



**Faculté des sciences  
Université Mohamed Premier  
Oujda**



## **MASTER SPECIALISE INGENIERIE INFORMATIQUE**

**2018-2019**

### **MEMOIRE DE FIN D'ETUDES**

# **Thème: Analyse des données génomiques à haut débit**

---

**Réalisé par : ABOUBAKAR AHAMADA**

Sous la supervision de :

M. MEZIANE Abdelouafi, Professeur à la faculté des sciences d'Oujda, Maroc.

M. Ben-Youssef NAIMI, PhD, Directeur d'IndaTamin, Oujda, MAROC.

**Membres du jury :**

M. MAZROUI Azzeddine, Professeur à la faculté des sciences d'Oujda

M. MEZIANE Abdelouafi, Professeur à la faculté des sciences d'Oujda

M. GABLI Mohammed, Professeur à la faculté des sciences d'Oujda

Mme. KASSOU Zineb, Responsable chez IndaTamine

Faculté des sciences de l'université Mohamed Premier

Le 23/07/2019



## Résumé

Ce rapport est la synthèse des projets réalisés lors de mon stage de fin d'études chez **IndaTamin®**. Il porte sur l'analyse des données génomiques à haut débit. Cette analyse consiste à mettre en pratique différentes méthodes statistiques et de « data mining » sur les données générées à partir des expériences biologiques pour en extraire des nouvelles connaissances pour les biologistes. La nature de ces données exige une attention particulière sur leur prétraitement avant de procéder à une quelconque analyse.

Durant ce stage j'ai travaillé sur trois projets qui font l'objet de ce rapport. Le premier projet consistait à identifier les gènes impliqués dans le diabète de type 2 chez un groupe d'individus diagnostiqués de cette maladie en se basant sur l'article «*PGC-1α-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes*»[6]. Le deuxième projet consiste à étudier l'effet d'une molécule dite Ganoderma sur des cellules cancéreuses. Nos résultats ont montré que la molécule a un effet direct sur le cycle cellulaire après 36 heures. Enfin le troisième projet consiste à la mise en place d'un modèle prédictif de l'échec tardif du greffe chez des patients ayant reçu une transplantation rénale. Après une phase de sélection des gènes les plus discriminants pour la classification et une phase d'enrichissement des données, nous avons utilisé les 4 méthodes de classification SVM, KNN, PAM(Predcive Analysis of Microarrays) et Random Forest pour entraîner des modèles d'apprentissage que nous avons évalués sur un jeu de données de test. D'une part les modèles de SVM et de Random Forest ont montré une capacité de généralisation par rapport aux autres méthodes. D'autre part nous avons observé une amélioration de la méthode de Random Forest une fois qu'on a réduit le nombre de variables à 20 gènes.

## **Remerciement**

En préambule à ce rapport, nous souhaitons adresser nos remerciements les plus sincères aux personnes ayant contribué au bon déroulement de ce stage.

Nos remerciements s'adressent à Monsieur le Directeur de IndaTamin, Monsieur Ben-Youssef NAIMI de nous avoir accueilli et supervisé durant ces mois à IndaTamin où nous avons passé des moments inoubliables. Ses recommandations, et ses remarques resteront pour nous une vraie leçon à vie.

Nous n'oublions pas de remercier Madame KASSOU Zineb de IndaTamin pour son accompagnement durant ce stage.

Nos remerciements s'adressent également à notre Cher Professeur encadrant, Monsieur MEZIANE Abdelouafi pour ses conseils.

Nos remerciements vont également envers le responsable du master M2I, Monsieur LAKHOUAJA Abdelhak ainsi que tous les Professeurs du master M2I.

Enfin, nous tenons à remercier tous les membres du jury pour avoir accepté d'évaluer notre travail.

# Table des matières

INTRODUCTION .....	2
Chapitre I : Présentation générale .....	3
1. Structure d'accueil.....	3
2. Données génomiques à haut début .....	3
2.1. Gène et niveau d'expression de gène .....	3
2.2. Les puces à ADN .....	4
2.3. Voies de signalisation et bases de données des voies de signalisation .....	8
3. Présentation du sujet .....	9
4. Environnement de travail.....	10
Chapitre II : Processus d'analyse des données à haut débit .....	11
Introduction.....	12
1. Prétraitement des données des puces à ADN.....	12
1.1. Elimination du bruit de fond .....	12
1.2. Normalisation des données.....	13
1.3. Quantification des données .....	15
1.4. Contrôle de qualité des données .....	15
2. Analyse différentielle d'expression de gènes.....	15
2.1. Notion de tests d'hypothèses et p-value .....	15
2.2. Méthodes statistiques pour l'analyse différentielle d'expression des gènes .....	17
3. Analyse d'enrichissement.....	18
3.1. GSEA (Gene Sets Enrichment Analysis) .....	18
3.2. Exemple de calcul du score d'enrichissement pour une voie de signalisation S.....	19
4. Apprentissage supervisé.....	20
4.1. Les machines à vecteurs supports(SVM) : .....	20
4.2. Le Random Forest (RF) : .....	21
4.3. La méthode des k-plus proches voisins (k-pp) : .....	21
4.4. PAM (Prediction Analysis of Microarrays):.....	21
Résumé.....	22
Chapitre III : .....	23
Identification des gènes impliqués dans le diabète de type 2 chez un groupe d'individus .....	23
Contexte .....	23
1. Données.....	23
2. Méthode .....	23
2.1. Prétraitement des données.....	23

2.2.	Contrôle de qualité des données .....	24
3.	Résultats .....	24
3.1.	Analyse différentielle avec SAM .....	24
3.2.	Différence entre les individus malades et les individus normaux.....	26
3.3.	Analyse d'enrichissement.....	27
	Conclusion .....	29
	<b>Chapitre IV :</b> .....	<b>30</b>
	Etude de l'effet de la molécule Ganoderma sur une lignée cellulaire .....	30
	Introduction.....	30
1.	Données.....	30
2.	Méthode .....	32
2.1.	Collection des <i>gene sets</i> dans les bases de données publiques.....	32
2.2.	Analyse d'enrichissement des gene sets.....	34
3.	Résultats .....	34
	Conclusion et perspectives.....	41
	<b>Chapitre V :</b> .....	<b>42</b>
	Mise en place d'un modèle de prédiction de l'échec tardif de la greffe rénale basé sur le profil d'expression des gènes .....	42
	Introduction.....	42
1.	Données.....	42
2.	Méthode .....	43
2.1.	Méthode de sélection des gènes (Features sélection).....	44
2.2.	Méthode adoptée pour la sélection des gènes.....	45
2.3.	SMOTE (Synthetic Minority Over-sampling Technique).....	46
2.4.	Méthode de k-fold cross validation.....	48
3.	Résultats .....	48
3.1.	Comparaison des modèles .....	48
3.2.	Utilisation de 20 tops gènes avec Random Forest .....	51
	Matrice de confusion du RF sur les données d'apprentissage.....	51
	Estimation de l'erreur pour le modèle de Random Forest.....	51
	Évaluation du modèle sur le jeu de données de test .....	52
	Conclusion et perspectives.....	52
	<b>Conclusion générale .....</b>	<b>53</b>
	<b>Références.....</b>	<b>54</b>

