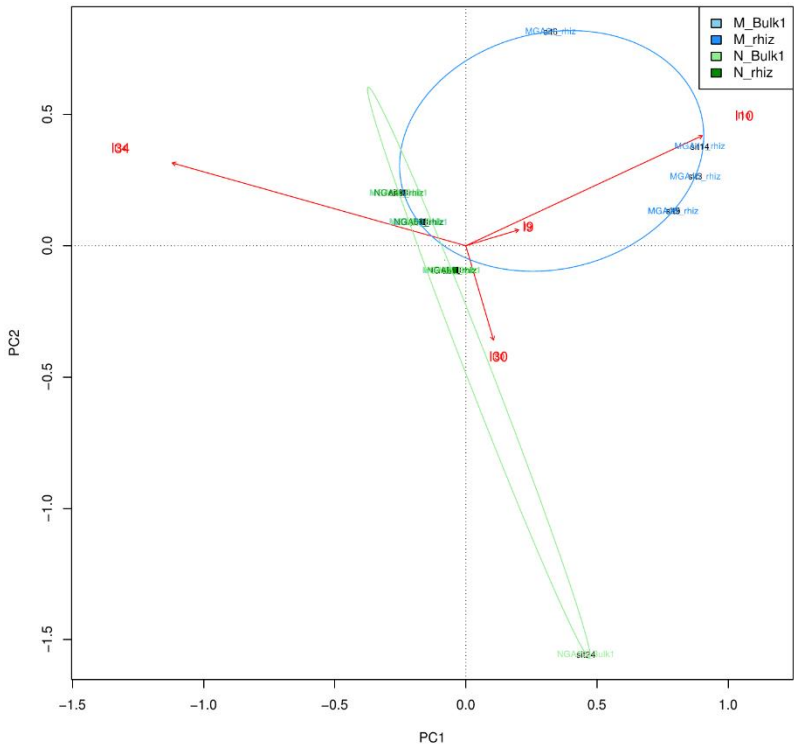


Figure 15 : Carte ACP du cluster lyse_ilt-P31545 (filtré)

La figure 15 semble montrer pour ce cluster bactérien et eucaryote un détachement des d'échantillons de sols métallifères des autres échantillons, bien que peu de données y figurent.

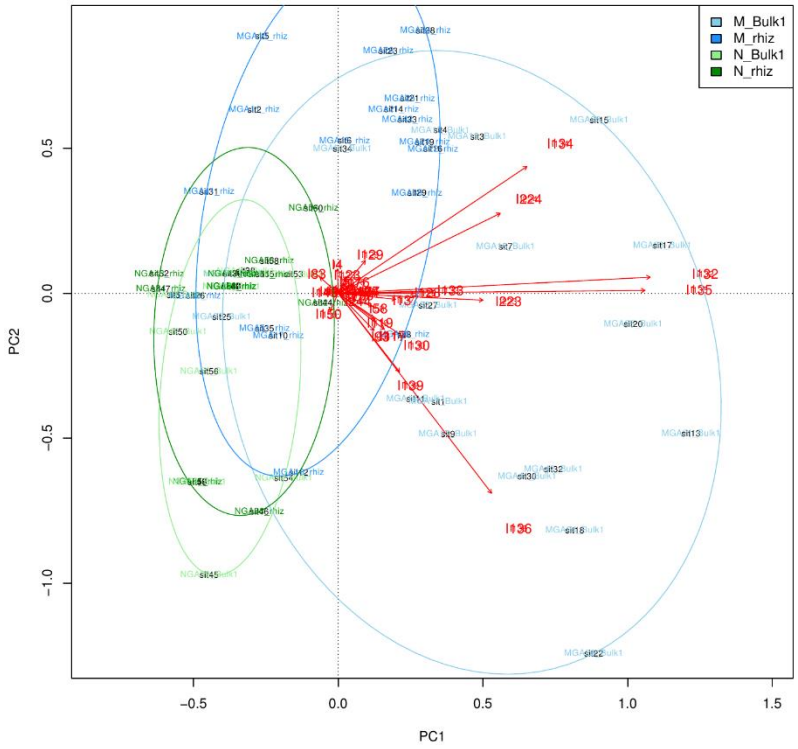


Figure 16 : Carte ACP du cluster lyse_ilt-P38933 (filtré)

Pour ce cluster, également de la famille Lyse_ILT (bactérien et eucaryote), une séparation est moins flagrante. On observe en effet un recouvrement plus important des catégories d'échantillons sur la figure 16. Cependant, là où les échantillons métalliques semblent dispersés, les échantillons non-métalliques semblent regroupés au bord de la distribution des autres échantillons.

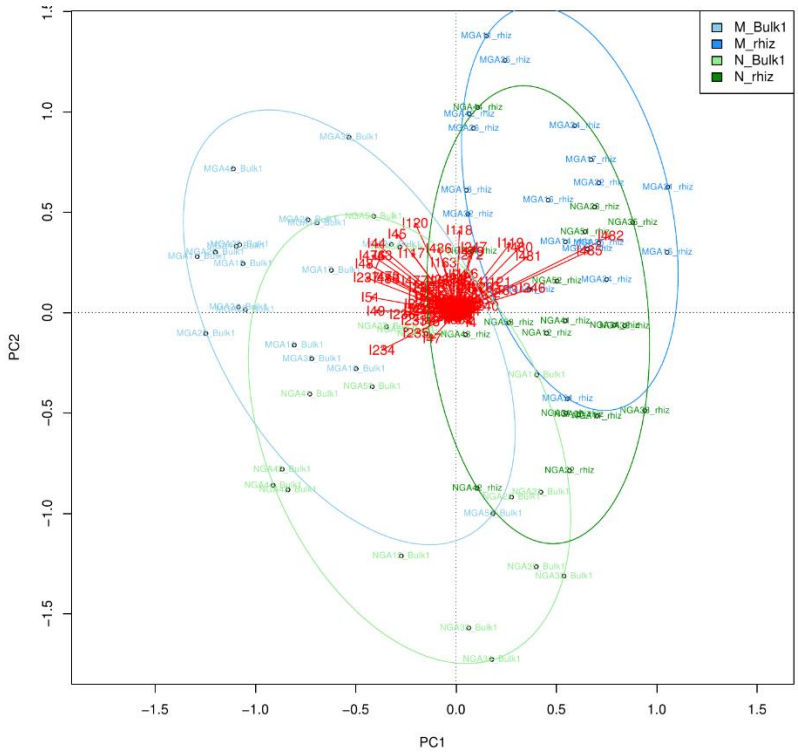


Figure 17 : Carte ACP du cluster nramp-P38925 (filtré)

En ce qui concerne ce cluster à trois domaines (figure 17), les catégories d'échantillons se regroupent partiellement, mais quatre pôles apparaissent. Ainsi, les quatre catégories d'échantillons semblent préférentiellement placées dans un cadrant différent des autres.

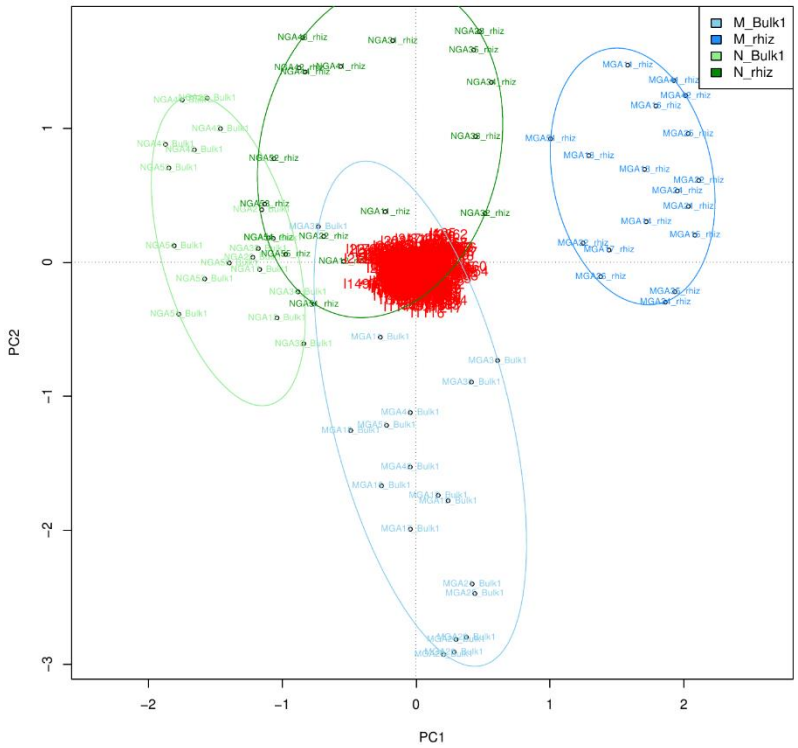


Figure 18 : Carte ACP du cluster ptype_atpase-P13587 (filtré)

La figure 18 montre une séparation quasiment parfaite des quatre catégories d'échantillons. Il n'y a presque aucun recouvrement des catégories sur cette carte ACP du cluster à trois domaines ptype_atpase-P13587.

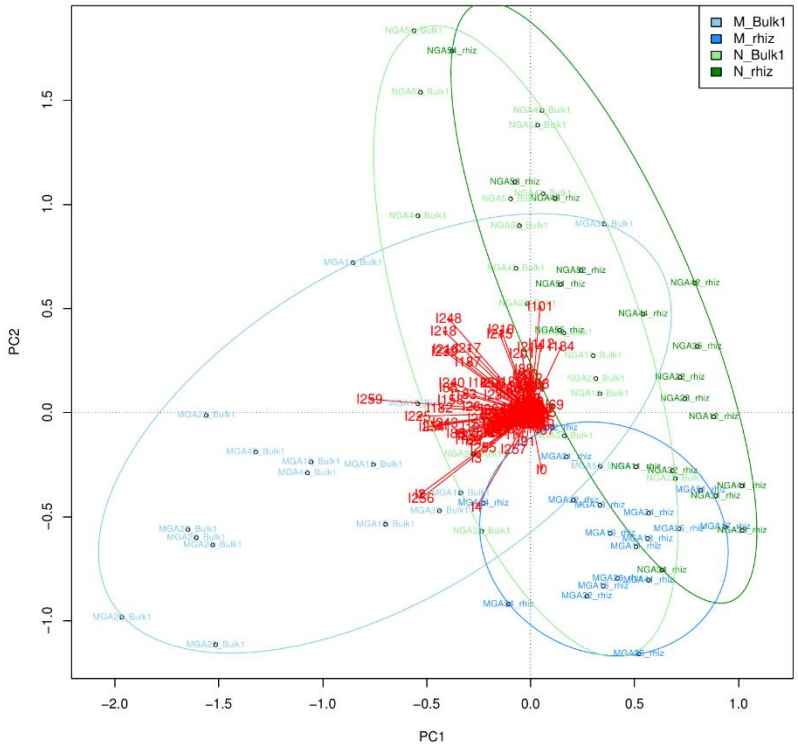


Figure 19 : Carte ACP du cluster rnd-P13510 (filtré)

La figure 19 semble présenter une séparation entre les échantillons métalliques et non-métalliques pour ce cluster bactérien. La majorité des échantillons de la catégorie M-Bulk1 est particulièrement mise à l'écart par l'ACP.

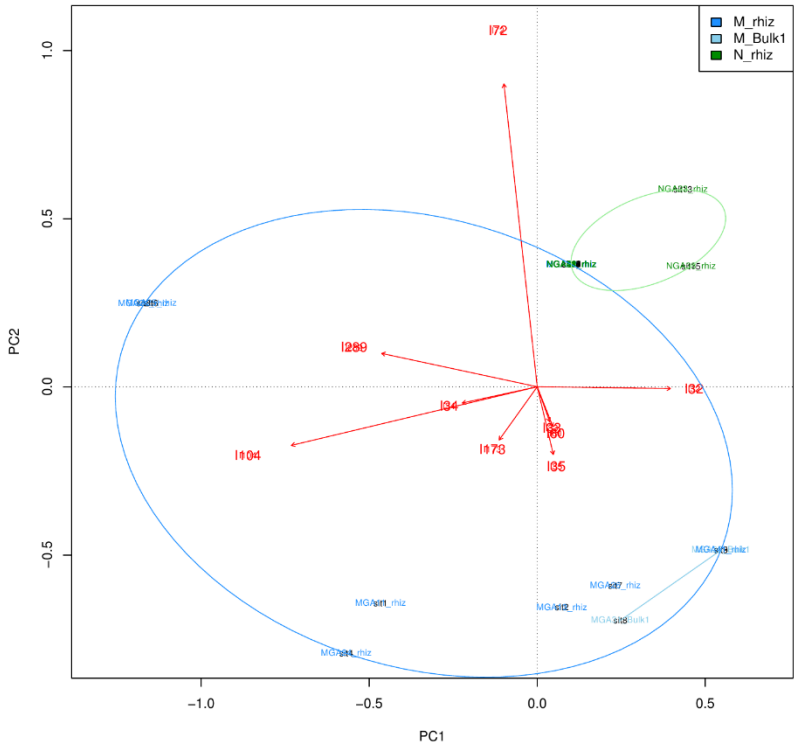


Figure 20 : Carte ACP du cluster zip-P0A8H3 (filtré)

L'ACP de ce cluster bactérien et eucaryote semble montrer une séparation entre des échantillons métalliques dispersés et des échantillons non-métalliques regroupés (figure 20).

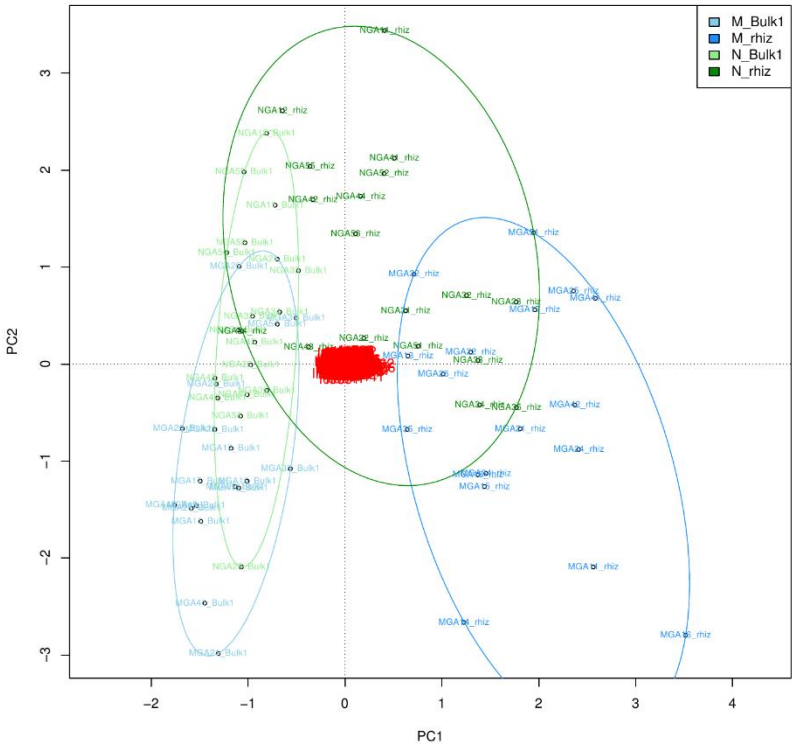


Figure 21 : Carte ACP du cluster abc_fet-P44513 (non-filtré)

Sur cette carte ACP d'un cluster à trois domaines (figure 21), les échantillons de rhizosphères métalliques semblent se détacher nettement des échantillons de rhizosphères non-métalliques. Les échantillons de sols ont plus tendance à être regroupés. Enfin, nous pouvons noter que la métallicité du sol apparaît accentuer les différences entre échantillons du sol et des rhizosphères.

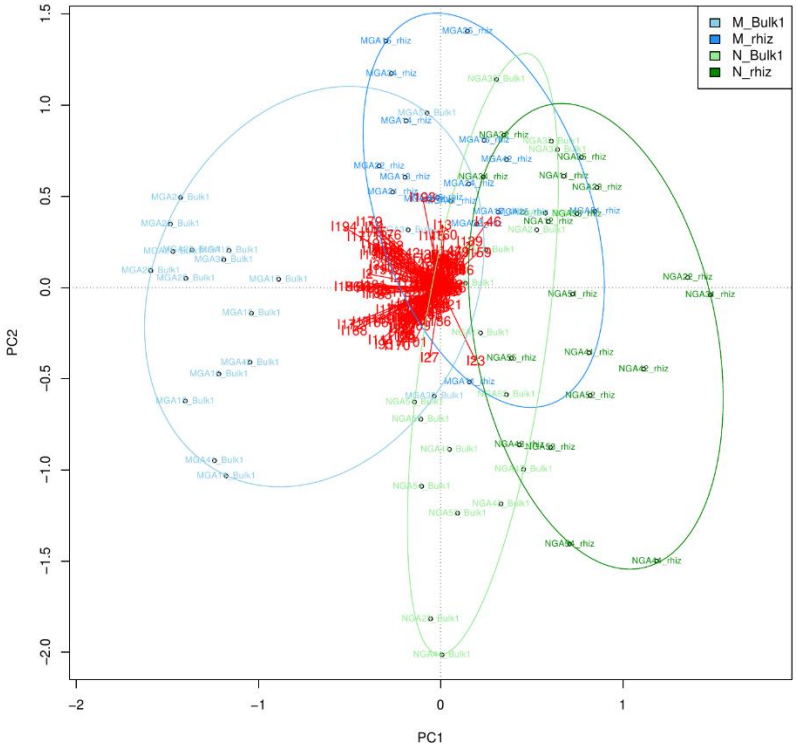


Figure 22 : Carte ACP du cluster rnd-P37972 (non-filtré)

Ce cluster bactérien, à nouveau de la famille RND semble, semble montrer une légère séparation des quatre catégories d'échantillons, bien que celles-ci se recouvrent partiellement (figure 22).

4.5. Tests de permutation

Pour appuyer les observations visuelles basées sur les cartes ACP, les résultats bruts de ces dernières ont servi à la réalisation d'un test statistique de permutation ou PERMANOVA, dont les résultats sont repris dans le tableau 4.

Cluster	Kaiser euclidean		2PC euclidean		Kaiser Manhattan		2PC Manhattan	
	N-M	R-B	N-M	R-B	N-M	R-B	N-M	R-B
cdf-P13512	0.001*	0.001*	0.001*	0.001*	0.001*	0.001*	0.001*	0.001*
lyse_ilt-P31545	/	/	0.001*	0.001*	/	/	0.001*	0.001*
lyse_ilt-P38993	0.001*	0.003*	0.001*	0.400	0.001*	0.005*	0.001*	0.389
abc_fet-P44513	0.260	0.419	0.002*	0.001*	0.258	0.440	0.002*	0.001*
rnd-P37972	0.002*	0.105	0.032	0.001*	0.001*	0.083	0.026	0.001*
nramp-P38925	0.001*	0.001*	0.002*	0.001*	0.005*	0.001*	0.004*	0.001*
ptype_atpase-P13587	0.456	0.701	0.008	0.001*	0.492	0.710	0.003*	0.001*
rnd-P13510	0.001*	0.002*	0.001*	0.002*	0.001*	0.001*	0.001*	0.002*
zip-P0A8H3	/	/	0.003*	0.527	/	/	0.001*	0.509

Tableau 4 : p-value des tests de permutation PERMANOVA calculés sur les résultats d'ACP des clusters prometteurs. Deux paramètres ont été soumis à variation : le nombre de composantes principales prises en compte et la méthode de calcul des distances. En orange, le partitionnement selon la métallicité des échantillons. En bleu, le partitionnement selon le sous-échantillon (rhiz/Bluk1). Les astérisques indiquent une significativité selon le seuil alpha de 0.05 / 9 = ***0.00556*** *(correction de Bonferroni).*

Premièrement, lorsque le test statistique est calculé à partir des deux premières composantes principales, les résultats confirment les observations visuelles de sept clusters, mais pas celles des clusters ptype_atpase-P13587 et rnd-P37972. En d'autres termes, les nœuds des arbres de référence sur lesquels des séquences ont été placées ont significativement contribué à différencier les échantillons métalliques des échantillons non-métalliques dans sept cas sur neuf. Néanmoins, à part une logique visuelle et intuitive, il n'y a pas de raison particulière à considérer seulement deux composantes principales, surtout lorsque la part de variance expliquée par celles-ci est faible. C'est pourquoi la règle de Kaiser a également été considérée. Dans ce cas, toutes les PC dont la eigenvalue est supérieure à 1 sont prises en compte. Les clusters n'ayant pas de résultat pour les tests paramétrés avec la règle de Kaiser sont ceux n'ayant pas de composante principale d'une eigenvalue de 1 ou plus. Toutes les composantes principales de ces clusters peuvent être considérées comme peu informatives. Remarquons que le partitionnement selon la métallicité des échantillons perd sa significativité pour le cluster abc_fet-P44513 avec ce paramètre, mais qu'il devient significatif pour le cluster rnd-P37927. Notons également que la méthode de calcul des distances n'a pas beaucoup d'effet sur les résultats. Seul le partitionnement selon la métallicité des échantillons (2 PC prises en compte) pour ptype_atpase-P13587, déjà à la limite de la significativité, l'a atteint.

Deuxièmement, le partitionnement des échantillons selon le type de sous-échantillon est significatif, peu importe le nombre de PC prises en compte dans le cas des clusters cdf-P13512, nramp-P38925 et rnd-P13510. Dans le cas de abc_fet-P44513, rnd-P37972 et de ptype_atpase-P13587, ce partitionnement n'est significatif qu'en considérant deux PC. À l'inverse, c'est la règle de Kaiser qui donne un test statistiquement significatif dans le cas du cluster lyse_ilt-P38993. Cependant, une seule PC possède une eigenvalue supérieure à un dans ce cluster, le test n'en considère alors qu'une. Les tests statistiques de permutation ont ici tendance à perdre leur significativité en prenant en compte plus de PC. En d'autres termes, ces tests sont moins puissants en rajoutant des paramètres expliquant de moins en moins de variance, ce qui correspond au problème de surajustement.

5. Discussion

5.1. Matériel de départ

5.1.1. Reconstitution du projet

Au début de ce mémoire, beaucoup de temps a été dépensé dans le but de retracer les acquisitions et transformations de données, ainsi que pour remettre de l'ordre dans les dossiers d'Amandine Bertrand. J'ai pu constater personnellement que de telles épreuves d' « archéologie » ralentissent considérablement la transition opérationnelle lors de la reprise d'un projet en cours. Bien que n'ayant pas de lien avec les conclusions techniques et biologiques de ce mémoire, cette constatation est importante pour mon développement professionnel.

5.1.2. Alignements de référence et clusters

Pour gagner du temps et consacrer plus d'efforts à la réalisation de nouvelles taches d'analyse, j'ai réutilisé les alignements de référence produits par Amandine Bertrand. Ces derniers résultent d'un enrichissement de 34 clusters issus de 21 familles de transporteurs de métaux contenant cent-trente-cinq séquences piochées dans la littérature par deux méthodes différentes (Forty-Two et hmmer) depuis la même base de données (life-tqmd-of73). Ces alignements n'étaient pas parfaits : Ils avaient été faits de manière automatique, sans curation manuelle, comportaient des doublons et plusieurs formats de noms différents. L'idéal aurait été de reconstruire les alignements un par un jusqu'à obtention de clusters enrichis d'un nombre satisfaisant de séquences vérifiées. Cependant, cette option n'était pas viable dans le cadre d'un mémoire. A la place, un effort a été fait pour nettoyer les alignements des éventuels doublons (CD-HIT), formater correctement et de manière unique les noms des séquences et les réaligner avec un algorithme plus lent, mais produisant des alignements multiples de qualité supérieure (MAFFT L-INS-I). Dans de futures itérations du projet, des alignements de référence pourraient être recréés de 0, à partir de banques de données plus riches, pour une meilleure résolution des arbres de référence.

5.2. Standardisation des données

La standardisation des données a été plutôt conservative. Quatorze échantillons sur quatre-vingt-six ont été écartés, laissant dix-huit échantillons par catégorie pour la suite des analyses. Une attention particulière a été portée, d'une part, à ne pas faire disparaître un site d'échantillonnage complet du set d'échantillons et, d'autre part, à ne pas réduire à 1 le nombre d'échantillons d'un sous-type (rhiz/Bulk1) représentant un site.

Le sous-échantillonnage aléatoire des *scaffolds* des échantillons retenus a mené à la perte de nombreuses séquences, réduisant le nombre de *scaffolds* parfois d'un facteur 3. Cependant, la quantité de données restantes était plutôt conséquente, avec environ 1 400 000 *scaffolds* par échantillon. Le sous-échantillonnage aléatoire s'est basé sur le nombre de *scaffolds* contenus dans les différents fichiers de séquençage, mais il aurait pu se baser sur le nombre de bases. Les résultats auraient également pu être standardisés en aval plutôt que de standardiser les données en amont. Ces alternatives présentent toutes des avantages et des inconvénients, comme la perte de données pour

la standardisation des données en amont, ou l'impossibilité de comparer les résultats bruts ou de faire une analyse visuelle, pour la normalisation des résultats en aval.

5.3. Rajouts de séquences environnementales orthologues

L'utilisation du programme Forty-Two peut encore être raffinée. Le paramétrage peut être poussé jusqu'à la personnalisation pour chaque alignement multiple. La solution de répartition en cinq groupes retenue ici était un bon compromis entre temps disponible et paramétrage fin. L'idéal serait d'exécuter le programme sans aucun avertissement lié aux organismes de référence problématiques. Il faudrait pour cela utiliser un set de *queries* et un set d'organismes de référence adapté à chaque alignement afin que les *Best hits* trouvés par les *queries* dans chaque organisme de référence soit en BRH avec les *Best hits* trouvés dans les autres organismes de référence. Ce problème des BRH peut également être dû à la structure des arbres de référence. Si ceux-ci incluent des paralogues, des erreurs de BRH sont inévitables. Le raffinement des alignements de référence pourrait également apporter une assurance d'éviter l'ajout de séquences paralogues.

Une autre imperfection réside dans la grande quantité de placements alternatifs dans l'arbre de référence attribués par RAxML aux nouvelles séquences. Un arbre de référence de meilleure résolution, et/ou un placement des séquences environnementales par ML plutôt que par MP devraient aider à résoudre le problème. Un réglage plus agressif de la filtration des séquences par ali2phylip.pl pourrait également être envisagé, bien que quelques tentatives préliminaires (non montrées ici) ont mené à une réduction trop importante du nombre de séquences. Un traitement itératif de plusieurs filtrations pourrait être envisagé pour conserver le plus de colonnes informatives en retirant le plus de séquences problématiques.

5.4. Analyse en composantes principales

L'analyse en composantes principales est une méthode d'analyse conceptuellement bien adaptée à notre cas dans ce projet. Il faut néanmoins garder à l'esprit que l'ACP présente quelques conditions d'applications. L'ACP assume la linéarité des variables. Celles-ci devraient avoir des relations linéaires entre elles. Dans notre cas, ces relations sont difficiles à évaluer de par le faible volume de données contenu dans chaque variable. Ainsi, la condition de linéarité des variables n'est pas forcément respectée. Une analyse NMDS (Non-Metric MultiDimentional Scaling) constituerait une alternative rigoureuse. En effet, cette méthode d'ordination des données est non-métrique, elle ne nécessite donc pas la linéarité des données, ni la normalité de celles-ci. Elle capture les relations complexes non-linéaires et respecte les différences entre les individus statistiques. La NMDS étire et comprime certaines distances de sorte à ordiner au mieux les données tout en respectant le rang de chaque différence (méthode non-métrique). La raison du choix de l'ACP tient simplement au fait qu'elle marchait mieux que la NMDS, c'est-à-dire pour plus de clusters, tout en donnant les mêmes résultats pour ceux où les deux méthodes marchaient.

Une transformation logarithme en base 10 a été appliquée sur les données dans le but d'éviter la domination de l'influence des nœuds fréquents. Cette transformation génère des différences d'ordres de grandeur plutôt que d'analyser les différences brutes, ce qui laisse plus de place à l'expression de l'influence d'espèces rares. Un pseudocompte de 0.1 a également été ajouté pour éviter les valeurs

nulles problématiques dans une transformation logarithmique. La valeur de 0.1 a été choisie pour éviter de trop fausser les données parfois de cet ordre de grandeur.

Un problème de cette analyse est l'interprétation des nœuds des arbres de référence sur lesquels des séquences environnementales orthologues ont été placées. En effet, il est souvent difficile de distinguer des flèches se démarquant et expliquant une certaine part de variance sur les cartes ACP montrées dans ce travail. Les nœuds sont alors difficiles à lire et les organismes associés ne sont pas repris dans les interprétations biologiques. Une solution envisagée pour résoudre ce problème serait de traiter les données de placement des nouvelles séquences dans les arbres de référence par groupes de nœuds. Ces groupes seraient formés par logique taxonomique et renverraient à une interprétation biologique plus aisée. De manière générale, des groupes de nœuds proches se retrouvent souvent enrichis dans nos arbres, ce qui est accentué par les placements alternatifs de certaines séquences environnementales. Celles-ci concernent très souvent des nœuds proches.

Les tests statistiques utilisés pour soutenir les observations visuelles semblent confirmer un partitionnement des échantillons selon la métallicité des sols en ce qui concerne le contenu en transporteurs de métaux des clusters cdf-P13512, lyse_ilt-P38993, nramp-P38925, rnd-P37972 et rnd-P13510. Ces clusters ont tous en commun une partie bactérienne dans leurs alignements et arbres de référence.