## Liste des figures

Figure 1: Synthèse de la protéine (from DNA to protein) .....	4
Figure 2: exemple d'une plaque de puce à ADN .....	5
Figure 3: Illustration du processus de la puce à ADN.....	6
Figure 4: dimension d'une puce affymetrix, source [2] .....	7
Figure 5: matrice M d'expression .....	8
Figure 6: Environnement de travail.....	11
Figure 7: exemple de calcul du score d'enrichissement.....	20
Figure 8: densité du niveau d'expression des 35 biopsies. (bg : background).....	24
Figure 9: SAM plot pour les GDE entre les groupes NGT et DM2 .....	25
Figure 10: GDE et FDR en fonction du paramètre delta.....	25
Figure 11: Différence du niveau d'expression entre les groupes NGT et DM2 .....	26
Figure 12: Cenet plot des 21 gene sets .....	27
Figure 13: FDR des 21 voies de signalisation enrichies .....	27
Figure 14: Courbe du score d'enrichissement d'Oxidative phosphorylation .....	28
Figure 15: Gènes dans Oxidative phosphorylation .....	29
Figure 16: extrait de la matrice de données expérimentales .....	31
Figure 17: clustering hierarchique sur les données expérimentales .....	32
Figure 18: proportion des gènes dans les gene sets .....	33
Figure 19: Processus de collection et traitement des données de AnyGenes.....	34
Figure 20: Courbe du score d'enrichissement du cycle cellulaire. Base de données : AnyGenes .....	35
Figure 21: Courbe du score d'enrichissement du cycle cellulaire. Base de données : Reactome .....	35
Figure 22: Courbe du score d'enrichissement du cycle cellulaire. Base de données : BioCarta.....	36
Figure 23: Courbe du score d'enrichissement du cycle cellulaire. Base de données : WikiPathways .....	36
Figure 24: Courbe du score d'enrichissement du cycle cellulaire. Base de données : GO.....	36
Figure 25: Courbe du score d'enrichissement du cycle cellulaire. Base de données : KEEG .....	36
Figure 26: Heatmap des voies de signalisation en réponse de la molécule de Ganoderma.....	37
Figure 27: Dotplot des voies de signalisations enrichies.....	38
Figure 28: EnrichMap des pathways de Reactom .....	39
Figure 29: Cnet plot des facteurs exprimés en réponse de la molécule. ....	39
Figure 30: Cenet plot pour les gene sets de AnyGenes.....	41
Figure 31: Worflow du modèle prédictif .....	43
Figure 32: SAM Plots .....	45
Figure 33: Pseudo code de SVM-RFE (source [19], page 3) .....	46
Figure 34: SMOTE algorithme, source [27]. .....	47
Figure 35: Données avant et après SMOTE .....	47
Figure 36: Illustration de la méthode de k-fold cross validation.....	48
Figure 37: Mesures de performances des modèles .....	49
Figure 38: Matrices de confusion et statistiques des Random Forest (gauche) et SVM (droite) .....	50
Figure 39: Importance des variables pour Random Forest .....	50
Figure 40: Matrice de confusion de Random Forest pour 20 variables .....	51
Figure 41: Out-Of-Bag Error.....	51
Figure 42: Matrice de confusion et visualisation pour le jeu de données de test .....	52

## Sigles et abréviations

INDATAMIN	International Data Mining
ADN	Acide Désoxyribonucléique
ARNm	Acide Ribonucléique messager
MSigDB	Molecular Signature Database
KEEG	Kyoto Encyclopedia of Gene and Genomes
GO	Gene Ontology
PID	Pathways Interactive Database
RMA	Robust microarrays of Analysis
MAS5.0	MicroArrays Suit 5.0
GDE	Gene différentiellement exprimé
FC	Fold change
SAM	Significance analysis of microarrays
FDR	False discovery rate
NGT	Normal glucose tolerance
DM2	Diabète mellitus 2
GSEA	Gene set enrichment analysis
ES	Enrichment score
NES	Normal enrichment score
SVM	Support vectors machine
PAM	Predictive analysis of microarrays
k-NN	k-Nearest Neighbors
RF	Random Forest
OOB	Out-Of-Bag



# INTRODUCTION

Comprendre les causes principales des maladies chez les êtres vivants a toujours été au centre de la recherche médicale et biologique. L'arrivée des technologies à haut débit notamment les puces à ADN a grandement contribué à la compréhension des mécanismes biologiques dans un organisme, ce qui permet de développer des stratégies thérapeutiques. Grâce à ces technologies, le génome humain ainsi que celui d'autres espèces notamment les modèles du laboratoire ont été complètement séquencés. Ce qui a généré une énorme quantité de données répertoriées dans des bases de données publiques. L'exploration de ces données permet d'apporter des réponses aux questions les plus préoccupantes chez les biologistes. Il est désormais possible, grâce à un simple prélèvement d'un échantillon biologique, d'identifier les gènes responsables d'une maladie comme le cancer ou le diabète, ou encore de prédire l'état de santé d'un individu dans le futur, rien qu'en se basant sur le profil de ces biomarqueurs (gènes). Ces données sont soumises à des multiples contraintes qu'il faut en tenir compte notamment le bruit de fond lié aux instruments de mesure et technologies utilisés et leur grande voluminosité.

J'ai effectué mon stage de fin d'études au sein de la société **IndaTamin®**, une société de « data mining » en biotechnologie basée à Oujda. Au cours de cette période, je me suis intéressé à l'analyse des données génomiques dites "à haut débit". Ma mission principale consistait à mettre en œuvre les méthodes et outils d'analyse et de fouille de données sur des données issues des expériences biologiques pour accompagner l'équipe des biologistes dans la prise des décisions. Trois projets ont été réalisés durant cette période. Un premier projet consistait à identifier les gènes en corrélation avec le diabète de type 2 chez un groupe de patients. L'utilisation des méthodes statistiques telles que SAM (Significance Analysis of Microarrays) et GSEA (Gene Set Enrichment Analysis) a permis d'aboutir à des résultats similaires à ceux d'un article de référence sur lequel nous nous sommes basés [6]. Un deuxième projet portant sur l'étude de l'effet d'une molécule testée sur une lignée cellulaire, au cours duquel nous avons exploré les ressources génomiques publiques pour confirmer nos résultats obtenus suite à une analyse d'enrichissement avec la méthode GSEA. Enfin, un troisième projet qui consistait à mettre en place un modèle prédictif pour prédire l'échec tardif du greffage rénale chez des patients à partir du profil d'expression des gènes. Les quatre méthodes de classification SVM, k-NN, PAM (Predictive Analysis of Microarrays) et Random Forest ont été utilisées et comparées pour retenir le modèle ayant une meilleure capacité de généralisation permettant de mieux prédire des nouveaux cas.

Dans l'objectif de faciliter la compréhension de notre travail, nous avons divisé ce document en cinq chapitres. Dans le premier chapitre, nous avons présenté la structure d'accueil, le projet du stage, la nature des données sur lesquelles nous avons travaillé et l'environnement de travail. Le deuxième chapitre est consacré à une présentation théorique des méthodes utilisées pour les différents projets. Les trois derniers chapitres présentent la méthodologie et les résultats obtenus pour les trois projets respectivement.

# Chapitre I : Présentation générale

Dans ce premier chapitre nous présentons la structure d'accueil, la nature des données sur lesquelles nous avons travaillé, les différentes missions menées durant ce stage et enfin l'environnement de travail.

## 1. Structure d'accueil



Créée en 2010, **IndaTamin®** est une société de « data mining » en biotechnologie, située à Oujda, au Maroc.

Elle est une filiale de la société **AnyGenes**, une société de recherches et développement en biotechnologie basée à Paris. **IndaTamin®** se spécialise dans la recherche d'informations génétiques au niveau de différentes bases de données génomiques publiques. L'objectif est de permettre l'analyse haut-débit du transcriptome de différentes espèces. Cette analyse haut débit est basée sur l'utilisation de différentes technologies telles que les puces à ADN ou PCR quantitative en temps réel. A travers une recherche d'informations approfondie du transcriptome, la société **IndaTamin®** propose des schémas de cascades signalétiques qu'empruntent les réseaux de gènes, et des outils (logiciels) permettant l'interprétation des données.

## 2. Données génomiques à haut début

Durant notre stage, nous avons travaillé avec des données génomiques à haut débit. L'appellation "données à haut débit" désigne les données générées suite aux expériences sur des échantillons biologiques. Ces expériences impliquant différentes technologies telles que la robotique, l'optique, le traitement d'images, etc. L'objectif est de mesurer le niveau d'expression des milliers de gènes pour ensuite identifier les gènes dont le niveau d'expression varie considérablement d'un état à un autre. La technologie à haut débit qui prédomine pour la mesure du niveau d'expression des gènes est celle des puces à ADN (DNA microarrays ou Gene Chip) que nous introduisons dans les paragraphes qui suivent.

### 2.1. Gène et niveau d'expression de gène

L'acide désoxyribonucléique (ADN) est une molécule présente dans les cellules vivantes. C'est cette molécule qui contient toute l'information génétique appelée **génome**, permettant le développement, le fonctionnement, et la reproduction des êtres vivants. La molécule ADN est structurée en une double hélice composée de deux brins complémentaires. **L'ADN est à l'origine de la synthèse des protéines par l'intermédiaire de l'acide ribonucléique messager**

**(ARNm).** Ce dernier est une copie transitoire d'une proportion de l'ADN correspondant à un gène. Le gène est alors considéré comme un fragment d'ADN. La synthèse de la protéine se fait en deux étapes :

1. La transcription, qui est le passage de la molécule d'ADN à une autre molécule, l'ARNm.
2. La traduction, qui est la traduction de l'ARNm en protéine.

**Le niveau d'expression d'un gène représente le nombre de copies d'ARNm obtenu par transcription de ce gène à un temps précis.** Le gène est dit surexprimé ou sous-exprimé si son niveau d'expression a augmenté ou a diminué par rapport à un état initial. Le changement du niveau d'expression de gène a une grande influence dans l'organisme. La mesure de l'expression des gènes permet, suite à des analyses, d'identifier les biomarqueurs (gènes) impliqués dans différentes maladies pour envisager une solution médicale.

### Schéma d'illustration

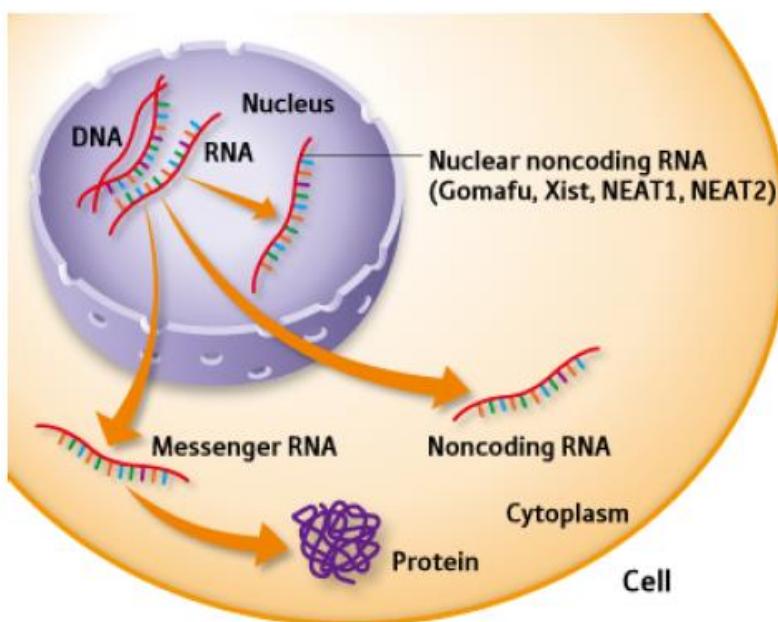


Figure 1: Synthèse de la protéine (from DNA to protein)

## 2.2. Les puces à ADN

De sa nature physique, une puce à ADN est constituée d'un support solide, souvent une lame de verre ou du silicium sur lequel sont déposés des fragments d'ADN (les sondes). Les puces à ADN sont utilisées pour mesurer simultanément le niveau d'expression de plusieurs milliers de gènes dans un contexte biologique particulier. Les puces à ADN reposent sur le principe

d'hybridation. En effet deux simples brins complémentaires d'ADN ont la propriété de s'hybrider pour former un ADN complet deux brins.

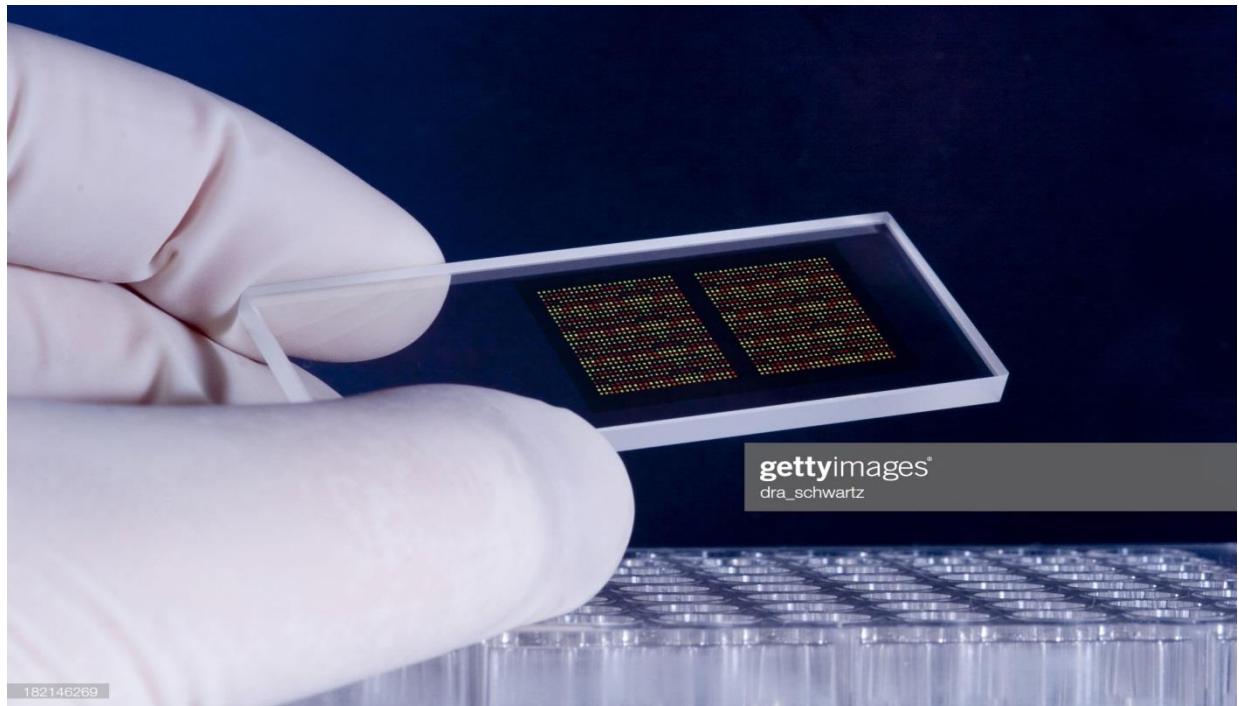


Figure 2: exemple d'une plaque de puce à ADN

### a) Mise en place d'une puce à ADN

#### **Fabrication de la puce :**

Partant du principe d'hybridation, les sondes déposées sur la puce sont ensuite dénaturées afin que leurs fragments d'ADN se retrouvent sous forme simple brin et soient en mesure de recevoir leur brin complémentaire (la cible) présent dans l'échantillon à analyser.