La non-significativité du test de permutations du cluster ptype_atpase-P13587 est surprenante. La famille des ATPases de type P a en effet déjà été liée à l'hyperaccumulation des métaux lourds chez les plantes, notamment via le gène *HMA4* (Claus et al., 2013; Corso et al., 2018; Hanikenne et al., 2008; Hanikenne & Nouet, 2011; Pollard et al., 2014). Il s'agissait également du cluster où le partitionnement selon la catégorie d'échantillons semblait visuellement le plus clair sur la carte ACP. Une explication à cela pourrait être le nombre démesuré de composantes principales dont la eigenvalue est supérieure à 1 et qui sont par conséquent prises en compte dans la règle de Kaiser. En effet, 68 PC sont utilisées dans les calculs par le test statistique. Ceci correspond à un problème bien connu qu'est le surajustement. Ici, le modèle est basé sur un grand nombre de paramètre, qui peuvent bruiter l'analyse s'ils sont non pertinents. Ajoutons à cela une correction agressive et conservative des seuils de significativité dans le contexte des tests PERMANOVA multiples. En effet, la correction de Bonferroni a été appliquée, comme plus tôt dans ce mémoire lors de l'identification des échantillons métagénomiques *outliers*. D'autres méthodes de correction du seuil de significativité moins strictes comme la méthode de Benjamini-Hochberg (ordonner les *p-values* et les ajuster en fonction de leur rang et du nombre total de tests) ou la méthode de permutation (permuter aléatoirement les données et créer une distribution nulle des *p-values* pour évaluer la puissance de nos *p-values* et les ajuster) existent et pourraient être utilisées. Ces méthodes éliminent un nombre modeste de vrais positifs en éliminant une large proportion de faux positifs. La plus sensible et performantes des deux étant la méthode de permutation (Shuken & McNerney, 2023). Une étude plus approfondie de la famille des ATPases de type P pourrait aider à tirer des conclusions claires.

5.5. Modèle d'analyse produit

En ce qui concerne l'objectif technique du mémoire, soit tester la méthode d'analyse et établir une sorte de protocole d'analyse, le modèle initial du *workflow* du projet peut être étendu et inclure les parties « standardisation des données » et « Criblage des familles de transporteurs de métaux » (figure 23).

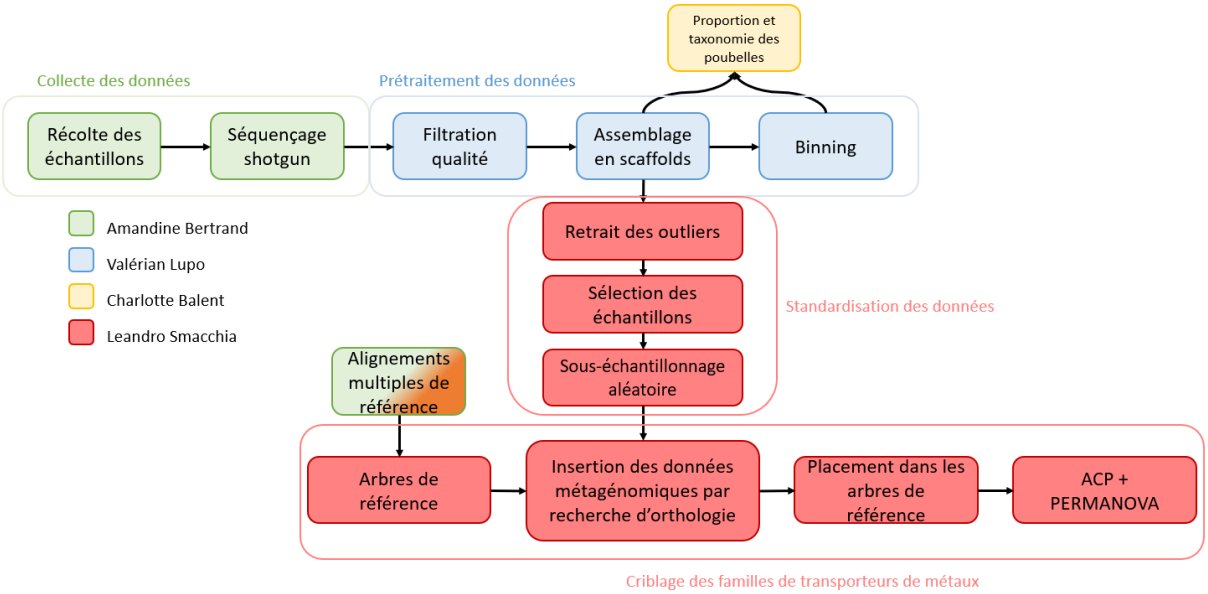


Figure 23 : Pipeline d'analyse de ce mémoire, inscrit dans la continuité du projet MetaRhizoMet+ entrepris par Amandine Bertrand

La partie criblage des familles de transporteurs de métaux peut être personnalisée et affinée pour correspondre à une famille ou un cluster particulier, objet d'un approfondissement de l'analyse.

5.6. Perspectives

Les familles CDF, Lyse_ILT, Nramp, RND et P-type ATPase semblent les plus prometteuses. C'est sur ces familles que les analyses futures devront se concentrer dans le cadre de ce projet. Il sera alors surement profitable de recréer un alignement enrichi en séquences de référence pour les groupes taxonomiques correspondant aux rajouts et de refaire les différentes analyses avec des paramétrages personnalisés à chaque alignement multiple.

Les organismes sous-tendant les différences entre les catégories d'échantillons des clusters prometteurs devront être investigués dans de futures analyses. Il s'agit en réalité de l'étape suivante dans l'analyse des placements par ACP, je n'ai simplement pas eu le temps de rajouter ce volet à ce mémoire.

6. Conclusion

Au cours de ce mémoire, les données métagénomiques assemblées à partir de séquençage *shotgun* d'échantillons de rhizosphères et de sols métallifères et non-métallifères ont été standardisées, rendant l'analyse des quatre groupes d'échantillons possible par comparaison directe.

L'insertion de séquences orthologues issues de ces échantillons métagénomiques dans des arbres de référence largement échantillonnés a mené à l'identification de familles de transporteurs de métaux prometteurs dans le cadre de l'investigation de la composante microbienne du caractère d'hyperaccumulation facultative d'*Arabidopsis halleri*. Les analyses en composantes principales ont montré que les clusters des familles Nramp, Lyse_ILT, ATPase de type P, CDF et RND, tous trois ayant une composante bactérienne, présentent un partitionnement des échantillons en accord avec la métallicité de l'environnement dont ils sont issus. Les tests statistiques de permutation ou PERMANOVA ont soutenu ces observations dans le cas des clusters des familles Nramp, Lyse_ILT, CDF et RND.

Aucune conclusion claire n'a pour l'instant été tirée sur les organismes ou groupes microbiens exerçant une influence sur cette séparation des différentes catégories d'échantillons (M-Bulk1 ; M-rhiz ; N-Bulk1 ; N-rhiz). L'échantillonnage taxonomique large dans les trois domaines implique une faible résolution des organismes de référence, ce qui explique en partie cette dernière constatation. Une grande proportion de *reads* et de *contigs* non identifiés a été écartée lors de l'assemblage des données métagénomiques. Ce manque de données pourrait également avoir un impact sur la difficulté à tirer des conclusions biologiques sur des taxons microbiens.

Ce mémoire valide une technique d'analyse innovante et jette les bases de futures recherches sur les familles de transporteurs de métaux prometteuses identifiées. Un échantillonnage plus dense et spécifique à chaque famille, ainsi qu'un réglage fin de la recherche par orthologie sont deux pistes faciles à explorer en se basant sur le *pipeline* d'analyse proposé dans ce mémoire.

L'hyperaccumulation facultative est un caractère fascinant et rempli de promesses écologiques de dépollution et de phytoremédiation. Comprendre ce caractère et son fonctionnement permettrait d'en tirer parti pour compenser les effets néfastes de l'industrie sur l'environnement. Ce projet est un pas de plus vers une compréhension de ce phénomène, qui ne dépend pas uniquement de la plante, dans sa globalité.

7. Bibliographie

Becher, M., Talke, I. N., Krall, L., & Krämer, U. (2004). Cross-species microarray transcript profiling reveals high constitutive expression of metal homeostasis genes in shoots of the zinc hyperaccumulator Arabidopsis halleri. *The Plant Journal: For Cell and Molecular Biology*, *37*(2), 251-268. https://doi.org/10.1046/j.1365-313x.2003.01959.x

Benizri, E., & Kidd, P. S. (2018). The Role of the Rhizosphere and Microbes Associated with Hyperaccumulator Plants in Metal Accumulation. In A. Van der Ent, G. Echevarria, A. J. M. Baker, & J. L. Morel (Éds.), *Agromining : Farming for Metals : Extracting Unconventional Resources Using Plants* (p. 157-188). Springer International Publishing. https://doi.org/10.1007/978-3-319-61899-9_9

Boyd, R., & Martens, S. (1998). Nickel hyperaccumulation by Thlaspi montanum var. montanum (Brassicaceae) : A constitutive trait. *American Journal of Botany*, *85*(2), 259.

Claus, J., Bohmann, A., & Chavarría-Krauser, A. (2013). Zinc uptake and radial transport in roots of Arabidopsis thaliana : A modelling approach to understand accumulation. *Annals of Botany*, *112*(2), 369-380. https://doi.org/10.1093/aob/mcs263

Clemens, S., Aarts, M. G. M., Thomine, S., & Verbruggen, N. (2013). Plant science : The key to preventing slow cadmium poisoning. *Trends in Plant Science*, *18*(2), 92-99. https://doi.org/10.1016/j.tplants.2012.08.003

Corso, M., Schvartzman, M. S., Guzzo, F., Souard, F., Malkowski, E., Hanikenne, M., & Verbruggen, N. (2018). Contrasting cadmium resistance strategies in two metallicolous populations of Arabidopsis halleri. *The New Phytologist*, *218*(1), 283-297. https://doi.org/10.1111/nph.14948

de Souza, M. P., Huang, C. P. A., Chee, N., & Terry, N. (1999). Rhizosphere bacteria enhance the accumulation of selenium and mercury in wetland plants. *Planta*, *209*(2), 259-263. https://doi.org/10.1007/s004250050630

Docherty, K. M., Borton, H. M., Espinosa, N., Gebhardt, M., Gil-Loaiza, J., Gutknecht, J. L. M., Maes, P. W., Mott, B. M., Parnell, J. J., Purdy, G., Rodrigues, P. A. P., Stanish, L. F., Walser, O. N., &

Gallery, R. E. (2015). Key Edaphic Properties Largely Explain Temporal and Geographic Variation in Soil Microbial Communities across Four Biomes. *PloS One*, *10*(11), e0135352. https://doi.org/10.1371/journal.pone.0135352

Hall, J. L., & Williams, L. E. (2003). Transition metal transporters in plants. *Journal of Experimental Botany*, *54*(393), 2601-2613. https://doi.org/10.1093/jxb/erg303

Hanikenne, M., & Nouet, C. (2011). Metal hyperaccumulation and hypertolerance : A model for plant evolutionary genomics. *Current Opinion in Plant Biology*, *14*(3), 252-259. https://doi.org/10.1016/j.pbi.2011.04.003

Hanikenne, M., Talke, I. N., Haydon, M. J., Lanz, C., Nolte, A., Motte, P., Kroymann, J., Weigel, D., & Krämer, U. (2008). Evolution of metal hyperaccumulation required cis-regulatory changes and triplication of HMA4. *Nature*, *453*(7193), 391-395. https://doi.org/10.1038/nature06877

Hautekeete, N., Decombeix, I., Bouchet, M.-H., CREACH, A., Saumitou-Laprade, P., Piquot, Y., & Pauwels, M. (2018). Habitat heterogeneity in the pseudometallophyte Arabidopsis halleri and its structuring effect on natural variation of zinc and cadmium hyperaccumulation. *Plant and Soil*, *423*. https://doi.org/10.1007/s11104-017-3509-1

Hiraoka, S., Yang, C.-C., & Iwasaki, W. (2016). Metagenomics and Bioinformatics in Microbial Ecology : Current Status and Beyond. *Microbes and Environments*, *31*(3), 204-212. https://doi.org/10.1264/jsme2.ME16024

Honeker, L. K., Gullo, C. F., Neilson, J. W., Chorover, J., & Maier, R. M. (2019). Effect of Re-acidification on Buffalo Grass Rhizosphere and Bulk Microbial Communities During Phytostabilization of Metalliferous Mine Tailings. *Frontiers in Microbiology*, *10*, 1209. https://doi.org/10.3389/fmicb.2019.01209

Islam, W., Noman, A., Naveed, H., Huang, Z., & Chen, H. Y. H. (2020). Role of environmental factors in shaping the soil microbiome. *Environmental Science and Pollution Research International*, *27*(33), 41225-41247. https://doi.org/10.1007/s11356-020-10471-2

Klimek, B., Stępniewska, K., Seget, B., Pandey, V. C., & Babst-Kostecka, A. (2023). Diversity and activity of soil biota at a post-mining site highly contaminated with Zn and Cd are enhanced by metallicolous compared to non-metallicolous Arabidopsis halleri ecotypes. *Land Degradation & Development*, *34*(5), 1538-1548. https://doi.org/10.1002/ldr.4551

Krämer, U. (2010). Metal hyperaccumulation in plants. *Annual Review of Plant Biology*, *61*, 517-534. https://doi.org/10.1146/annurev-arplant-042809-112156

Kumar, A., & Verma, J. P. (2018). Does plant-Microbe interaction confer stress tolerance in plants : A review? *Microbiological Research*, *207*, 41-52. https://doi.org/10.1016/j.micres.2017.11.004

Kushwaha, P., Neilson, J. W., Maier, R. M., & Babst-Kostecka, A. (2022). Soil microbial community and abiotic soil properties influence Zn and Cd hyperaccumulation differently in Arabidopsis halleri. *The Science of the Total Environment*, *803*, 150006. https://doi.org/10.1016/j.scitotenv.2021.150006

Lazzaro, A., Widmer, F., Sperisen, C., & Frey, B. (2008). Identification of dominant bacterial phylotypes in a cadmium-treated forest soil. *FEMS Microbiology Ecology*, *63*(2), 143-155. https://doi.org/10.1111/j.1574-6941.2007.00417.x

Léonard, R. R., Leleu, M., Van Vlierberghe, M., Cornet, L., Kerff, F., & Baurain, D. (2021). ToRQuEMaDA : Tool for retrieving queried Eubacteria, metadata and dereplicating assemblies. *PeerJ*, *9*, e11348. https://doi.org/10.7717/peerj.11348

Liddicoat, C., Bi, P., Waycott, M., Glover, J., Breed, M., & Weinstein, P. (2018). Ambient soil cation exchange capacity inversely associates with infectious and parasitic disease risk in regional Australia. *The Science of the Total Environment*, *626*, 117-125. https://doi.org/10.1016/j.scitotenv.2018.01.077

Lopez, S., Goux, X., Echevarria, G., Calusinska, M., Morel, J. L., & Benizri, E. (2019). Community diversity and potential functions of rhizosphere-associated bacteria of nickel hyperaccumulators found in Albania. *The Science of the Total Environment*, *654*, 237-249. https://doi.org/10.1016/j.scitotenv.2018.11.056

Merlot, S., Garcia de la Torre, V. S., & Hanikenne, M. (2021). Physiology and Molecular Biology of Trace Element Hyperaccumulation. In A. van der Ent, A. J. M. Baker, G. Echevarria, M.-O. Simonnot, & J. L. Morel (Éds.), *Agromining : Farming for Metals : Extracting Unconventional Resources Using Plants* (p. 155-181). Springer International Publishing. https://doi.org/10.1007/978-3-030-58904-2_8

Meyer, C.-L., Juraniec, M., Huguet, S., Chaves-Rodriguez, E., Salis, P., Isaure, M.-P., Goormaghtigh, E., & Verbruggen, N. (2015). Intraspecific variability of cadmium tolerance and accumulation, and cadmium-induced cell wall modifications in the metal hyperaccumulator Arabidopsis halleri. *Journal of Experimental Botany*, *66*(11), 3215-3227. https://doi.org/10.1093/jxb/erv144

Meyer, C.-L., Pauwels, M., Briset, L., Godé, C., Salis, P., Bourceaux, A., Souleman, D., Frérot, H., & Verbruggen, N. (2016). Potential preadaptation to anthropogenic pollution : Evidence from a common quantitative trait locus for zinc and cadmium tolerance in metallicolous and nonmetallicolous accessions of Arabidopsis halleri. *The New Phytologist*, *212*(4), 934-943. https://doi.org/10.1111/nph.14093

Mishra, A., Singh, L., & Singh, D. (2023). Unboxing the black box-one step forward to understand the soil microbiome : A systematic review. *Microbial Ecology*, *85*(2), 669-683. https://doi.org/10.1007/s00248-022-01962-5

Murawska-Wlodarczyk, K., Korzeniak, U., Chlebicki, A., Mazur, E., Dietrich, C. C., & Babst-Kostecka, A. (2022). Metalliferous habitats and seed microbes affect the seed morphology and reproductive strategy of Arabidopsis halleri. *Plant and Soil*, *472*(1-2), 175-192. https://doi.org/10.1007/s11104-021-05203-5

Myrold, D. D., & Nannipieri, P. (2013). *Classical Techniques Versus Omics Approaches*.

Pollard, A. J., Reeves, R. D., & Baker, A. J. M. (2014). Facultative hyperaccumulation of heavy metals and metalloids. *Plant Science: An International Journal of Experimental Plant Biology*, *217-218*, 8-17. https://doi.org/10.1016/j.plantsci.2013.11.011

Putra, R., & Müller, C. (2023). Extending the elemental defence hypothesis in the light of plant chemodiversity. *New Phytologist*, *239*(5), 1545-1555. https://doi.org/10.1111/nph.19071

Qian, J., Li, D., Zhan, G., Zhang, L., Su, W., & Gao, P. (2012). Simultaneous biodegradation of Ni–citrate complexes and removal of nickel from solutions by Pseudomonas alcaliphila. *Bioresource Technology*, *116*, 66-73. https://doi.org/10.1016/j.biortech.2012.04.017

Schvartzman, M. S., Corso, M., Fataftah, N., Scheepers, M., Nouet, C., Bosman, B., Carnol, M., Motte, P., Verbruggen, N., & Hanikenne, M. (2018). Adaptation to high zinc depends on distinct mechanisms in metallicolous populations of Arabidopsis halleri. *The New Phytologist*, *218*(1), 269-282. https://doi.org/10.1111/nph.14949

Shuken, S. R., & McNerney, M. W. (2023). Costs and Benefits of Popular P-Value Correction Methods in Three Models of Quantitative Omic Experiments. *Analytical Chemistry*, *95*(5), 2732-2740. https://doi.org/10.1021/acs.analchem.2c03719

Talke, I. N., Hanikenne, M., & Krämer, U. (2006). Zinc-dependent global transcriptional control, transcriptional deregulation, and higher gene copy number for genes in metal homeostasis of the hyperaccumulator Arabidopsis halleri. *Plant Physiology*, *142*(1), 148-167. https://doi.org/10.1104/pp.105.076232

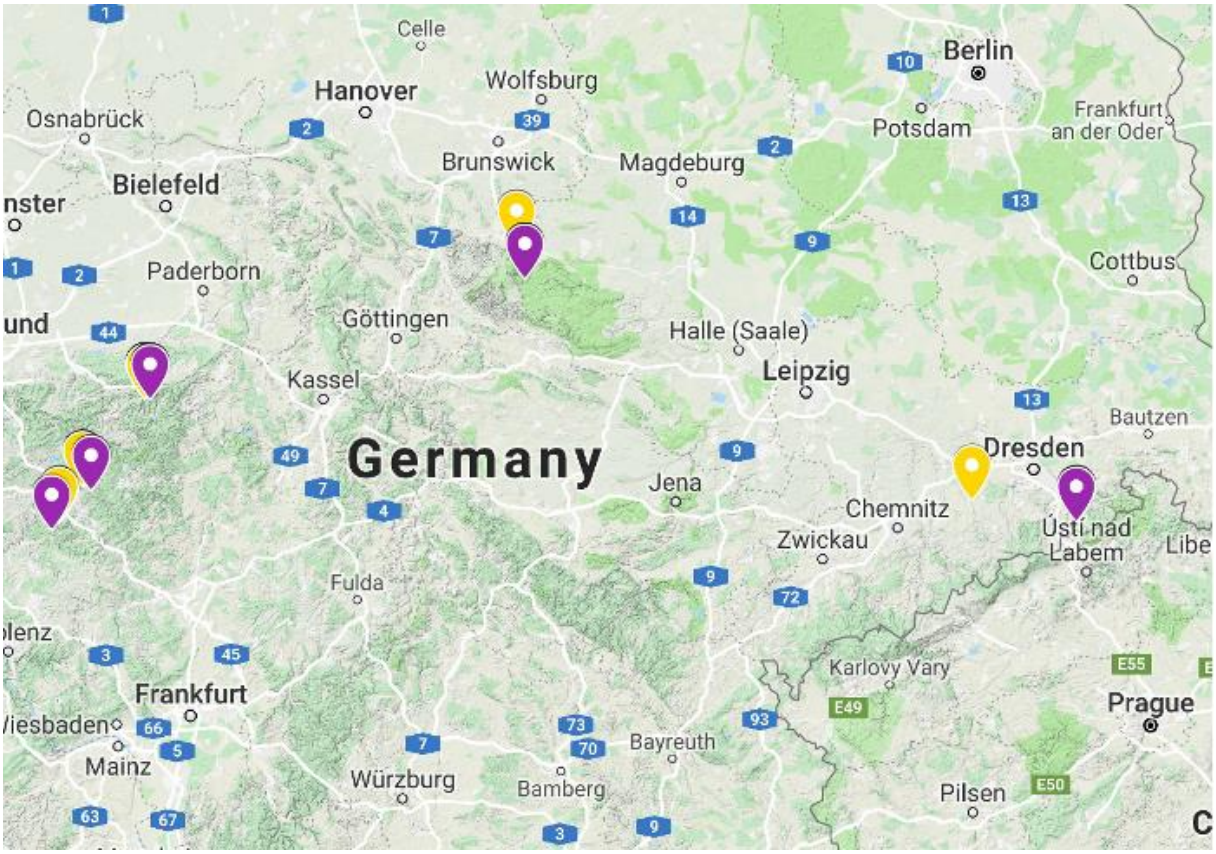
Tipayno, S., Kim, C.-G., & Sa, T. (2012). T-RFLP analysis of structural changes in soil bacterial communities in response to metal and metalloid contamination and initial phytoremediation. *Applied Soil Ecology*, *61*, 137-146. https://doi.org/10.1016/j.apsoil.2012.06.001

Valentín-Vargas, A., Neilson, J. W., Root, R. A., Chorover, J., & Maier, R. M. (2018). Treatment impacts on temporal microbial community dynamics during phytostabilization of acid-generating mine tailings in semiarid regions. *The Science of the Total Environment*, *618*, 357-368. https://doi.org/10.1016/j.scitotenv.2017.11.010

Van Vlierberghe, M., Philippe, H., & Baurain, D. (2021). Broadly sampled orthologous groups of eukaryotic proteins for the phylogenetic study of plastid-bearing lineages. *BMC Research Notes*, *14*(1), 143. https://doi.org/10.1186/s13104-021-05553-4

8. Annexes

8.1. Liste des échantillons et sites d'échantillonnage



Carte des sites d'échantillonnage des sols pollués aux métaux (jaune) et des sols non pollués aux métaux (mauve) sur lesquels poussent Arabidopsis halleri.

Soil_type	Zip_code	Square	Subsample	Soil_type	Zip_code	Square	Subsample
M	9599	51	bulk1	N	1816	51	bulk1
M	9599	51	bulk2	N	1816	51	bulk2
M	9599	51	root	N	1816	51	root
M	9599	51	shoot	N	1816	51	shoot
M	38871	41	bulk1	N	1816	52	bulk1
M	38871	41	bulk2	N	1816	52	bulk2
M	38871	41	root	N	1816	52	root
M	38871	41	shoot	N	1816	52	shoot
M	38871	42	bulk1	N	1816	53	bulk1
M	38871	42	bulk2	N	1816	53	bulk2
M	38871	42	root	N	1816	53	root
M	38871	42	shoot	N	1816	53	shoot
M	57223	21	bulk1	N	1816	54	bulk1
M	57223	21	bulk2	N	1816	54	bulk2
M	57223	21	root	N	1816	54	root
M	57223	21	shoot	N	1816	54	shoot
M	57223	22	bulk1	N	1816	55	bulk1

M	57223	22	bulk2	N	1816	55	bulk2
M	57223	22	root	N	1816	55	root
M	57223	22	shoot	N	1816	55	shoot
M	57223	23	bulk1	N	38879	41	bulk1
M	57223	23	bulk2	N	38879	41	bulk2
M	57223	23	root	N	38879	41	root
M	57223	23	shoot	N	38879	41	shoot
M	57223	24	bulk1	N	38879	42	bulk1
M	57223	24	bulk2	N	38879	42	bulk2
M	57223	24	root	N	38879	42	root
M	57223	24	shoot	N	38879	42	shoot
M	57223	25	bulk1	N	38879	43	bulk1
M	57223	25	bulk2	N	38879	43	bulk2
M	57223	25	root	N	38879	43	root
M	57223	25	shoot	N	38879	43	shoot
M	57223	26	bulk1	N	38879	44	bulk1
M	57223	26	bulk2	N	38879	44	bulk2
M	57223	26	root	N	38879	44	root
M	57223	26	shoot	N	38879	44	shoot
M	57572	11	bulk1	N	57250	21	bulk1
M	57572	11	bulk2	N	57250	21	root
M	57572	11	root	N	57250	21	shoot
M	57572	11	shoot	N	57250	22	bulk1
M	57572	12	bulk1	N	57250	22	root
M	57572	12	bulk2	N	57250	22	shoot
M	57572	12	root	N	57250	23	bulk1
M	57572	12	shoot	N	57250	23	bulk2
M	57572	13	bulk1	N	57250	23	root
M	57572	13	bulk2	N	57250	23	shoot
M	57572	13	root	N	57548	11	bulk1
M	57572	13	shoot	N	57548	11	root
M	57572	14	bulk1	N	57548	11	shoot
M	57572	14	bulk2	N	57548	12	bulk1
M	57572	14	root	N	57548	12	root
M	57572	14	shoot	N	57548	12	shoot
M	57572	15	bulk1	N	57548	13	bulk1
M	57572	15	bulk2	N	57548	13	root
M	57572	15	root	N	57548	13	shoot
M	57572	15	shoot	N	59909	31	bulk1
M	57572	16	bulk1	N	59909	31	bulk2
M	57572	16	bulk2	N	59909	31	root
M	57572	16	root	N	59909	31	shoot
M	57572	16	shoot	N	59909	32	bulk1
M	57572	17	bulk1	N	59909	32	root
M	57572	17	bulk2	N	59909	32	shoot
M	57572	17	root	N	59909	33	bulk1
M	57572	17	shoot	N	59909	33	root
M	57572	18	bulk1	N	59909	33	shoot

M	57572	18	bulk2	N	59909	34	bulk1
M	57572	18	root	N	59909	34	bulk2
M	57572	18	shoot	N	59909	34	root
M	59909	31	bulk1	N	59909	34	shoot
M	59909	31	root	N	59909	35	bulk1
M	59909	31	shoot	N	59909	35	bulk2
M	59909	32	bulk1	N	59909	35	root
M	59909	32	bulk2	N	59909	35	shoot
M	59909	32	root				
M	59909	32	shoot				
M	59909	33	bulk1				
M	59909	33	bulk2				
M	59909	33	root				
M	59909	33	shoot				
M	59909	34	bulk1				
M	59909	34	bulk2				
M	59909	34	root				
M	59909	34	shoot				
M	59909	35	bulk1				
M	59909	35	root				
M	59909	35	shoot				
M	59909	36	bulk1				
M	59909	36	bulk2				
M	59909	36	root				
M	59909	36	shoot				

Liste des échantillons prélevés en Allemagne. À gauche, les échantillons provenant de sites pollués aux métaux. À droite, les échantillons provenant de sites non pollués aux métaux. Le code postal de chaque site est inscrit dans la colonne Zip_code. Le code à deux chiffres de chaque carré correspond à l'assemblage d'un numéro représentant le site d'échantillonnage et d'un numéro représentant un carré spécifique du site. 23 Carrés pollués aux métaux ont été échantillonnés contre 20 carrés non pollués aux métaux. 7 échantillons Bulk2 sont manquant dans 7 carrés non pollués, ainsi que 2 échantillons Bulk2 dans 2 carrés pollués.

8.2. Liste des familles de transporteurs de métaux retenues

- abc_fet
- abc_hmt
- abc_mzt
- abc_nate
- abc_nicot
- abc_pept
- cdf
- copd
- ctr
- lyse_ilt
- lyse_mntp
- lyse_nico
- lyse_terc
- mit
- nramp
- omf
- p-type_atpase
- rnd
- tog_mgte
- tog_nicot
- zip

8.3. Liste des protéines retenues pour chaque famille

abc_fet
Haemophilus influenzae@P35755
Haemophilus influenzae@P44513
Neisseria gonorrhoeae@Q5FA19
Serratia marcescens@P21408
Synechocystis sp.@P72827
Synechocystis sp.@Q55835

abc_hmt
Saccharomyces cerevisiae@P40416
Schizosaccharomyces pombe@Q02592

abc_mzt
Bacillus subtilis@O34610
Bacillus subtilis@O34946
Bacillus subtilis@O34966
Escherichia coli@P0A9X1
Escherichia coli@P39172
Escherichia coli@P39832
Salmonella typhimurium@Q8ZNV7
Streptococcus pneumoniae@P0A4G2
Streptococcus pyogenes@P0A4G4
Synechocystis sp.@Q55282
Treponema pallidum@P96116
Yersinia pestis@Q56952

abc_nate
Bacillus subtilis@P46903
Bacillus subtilis@P46904

abc_nicot
Rhodobacter capsulatus@D5AQY6
Rhodobacter capsulatus@D5AQY7
Rhodobacter capsulatus@D5AQY8
Rhodobacter capsulatus@D5AUZ7
Rhodobacter capsulatus@D5AUZ9
Rhodobacter capsulatus@O68104
Rhodobacter capsulatus@O68106
Salmonella typhimurium@Q05594
Salmonella typhimurium@Q05595
Salmonella typhimurium@Q05596
Salmonella typhimurium@Q05598

abc_pept
Escherichia coli@P0AFA9
Escherichia coli@P33590
Escherichia coli@P33591

Cdf
Bacillus subtilis@O07084
Cupriavidus metallidurans@P13512
Escherichia coli@P69380
Escherichia coli@P75757
Saccharomyces cerevisiae@P20107
Saccharomyces cerevisiae@P32798
Saccharomyces cerevisiae@P53735
Saccharomyces cerevisiae@Q03455
Schizosaccharomyces pombe@O14329

Copd
Escherichia coli@P76278
Pseudomonas syringae@P12377

Ctr
Saccharomyces cerevisiae@P38865
Saccharomyces cerevisiae@P49573
Saccharomyces cerevisiae@Q06686
Schizosaccharomyces pombe@O94722
Schizosaccharomyces pombe@Q9P7F9
Schizosaccharomyces pombe@Q9USV7

lyse_ilt
Bacillus subtilis@P39595
Escherichia coli@P0AB24
Escherichia coli@P31545
Escherichia coli@Q8XAS8
Saccharomyces cerevisiae@P38310
Saccharomyces cerevisiae@P38993
Saccharomyces cerevisiae@P40088
Saccharomyces cerevisiae@P43561
Schizosaccharomyces pombe@Q09919
Yersinia pestis@Q0WFT9

lyse_mntp
Escherichia coli@P76264

lyse_nico
Escherichia coli@P76425

lyse_terc
Escherichia coli@P42601

Mit
Bacillus subtilis@P40948
Escherichia coli@P0ABI4
Escherichia coli@P64423
Methanocaldococcus jannaschii@Q58439
Saccharomyces cerevisiae@P35724
Saccharomyces cerevisiae@Q01926
Saccharomyces cerevisiae@Q02783
Saccharomyces cerevisiae@Q08269
Salmonella typhimurium@P0A2R8
Salmonella typhimurium@Q9EYX5
Thermotoga maritima@Q9WZ31

Nramp
Bacillus subtilis@P96593
Deinococcus radiodurans@Q9RTP8
Escherichia coli@P0A769
Lactobacillus brevis@Q93V04
Mycobacterium tuberculosis@P9WIZ5
Saccharomyces cerevisiae@P38925
Saccharomyces cerevisiae@Q12078
Salmonella typhimurium@Q9RPF4
Schizosaccharomyces pombe@Q10177
Staphylococcus aureus@Q99UZ

Omf
Cupriavidus metallidurans@P13509
Cupriavidus metallidurans@P37974
Escherichia coli@P77211

p-type_atpase
Archaeoglobus fulgidus@O29777
Archaeoglobus fulgidus@O30085
Bacillus subtilis@O31688
Bacillus subtilis@O32219
Bacillus subtilis@O32220
Enterococcus hirae@P05425
Enterococcus hirae@P32113
Escherichia coli@P03960
Escherichia coli@P0ABB8
Escherichia coli@P37617
Escherichia coli@Q59385
Halobacterium salinarum@B0R9M0
Helicobacter pylori@Q59465
Legionella pneumophila@Q5ZWR1
Listeria monocytogenes@Q60048
Saccharomyces cerevisiae@P13586
Saccharomyces cerevisiae@P13587
Saccharomyces cerevisiae@P38360
Saccharomyces cerevisiae@P38929
Saccharomyces cerevisiae@P38995
Saccharomyces cerevisiae@P39986
Salmonella typhimurium@P36640
Salmonella typhimurium@Q8ZR95
Salmonella typhimurium@Q9ZHC7
Schizosaccharomyces pombe@O59868
Schizosaccharomyces pombe@P22189
Shigella sonnei@Q3YW59
Staphylococcus aureus@P20021
Staphylococcus aureus@Q7A3E6

Rnd
Cupriavidus metallidurans@P13510
Cupriavidus metallidurans@P13511
Cupriavidus metallidurans@P37972
Escherichia coli@P38054
Escherichia coli@P77214
Escherichia coli@P77239

tog_mgte
Enterococcus faecalis@Q830V1
Thermus thermophilus@Q5SMG8

tog_nicot
Cupriavidus necator@P23516
Helicobacter pylori@Q48262
Rhodococcus rhodochrous@P96454

Zip
Escherichia coli@P0A8H3
Saccharomyces cerevisiae@P32804
Saccharomyces cerevisiae@P34240
Saccharomyces cerevisiae@Q12436
Schizosaccharomyces pombe@O94639

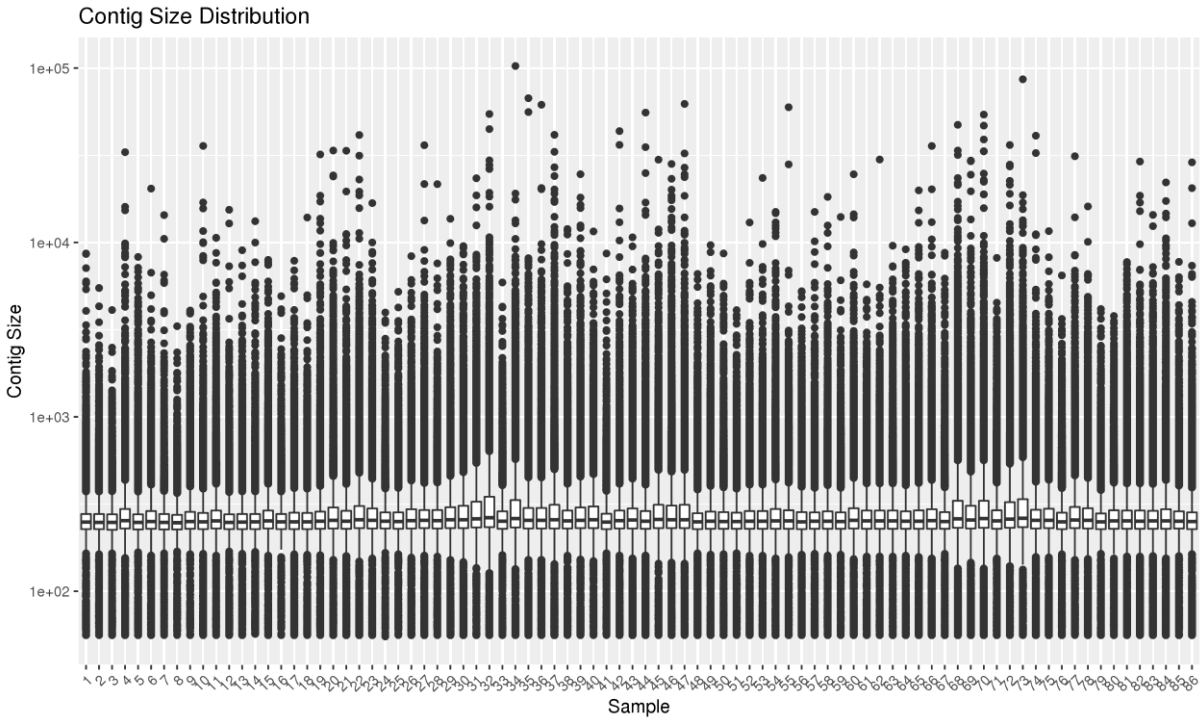
8.4. Liste des 34 clusters de séquences de référence initiales

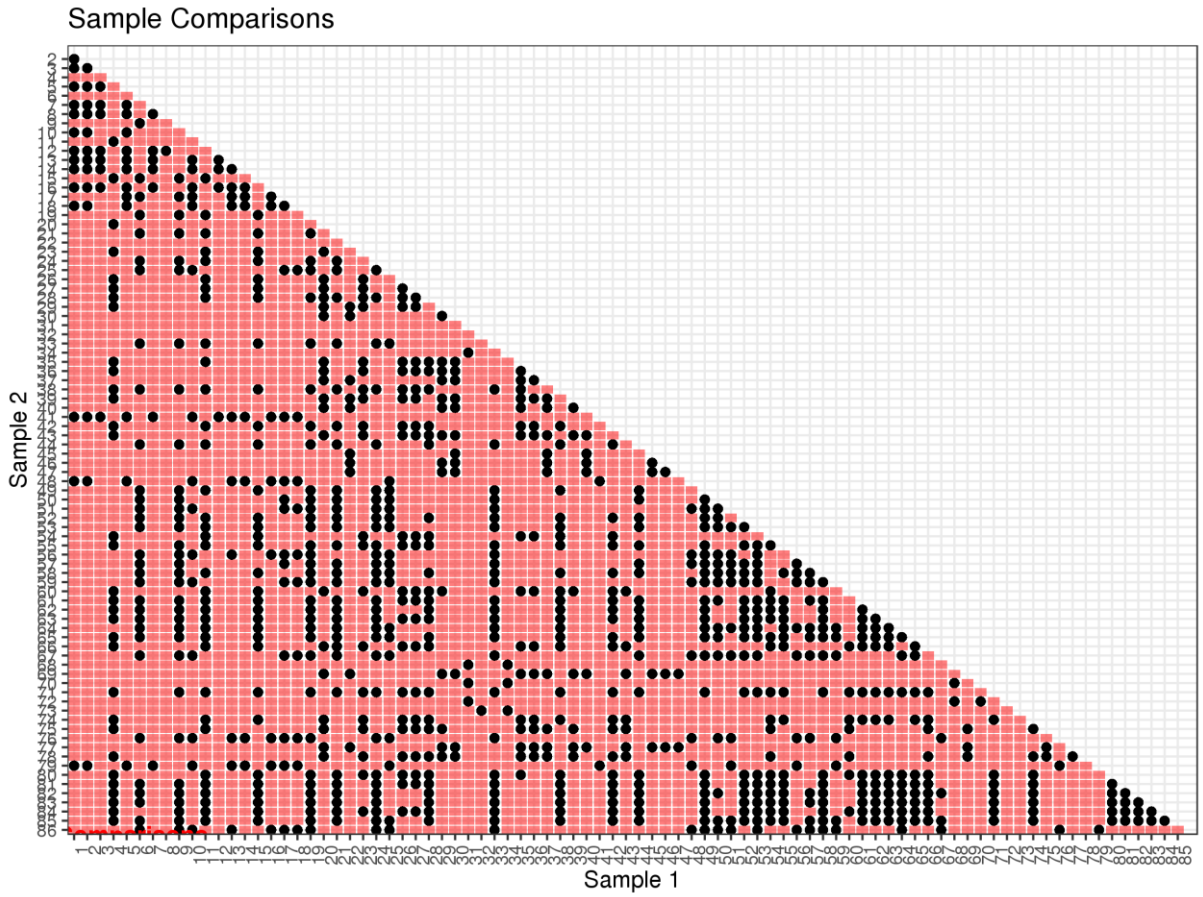
- abc_fet-P44513
- abc_fet-P72827
- abc_mzt-O34610
- abc_mzt-P0A4G4
- abc_nate-P46904
- abc_nicot-O68104
- abc_nicot-Q05594
- abc_nicot-Q05598
- abc_pept-P0AFA9
- abc_pept-P33590
- abc_pept-P33591
- cdf-P13512
- cdf-P53735
- copd-P12377
- ctr-Q9P7F9
- lyse_ilt-P31545
- lyse_ilt-P38993
- lyse_ilt-P40088
- lyse_ilt-Q0WFT9
- lyse_mntp-P76264
- lyse_nico-P76425
- lyse_terc-P42601
- mit-Q01926
- mit-Q58439
- nramp-P38925
- omf-P37974
- ptype_atpase-P13587
- rnd-P13510
- rnd-P37972
- rnd-P77214
- tog_mgte-Q5SMG8
- tog_nicot-P23516
- zip-O94639
- zip-P0A8H3

8.5. Liste des organismes constituant la *query* et les protéomes de référence (construction des alignements de référence)

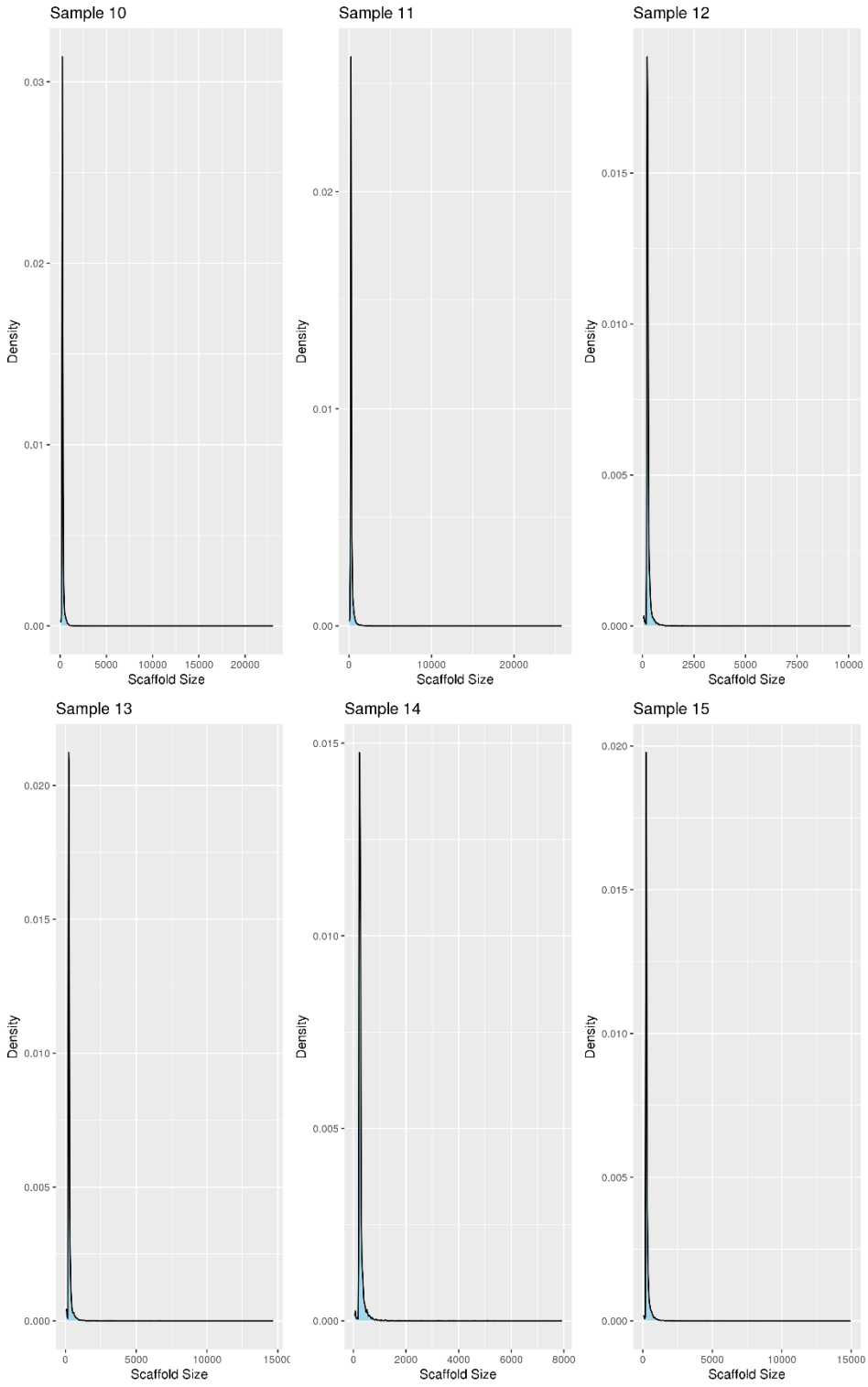
- Archaeoglobus fulgidus_224325
- Bacillus subtilis_224308
- Cupriavidus metallidurans_266264
- Cupriavidus necator_381666
- Deinococcus radiodurans_243230
- Enterococcus faecalis_226185
- Enterococcus hirae_768486
- Escherichia coli_83333
- Escherichia coli_83334
- Haemophilus influenzae_71421
- Halobacterium salinarum_478009
- Helicobacter pylori_85962
- Lactobacillus brevis_1580
- Legionella pneumophila_272624
- Listeria monocytogenes_1639
- Methanocaldococcus jannaschii_243232
- Mycobacterium tuberculosis_83332
- Neisseria gonorrhoeae_242231
- Pseudomonas syringae_323
- Rhodobacter capsulatus_272942
- Rhodococcus rhodochrous_1829
- Saccharomyces cerevisiae_559292
- Salmonella typhimurium_90371
- Salmonella typhimurium_99287
- Schizosaccharomyces pombe_284812
- Serratia marcescens_615
- Shigella sonnei_300269
- Staphylococcus aureus_1280
- Staphylococcus aureus_158879
- Streptococcus pneumoniae_170187
- Streptococcus pyogenes_301447
- Synechocystis sp._1111708
- Thermotoga maritima_243274
- Thermus thermophilus_300852
- Treponema pallidum_243276
- Yersinia pestis_632

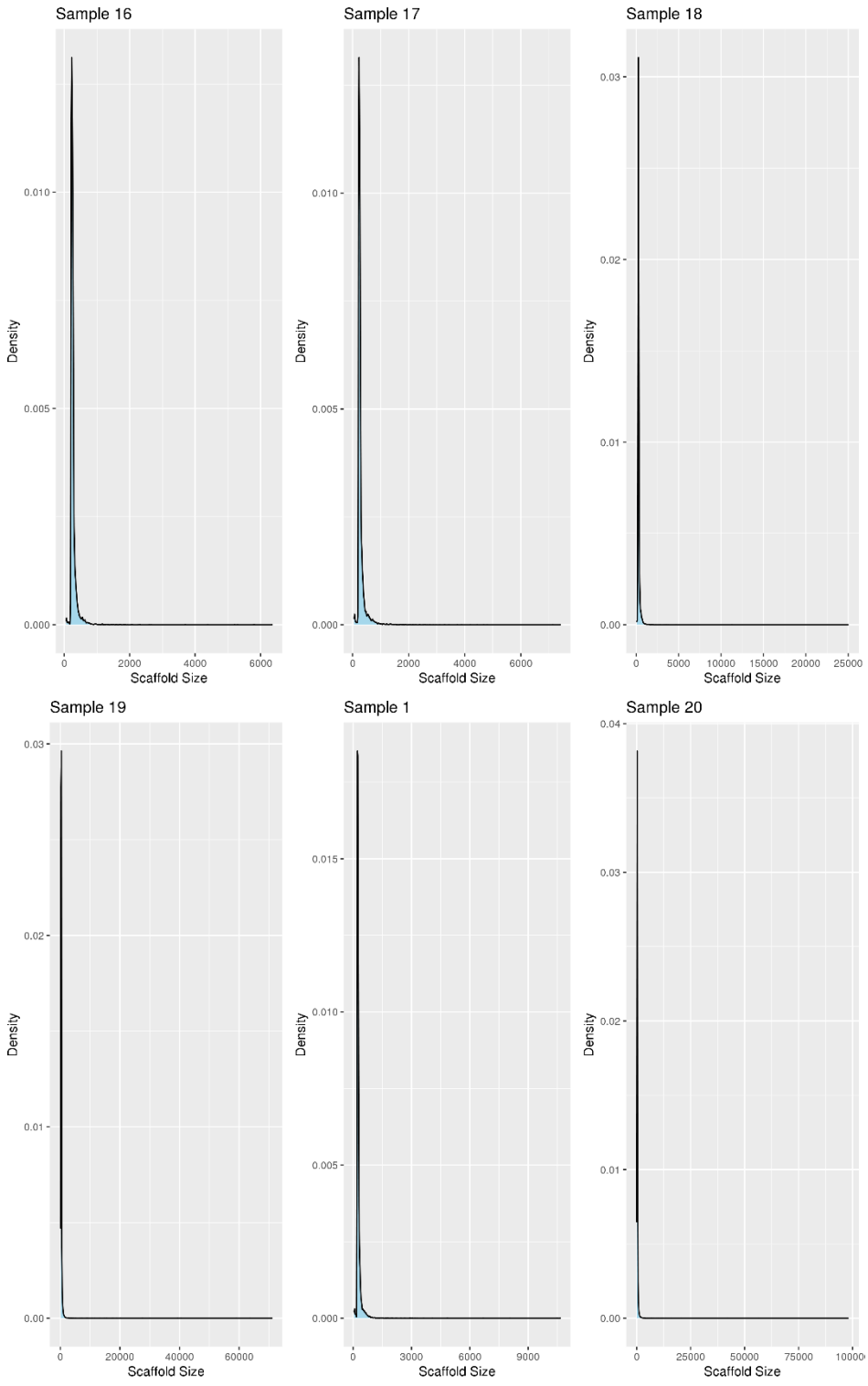
8.6. Analyse de la distribution des tailles de *contigs*



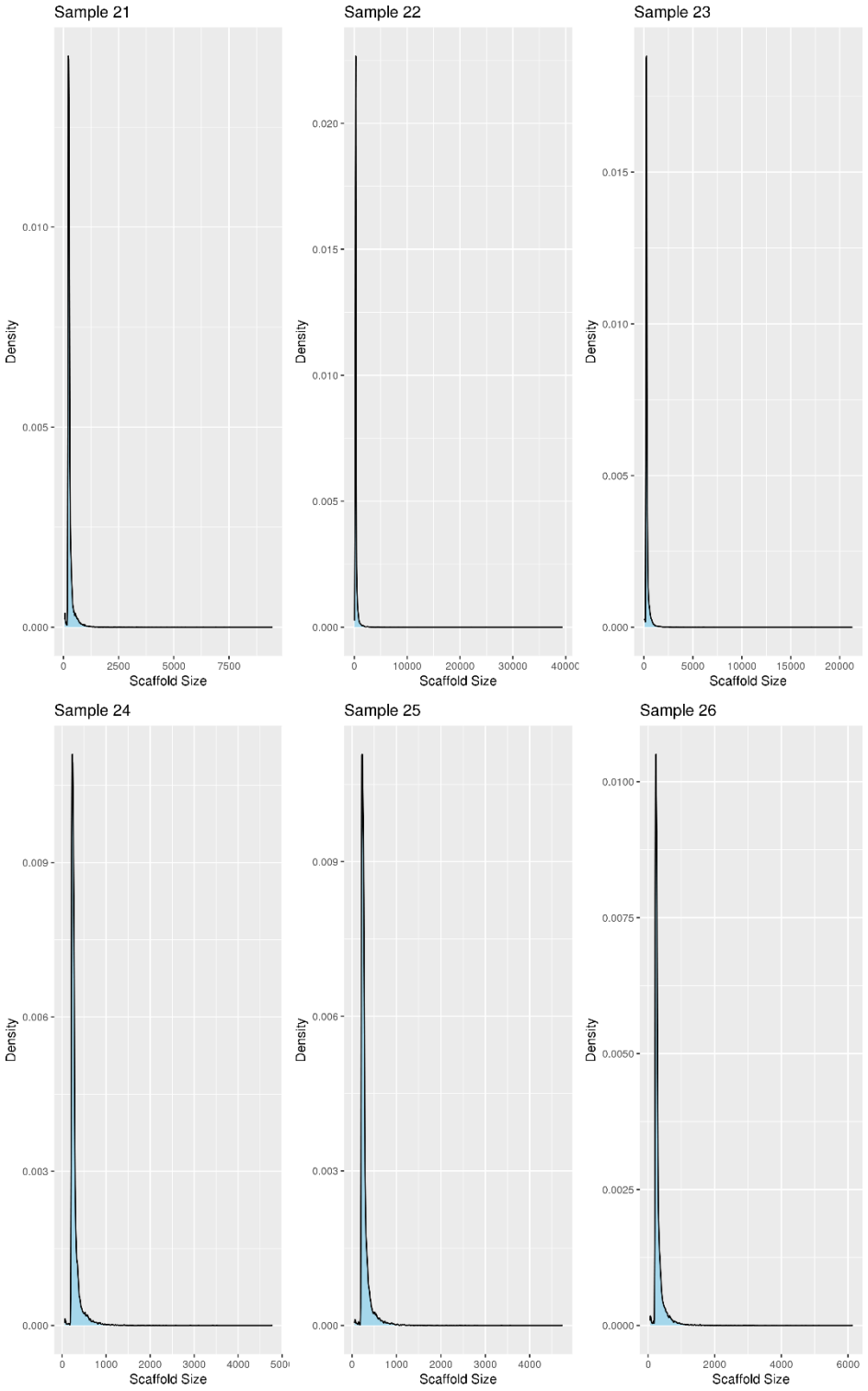


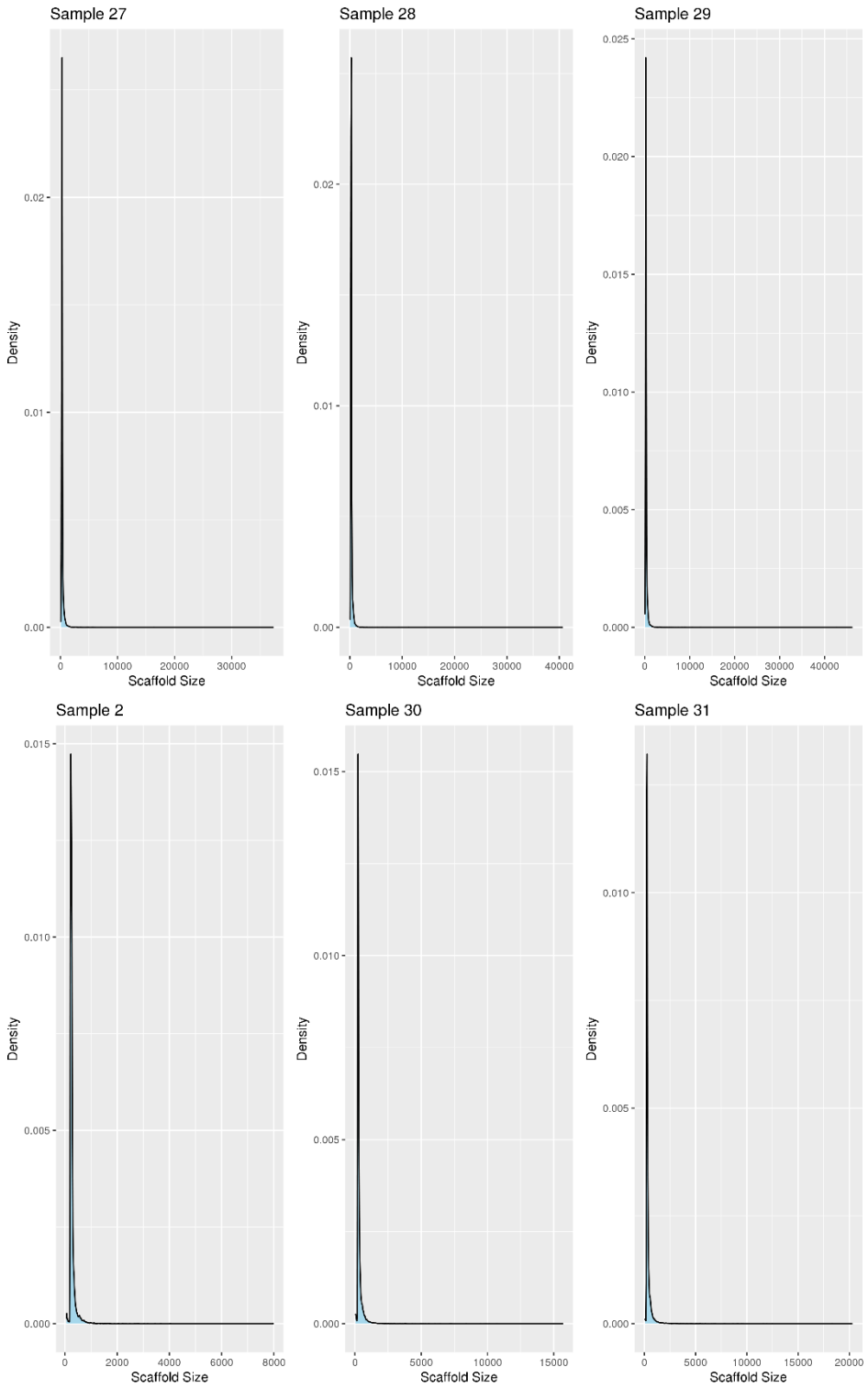
8.7. Distributions des tailles de *scaffolds*