#### **Préparation de la cible :**

Les ARNs extraits de l'échantillon à analyser sont transformés en ADNc (complémentaire au sonde) par transcription inverse. L'ADN ainsi formé est marqué par un nucléotide radioactif (ou fluorochrome) en vue de sa révélation une fois fixé sur une sonde de la puce.

#### **Hybridation :**

Les ADN marqués sont placés sur la puce et celle-ci est mise en incubation pendant une durée au bout de laquelle les sondes et les cibles vont s'apparier pour former l'ADN double brins.

#### **Lecture et acquisition des données :**

Les spots sont excités par un laser et la fluorescence émise est visualisée via un photomultiplicateur (PMT). Une image dont le niveau de gris représente l'intensité de la fluorescence lue sur chaque spot est obtenue. **L'image ainsi obtenue est analysée par un logiciel d'analyse d'image pour produire un fichier binaire contenant les intensités des**

**sondes.** C'est ce fichier binaire qui nous est transmis pour commencer les différentes analyses.

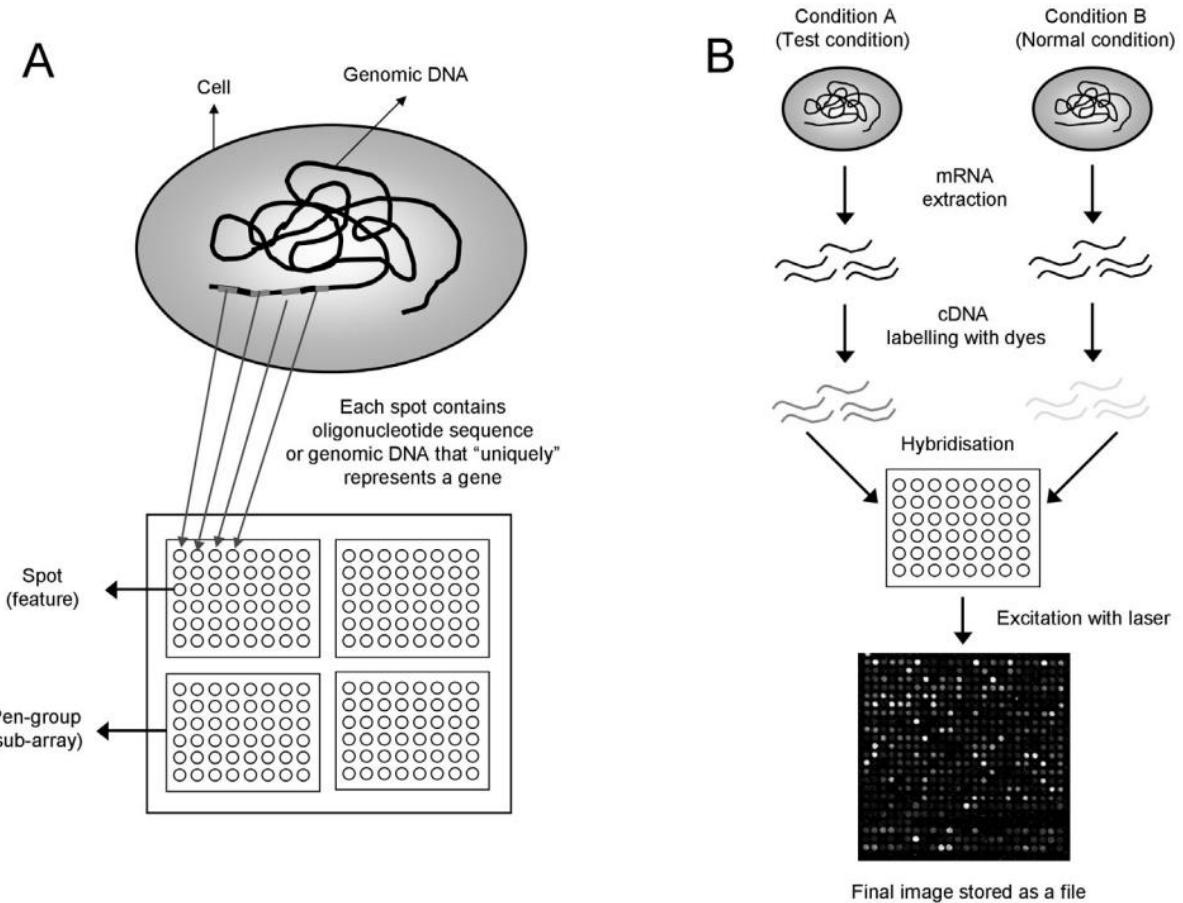


Figure 3: Illustration du processus de la puce à ADN

### b) Les puces affymetrix

Les puces affymetrix sont une des types des puces à ADN qui existent sur le marché et font l'objet de ce document. Ils sont à l'origine fabriquées par la société américaine Affymetrix, Inc qui est rachetée par la société américaine Thermo Fisher Scientific en Mars 2016. Les puces affymetrix sont utilisées pour l'analyse d'expression et détecter les gènes particuliers dans plusieurs milliers de gènes. Dans les puces affymetrix on utilise la notion de "*probe set*" pour désigner un gène. Un "*probe set*" contient entre 11 à 20 '*probes*' qui représentent différentes mesures de l'expression d'un gène [1]. Ce qui fait que malgré sa petite dimension d'environ  $1.28 \times 1.28 \text{ cm}^2$ , une puce affymetrix peut contenir plusieurs millions des "*probes*" et peut mesurer jusqu'à plus de 16 Méga octets.

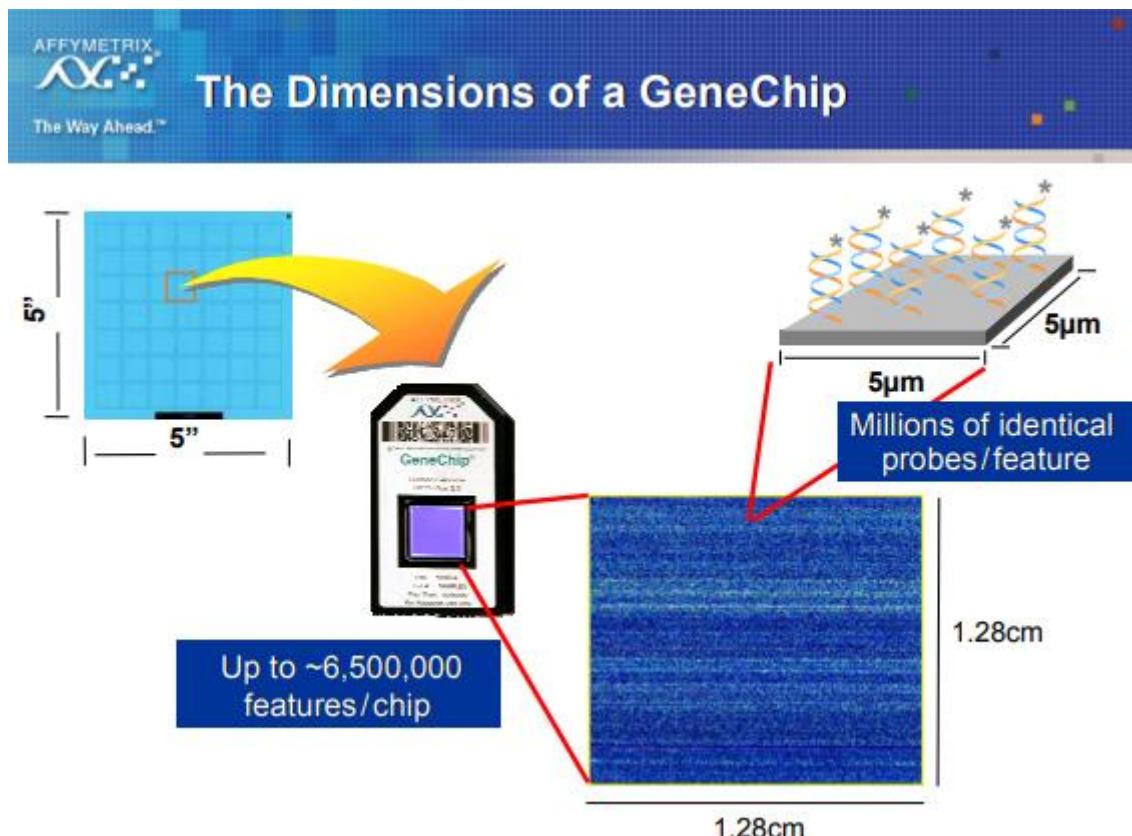


Figure 4: dimension d'une puce affymetrix, source [2]