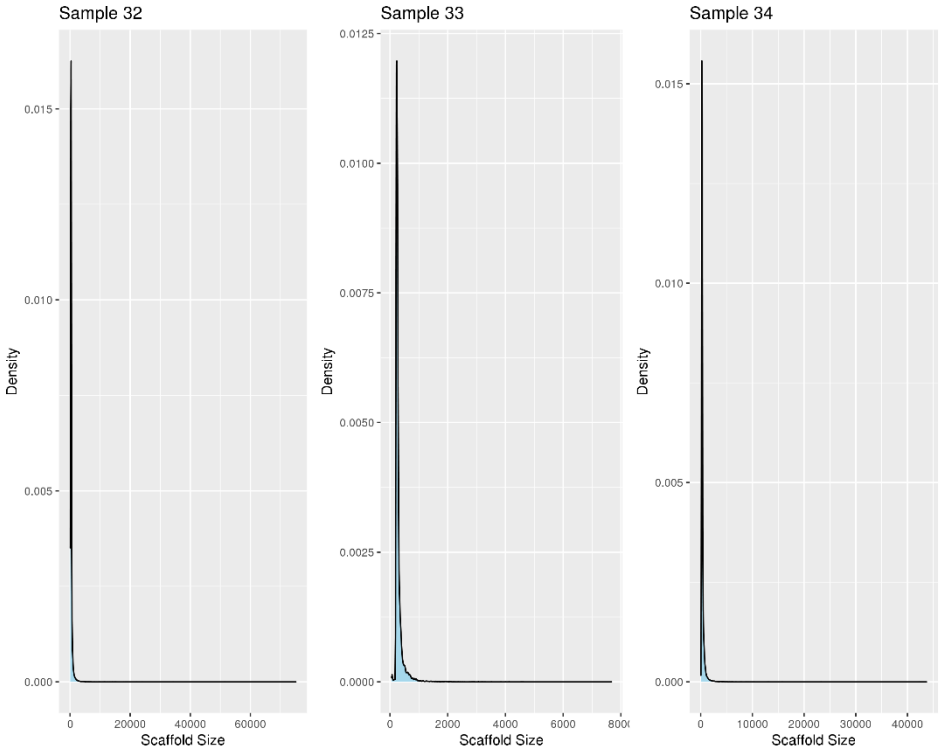


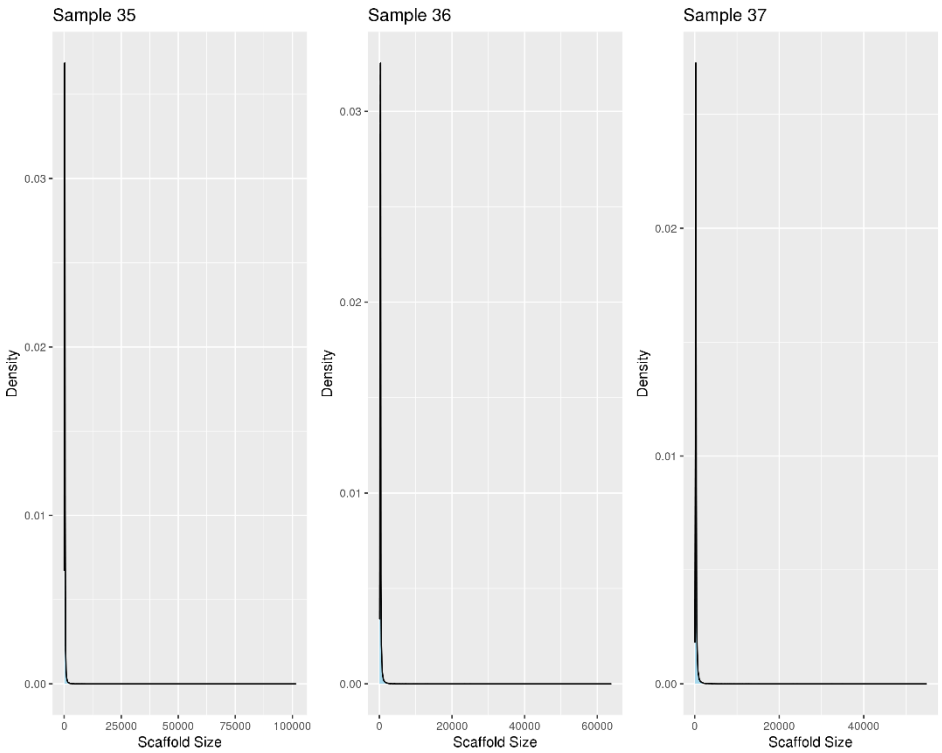

Sample 16
Density
0.010
0.005
0.000
0
2000
4000
6000
Scaffold Size
Sample 17
Density
0.010
0.005
0.000
0
2000
4000
6000
Scaffold Size
Sample 18
Density
0.03
0.02
0.01
0.00
0
5000
10000
15000
20000
25000
Scaffold Size
Sample 19
Density
0.03
0.02
0.01
0.00
0
20000
40000
60000
Scaffold Size
Sample 1
Density
0.015
0.010
0.005
0.000
0
3000
6000
9000
Scaffold Size
Sample 20
Density
0.04
0.03
0.02
0.01
0.00
0
25000
50000
75000
Scaffold Size

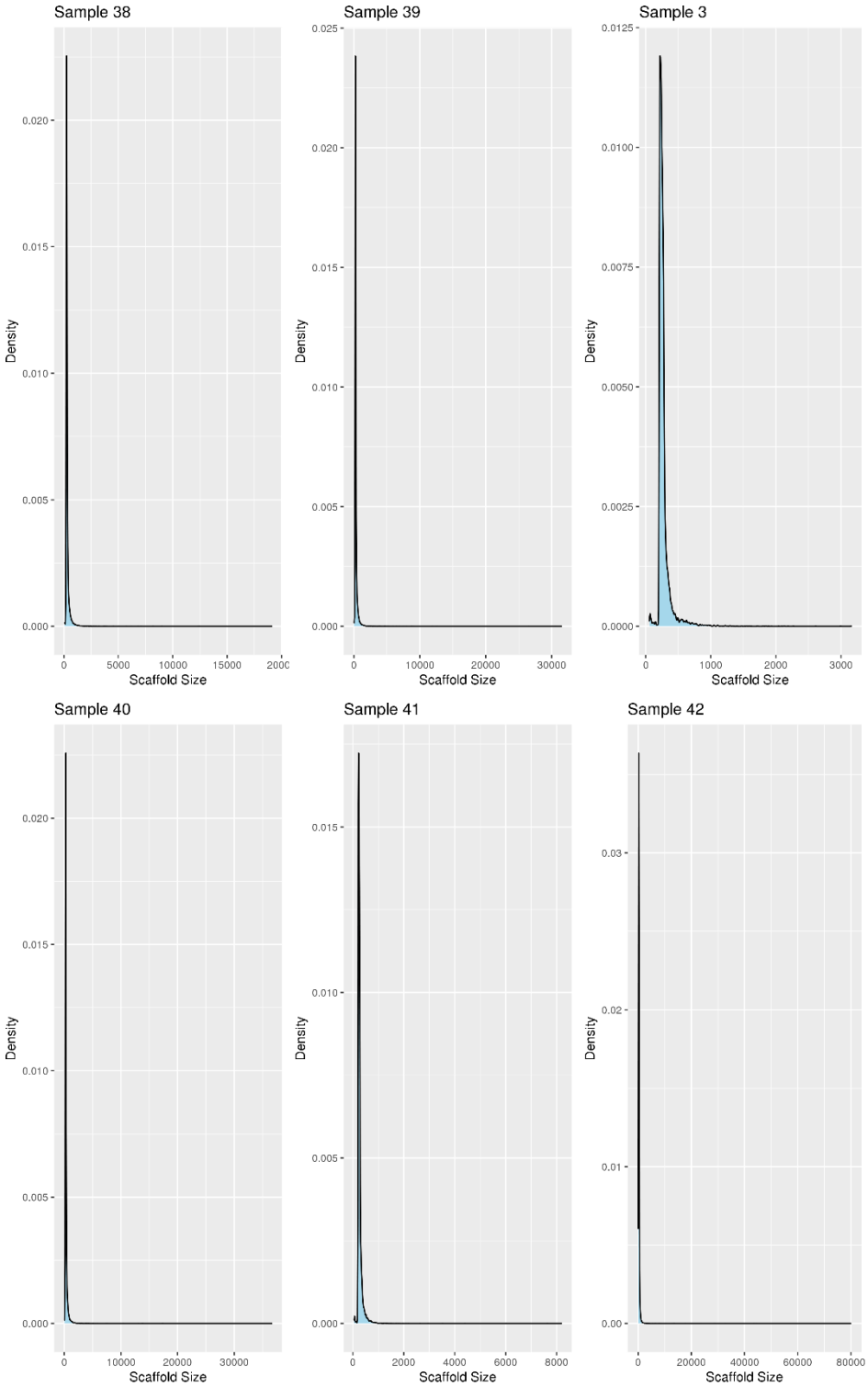
X

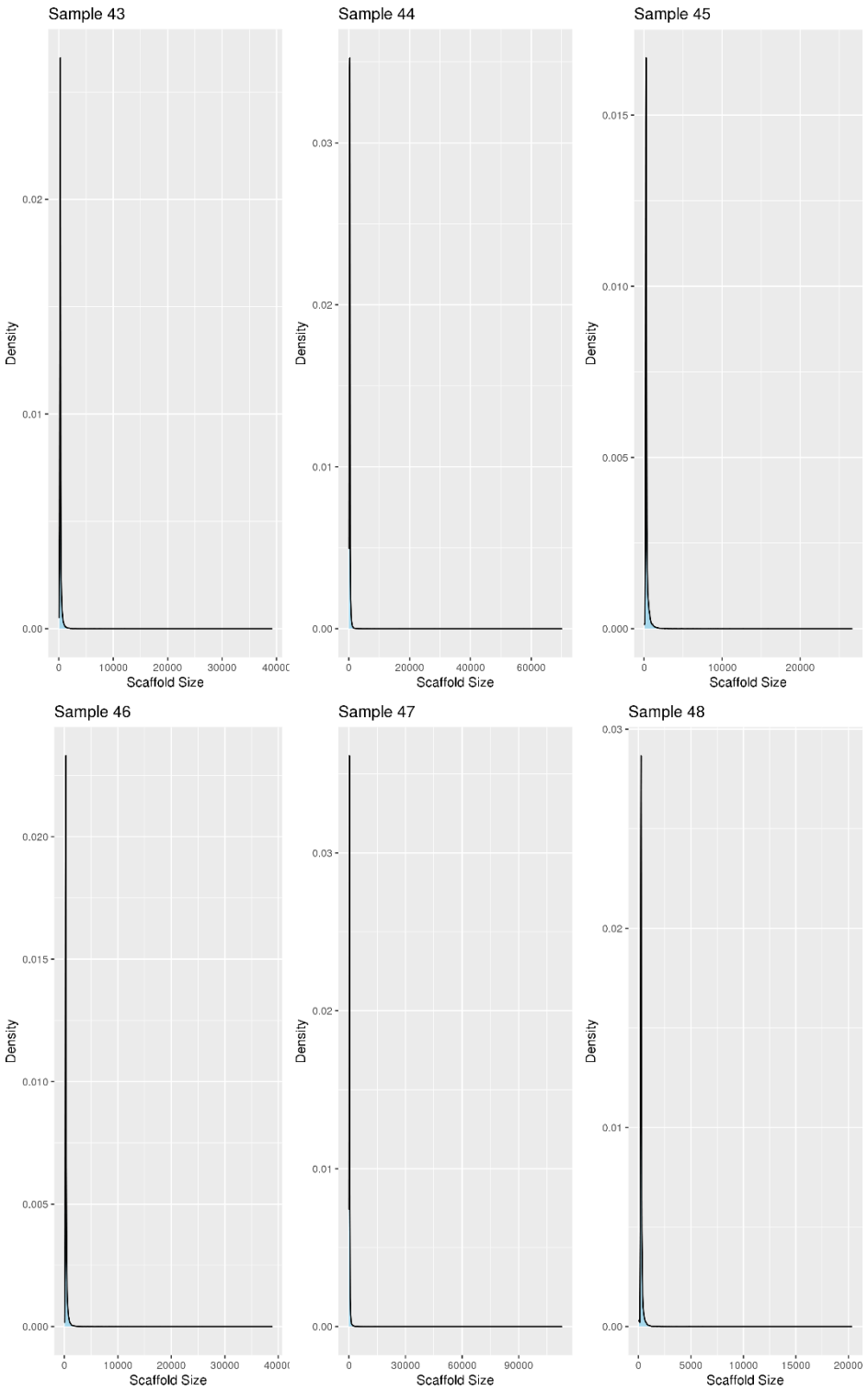

Sample 21
Density
0.010
0.005
0.000
0
2500
5000
7500
Scaffold Size
Sample 22
Density
0.020
0.015
0.010
0.005
0.000
0
10000
20000
30000
Scaffold Size
Sample 23
Density
0.015
0.010
0.005
0.000
0
5000
10000
15000
20000
Scaffold Size
Sample 24
Density
0.009
0.006
0.003
0.000
0
1000
2000
3000
4000
Scaffold Size
Sample 25
Density
0.009
0.006
0.003
0.000
0
1000
2000
3000
4000
Scaffold Size
Sample 26
Density
0.0100
0.0075
0.0050
0.0025
0.0000
0
2000
4000
6000
Scaffold Size

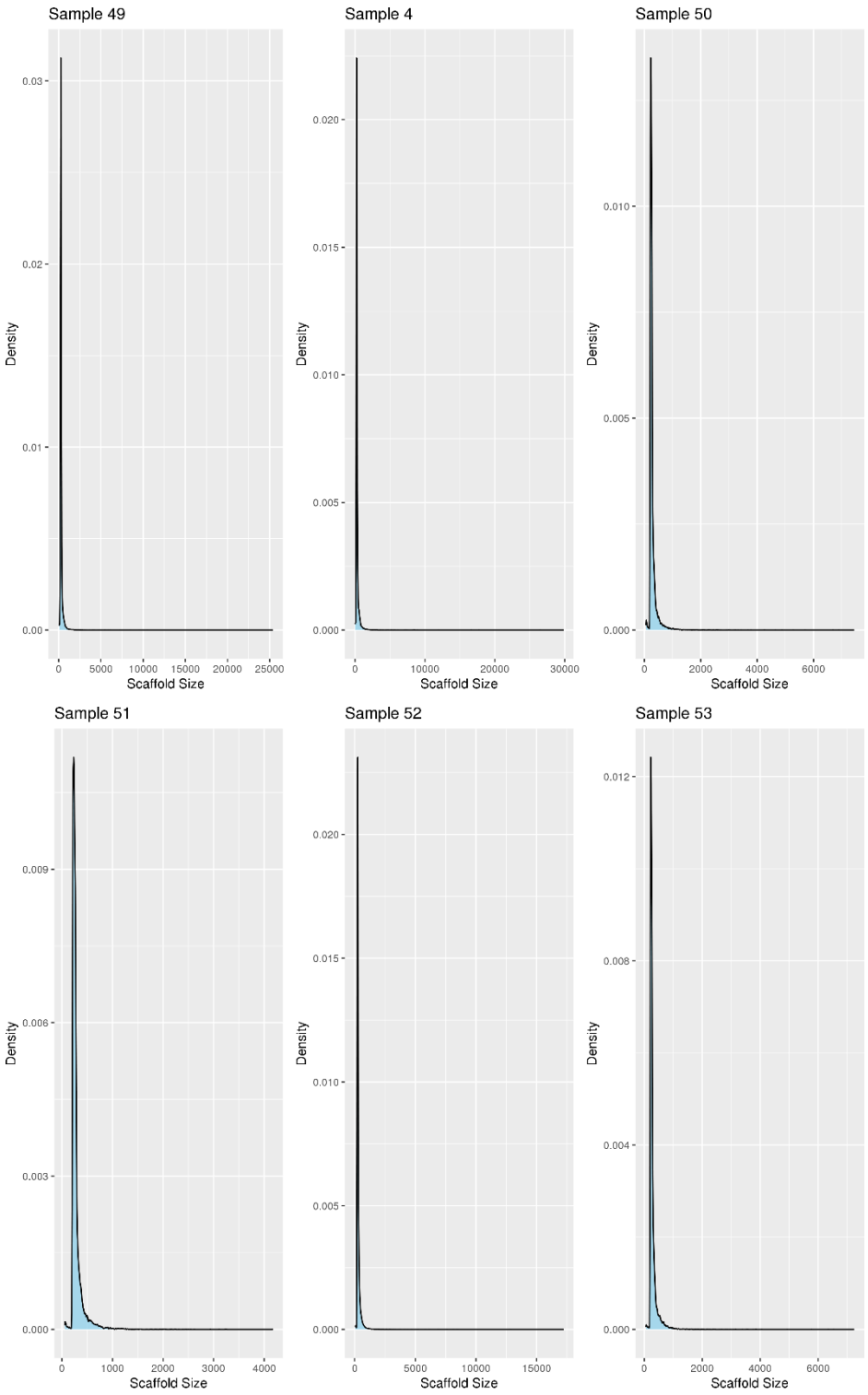

Sample 27
Sample 28
Sample 29
Sample 2
Sample 30
Sample 31
Density
Scaffold Size

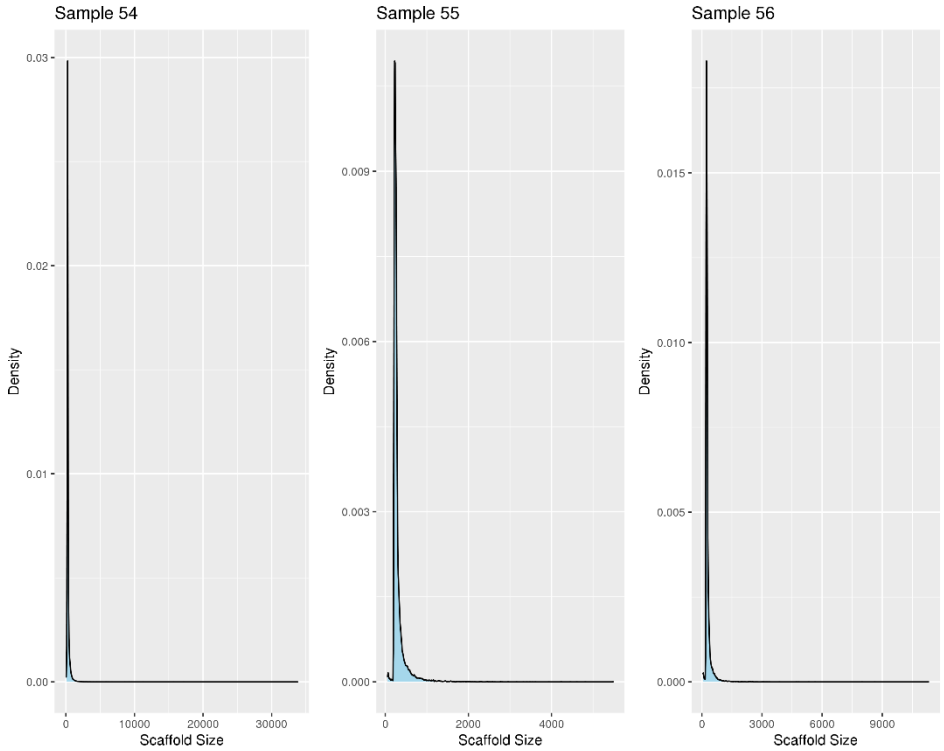

Sample 32
Density
0.015
0.010
0.005
0.000
0
20000
40000
60000
Scaffold Size
Sample 33
Density
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
0
2000
4000
6000
Scaffold Size
Sample 34
Density
0.015
0.010
0.005
0.000
0
10000
20000
30000
40000
Scaffold Size


Sample 35
Density
0.03
0.02
0.01
0.00
0
25000
50000
75000
100000
Scaffold Size
Sample 36
Density
0.03
0.02
0.01
0.00
0
20000
40000
60000
Scaffold Size
Sample 37
Density
0.02
0.01
0.00
0
20000
40000
Scaffold Size


Sample 38
Density
Scaffold Size
Sample 39
Density
Scaffold Size
Sample 3
Density
Scaffold Size
Sample 40
Density
Scaffold Size
Sample 41
Density
Scaffold Size
Sample 42
Density
Scaffold Size

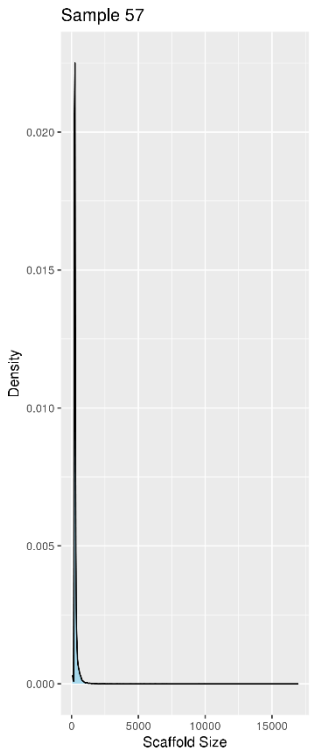

Sample 43
Sample 44
Sample 45
Sample 46
Sample 47
Sample 48
Density
Scaffold Size

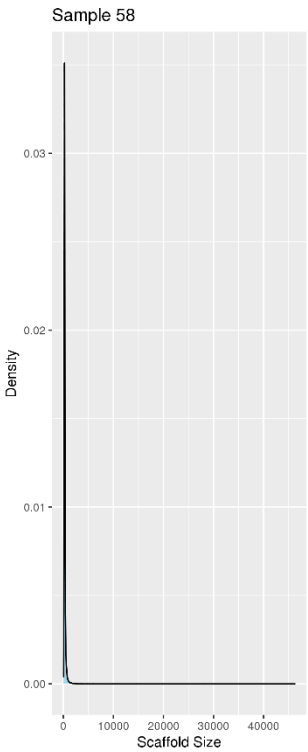

Sample 49
Density
0.03
0.02
0.01
0.00
0
5000
10000
15000
20000
25000
Scaffold Size
Sample 4
Density
0.020
0.015
0.010
0.005
0.000
0
10000
20000
30000
Scaffold Size
Sample 50
Density
0.010
0.005
0.000
0
2000
4000
6000
Scaffold Size
Sample 51
Density
0.009
0.006
0.003
0.000
0
1000
2000
3000
4000
Scaffold Size
Sample 52
Density
0.020
0.015
0.010
0.005
0.000
0
5000
10000
15000
Scaffold Size
Sample 53
Density
0.012
0.008
0.004
0.000
0
2000
4000
6000
Scaffold Size

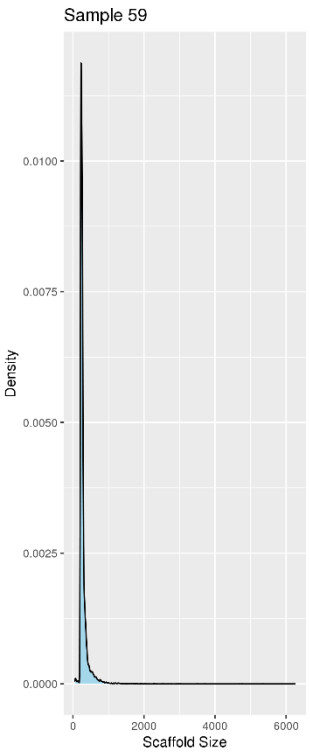

Sample 54
Density
0.03
0.02
0.01
0.00
0
10000
20000
30000
Scaffold Size

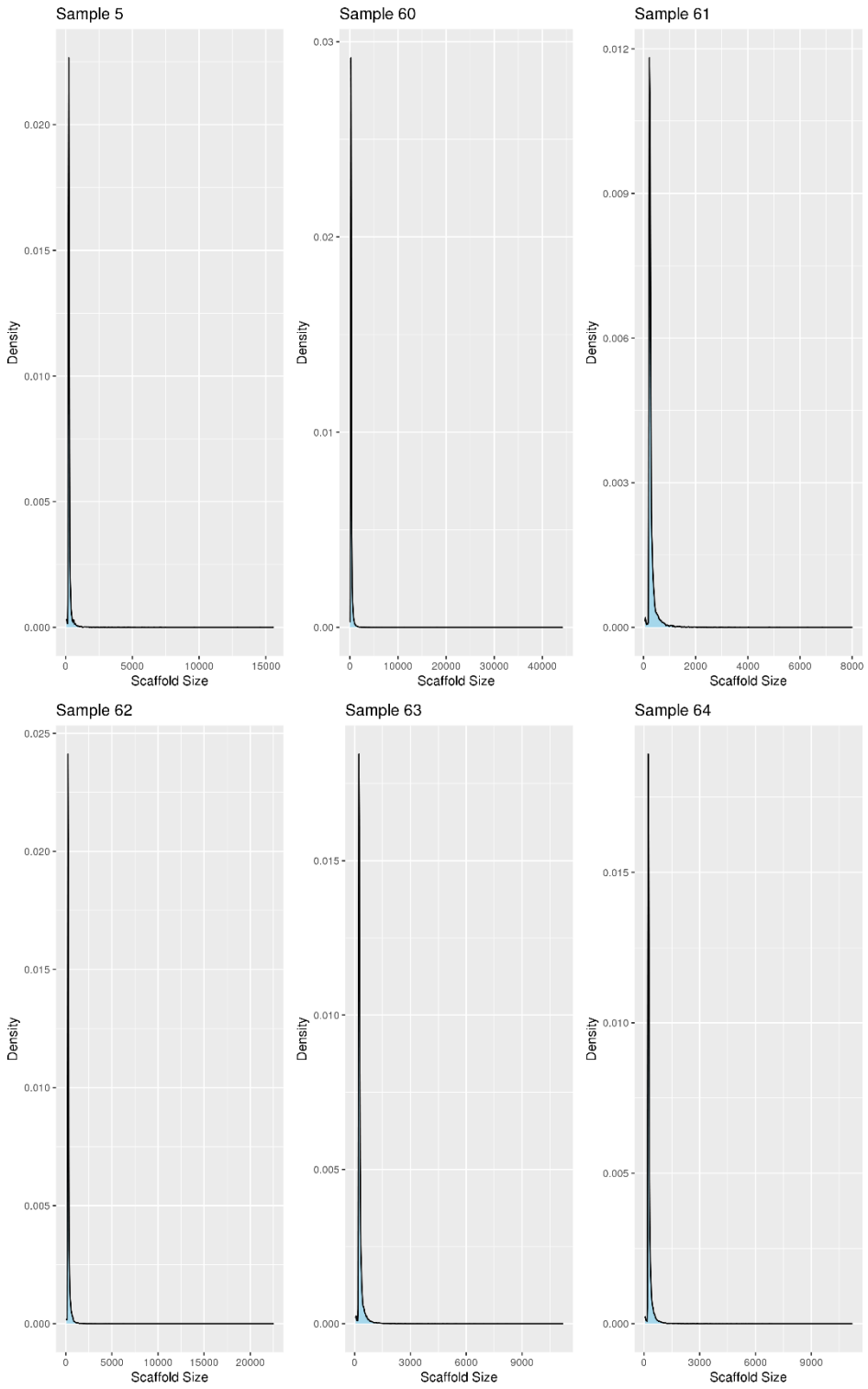
Sample 55
Density
0.009
0.006
0.003
0.000
0
2000
4000
Scaffold Size

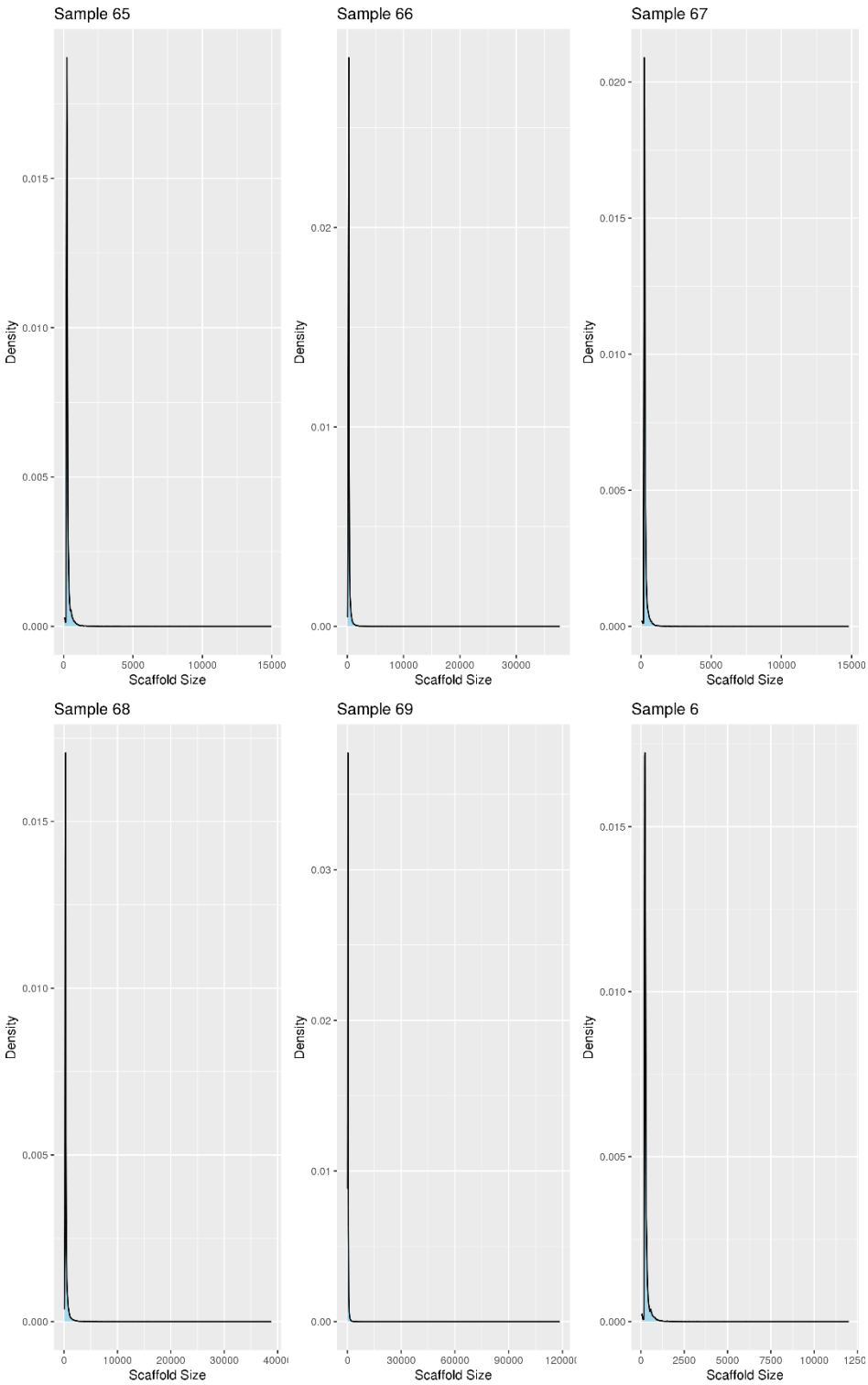
Sample 56
Density
0.015
0.010
0.005
0.000
0
3000
6000
9000
Scaffold Size

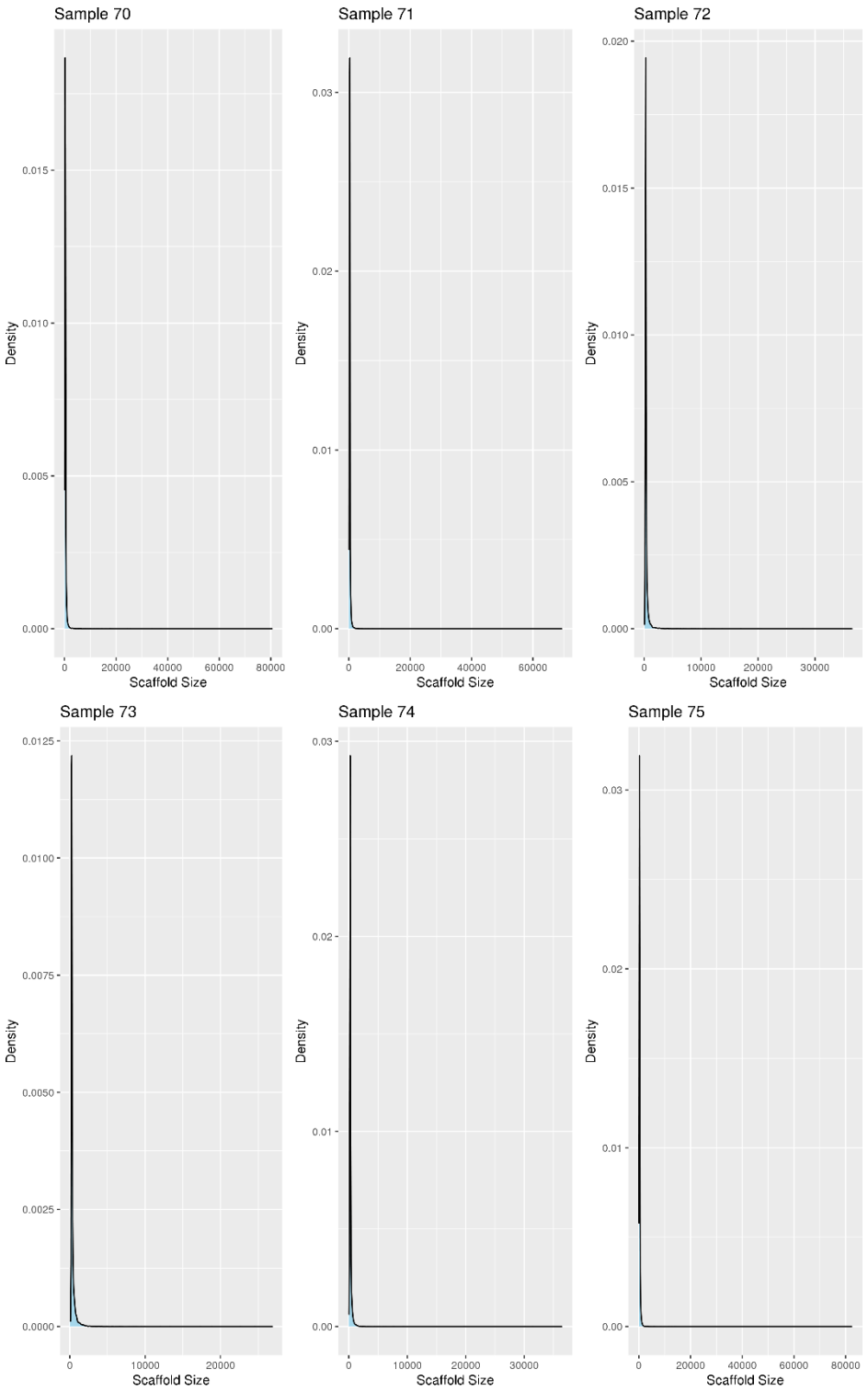

Sample 57
Density
0.020
0.015
0.010
0.005
0.000
0
5000
10000
15000
Scaffold Size

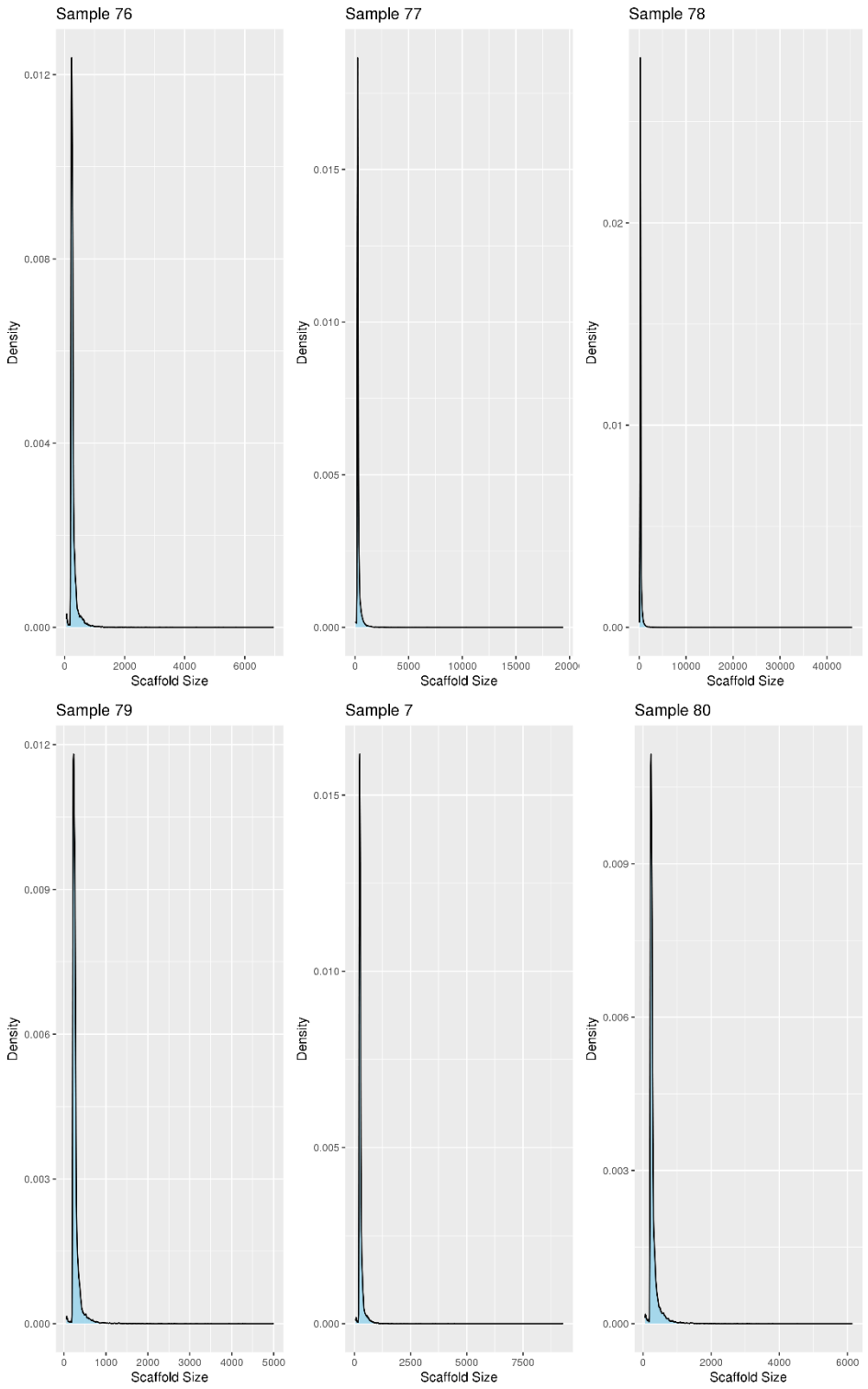

Sample 58
Density
0.03
0.02
0.01
0.00
0
10000
20000
30000
40000
Scaffold Size

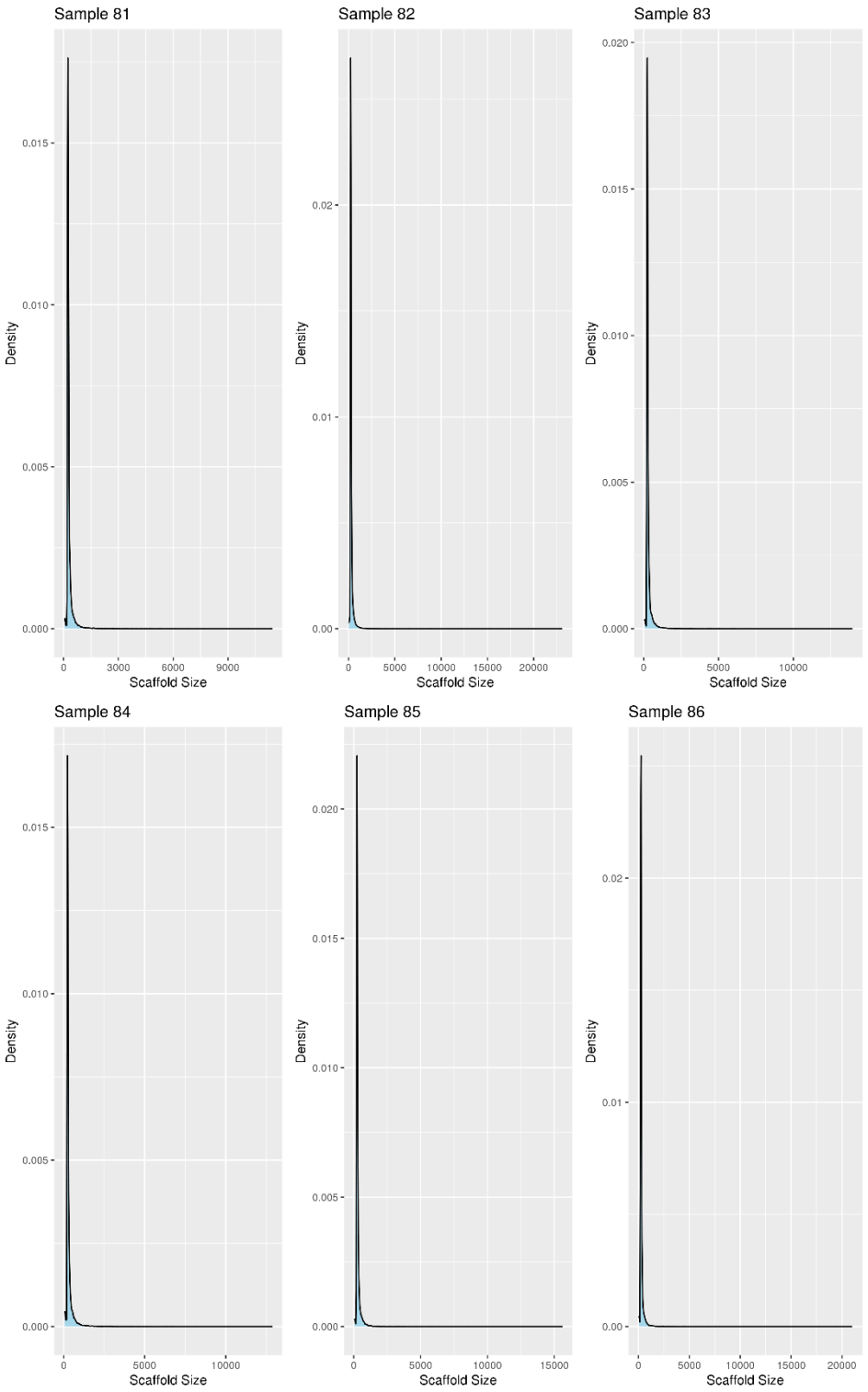

Sample 59
Density
0.0100
0.0075
0.0050
0.0025
0.0000
0
2000
4000
6000
Scaffold Size

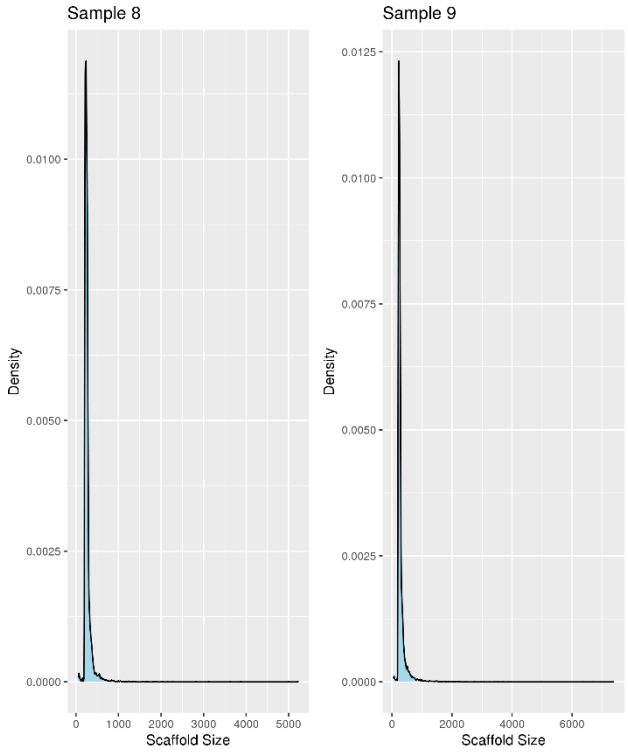

Sample 5
Density
0.020
0.015
0.010
0.005
0.000
0
5000
10000
15000
Scaffold Size
Sample 60
Density
0.03
0.02
0.01
0.00
0
10000
20000
30000
40000
Scaffold Size
Sample 61
Density
0.012
0.009
0.006
0.003
0.000
0
2000
4000
6000
8000
Scaffold Size
Sample 62
Density
0.025
0.020
0.015
0.010
0.005
0.000
0
5000
10000
15000
20000
Scaffold Size
Sample 63
Density
0.015
0.010
0.005
0.000
0
3000
6000
9000
Scaffold Size
Sample 64
Density
0.015
0.010
0.005
0.000
0
3000
6000
9000
Scaffold Size


Sample 65
Density
0.015
0.010
0.005
0.000
0
5000
10000
15000
Scaffold Size
Sample 66
Density
0.02
0.01
0.00
0
10000
20000
30000
Scaffold Size
Sample 67
Density
0.020
0.015
0.010
0.005
0.000
0
5000
10000
15000
Scaffold Size
Sample 68
Density
0.015
0.010
0.005
0.000
0
10000
20000
30000
Scaffold Size
Sample 69
Density
0.03
0.02
0.01
0.00
0
30000
60000
90000
Scaffold Size
Sample 6
Density
0.015
0.010
0.005
0.000
0
2500
5000
7500
10000
Scaffold Size

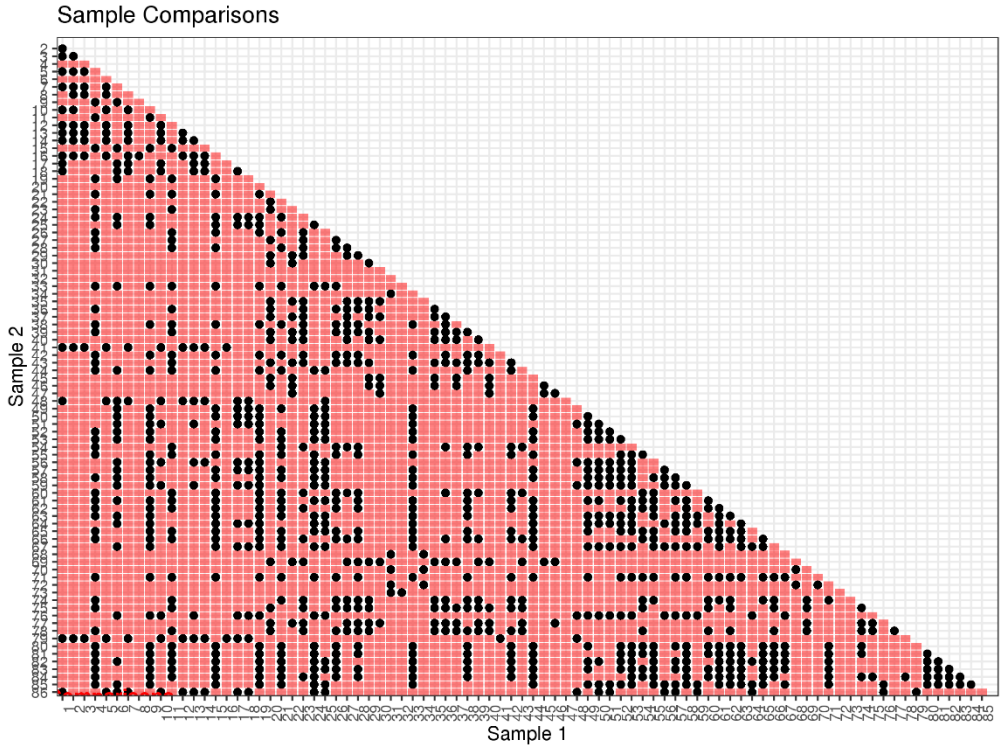

Sample 70
Density
0.015
0.010
0.005
0.000
0
20000
40000
60000
80000
Scaffold Size
Sample 71
Density
0.03
0.02
0.01
0.00
0
20000
40000
60000
Scaffold Size
Sample 72
Density
0.020
0.015
0.010
0.005
0.000
0
10000
20000
30000
Scaffold Size
Sample 73
Density
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
0
10000
20000
Scaffold Size
Sample 74
Density
0.03
0.02
0.01
0.00
0
10000
20000
30000
Scaffold Size
Sample 75
Density
0.03
0.02
0.01
0.00
0
20000
40000
60000
80000
Scaffold Size


Sample 76
Density
0.012
0.008
0.004
0.000
0
2000
4000
6000
Scaffold Size
Sample 77
Density
0.015
0.010
0.005
0.000
0
5000
10000
15000
Scaffold Size
Sample 78
Density
0.02
0.01
0.00
0
10000
20000
30000
40000
Scaffold Size
Sample 79
Density
0.012
0.009
0.006
0.003
0.000
0
1000
2000
3000
4000
5000
Scaffold Size
Sample 7
Density
0.015
0.010
0.005
0.000
0
2500
5000
7500
Scaffold Size
Sample 80
Density
0.009
0.006
0.003
0.000
0
2000
4000
6000
Scaffold Size

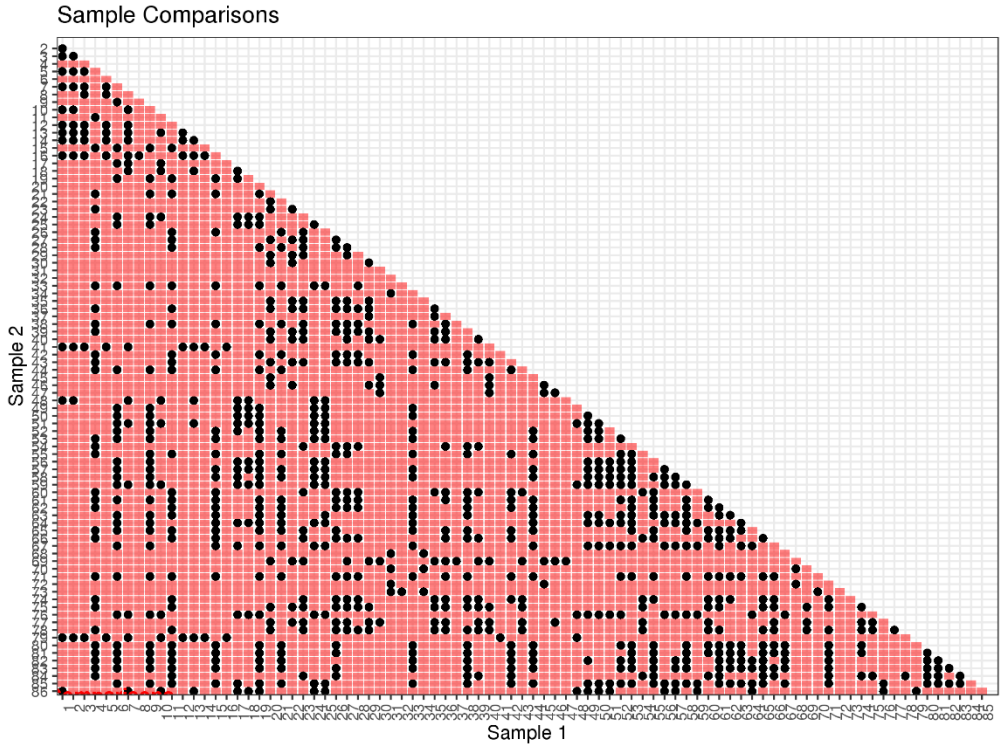

Sample 81
Sample 82
Sample 83
Sample 84
Sample 85
Sample 86
Density
Scaffold Size


Sample 8
Sample 9
Density
Scaffold Size
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
0
1000
2000
3000
4000
5000
6000

8.8. Comparaisons multiples sur 1/100 des données

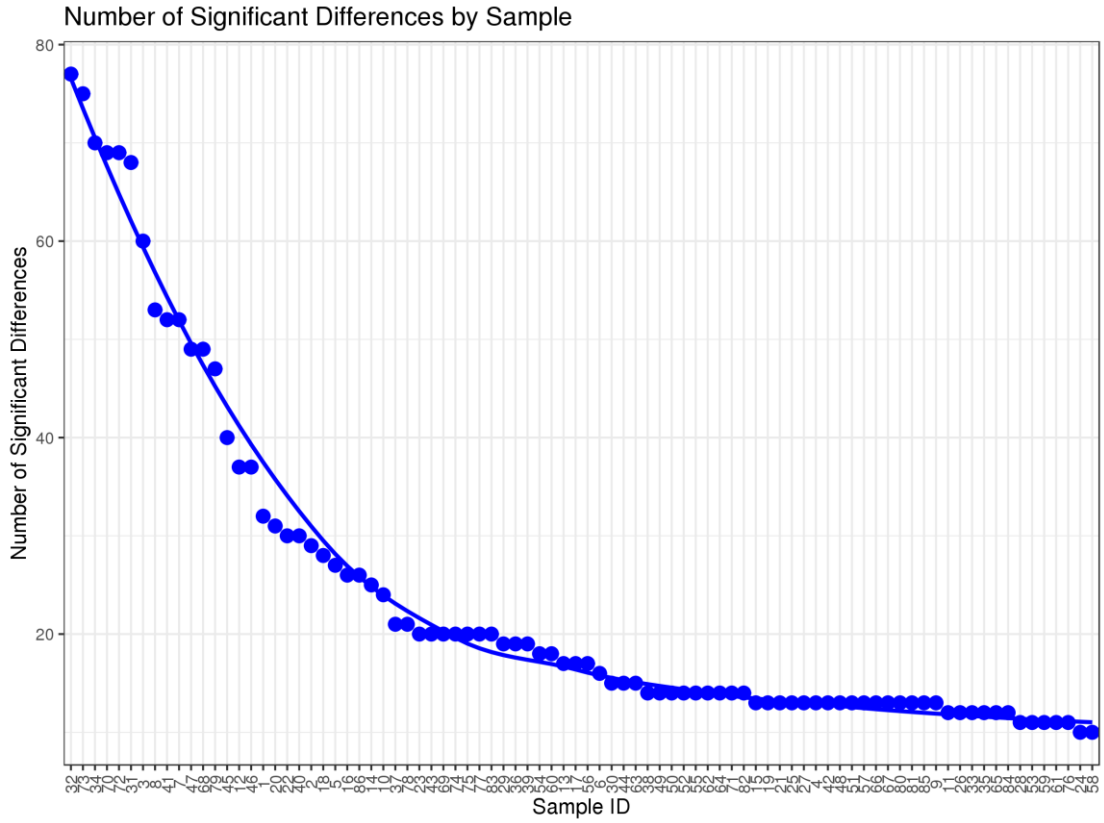


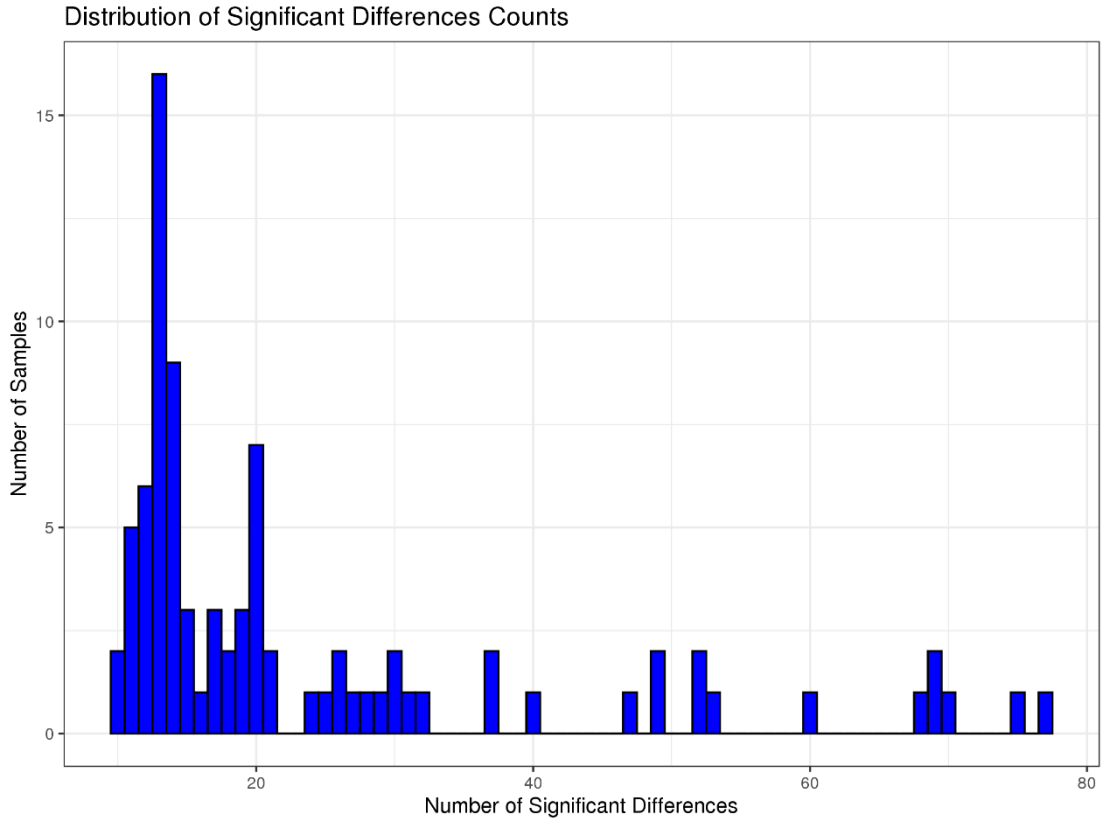
Wilcoxon



KS

8.9. Distribution des échantillons sur base de leurs différences





8.10. Code bash ayant servi à l'homogénéisation du nombre d'échantillons par catégorie

```
cut -f1 scaffolds_size_all.tsv | uniq -c | sort -rn > scaffolds_number_all.tsv
perl -nle 's/([0-9]+)\t([0-9]+)/$2\t$1/; print' scaffolds_number_all.tsv | \
  sort -n > scaffolds_number-per-sample.tsv
paste scaffolds_number-per-sample.tsv info_samples_novaseq_germany_formated.tab |\
  perl -nle 's/([0-9]+)\s+([0-9]+)\s+[0-9]+\s+(.+)/$1\t$2\t$3/; print' | \
  sort -n -k2 > samples_scaffolds-counts_info.tsv

egrep -v "NGA13_rhiz|MGA14_Bulk1|MGA36_rhiz|MGA34_Bulk1|MGA35_Bulk1|MGA36_Bulk1|MG
A15_Bulk1" samples_scaffolds-counts_info.tsv \
  > samples_wo-outliers_scaffolds-counts_info.tsv

sort -k2 -rn samples_wo-outliers_scaffolds-counts_info.tsv
cut -f3 samples_wo-outliers_scaffolds-counts_info.tsv | \
  cut -c1,4,7 | sort | uniq -c
cut -f3 samples_wo-outliers_scaffolds-counts_info.tsv | \
  cut -c1,7 | sort | uniq -c
#     18 MB
#     22 Mr
#     20 NB
#     19 Nr


## Identifier de potentiels samples contenant peu de scaffolds qui correspondent a
ux samples à retirer pour XXVIrrive à 18 sample par catégorie

# 5       949071  NGA21_rhiz
# 8       1244308 NGA13_Bulk1

egrep -v "NGA21_rhiz|NGA13_Bulk1" samples_without-outliers \
  > samples_wo-outliers_wo-small.tsv




### Retraits des samples les plus différents dans les catégories contenant plus de
18 samples

cat samples_wo-outliers_wo-small.tsv
cut -f3 samples_wo-outliers_wo-small.tsv | cut -c1,4,7 | sort | uniq -c
cut -f3 samples_wo-outliers_wo-small.tsv | cut -c1,7 | sort | uniq -c
#     18 MB
#     22 Mr
#     19 NB
#     18 Nr


## Identification des samples les plus différents (4Mr 1NB)

# 7       52      NGA12_Bulk1
# 68      49      MGA33_rhiz
# 20      31      MGA12_rhiz
# 22      30      MGA31_rhiz
```

```
# 86      26      MGA51_rhiz

#N1B -> 1
#M5r -> 0
#M3r -> 3
#M1r -> 7

# Problème de représentent unique et de réplicas -> changement de sélection

# 68      49      MGA33_rhiz
# 20      31      MGA12_rhiz
# 22      30      MGA31_rhiz
# 18      28      NGA31_Bulk1
# 37      21      MGA23_rhiz

egrep -v "MGA33_rhiz|MGA12_rhiz|MGA31_rhiz|NGA31_Bulk1|MGA23_rhiz" samples_wo-outl
iers_wo-small.tsv > samples_selected.tsv

cat samples_selected.tsv
cut -f3 samples_selected.tsv | cut -c1,4,7 | sort | uniq -c
cut -f3 samples_selected.tsv | cut -c1,7 | sort | uniq -c
```