*Sur ce schéma, on a représenté 49 plaquettes en verre de 5 pouces x 5 pouces. C'est sur ces plaquettes que les fragments d'ADN sont déposés. Chaque plaquette constituera une puce. Après la phase d'hybridation, la plaquette est transférée dans la puce affymetrix (GeneChip) où l'intensité de chaque probe sera lue pour produire une image. Une puce standard mesure environ 1.28 cm x 1.28 cm et peut contenir plus de 6 millions de sections. Chaque section peut contenir près d'un million de probes identiques.*

Les données des puces affymetrix sont des fichiers binaires au format .CEL. Il existe des programmes permettant de lire ces fichiers et les transformer en une matrice d'expression lisible pour l'être humain comme un fichier Excel.

Au final on se retrouve avec une matrice dite "matrice d'expression" comme le montre la figure suivante.

	sample1	sample2	sample3	sample4
1007_s_at	8.575758	8.915618	9.150667	8.967870
1053_at	6.959002	7.039825	6.898245	7.136316
117_at	7.738714	7.618013	7.499127	7.610726
121_at	10.114529	10.018231	10.003332	9.809068
1255_g_at	5.056204	4.759066	4.629297	4.673458
1294_at	8.009337	7.980694	8.343183	8.025335
1316_at	6.899290	7.045843	6.976185	7.063050
1320_at	7.218898	7.600437	7.433031	7.201984
1405_i_at	6.861933	6.042179	6.165090	6.200671
1431_at	5.073265	5.114023	5.159933	5.063821
...				

Figure 5: matrice M d'expression

Sur cette matrice M, les lignes représentent les gènes (probes) et les colonnes représentent les échantillons, c'est-à-dire les puces à ADN. Une valeur  $M(i,j)$  représente le niveau d'expression du  $i$ -ème gène dans le  $j$ -ème échantillon.

**Remarque :** La technologie des puces à ADN (microarrays) n'est pas l'unique technologie utilisée pour mesurer l'expression des gènes et donc de générer grande quantité de données. Il existe cependant d'autres technologies telles que les qPCR (quantitative real-time polymerase chain reaction) et RNA-Sequencing. Le lecteur intéressé peut consulter la référence [31].

### 2.3. Voies de signalisation et bases de données des voies de signalisation

#### a) Voie de signalisation (Signaling pathway)

Par définition, une voie de signalisation (signaling pathway) décrit un groupe de molécules (gènes) dans une cellule qui travaillent ensemble pour contrôler une ou plusieurs fonctions cellulaires, telles que la division cellulaire ou la mort cellulaire, l'activation ou la désactivation de certains gènes. Lorsque la première molécule d'une voie de signalisation reçoit un signal, elle active une autre. Ce processus est répété jusqu'à ce que la dernière molécule soit activée et la fonction cellulaire est réalisée [3]. Dans l'objectif de faciliter l'accès à ces informations pour la recherche médicale, des institutions publiques et privées mettent en place des bases de données dédiées aux voies de signalisation.

**NB : Sauf mention contraire, dans la suite de ce document "voie de signalisation", "signaling pathway" et "gene set" sont employés pour signifier la même chose.**

### **b) Base de données des voies de signalisation**

Les bases de données de voies de signalisation (signaling pathways databases) sont conçues pour stocker les informations relatives aux voies de signalisation. Une voie de signalisation est caractérisée par un nom et les gènes qu'elle contient. Parmi les institutions les plus connues offrant un accès à ces données on cite :

- MSigDB (Molecular Signature Database)
- KEEG (Kyoto Encyclopedia of Gene and Genomes)
- Reactome
- GO (Gene Ontology)
- WikiPathways

Etc.

L'accès aux voies de signalisation de ces bases de données peut se faire via une interface web directement sur le site de la base de données ou via une API (Application Programming Interface) qui permet de travailler directement avec les voies de signalisation d'un organisme.

## **3. Présentation du sujet**

Le sujet faisant l'objectif de ce stage s'intitule « **Analyse des données génomique à haut débit** ». Dans cette section nous présentons d'une manière générale l'objectif du sujet.

### **Contexte**

Etant un acteur de la biotechnologie, **IndaTamin®** a toujours à sa disposition des données issues des expérimentations internes ou externes. **IndaTamin®** se donne la mission, parmi tant d'autres, d'analyser ces données pour répondre aux questions soulevées par les équipes de recherche, les accompagner dans leurs décisions et la publication des articles. C'est dans ce contexte que vient mon rôle en tant que "**data analyst**".

### **Projets et objectifs**

Notre stage faisait l'objet d'une succession de projets à réaliser. Dans cette section nous présentons les différents projets réalisés durant notre stage.

#### **Projet 1 : Identification des gènes impliqués dans le diabète de type 2**

Des échantillons ont été prélevés chez deux groupes d'individus, normaux et diabétiques. Le niveau d'expression des gènes présents dans chaque échantillon a été mesuré via la technologie des puces affymetrix. L'objectif est de découvrir les gènes impliqués dans la maladie. En effet ce même travail a été déjà réalisé par une équipe de chercheurs et ils ont rédigé un article sur leur méthodologie et les résultats obtenus [6]. Il s'agit donc de reprendre

ce même travail et comparer les deux résultats. Ce projet fait l'objet du **chapitre 3** de ce document.

**Projet 2 : Etude de l'effet de la molécule de Ganoderma sur des cellules cancéreuses.**

Le Ganoderma est une molécule extraite d'une plante. La molécule est testée in vitro sur une lignée cellulaire provenant d'un organe diagnostiqué du cancer. Après traitement, le niveau d'expression des gènes dans cette lignée cellulaire est mesuré par rapport à des gènes de référence (non traités). Cette opération est réalisée plusieurs fois en variant à chaque fois la quantité de la molécule et le nombre d'heures de l'expérience. A partir des données issues de cette expérience, nous voulons savoir l'effet de la molécule sur la lignée cellulaire et en particulier les gènes sensibles à la molécule. Ce projet fait l'objet du **chapitre 4** de ce document.

**Projet 3 : Mise en place d'un modèle prédictif pour la prédiction de l'échec tardif du greffage rénal.**

La transplantation est l'opération qui consiste à transplanter chez un être vivant un organe provenant d'un individu de la même espèce. La transplantation rénale en est l'exemple. Malheureusement, pour des raisons beaucoup plus complexes, la transplantation peut échouer un peu plus tard et causer des complications sanitaires chez le patient. Connaitre les patients à risque permettra de mieux les surveiller et d'envisager une solution médicale. C'est l'objectif de ce projet. A partir du profil d'expression des gènes du patient, nous voulons être capables de décider si celui-ci connaît un échec tardif ou non. Les méthodes d'apprentissage supervisé porteront une réponse à cette problématique. Ce projet fait l'objet du **chapitre 5** de ce document.