8.11. Liste des paramètres ayant servi à la création des fichiers de configuration de Forty-Two

MGA_Bulk1	3dom	0.25
MGA_Bulk1	Bact_Euk	0.33
MGA_Bulk1	Bact	0.61
MGA_Bulk1	Euk	0.72
MGA_Bulk1	Prok	0.38
MGA_rhiz	3dom	0.25
MGA_rhiz	Bact_Euk	0.33
MGA_rhiz	Bact	0.61
MGA_rhiz	Euk	0.72
MGA_rhiz	Prok	0.38
NGA_Bulk1	3dom	0.25
NGA_Bulk1	Bact_Euk	0.33
NGA_Bulk1	Bact	0.61
NGA_Bulk1	Euk	0.72
NGA_Bulk1	Prok	0.38
NGA_rhiz	3dom	0.25
NGA_rhiz	Bact_Euk	0.33
NGA_rhiz	Bact	0.61
NGA_rhiz	Euk	0.72
NGA_rhiz	Prok	0.38

8.12. Liste des paramètres ayant servi à la création des jobs Forty-Two

MGA_Bulk1-3dom	3dom_clusters	12
MGA_Bulk1-Bact	Bact_clusters	4
MGA_Bulk1-Bact_Euk	Bact_Euk_clusters	4
MGA_Bulk1-Euk	Euk_clusters	3
MGA_Bulk1-Prok	Prok_clusters	10
MGA_rhiz-3dom	3dom_clusters	12
MGA_rhiz-Bact	Bact_clusters	4
MGA_rhiz-Bact_Euk	Bact_Euk_clusters	4
MGA_rhiz-Euk	Euk_clusters	3
MGA_rhiz-Prok	Prok_clusters	10
NGA_Bulk1-3dom	3dom_clusters	12
NGA_Bulk1-Bact	Bact_clusters	4
NGA_Bulk1-Bact_Euk	Bact_Euk_clusters	4
NGA_Bulk1-Euk	Euk_clusters	3
NGA_Bulk1-Prok	Prok_clusters	10
NGA_rhiz-3dom	3dom_clusters	12
NGA_rhiz-Bact	Bact_clusters	4
NGA_rhiz-Bact_Euk	Bact_Euk_clusters	4
NGA_rhiz-Euk	Euk_clusters	3
NGA_rhiz-Prok	Prok_clusters	10

8.13. Liste des paires nom de site – code en quatre lettres

MGA11_Bulk1	maab	MGA42_rhiz	mdbr
MGA11_rhiz	maar	MGA51_Bulk1	meab
MGA12_Bulk1	mabb	MGA51_rhiz	mear
MGA12_rhiz	mabr	NGA11_Bulk1	naab
MGA13_Bulk1	macb	NGA11_rhiz	naar
MGA13_rhiz	macr	NGA12_Bulk1	nabb
MGA14_Bulk1	madb	NGA12_rhiz	nabr
MGA14_rhiz	madr	NGA13_Bulk1	nacb
MGA15_Bulk1	maeb	NGA13_rhiz	nacr
MGA15_rhiz	maer	NGA21_Bulk1	nbab
MGA16_Bulk1	mafb	NGA21_rhiz	nbar
MGA16_rhiz	mafr	NGA22_Bulk1	nbbb
MGA17_Bulk1	magb	NGA22_rhiz	nbbr
MGA17_rhiz	magr	NGA23_Bulk1	nbcb
MGA18_Bulk1	mahb	NGA23_rhiz	nbcr
MGA18_rhiz	mahr	NGA31_Bulk1	ncab
MGA21_Bulk1	mbab	NGA31_rhiz	ncar
MGA21_rhiz	mbar	NGA32_Bulk1	ncbb
MGA22_Bulk1	mbbb	NGA32_rhiz	ncbr
MGA22_rhiz	mbbr	NGA33_Bulk1	nccb
MGA23_Bulk1	mbcb	NGA33_rhiz	nccr
MGA23_rhiz	mbcr	NGA34_Bulk1	ncdb
MGA24_Bulk1	mbdb	NGA34_rhiz	ncdr
MGA24_rhiz	mbdr	NGA35_Bulk1	nceb
MGA25_Bulk1	mbeb	NGA35_rhiz	ncer
MGA25_rhiz	mber	NGA41_Bulk1	ndab
MGA26_Bulk1	mbfb	NGA41_rhiz	ndar
MGA26_rhiz	mbfr	NGA42_Bulk1	ndbb
MGA31_Bulk1	mcab	NGA42_rhiz	ndbr
MGA31_rhiz	mcar	NGA43_Bulk1	ndcb
MGA32_Bulk1	mcbb	NGA43_rhiz	ndcr
MGA32_rhiz	mcbr	NGA44_Bulk1	nddb
MGA33_Bulk1	mccb	NGA44_rhiz	nddr
MGA33_rhiz	mccr	NGA51_Bulk1	neab
MGA34_Bulk1	mcdb	NGA51_rhiz	near
MGA34_rhiz	mcdr	NGA52_Bulk1	nebb
MGA35_Bulk1	mceb	NGA52_rhiz	nebr
MGA35_rhiz	mcer	NGA53_Bulk1	necb
MGA36_Bulk1	mcfb	NGA53_rhiz	necr
MGA36_rhiz	mcfr	NGA54_Bulk1	nedb
MGA41_Bulk1	mdab	NGA54_rhiz	nedr
MGA41_rhiz	mdar	NGA55_Bulk1	neeb
MGA42_Bulk1	mdbb	NGA55_rhiz	neer

8.14. Configuration de la filtration des alignements multiples enrichis par Forty-Two

```
ali2phylip.pl \
--p80 \
--max=0.3 \
--min=0.3 \
--test-out=test-out-0.3-0.3 \
  *fasta \
  2> test-out-0.3-0.3.log

ali2phylip.pl \
--p80 \
--max=0.3 \
--min=0.2 \
--test-out=test-out-0.3-0.2 \
  *fasta  \
  2> test-out-0.3-0.2.log

ali2phylip.pl \
--p80 –max=0.3 \
--min=75 \
--test-out=test-out-0.3-75 \
  *fasta \
  2> test-out-0.3-75.log

ali2phylip.pl \
--p80 \
--max=0.3 \
--min=50 \
--test-out=test-out-0.3-50 \
  *fasta \
  2> test-out-0.3-50.log

perl -e ‘print « file\tin.seqs\tin.sites\t0.3-out.seqs\t0.3-out.sites\n »’ \
  > resume-test-out-0.3-0.3.txt
perl -e 'print "file\tin.seqs\tin.sites\t0.2-out.seqs\t0.2-out.sites\n"' \
  > resume-test-out-0.3-0.2.txt
perl -e 'print "file\tin.seqs\tin.sites\t75-out.seqs\t75-out.sites\n"' \
  > resume-test-out-0.3-75.txt
perl -e 'print "file\tin.seqs\tin.sites\t50-out.seqs\t50-out.sites\n"' \
  > resume-test-out-0.3-50.txt

egrep -v "test|file" test-out-0.3-0.3 | cut -f1,12,13,15,16 | cut -d'/' -f2 | \
  Cut -d’-‘ -f1,2,11 | perl -nle ‘s/-names.fasta// ; print’ | sort -n \
  >> resume-test-out-0.3-0.3.txt
egrep -v "test|file" test-out-0.3-0.2 | cut -f1,12,13,15,16 | cut -d'/' -f2 | \
  cut -d'-' -f1,2,11 | perl -nle 's/-names.fasta//; print' | sort -n \
  >> resume-test-out-0.3-0.2.txt
egrep -v "test|file" test-out-0.3-75 | cut -f1,12,13,15,16 | cut -d'/' -f2 | \
  cut -d'-' -f1,2,11 | perl -nle 's/-names.fasta//; print' | sort -n \
  >> resume-test-out-0.3-75.txt
egrep -v "test|file" test-out-0.3-50 | cut -f1,12,13,15,16 | cut -d'/' -f2 | \
  Cut -d’-‘ -f1,2,11 | perl -nle ‘s/-names.fasta// ; print’ | sort -n \
  >> resume-test-out-0.3-50.txt
```

```
perl -e 'print "file\tnew_seqs\n"' > added_sequences.txt
grep -c "\#NEW\#" ../../5-forty-two/*/*-42-*.fasta | \
  perl -nle 's/(?:[^\/]+\/){4}([^-]+-[^-]+)-[^:]+:([0-9]+)/$1\t$2/; print' | \
  egrep -v "abc_fet-P72827|cdf-P53735|ctr-Q9P7F9|lyse_ilt-P40088|mit-Q01926|tog_nicot-P23516" | sort -n \
  >> added_sequences.txt

paste resume-test-out-0.3-0.3.txt resume-test-out-0.3-0.2.txt \
  resume-test-out-0.3-75.txt resume-test-out-0.3-50.txt added_sequences.txt | \
  perl -anle 'print "$F[0]\t$F[1]\t$F[21]\t$F[3]\t$F[8]\t$F[13]\t$F[18]\t$F[2]\t$F[4]\t$F[9]\t$F[14]\t$F[19]"' \
  > comparative_tests.txt

perl -ane ' {
        print $F[0] . "\t" . $F[1] . "\t" . $F[2] . "\t"
    }
    if ($F[3] > (2 * ($F[1]) / 3)) {
        print $F[3] . " OK" . "\t"
    }
    else {
        print $F[3] . "\t"
    }
    if ($F[4] > (2 * ($F[1]) / 3)) {
        print $F[4] . " OK" . "\t"
    }
    else {
        print $F[4] . "\t"
    }
    if ($F[5] > (2 * ($F[1]) / 3)) {
        print $F[5] . " OK" . "\t"
    }
    else {
        print $F[5] . "\t"
    }
    if ($F[6] > (2 * ($F[1]) / 3)) {
        print $F[6] . " OK" . "\n"
    }
    else {
        print $F[6] . "\n"
    }
' comparative_tests.txt > multiple_parameters_tests.txt
```

file	in.seqs	new_seqs	0.3-out.seqs	0.2-out.seqs OK	75-out.seqs OK	50-out.seqs OK
abc_fet-P44513	55360	53420	48388 OK	52882 OK	31153	51003 OK
abc_fet-P44513	55566	53626	48550 OK	53122 OK	29658	50953 OK
abc_fet-P44513	55819	53879	48884 OK	53144 OK	28901	51040 OK
abc_fet-P44513	57069	55129	49286 OK	54160 OK	33150	52251 OK
abc_mzt-O34610	353	44	347 OK	353 OK	348 OK	353 OK

abc_mzt-O34610	360	51	355 OK	359 OK	358 OK	360 OK
abc_mzt-O34610	369	60	363 OK	368 OK	364 OK	369 OK
abc_mzt-O34610	371	62	369 OK	370 OK	370 OK	371 OK
abc_mzt-P0A4G4	361	108	346 OK	361 OK	359 OK	361 OK
abc_mzt-P0A4G4	374	121	350 OK	374 OK	367 OK	374 OK
abc_mzt-P0A4G4	386	133	366 OK	386 OK	381 OK	386 OK
abc_mzt-P0A4G4	388	135	364 OK	388 OK	380 OK	388 OK
abc_nate-P46904	155	0	154 OK	155 OK	155 OK	155 OK
abc_nate-P46904	155	0	154 OK	155 OK	155 OK	155 OK
abc_nate-P46904	156	1	155 OK	156 OK	156 OK	156 OK
abc_nate-P46904	159	4	158 OK	159 OK	159 OK	159 OK
abc_nicot-O68104	24	1	23 OK	23 OK	23 OK	23 OK
abc_nicot-O68104	26	3	23 OK	23 OK	23 OK	23 OK
abc_nicot-O68104	26	3	23 OK	23 OK	23 OK	23 OK
abc_nicot-O68104	31	8	31 OK	31 OK	31 OK	31 OK
abc_nicot-Q05594	113	22	112 OK	113 OK	108 OK	113 OK
abc_nicot-Q05594	113	22	111 OK	113 OK	107 OK	113 OK
abc_nicot-Q05594	117	26	113 OK	117 OK	111 OK	117 OK
abc_nicot-Q05594	129	38	127 OK	129 OK	124 OK	129 OK
abc_nicot-Q05598	213	17	211 OK	211 OK	211 OK	211 OK
abc_nicot-Q05598	216	2	211 OK	211 OK	211 OK	211 OK
abc_nicot-Q05598	220	5	212 OK	212 OK	212 OK	212 OK
abc_nicot-Q05598	228	9	217 OK	217 OK	217 OK	217 OK

abc_pept-P0AFA9	8818	8456	6968 OK	8370 OK	5491	8165 OK
abc_pept-P0AFA9	9016	8654	7019 OK	8504 OK	5091	8100 OK
abc_pept-P0AFA9	9403	9041	7266 OK	8873 OK	5678	8647 OK
abc_pept-P0AFA9	9847	9485	7685 OK	9286 OK	5683	8899 OK
abc_pept-P33590	1034	1140	593	706 OK	877 OK	1030 OK
abc_pept-P33590	1173	1510	756	892 OK	1037 OK	1171 OK
abc_pept-P33590	1490	684	873	1105 OK	1275 OK	1482 OK
abc_pept-P33590	1860	823	1206	1464 OK	1664 OK	1852 OK
abc_pept-P33591	9347	8992	6712 OK	8536 OK	4381	6861 OK
abc_pept-P33591	9389	9034	6996 OK	7184 OK	3954	6630 OK
abc_pept-P33591	9810	9455	6756 OK	9045 OK	4758	8306 OK
abc_pept-P33591	9990	9635	7287 OK	7538 OK	4127	6883 OK
cdf-P13512	1515	1045	1096 OK	1502 OK	1307 OK	1508 OK
cdf-P13512	1532	1062	1113 OK	1522 OK	1303 OK	1530 OK
cdf-P13512	2667	2197	1915 OK	2646 OK	2239 OK	2653 OK
cdf-P13512	3036	2566	2216 OK	3019 OK	2535 OK	3028 OK
copd-P12377	135	121	127 OK	130 OK	128 OK	131 OK
copd-P12377	281	267	271 OK	276 OK	275 OK	276 OK
copd-P12377	395	381	378 OK	390 OK	382 OK	390 OK
copd-P12377	64	50	58 OK	59 OK	59 OK	59 OK
lyse_ilt-P31545	46	18	38 OK	44 OK	44 OK	46 OK
lyse_ilt-P31545	53	25	43 OK	53 OK	52 OK	53 OK
lyse_ilt-P31545	53	25	45 OK	53 OK	51 OK	53 OK
lyse_ilt-P31545	57	29	51 OK	57 OK	55 OK	57 OK
lyse_ilt-P38993	336	106	233 OK	294 OK	332 OK	336 OK
lyse_ilt-P38993	347	198	235 OK	297 OK	341 OK	347 OK
lyse_ilt-P38993	439	472	306 OK	397 OK	433 OK	439 OK

lyse_ilt-P38993	713	95	505 OK	697 OK	697 OK	713 OK
lyse_ilt-Q0WFT9	451	437	334 OK	393 OK	170	352 OK
lyse_ilt-Q0WFT9	533	519	383 OK	455 OK	163	389 OK
lyse_ilt-Q0WFT9	963	949	685 OK	823 OK	368	739 OK
lyse_ilt-Q0WFT9	990	976	721 OK	874 OK	438	821 OK
lyse_mntp-P76264	102	10	102 OK	102 OK	95 OK	102 OK
lyse_mntp-P76264	112	25	112 OK	112 OK	102 OK	112 OK
lyse_mntp-P76264	49	63	49 OK	49 OK	48 OK	49 OK
lyse_mntp-P76264	64	73	64 OK	64 OK	57 OK	64 OK
lyse_terc-P42601	1190	1060	938 OK	1181 OK	791	1167 OK
lyse_terc-P42601	1302	1172	1024 OK	1288 OK	792	1251 OK
lyse_terc-P42601	1532	1402	1141 OK	1520 OK	1141 OK	1520 OK
lyse_terc-P42601	1609	1479	1260 OK	1595 OK	1017	1553 OK
mit-Q58439	293	111	239 OK	285 OK	273 OK	293 OK
mit-Q58439	295	76	241 OK	287 OK	275 OK	295 OK
mit-Q58439	299	78	235 OK	288 OK	267 OK	298 OK
mit-Q58439	328	82	247 OK	315 OK	293 OK	328 OK
nramp-P38925	1707	1448	604	1316 OK	1124	1563 OK
nramp-P38925	1797	1538	620	1340 OK	1198	1693 OK
nramp-P38925	1994	1735	752	1500 OK	1330 OK	1864 OK
nramp-P38925	2221	1962	765	1732 OK	1356	1978 OK
omf-P37974	116	13	103 OK	103 OK	103 OK	103 OK
omf-P37974	126	23	105 OK	105 OK	105 OK	105 OK
omf-P37974	127	24	104 OK	104 OK	104 OK	104 OK
omf-P37974	132	29	118 OK	118 OK	118 OK	118 OK
ptype_atpase-P13587	17585	15757	5944	13098 OK	3134	5944
ptype_atpase-P13587	18159	16331	7879	13548 OK	2138	5265

ptype_atpase-P13587	22256	20428	9000	16333 OK	5477	14247
ptype_atpase-P13587	25696	23868	11577	19154 OK	3109	11328
rnd-P13510	1305	1173	1250 OK	1303 OK	1291 OK	1305 OK
rnd-P13510	591	459	560 OK	589 OK	589 OK	591 OK
rnd-P13510	624	492	587 OK	623 OK	616 OK	624 OK
rnd-P13510	851	719	815 OK	849 OK	846 OK	851 OK
rnd-P37972	10527	10424	5846	5957	2893	5305
rnd-P37972	10996	10893	5841	5993	2672	5098
rnd-P37972	16285	16182	8389	8721	4721	7993
rnd-P37972	8447	8344	4536	4660	2043	4007
rnd-P77214	210	205	210 OK	210 OK	17	196 OK
rnd-P77214	233	228	233 OK	233 OK	66	222 OK
rnd-P77214	79	74	79 OK	79 OK	7	74 OK
rnd-P77214	91	86	91 OK	91 OK	10	88 OK
tog_mgte-Q5SMG8	379	203	240	311 OK	346 OK	379 OK
tog_mgte-Q5SMG8	413	237	274	382 OK	392 OK	413 OK
tog_mgte-Q5SMG8	492	316	316	455 OK	443 OK	492 OK
tog_mgte-Q5SMG8	503	327	319	443 OK	456 OK	502 OK
zip-O94639	242	0	241 OK	241 OK	241 OK	242 OK
zip-O94639	248	11	241 OK	247 OK	245 OK	248 OK
zip-O94639	253	13	246 OK	252 OK	249 OK	253 OK
zip-O94639	255	6	250 OK	254 OK	253 OK	255 OK
zip-P0A8H3	154	22	153 OK	154 OK	154 OK	154 OK
zip-P0A8H3	155	26	151 OK	155 OK	153 OK	155 OK
zip-P0A8H3	173	3	162 OK	172 OK	166 OK	173 OK
zip-P0A8H3	177	4	167 OK	177 OK	167 OK	177 OK

8.15. Liste des organismes de référence effectifs de la recherche par orthologie (Forty-Two)

abc_fet-P44513	**cdf-P13512**	**mit-Q58439**
Chlamydomonas reinhardtii_3055 Cyanophora paradoxa_2762 Dictyostelium discoideum_44689 Euglena gracilis_3039 Guillardia theta_905079 Klebsormidium flaccidum_3175 Porphyridium purpureum_35688 Symbiodinium microadriaticum_2951	Cyanophora paradoxa_2762 Dictyostelium discoideum_44689 Euglena gracilis_3039 Guillardia theta_905079 Klebsormidium flaccidum_3175 Methanothrix soehngenii_GCF_000204415.1 Nitrososphaera viennensis_GCF_000698785.1 Porphyridium purpureum_35688	Acetomicrobium mobile_GCF_000266925.1 Cyanophora paradoxa_2762 Dehalogenimonas lykanthroporepellens_GCF_000143165.1 Dictyostelium discoideum_44689 Euglena gracilis_3039 Symbiodinium microadriaticum_2951 Thermococcus sibiricus_GCF_000022545.1 Thermotoga sp._GCF_000832145.1
abc_fet-P72827	**cdf-P53735**	**nramp-P38925**
Acetomicrobium mobile_GCF_000266925.1 Bordetella pertussis_GCF_000195715.1 Escherichia coli_GCF_000800765.1 Fructobacillus pseudoficulneus_GCF_001047115.1 Methanothrix soehngenii_GCF_000204415.1 Pyrodictium delaneyi_GCF_001412615.1 Sphaerochaeta coccoides_GCF_000208385.1 Thermococcus sibiricus_GCF_000022545.1		Chlamydomonas reinhardtii_3055 Cyanophora paradoxa_2762 Dictyostelium discoideum_44689 Euglena gracilis_3039 Guillardia theta_905079 Klebsormidium flaccidum_3175 Monosiga brevicollis_81824 Symbiodinium microadriaticum_2951
abc_mzt-O34610	**copd-P12377**	**omf-P37974**
Acetomicrobium mobile_GCF_000266925.1 Corynebacterium kutscheri_GCF_000980835.1 Fructobacillus pseudoficulneus_GCF_001047115.1 Methanobacterium lacus_GCF_000191585.1 Methanothrix soehngenii_GCF_000204415.1 Nitrososphaera viennensis_GCF_000698785.1 Thermotoga sp._GCF_000832145.1 Waddlia chondrophila_GCF_000092785.1	Bordetella pertussis_GCF_000195715.1 Corynebacterium kutscheri_GCF_000980835.1 Escherichia coli_GCF_000800765.1 Nitrososphaera viennensis_GCF_000698785.1	Acetomicrobium mobile_GCF_000266925.1 Bordetella pertussis_GCF_000195715.1 Escherichia coli_GCF_000800765.1 Hydrogenobacter thermophilus_GCF_000010785.1 Porphyromonas sp._GCF_000768935.1 Thermosulfurimonas dismutans_GCF_001652585.1 Waddlia chondrophila_GCF_000092785.1 Wolinella succinogenes_GCF_000196135.1
abc_mzt-P0A4G4	**ctr-Q9P7F9**	**ptype_atpase-P13587**
Corynebacterium kutscheri_GCF_000980835.1 Fructobacillus pseudoficulneus_GCF_001047115.1 Hydrogenobacter thermophilus_GCF_000010785.1 Methanobacterium lacus_GCF_000191585.1 Methanothrix soehngenii_GCF_000204415.1 Nitrososphaera viennensis_GCF_000698785.1 Thermosulfurimonas dismutans_GCF_001652585.1 Waddlia chondrophila_GCF_000092785.1	Chlamydomonas reinhardtii_3055 Cyanophora paradoxa_2762 Dictyostelium discoideum_44689 Emiliania huxleyi_280463 Euglena gracilis_3039 Guillardia theta_905079 Pseudo-nitzschia multiseries_37319 Symbiodinium microadriaticum_2951	Chlamydomonas reinhardtii_3055 Cyanophora paradoxa_2762 Dictyostelium discoideum_44689 Euglena gracilis_3039 Guillardia theta_905079 Klebsormidium flaccidum_3175 Monosiga brevicollis_81824 Pseudo-nitzschia multiseries_37319
abc_nate-P46904	**lyse_ilt-P31545**	**rnd-P13510**
Dehalogenimonas lykanthroporepellens_GCF_000143165.1 Korarchaeum cryptofilum_GCF_000019605.1 Methanobacterium lacus_GCF_000191585.1 Methanomethylophilus alvus_GCF_000300255.2 Porphyromonas sp._GCF_000768935.1 Pyrodictium delaneyi_GCF_001412615.1 Thermotoga sp._GCF_000832145.1 Waddlia chondrophila_GCF_000092785.1	Corynebacterium kutscheri_GCF_000980835.1 Dictyostelium discoideum_44689 Escherichia coli_GCF_000800765.1 Fructobacillus pseudoficulneus_GCF_001047115.1	Acetomicrobium mobile_GCF_000266925.1 Bordetella pertussis_GCF_000195715.1 Escherichia coli_GCF_000800765.1 Hydrogenobacter thermophilus_GCF_000010785.1 Porphyromonas sp._GCF_000768935.1 Thermosulfurimonas dismutans_GCF_001652585.1 Waddlia chondrophila_GCF_000092785.1 Wolinella succinogenes_GCF_000196135.1
abc_nicot-O68104	**lyse_ilt-P38993**	**rnd-P37972**
Aciduliprofundum sp._GCF_000327505.1 Corynebacterium kutscheri_GCF_000980835.1 Methanobacterium lacus_GCF_000191585.1 Methanomethylophilus alvus_GCF_000300255.2 Methanothrix soehngenii_GCF_000204415.1 Porphyromonas sp._GCF_000768935.1 Pyrodictium delaneyi_GCF_001412615.1 Thermococcus sibiricus_GCF_000022545.1	Bordetella pertussis_GCF_000195715.1 Chlamydomonas reinhardtii_3055 Emiliania huxleyi_280463 Euglena gracilis_3039 Klebsormidium flaccidum_3175 Monosiga brevicollis_81824 Symbiodinium microadriaticum_2951 Thermosulfurimonas dismutans_GCF_001652585.1	Acetomicrobium mobile_GCF_000266925.1 Bordetella pertussis_GCF_000195715.1 Escherichia coli_GCF_000800765.1 Hydrogenobacter thermophilus_GCF_000010785.1 Porphyromonas sp._GCF_000768935.1 Thermosulfurimonas dismutans_GCF_001652585.1 Waddlia chondrophila_GCF_000092785.1 Wolinella succinogenes_GCF_000196135.1
abc_nicot-Q05594	**lyse_ilt-P40088**	**rnd-P77214**
Aciduliprofundum sp._GCF_000327505.1 Corynebacterium kutscheri_GCF_000980835.1 Methanobacterium lacus_GCF_000191585.1 Methanomethylophilus alvus_GCF_000300255.2 Methanothrix soehngenii_GCF_000204415.1 Porphyromonas sp._GCF_000768935.1 Pyrodictium delaneyi_GCF_001412615.1 Thermococcus sibiricus_GCF_000022545.1	Chlamydomonas reinhardtii_3055 Cyanophora paradoxa_2762 Guillardia theta_905079 Klebsormidium flaccidum_3175 Porphyridium purpureum_35688 Symbiodinium microadriaticum_2951 Thermococcus sibiricus_GCF_000022545.1 Wolinella succinogenes_GCF_000196135.1	Bordetella pertussis_GCF_000195715.1 Escherichia coli_GCF_000800765.1
abc_nicot-Q05598	**lyse_ilt-Q0WFT9**	**tog_mgte-Q5SMG8**
Acetomicrobium mobile_GCF_000266925.1 Aciduliprofundum sp._GCF_000327505.1 Korarchaeum cryptofilum_GCF_000019605.1 Methanobacterium lacus_GCF_000191585.1 Methanomethylophilus alvus_GCF_000300255.2 Methanothrix soehngenii_GCF_000204415.1 Sphaerochaeta coccoides_GCF_000208385.1 Thermococcus sibiricus_GCF_000022545.1	Escherichia coli_GCF_000800765.1	Acetomicrobium mobile_GCF_000266925.1 Bordetella pertussis_GCF_000195715.1 Cyanophora paradoxa_2762 Emiliania huxleyi_280463 Pseudo-nitzschia multiseries_37319 Pyrodictium delaneyi_GCF_001412615.1 Symbiodinium microadriaticum_2951 Thermotoga sp._GCF_000832145.1
abc_pept-P0AFA9	**lyse_mntp-P76264**	**tog_nicot-P23516**
Acetomicrobium mobile_GCF_000266925.1	Acetomicrobium mobile_GCF_000266925.1	Emiliania huxleyi_280463