## 4. Environnement de travail

Durant ce stage nous avons travaillé sur une station Linux de processeur Intel (53.5 GHz 4 cores) et une RAM de 16 GB avec les différentes outils et technologies suivantes :

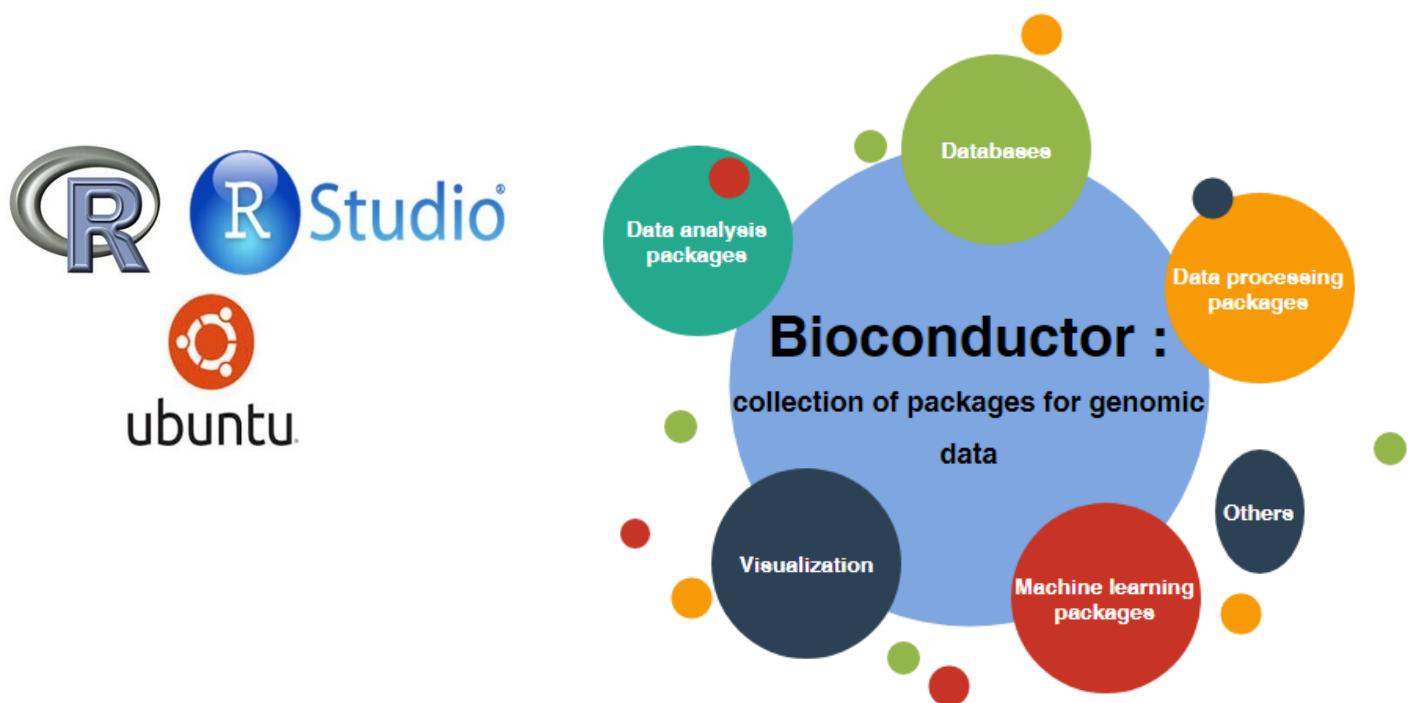
**R et RStudio :**

- Le R est un langage de programmation multi-paradigme, multiplateforme et libre destiné à la manipulation des données. Il a été créé en 1996 par Ross Ihaka et Robert Gentleman qui se sont inspiré du langage S (pour statistique). R est aussi un langage de script, interprété et non typé. Il a été conçu pour faciliter l'exploration de données et la visualisation. Il regroupe plusieurs libraires dédiées au data mining, à l'apprentissage automatique, text mining, web scraping, visualisation, etc. Il est l'un des langages les plus utilisés dans le domaine de la science de données. R est soutenu par la **Fondation for Statistical Computing** et bénéficie d'une communauté active qui continue de l'améliorer en proposant des nouveaux packages.

- Le RStudio est un environnement de développement open source et une multiplateforme pour le langage R. Il fournit une interface interactive qui facilite l'utilisateur à effectuer différentes opérations sur les données et les graphes. Une partie de RStudio est écrite en C++, Java, Javascript et son interface graphique utilise l'interface de programmation Qt de C++.

### Bioconductor :

Bioconductor est un projet pour le développement et la diffusion d'outils d'analyse statistiques et graphique en langage R (modules ou packages). Ces outils sont destinés à l'analyse de données issues des expériences biologiques en particulier les données générées par des méthodes d'analyse haut débit. Bioconductor se présente donc comme une collection de packages R pour l'analyse des données des expériences biologiques.



*Figure 6: Environnement de travail.*

# Chapitre II : Processus d'analyse des données à haut débit

## Introduction

L'objectif final des expériences des puces à ADN est l'exploration des données y résultant pour une meilleure compréhension du mécanisme de régulation de l'expression des gènes. Dans ce but, les expériences sont réalisées pour découvrir des nouvelles connaissances. A partir de ces données, les biologistes s'intéressent principalement aux gènes différentiellement exprimés (GDE) entre les échantillons étudiés ou les traitements testés, les gènes ou les échantillons ayant des profils similaires. Une fois identifiés, les biomarqueurs (gènes) associés à un phénotype observé (une maladie par exemple) chez un groupe d'individus peuvent être utilisés pour prédire l'état des nouveaux individus en se basant sur le niveau d'expression de ces gènes. Dans ce contexte, les méthodes d'analyse et de fouille des données restent indispensables. Cependant, l'analyse des données des puces à ADN est soumise à de multiples contraintes liées à la nature même de ces données. Le bruit de fond lié aux différentes technologies de mesure d'expression et la grande dimension de ces données doivent être pris en considération avant de réaliser une quelconque analyse.

Dans ce chapitre nous présentons le processus d'analyse de données à haut débit, allant du prétraitement aux méthodes d'analyse statistique et data mining.

### 1. Prétraitement des données des puces à ADN

Les données issues des technologies de puces à ADN sont considérées comme des données bruitées. Ce bruit de fond peut provenir des instruments utilisés lors de l'expérience. Par conséquent elles doivent être nettoyées et normalisées avant de réaliser une quelconque analyse. Cela permet de garantir une meilleure qualité de données pour l'analyse.

#### 1.1. Elimination du bruit de fond

L'intensité observée de chaque gène est composée d'un signal et d'un bruit de fond. Ce bruit de fond peut provenir de l'hybridation, de bruit dans l'instrumentation, du scanner, etc. L'objectif des méthodes de nettoyage de bruit de fond est donc d'enlever ce bruit pour ne garder que le vrai signal.

##### a) Méthode RMA (*Robust Micro-Array Average*)

Cette méthode modélise l'intensité observée comme  $O=S+N$ ,  $S$  et  $N$  étant le signal et le bruit (noise) respectivement, sous l'hypothèse que  $S$  et  $N$  suivent une distribution exponentielle  $Exp(\alpha)$  et une distribution normale  $N(\mu, \sigma^2)$ , respectivement [4,8].

RMA essaie de faire une estimation de la valeur de  $S$  connaissant une valeur  $O$  donnée (intensité observée).

$$E(S|O = o) = a + \sigma \frac{\phi\left(\frac{a}{\sigma}\right) + \phi\left(\frac{o-a}{\sigma}\right)}{\Phi\left(\frac{a}{\sigma}\right) + \Phi\left(\frac{o-a}{\sigma}\right) - 1}$$

Où :

- $a = o - \mu - \sigma^2 \alpha$
- $\Phi$  la fonction de répartition de la loi normale :  $\Phi(z) = \int_{-\infty}^z \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}w^2\right) dw$
- $\phi$  la densité de la loi normale centrée réduite :  $\phi(z) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}z^2\right)$

### b) Méthode MAS 5.0 :

C'est l'algorithme proposé par Affymetrix pour éliminer du bruit de fond [28]. Pour estimer le niveau du bruit de fond, la méthode **MAS 5.0** (MicroArrays Suit version 5) procède selon les étapes suivantes :

- 1) La puce est divisée en  $K$  différentes zones (16 par défaut).
- 2) Pour chaque zone  $i$ ,  $i \in 1, \dots, K$  il calcule le fond (background) et le bruit (noise) :
  - i) La moyenne de 2 % des valeurs d'intensités les plus basses est considérée comme fond ( $b_k$ )
  - ii) L'écart-type de 2 % des valeurs d'intensités les plus basses est choisis comme le bruit ( $n_k$ )
- 3) Enfin, l'intensité ajustée du fond est donnée par:

$I_{New}(x,y)=\text{Max } \{I(x,y)-b(x,y), 0.5*n(x,y)\}$ , avec  $I(x,y)$  l'ancienne valeur,  $b(x,y)$  et  $n(x,y)$  une moyenne pondérée des  $b_k$  et  $n_k$  [28].

**Note :  $x$  et  $y$  désignent l'emplacement géométrique du probe dans la puce.**

## 1.2. Normalisation des données

La normalisation consiste à rendre toutes les puces dans une échelle comparable en éliminant les différentes variations non biologiques qui peuvent exister entre les puces. Par exemple les variations causées par les paramètres du scanner, etc. Pour cette raison, nous avons besoin d'une méthode pour ramener toutes les puces à une échelle similaire.

### a) Normalisation par la méthode quantiles :

La normalisation par quantile est une méthode de normalisation qui rend identiques les distributions d'intensités de toutes les puces en remplaçant les valeurs de chaque ligne par la moyenne des valeurs les plus élevées des toutes colonnes. Ci-dessous nous présentons un pseudo-code et un exemple qui illustrent l'algorithme de normalisation par la méthode quantile.

## Pseudo code de la méthode quantile:

**Input :**  $M_{n,p}$  #Une matrice de  $n$  ligne et  $p$  colonnes.

**Output :**  $M_{n,p}$  #Une matrice de  $n$  ligne et  $p$  colonnes normalisée.

**Begin :**

**For**  $j \leftarrow 1 : p$  **do**  
 $M[j] \leftarrow \text{sort}(M[j], \text{decreasing}=\text{TRUE})$  # tri ascendant de tous les éléments de la colonne  $j$ .  
**End for**

**For**  $i \leftarrow 1 : n$  **do**  
 $\text{Moy}[i] \leftarrow \text{Mean}(M[i, :])$  # moyenne de tous les éléments de la ligne  $i$ .

$M[i, :] \leftarrow \text{Moy}[i]$  # Remplacer tous les éléments de la ligne  $i$  par sa moyenne.  
**End for**

Reorder each column  $M[, j]$  to its original order.

**End**

**Exemple:**

Nous donnons un exemple avec la matrice d'expression  $M = \begin{bmatrix} 2 & 7 & 4 \\ 5 & 1 & 6 \\ 3 & 2 & 1 \end{bmatrix}$

$$\begin{bmatrix} 2 & 7 & 4 \\ 5 & 1 & 6 \\ 3 & 2 & 1 \end{bmatrix} \xrightarrow{\text{sort columns}} \begin{bmatrix} 5 & 7 & 6 \\ 3 & 2 & 4 \\ 2 & 1 & 1 \end{bmatrix} \xrightarrow{\text{calcul rows means}} \begin{bmatrix} 6 \\ 3 \\ 1.33 \end{bmatrix} \xrightarrow{\text{Replace each value by row mean}} \begin{bmatrix} 6 & 6 & 6 \\ 3 & 3 & 3 \\ 1.33 & 1.33 & 1.33 \end{bmatrix} \xrightarrow{\text{Reorder columns to initial order}} \begin{bmatrix} 1.33 & 6 & 6 \\ 6 & 1.33 & 3 \\ 3 & 3 & 1.33 \end{bmatrix}$$

On voit qu'au final toutes les colonnes (qui représentent les puces) ont les mêmes mesures statistiques.

### b) Normalisation par la méthode Scaling :

Le principe de cette méthode est de transformer les données de chaque puce de sorte que toutes les puces aient les mêmes mesures statistiques : moyenne, médiane, etc. [8,28]

La valeur de l'intensité de probe dans chaque puce est multipliée par un facteur  $f_j$  différent pour chaque puce  $j$ . La valeur de  $f_j$  est donnée par :

$$f_j = \frac{S_c}{\text{TrimMean}(S_j, 0.2, 0.98)}$$

Par défaut, la valeur de  $S_c$  est de 500.  $\text{TrimMean}(S_j, 0.2, 0.98)$  est la moyenne des valeurs dans la puce  $j$  après une troncature de 2% pour les intensités les basses et les plus élevées.

**Rappel :** Pour un tableau trié, la moyenne tronquée (ajustée) à 2% est la moyenne obtenue après suppression de 2% des valeurs au début et de 2% à la fin du tableau.

### 1.3. Quantification des données

Dans les microarrays, nous avons de nombreuses sondes (*probes*) mesurant l'expression de chaque gène. La quantification (summarization) consiste à résumer en une seule valeur les niveaux d'intensité des sondes mesurées. Les méthodes **medianpolis** et **Tukey** sont plus utilisées [8].

### 1.4. Contrôle de qualité des données

Le contrôle de qualité a pour objectif de s'assurer que les données sont suffisamment bien nettoyées et normalisées pour l'analyse. Ce contrôle est fait via des outils de visualisation tels que les box plots, histogrammes, nuages des points, etc.

### 1.5. Filtrage

Les données nettoyées et normalisées sont prêtes pour une analyse. Mais vue la voluminosité des données, il est beaucoup plus recommandé de procéder à un filtrage pour éliminer les gènes non significatifs. Le filtrage consiste à éliminer les gènes ne présentant pas une variation significative du niveau d'expression selon une valeur seuil choisie. Exemple, les gènes dont leur *fold change* ou la différence de moyenne ou encore leur intervalle interquartile ne dépasse pas une valeur seuil.

## 2. Analyse différentielle d'expression de gènes

L'analyse différentielle d'expression de gènes a pour objectif d'analyser la variation de l'expression de gènes d'un échantillon par rapport à échantillon de référence pour identifier les gènes qui entrent en jeu du phénomène biologique.

L'idée principale est de vérifier pour chaque gène, s'il y a eu un changement significatif de son niveau d'expression. Pour cela, différentes méthodes statistiques sont utilisées notamment les tests d'hypothèses.

### 2.1. Notion de tests d'hypothèses et p-value

La mise en évidence des GDE conduit souvent à réaliser des tests statistiques pour évaluer s'il existe ou non une différence significative (effet biologique) du niveau d'expression des gènes entre deux ou plusieurs conditions. Deux hypothèses sont ainsi formulées avant l'analyse.

- L'hypothèse nulle ( $H_0$ ) selon laquelle il n'existe pas une différence significative du niveau d'expression d'un gène entre les deux conditions au risque  $\alpha$  de se tromper. Dans ce cas la différence observée est due au hasard ou liée aux erreurs de mesures.
- L'hypothèse alternative ( $H_1$ ) est qu'il existe une réelle différence non liée au hasard ou aux erreurs de mesures.