Aciduliprofundum sp._GCF_000327505.1 Bordetella pertussis_GCF_000195715.1 Korarchaeum cryptofilum_GCF_000019605.1 Pyrodictium delaneyi_GCF_001412615.1 Thermococcus sibiricus_GCF_000022545.1 Thermosulfurimonas dismutans_GCF_001652585.1 Thermotoga sp._GCF_000832145.1	Dehalogenimonas lykanthroporepellens_GCF_000143165.1 Escherichia coli_GCF_000800765.1 Korarchaeum cryptofilum_GCF_000019605.1 Methanomethylophilus alvus_GCF_000300255.2 Wolinella succinogenes_GCF_000196135.1	Fructobacillus pseudoficulneus_GCF_001047115.1 Guillardia theta_905079 Klebsormidium flaccidum_3175 Monosiga brevicollis_81824 Pseudo-nitzschia multiseries_37319 Pyrodictium delaneyi_GCF_001412615.1 Symbiodinium microadriaticum_2951
abc_pept-P33590 Acetomicrobium mobile_GCF_000266925.1 Aciduliprofundum sp._GCF_000327505.1 Escherichia coli_GCF_000800765.1 Korarchaeum cryptofilum_GCF_000019605.1 Symbiodinium microadriaticum_2951 Thermococcus sibiricus_GCF_000022545.1 Thermotoga sp._GCF_000832145.1 Wolinella succinogenes_GCF_000196135.1	**lyse_terc-P42601** Dehalogenimonas lykanthroporepellens_GCF_000143165.1 Emiliania huxleyi_280463 Guillardia theta_905079 Hydrogenobacter thermophilus_GCF_000010785.1 Klebsormidium flaccidum_3175 Porphyridium purpureum_35688 Pseudo-nitzschia multiseries_37319 Symbiodinium microadriaticum_2951	**zip-O94639** Chlamydomonas reinhardtii_3055 Cyanophora paradoxa_2762 Dictyostelium discoideum_44689 Emiliania huxleyi_280463 Euglena gracilis_3039 Guillardia theta_905079 Klebsormidium flaccidum_3175 Symbiodinium microadriaticum_2951
abc_pept-P33591 Acetomicrobium mobile_GCF_000266925.1 Aciduliprofundum sp._GCF_000327505.1 Bordetella pertussis_GCF_000195715.1 Escherichia coli_GCF_000800765.1 Korarchaeum cryptofilum_GCF_000019605.1 Pyrodictium delaneyi_GCF_001412615.1 Thermococcus sibiricus_GCF_000022545.1 Thermotoga sp._GCF_000832145.1	**mit-Q01926** Chlamydomonas reinhardtii_3055 Cyanophora paradoxa_2762 Dictyostelium discoideum_44689 Euglena gracilis_3039 Guillardia theta_905079 Klebsormidium flaccidum_3175 Porphyridium purpureum_35688 Pseudo-nitzschia multiseries_37319	**zip-P0A8H3** Chlamydomonas reinhardtii_3055 Cyanophora paradoxa_2762 Emiliania huxleyi_280463 Euglena gracilis_3039 Guillardia theta_905079 Porphyridium purpureum_35688 Pseudo-nitzschia multiseries_37319 Symbiodinium microadriaticum_2951

8.16. Comparaison de la décisivité des placements des rajouts filtrés et non-filtrés

Un poids a d'abord été attribué aux placements des séquences. Ensuite, les données de chaque cluster ont été rassemblées en un fichier en trois colonnes : nom de l'échantillon, numéro de nœud, poids du placement.

```
for t in $(cat placement_list); do \
  perl -anle 'push @{$seen{$F[0]}}, $F[1];
  END{ for $seq (sort keys %seen) { @nodes = sort @{$seen{$seq}};
  for $node (@nodes) { print join q{ }, $seq, $node, 1/@nodes } } }' \
  rAxML_equallyParsimoniousPlacements.$t-names-RAXML-PROTGAMMALG4X-100xRAPIDBP \
  > $t-names-placement.list; \
done

for c in $(cut -d'-' -f-2 placement_list); do \
  grep -h \# $c*-placement.list | \
  perl -nle 'BEGIN{ open $in, "<", shift; while (<$in>)
  { chomp; ($old, $new) = split "\t"; $hash{$new} = $old } }
  s/([^#]+)#\S+ (I[0-9]+) (.+)/$hash{$1}\t$2\t$3/; print' \
  sample_codes_correspondance \
  > $c-placement-regrouped.list; \
done
```

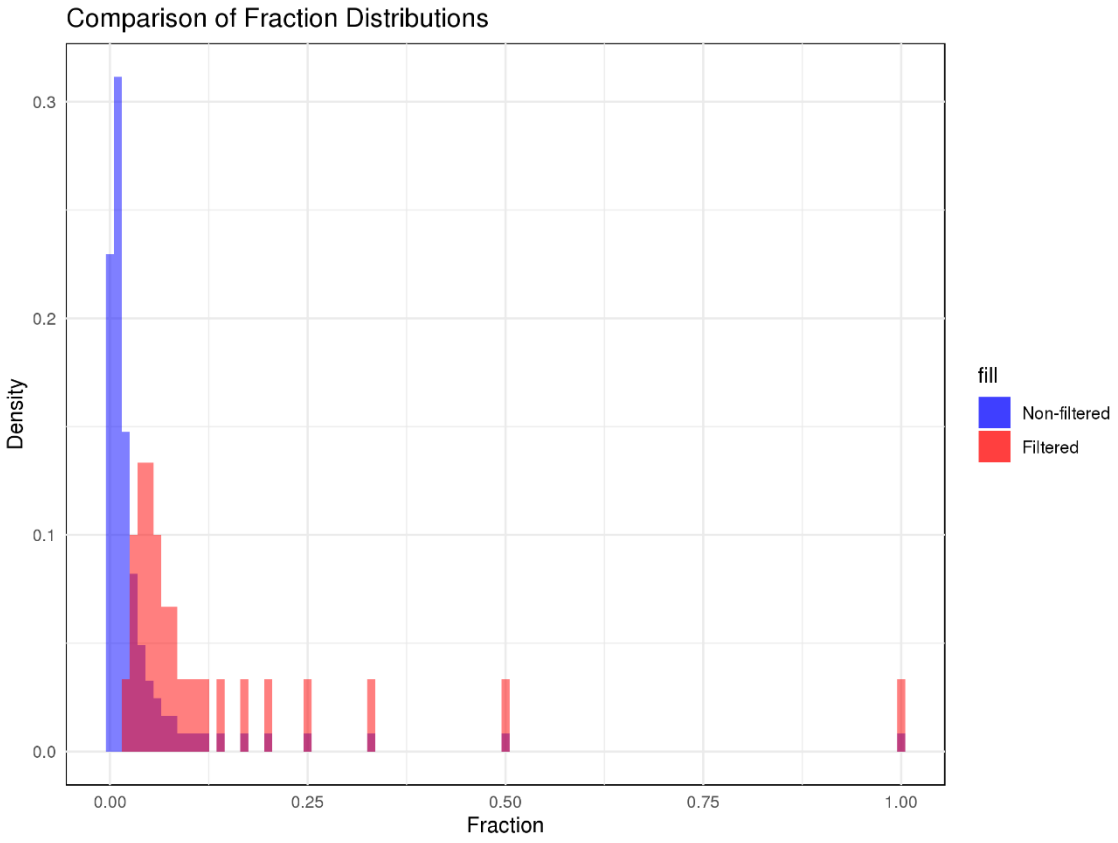
Un décompte des différents poids de placement a d'abord été réalisé.

```
Cut -f3 ../*regrouped.list | sort | uniq -c | \
  perl -nle 's/\s+(\S+)\s+(\S+)/$1\t$2/; print' > décisivité_filtered
cut -f3 *regrouped.list | sort | uniq -c | \
  perl -nle 's/\s+(\S+)\s+(\S+)/$1\t$2/; print' > décisivité_nonfiltered
```

Les histogrammes superposés ont ensuite été générés sous R.

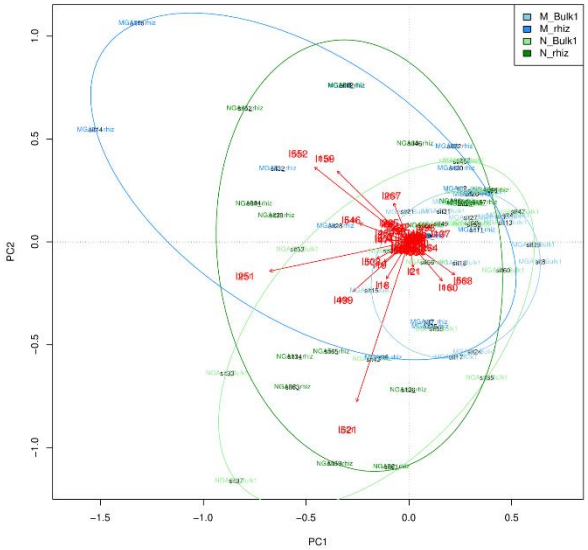
```
# Load necessary libraries
library(ggplot2)
```

```r
# Read data from files
décisivité_nonfiltered <- read.csv("décisivité_nonfiltered",
  Sep = “\t”, header = FALSE)
décisivité_filtered <- read.csv(“décisivité_filtered”,
  sep = "\t", header = FALSE)

# Rename columns
colnames(décisivité_nonfiltered) <- c("Count", "Fraction")
colnames(décisivité_filtered) <- c("Count", "Fraction")

# Density relative histogram

decisiveness_relative_histogram <- ggplot() +
  geom_histogram(data = décisivité_nonfiltered,
    aes(x = Fraction, y = after_stat(count / sum(count)),
    fill = "Non-filtered"), binwidth = 0.01, alpha = 0.5, position = "identity") +
  geom_histogram(data = décisivité_filtered,
    aes(x = Fraction, y = after_stat(count / sum(count)),
    fill = "Filtered"), binwidth = 0.01, alpha = 0.5, position = "identity") +
  labs(title = "Comparison of Fraction Distributions",
       x = "Fraction",
       y = "Density") +
  scale_fill_manual(values = c("Non-filtered" = "blue", "Filtered" = "red")) +
  theme_minimal() +
  theme(panel.background = element_rect(fill = "white"))

ggsave("decisiveness_relative_histogram.png",
  plot = decisiveness_relative_histogram, width = 8, height = 6, dpi = 300)
```

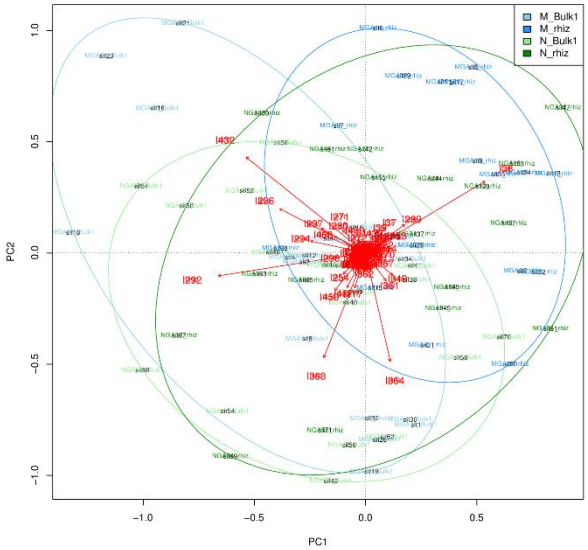


Ces histogrammes relatifs affichent un placement bruité et plus incertain pour les rajouts non-filtrés.

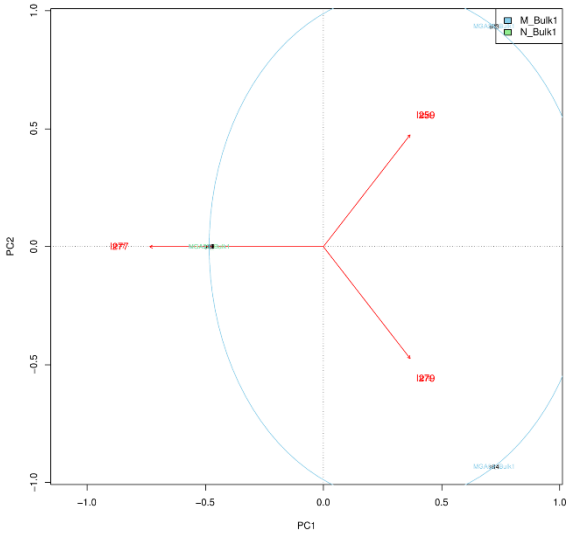
8.17. Cartes ACP



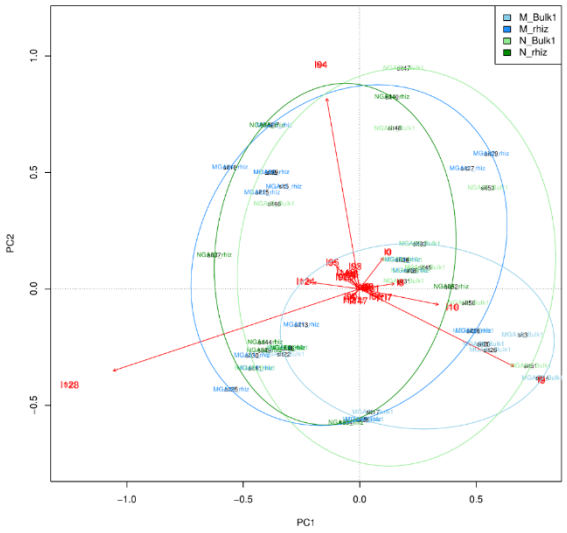
abc_mzt-O34610



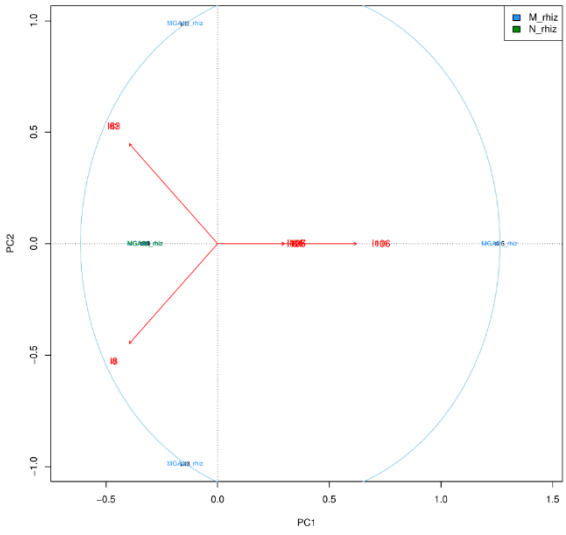
abc_mzt-P0A4G4



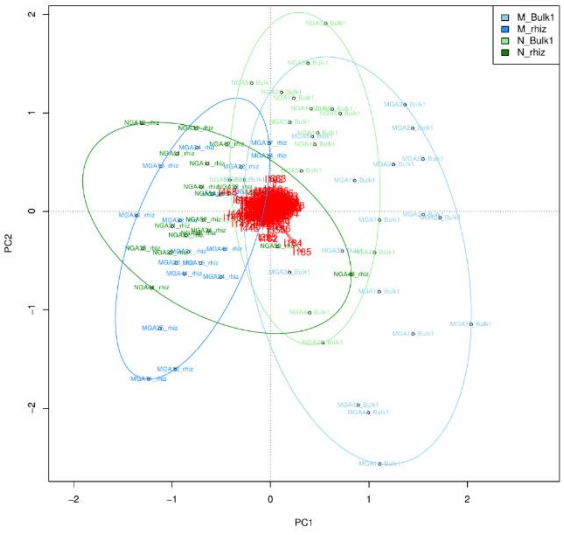
abc_nate-P46904



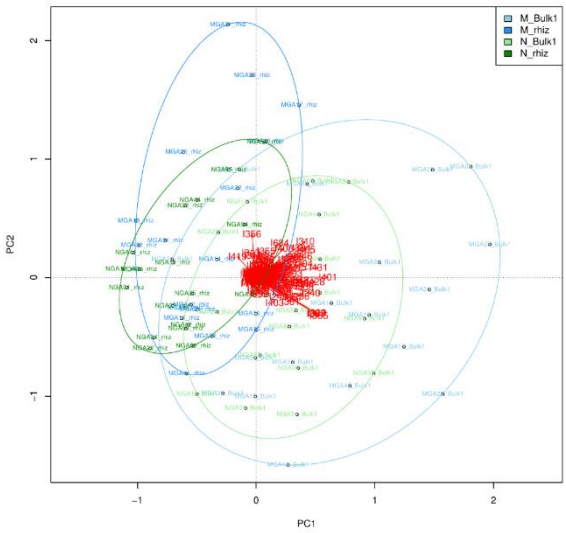
abc_nicot-Q05594



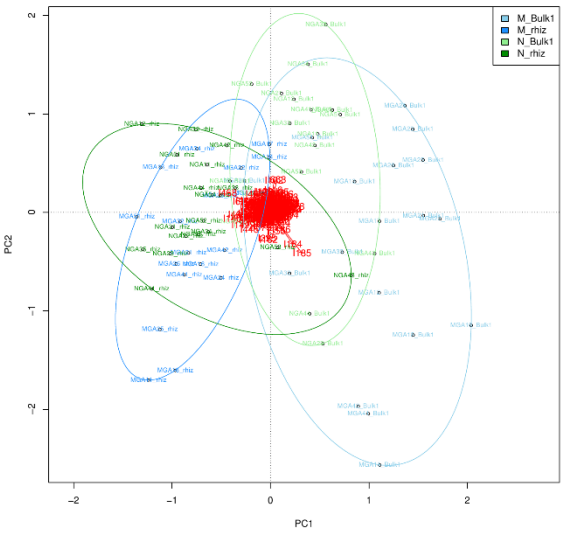
abc_nicot-Q05598



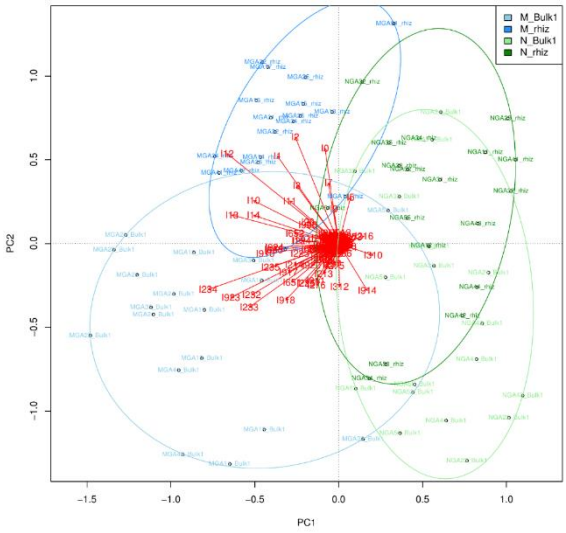
abc_pept-P0AFA9



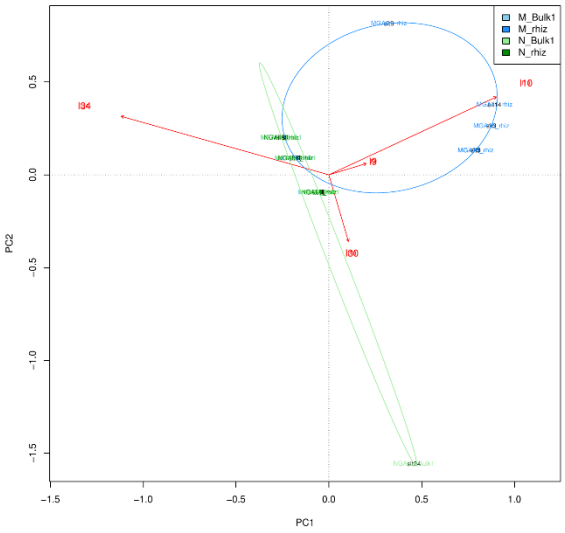
abc_pept-P33590



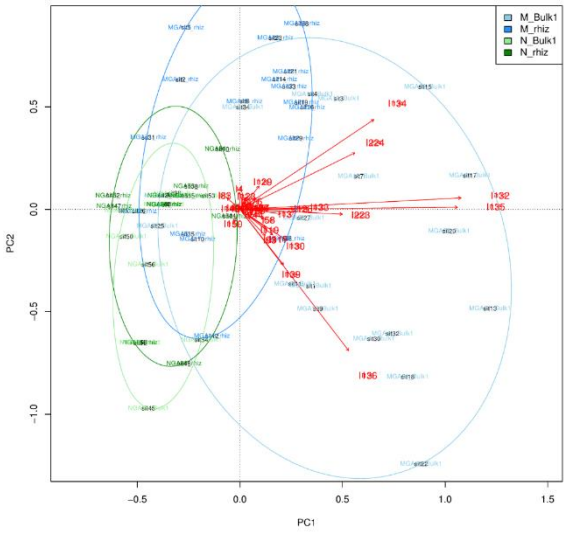
abc_pept-P33591



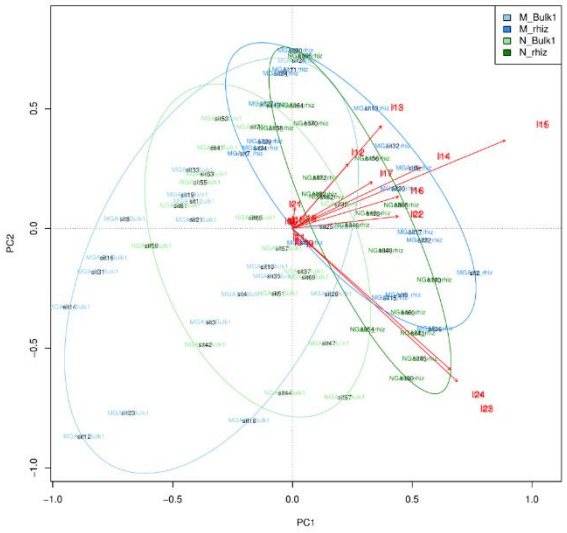
cdf-P13512



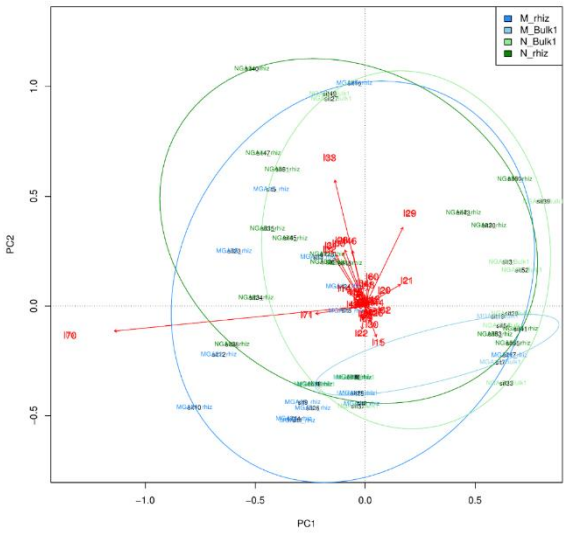
lyse_ilt-P31545



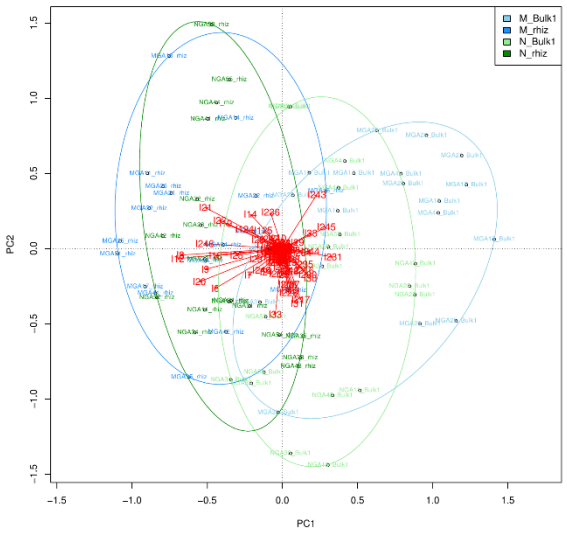
lyse_ilt-P38933



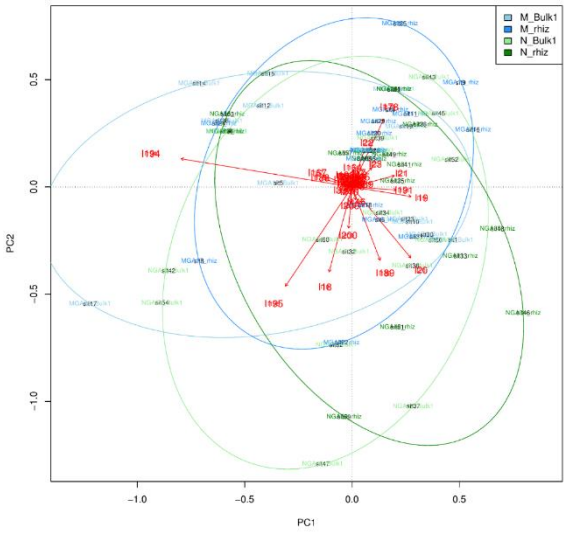
lyse_ilt-Q0WFT9



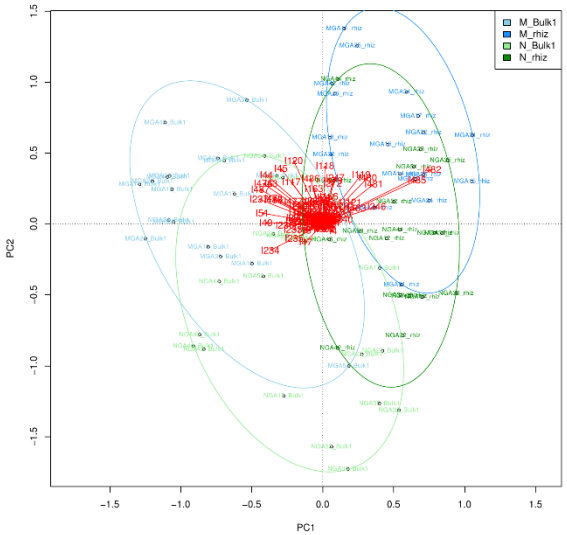
lyse_mntp-P76264



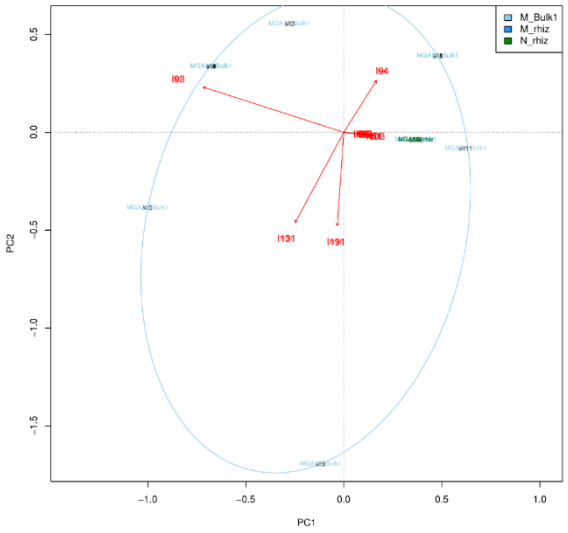
lyse_terc-P42601



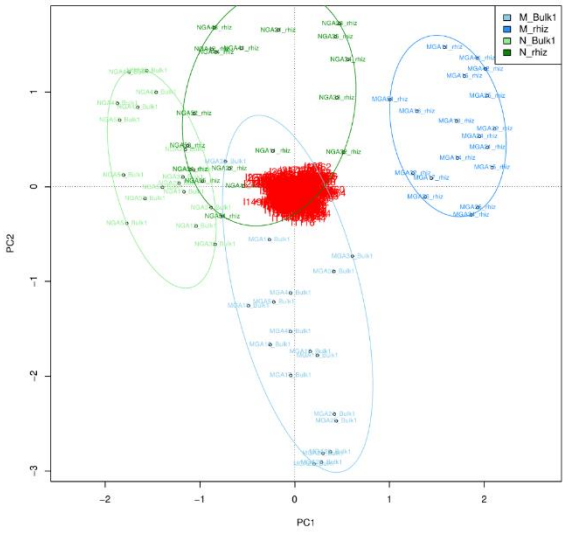
mit-Q58439



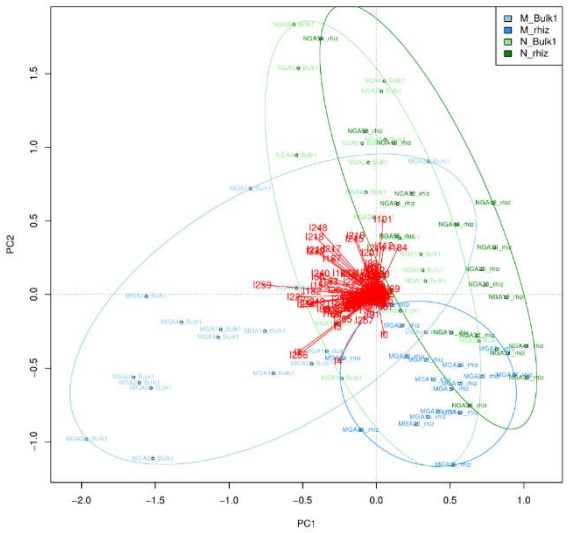
nramp-P38925



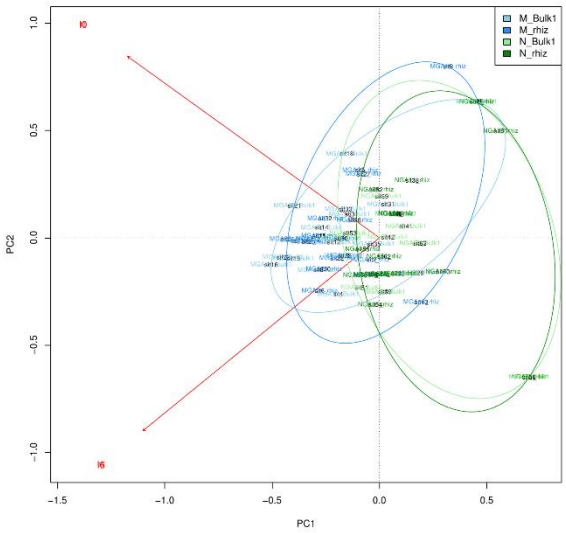
omf-P37974



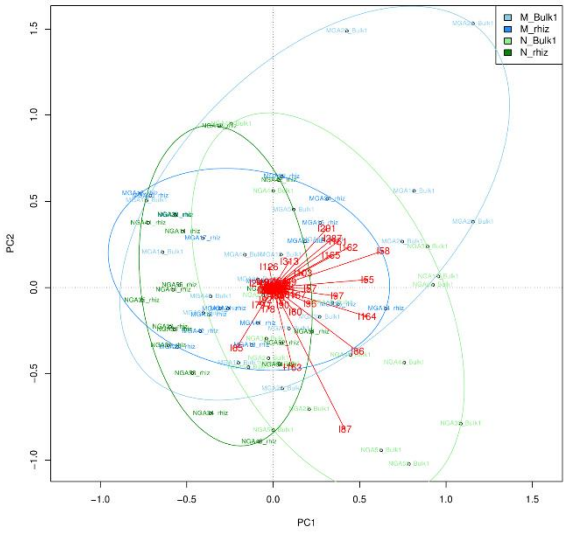
ptype_atpase-P13587



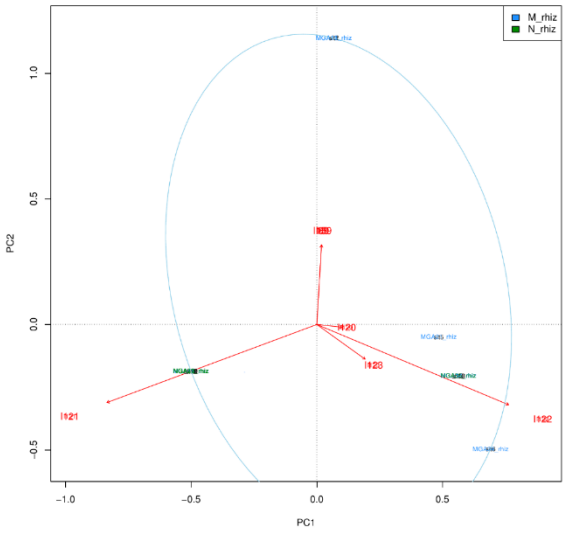
rnd-P13510



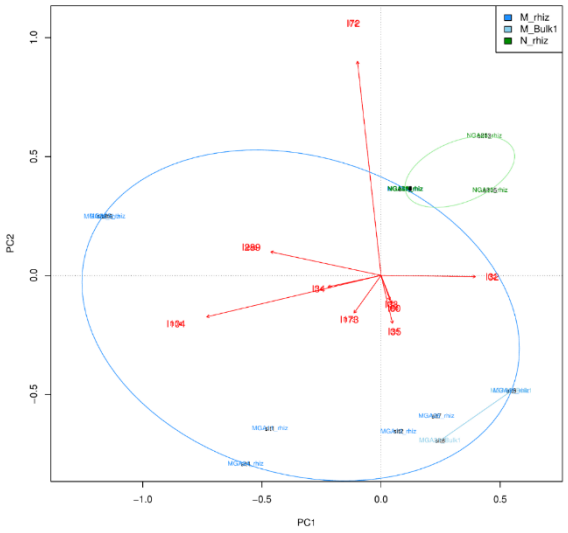
rnd-P77214



tog_mgte-Q5SMG8

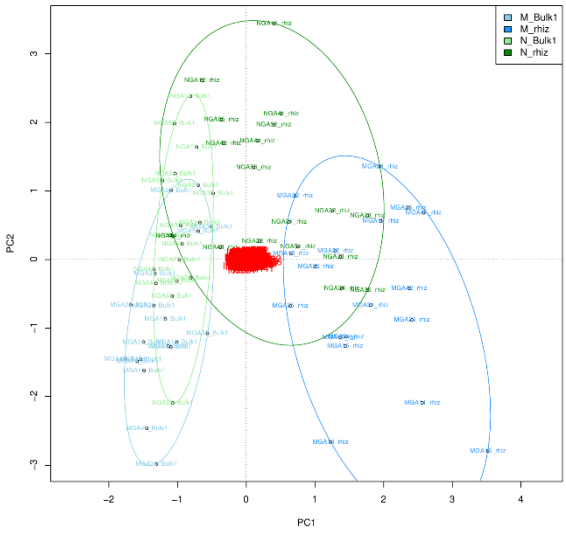


zip-O94639

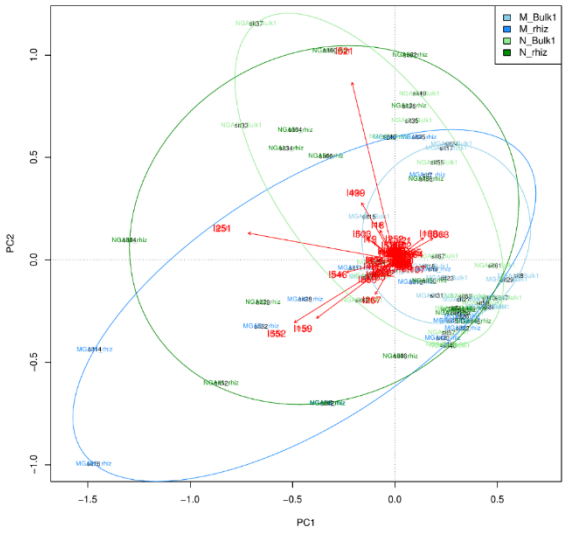


zip-P0A8H3

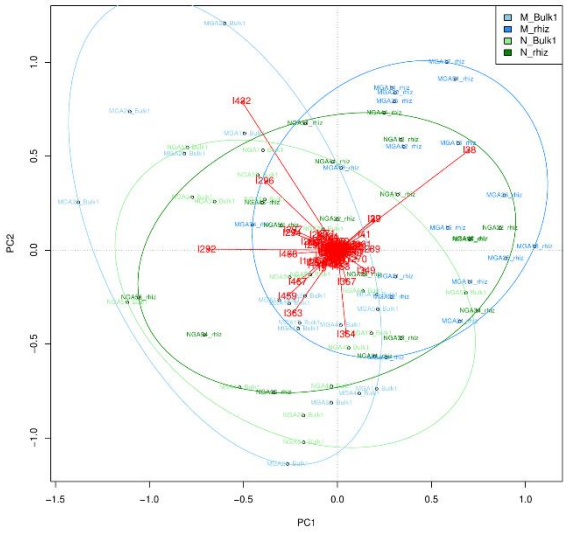
8.18. Cartes ACP des alignements enrichis non-filtrés



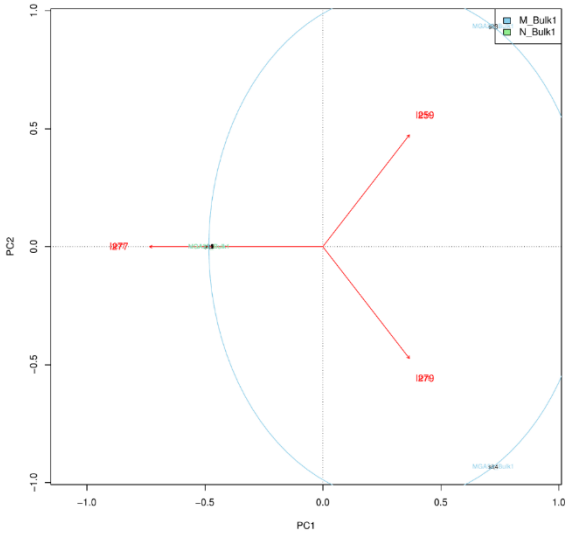
abc_fet-P44513



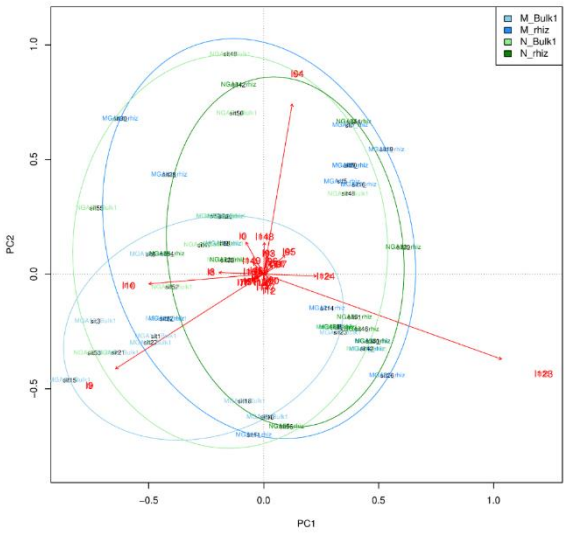
abc_mzt-O34610



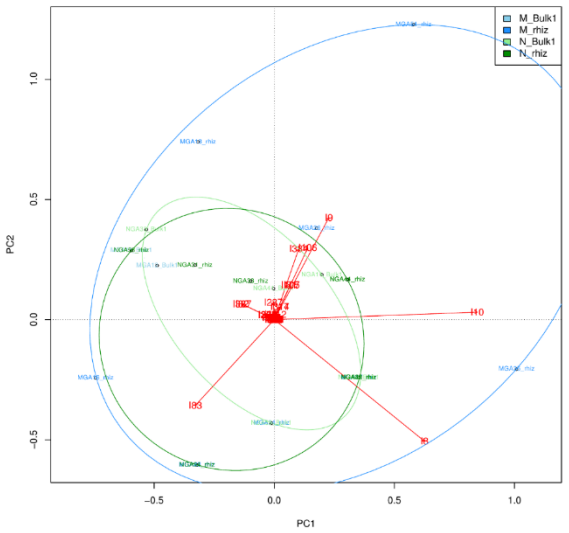
abc_mzt-P0A4G4



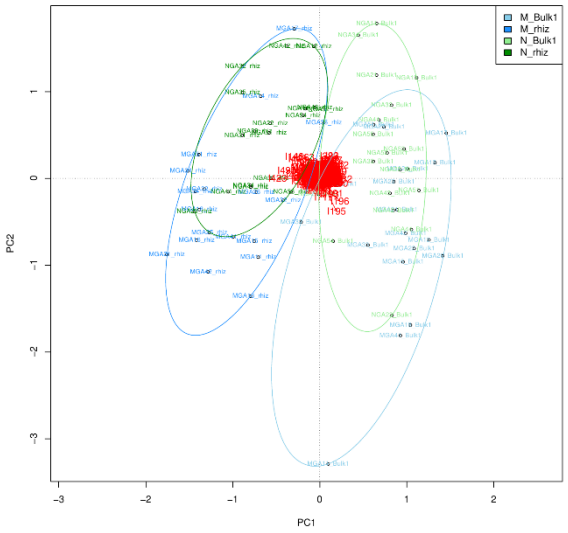
abc_nate-P46904



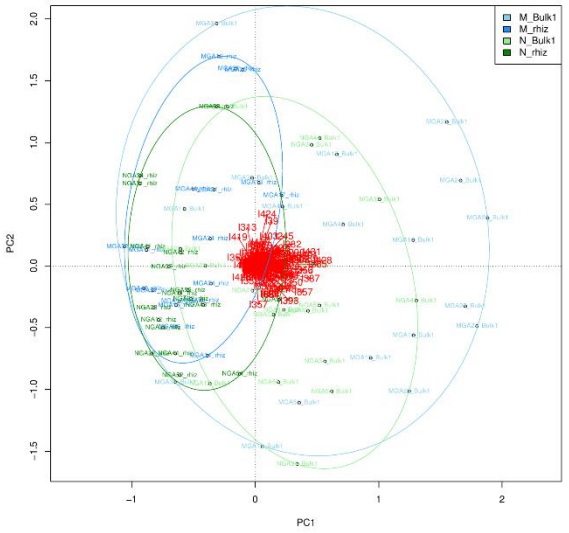
abc_nicot-Q05594



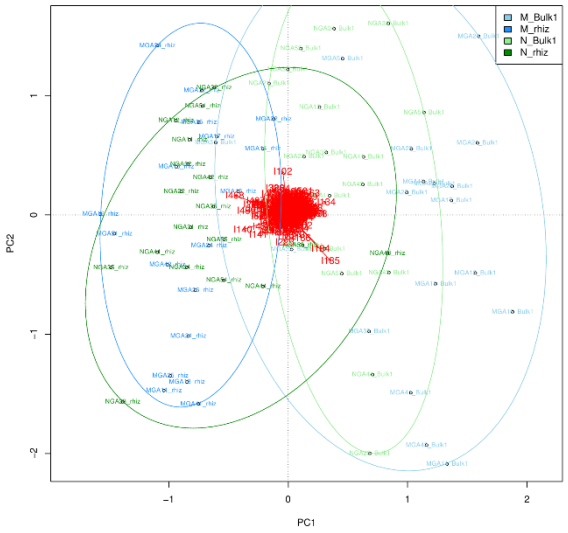
abc_nicot_Q05598



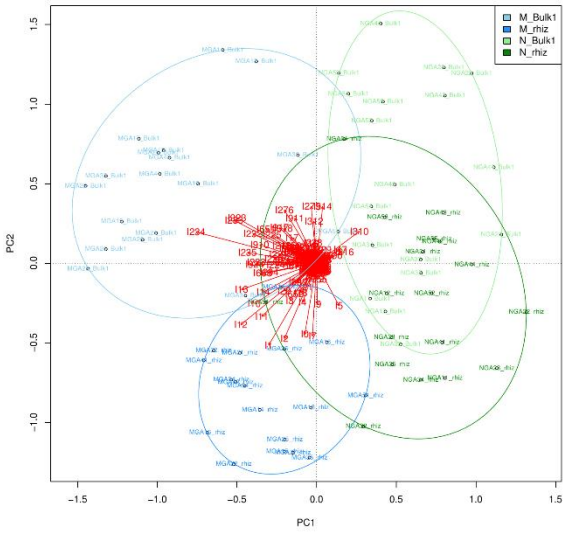
abc_pept-P0AFA9



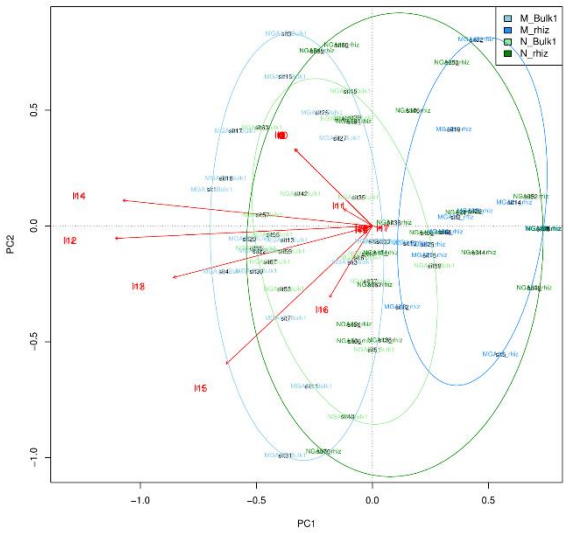
abc_pept-P33590



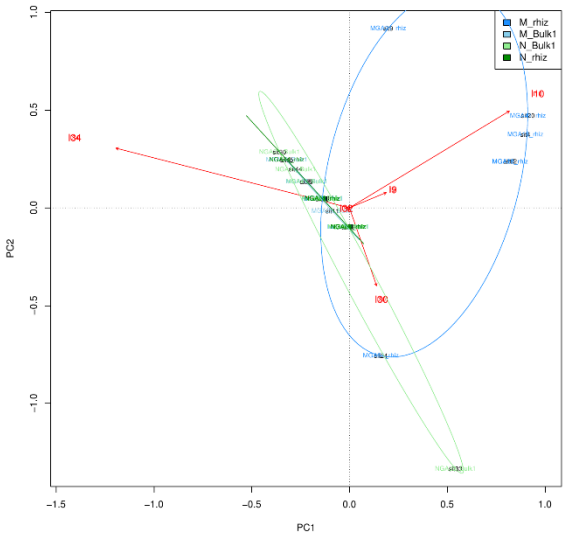
abc_pept-P33591



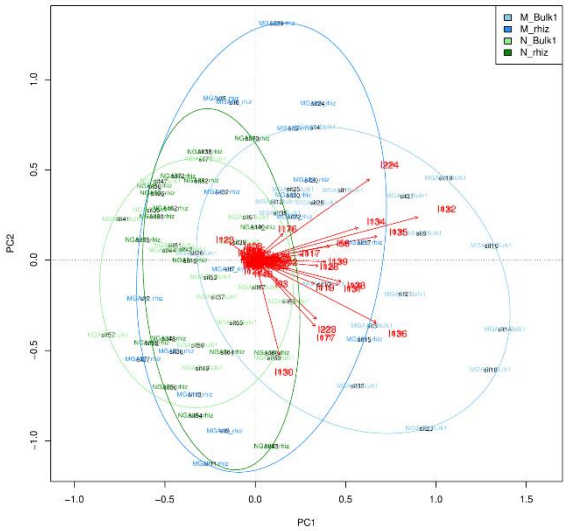
cdf-P13512



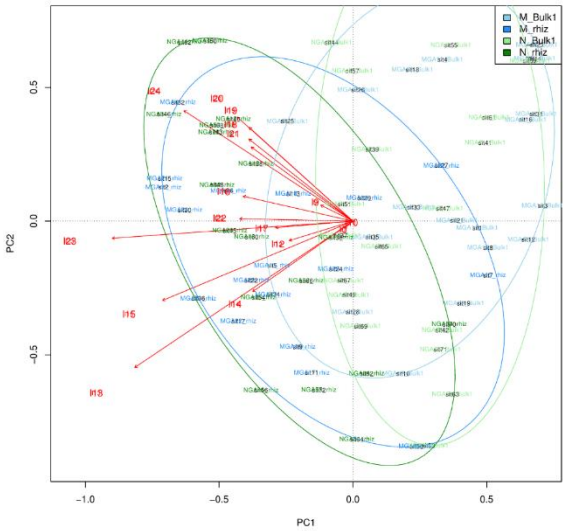
copd-P12377



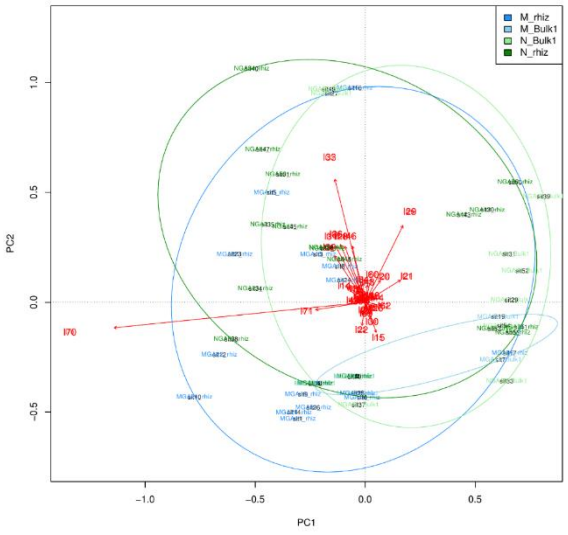
lyse_ilt-P31545



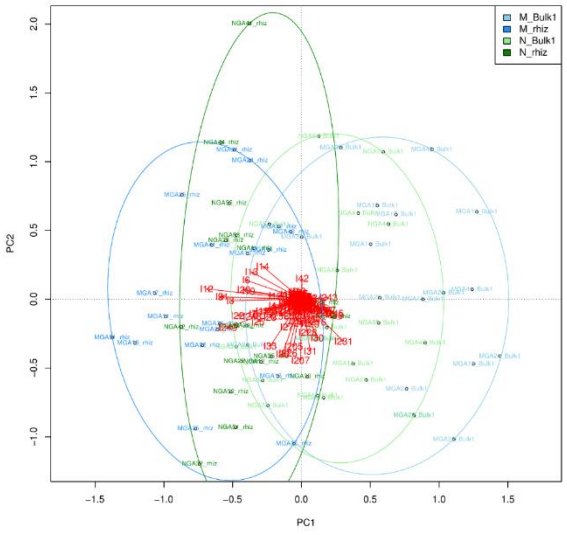
lyse_ilt-P38993



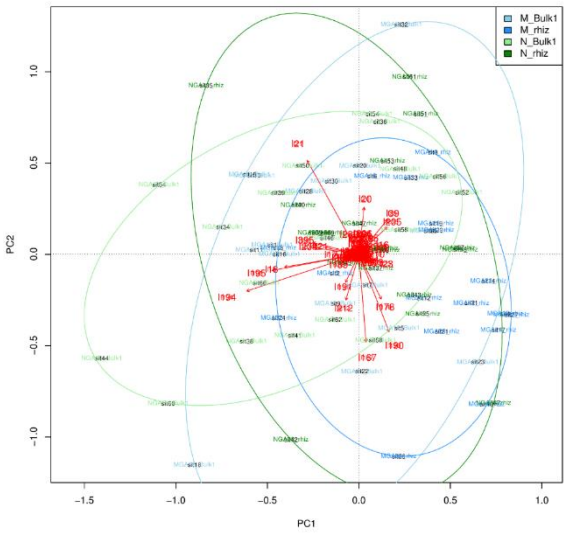
lyse_ilt-Q0WFT9



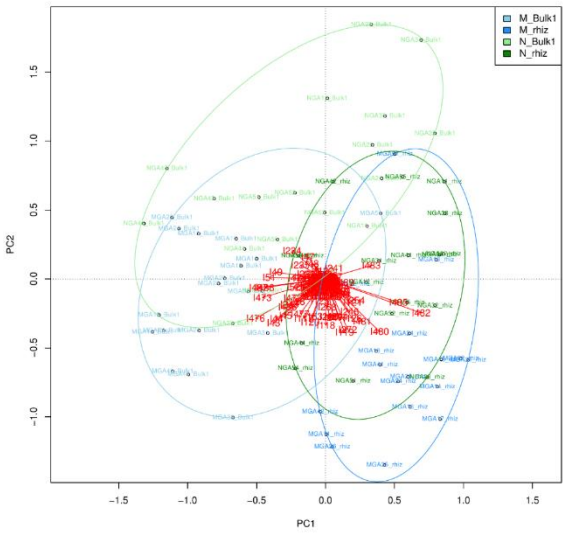
lyse_mntp-P76264



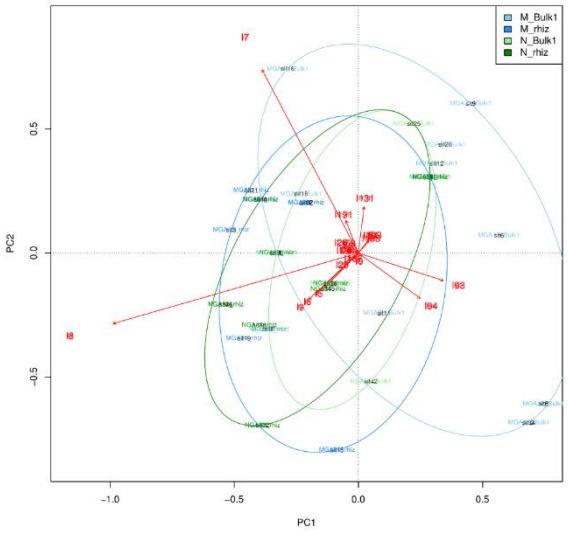
lyse_terc-P42601



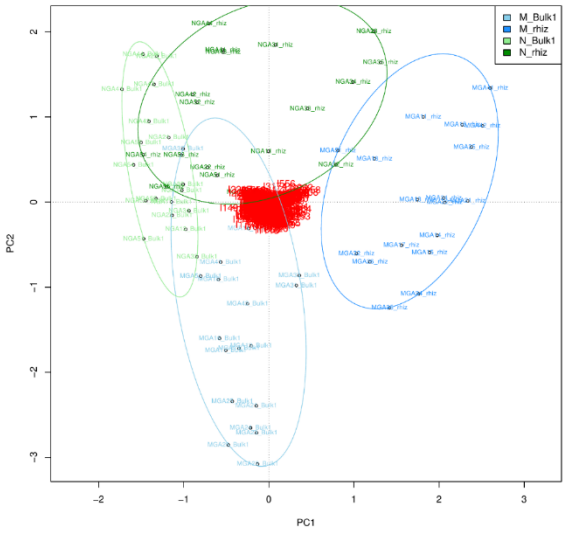
mit-Q58439



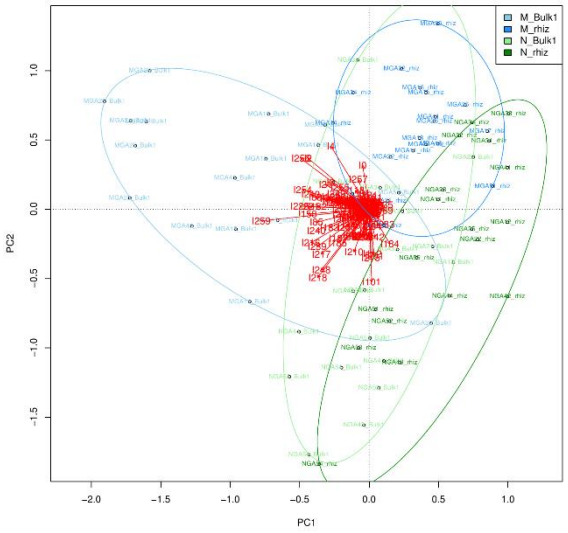
nramp-P38925



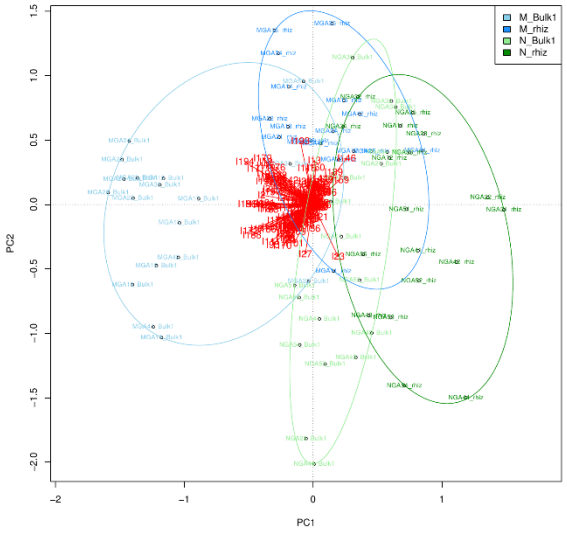
omf-P37974



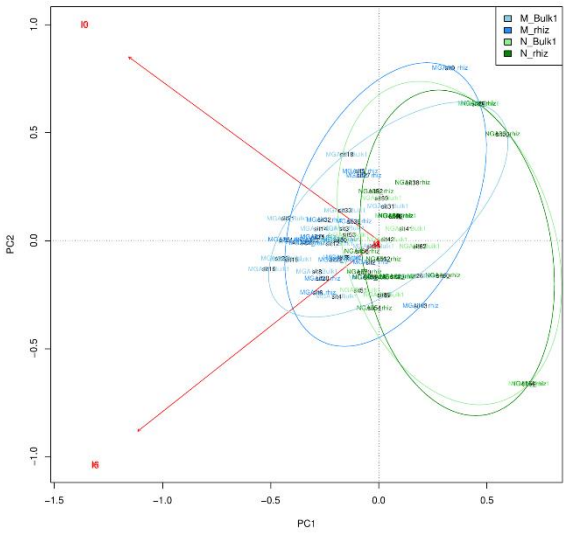
ptype_atpase-P13587



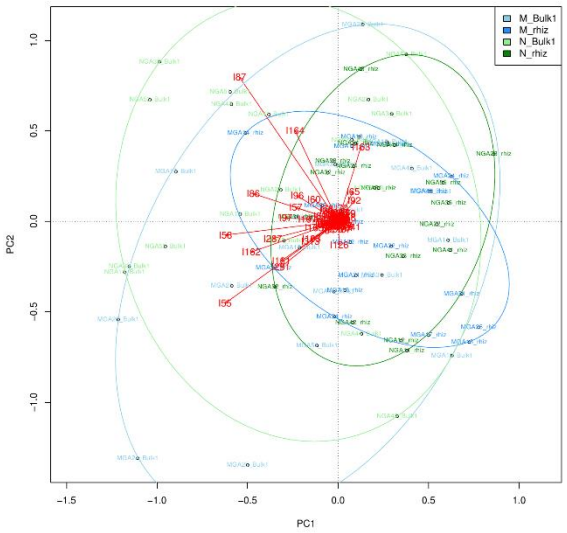
rnd-P13510



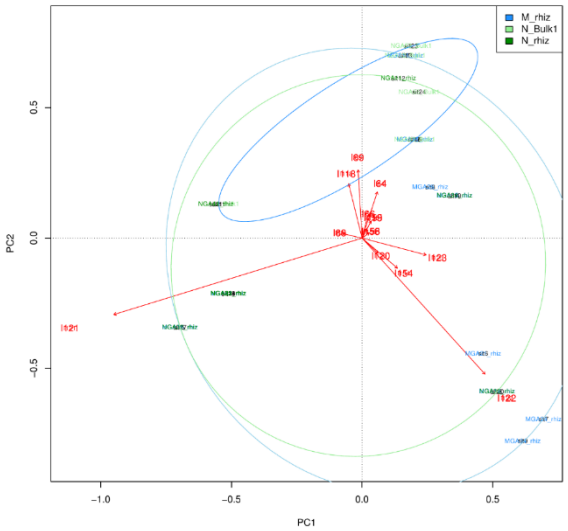
rnd-P37972



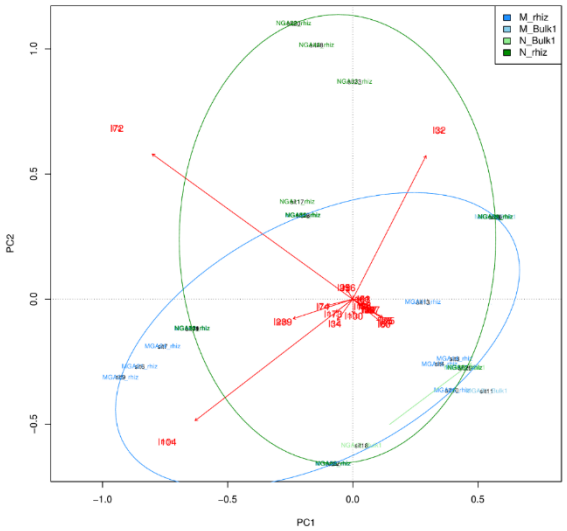
rnd-P77214



tog_mgte-Q5SMG8



zip-O94639



Zip-P0A8H3