Un test d'hypothèse consiste à vérifier la validité de l'hypothèse nulle.

Pour mettre en évidence les gènes exprimés (GDE), on calcule une statistique qui quantifie la différence du niveau d'expression entre les deux conditions. Ensuite on calcule, sous  $H_0$ , **la probabilité  $p$  d'obtenir une valeur aussi extrême que la statistique observée**. Si cette probabilité, encore appelée  **$p$ -value** est inférieure à la valeur seuil  $\alpha$ , l'hypothèse nulle  $H_0$  est rejetée. Autrement dit la différence est significative.

L'une des difficultés sur les tests d'hypothèse est le nombre des faux positifs. C'est-à-dire les gènes classés comme ayant connus une différence significative du niveau d'expression alors qu'ils ne le sont pas en réalité (les faux positifs). Ce phénomène est observé lors des tests d'hypothèses multiples et le nombre des faux positifs augmente quand le nombre de tests réalisé est important. C'est le cas des données des puces à ADN où nous sommes amenés à réaliser plusieurs milliers de tests. Il est donc important de réduire le nombre des faux positifs afin de minimiser l'*erreur de type I*, c'est-à-dire rejeter à tort l'hypothèse nulle.

## Correction pour les tests multiples

Chaque gène de la matrice d'expression fait l'objet d'un test statistique avec un risque  $\alpha$  d'avoir des faux positifs. Or, pour chaque test réalisé le risque  $\alpha$  se multiplie et le nombre des faux positifs augmente. Une attention particulière doit être portée pour contrôler ce type d'erreur

### ***False Discovery Rate (FDR)***

L'approche FDR permet d'estimer la proportion  $q$  des erreurs de type I parmi l'ensemble des hypothèses rejetées. Par analogie avec la p-value, le résultat du contrôle FDR est parfois appelé *q-value*.

### **FDR avec la méthode de Benjamini–Hochberg**

Considérons un ensemble d'hypothèses nulles  $H_1, \dots, H_m$  avec leurs correspondantes p-values  $P_{(1)}, \dots, P_{(m)}$  rangées de plus petite p-value au plus grande. La méthode de Benjamini-Hochberg contrôle le FDR à une valeur de signification  $\alpha$  donnée (exemple  $\alpha=5\%$ ) de la façon suivante :

1. Pour une valeur  $\alpha$  donnée, trouver la plus grande valeur  $k$  telle que
$$P_{(k)} \leq \frac{k}{m} \alpha$$
2. Rejeter l'hypothèse nulle pour tout  $H_{(i)}$  pour  $i=1, \dots, k$

## 2.2. Méthodes statistiques pour l'analyse différentielle d'expression des gènes

Soit  $X_i$  et  $Y_i$  les moyennes de l'expression du gène  $i$  ( $i=1, \dots, n$ ) dans le groupe 1 et groupe 2, respectivement.

### Fold change :

Le fold change du gène  $i$  est défini par le ratio des deux moyennes :

$$FC = \frac{X_i}{Y_i}$$

Une valeur seuil est choisi (souvent 2-fold ou 1,5-fold) pour qualifier un gène comme étant différentiellement exprimé.

$$\begin{cases} \text{Si } FC > 2, & \text{le gène est considéré surexprimé.} \\ \text{Si } FC < 0.5, & \text{le gène est considéré sous-exprimé.} \end{cases}$$

Le fold change est sans doute la méthode la plus simple et la plus catégorique pour décider si un gène est différentiellement exprimé d'une condition à une autre.

### Le test de Student ordinaire :

Le test de Student ordinaire (t-test) est l'une des méthodes utilisées pour identifier les gènes différentiellement exprimés dans deux groupes en calculant une statistique  $t_i$ . Pour deux classes non appariées (notre cas),  $t_i$  est donnée par :

$$t_i = \frac{X_i - Y_i}{\sqrt{s^2(\frac{1}{n_1} + \frac{1}{n_2})}}, \quad n_1 \text{ et } n_2 \text{ sont les nombres d'individus dans le groupe 1 et 2,}$$

respectivement.

$$\text{Avec } S^2 = \frac{\sum_{i=1}^{n_1} (x_i - X_i)^2 + \sum_{i=1}^{n_2} (y_i - Y_i)^2}{n_1 + n_2 - 2}$$

En plus de la normalité des échantillons, le t-test exige une égalité des variances. Cette dernière condition est rarement satisfaite, ce qui peut causer des résultats erronés si on tente de l'appliquer sur des données ne respectant pas ces conditions.

### SAM (Significance Analysis of Microarrays):

SAM est une méthode alternative de t-test utilisée pour l'analyse différentielle. Pour pallier au problème du t-test sur l'égalité des variances, SAM implémente le t-test modifié. Pour un gène  $i$ , une statistique  $d_i$  est calculée:

$$d_i = \frac{Y_i - X_i}{s_i + s_0}$$

Avec  $s_i$  est l'écart-type et  $s_0$  une valeur constante. Pour évaluer la significativité de la statistique  $d_i$ , SAM effectue une permutation des labels (noms des colonnes) de la matrice d'expression. Pour chaque permutation, une nouvelle statistique  $d_p$  est calculée pour chaque gène. La somme des  $d_p$  pour l'ensemble des permutations est calculée puis divisée par le nombre de permutations. Cette valeur notée  $d_E$  (*Expected*) sera comparée à la valeur  $d$  observée, sans permutation [4], (page 57,58).

### **Classification ascendante hiérarchique (CAH) :**

En analyse des données génomique, la méthode de classification hiérarchique est la méthode de classification non supervisée la plus utilisée. Bien qu'elle soit différente des méthodes statistiques introduites ici, la CAH permet d'étudier les profils d'expression des gènes pour comprendre les groupes des gènes ayant un profil similaire du niveau d'expression ou les échantillons ayant un des profils similaires. Cet algorithme travaille avec une matrice de distance dérivée de la matrice de données d'expression. L'algorithme construit des groupes par agrégation successives des éléments les plus proches deux à deux pour former une hiérarchie de partitions des éléments. Au départ chaque élément (individu ou échantillon) est considéré comme un groupe de taille 1. Puis, à chaque étape les deux groupes les plus proches (selon leur distance) sont cherchés et fusionnés jusqu'à avoir un seul groupe. Ce qui donne la CAH un critère agglomérant.

## **3. Analyse d'enrichissement**

Les méthodes vues dans la section précédente permettent d'identifier les GDE individuellement. Malheureusement, ces méthodes s'avèrent insuffisantes pour révéler les variations modestes de l'expression des gènes pris individuellement, car la plus part des maladies impliquent généralement des groupes entiers de gènes. Plusieurs gènes sont liés à une fonction biologique (voie de signalisation) et c'est souvent le changement additif d'expression dans l'ensemble de ces gènes qui conduit à la différence d'expression relative à la maladie [6]. L'analyse d'enrichissement a été introduite pour la variation du niveau d'expression au sein d'un groupe de gènes à priori (gene set ou voie de signalisation). Ce type d'analyse résout le problème de variation modeste d'expression, souvent indétectable avec les méthodes classiques.

### **3.1. GSEA (Gene Sets Enrichment Analysis)**

GSEA (Gene Set Enrichment Analyse) est une méthode statistique destinée à l'analyse d'enrichissement des voies de signalisation ou "pathway" (groupes de gènes). GSEA analyse systématiquement un ensemble de gènes à priori qui ont été regroupés pour leur implication dans une même voie de signalisation et attribue un score dit d'enrichissement à cette dernière. Ce score représente le degré d'enrichissement d'une voie de signalisation.

Une voie de signalisation sur-enrichie ou sous-enrichie signifie respectivement une multiplication ou une baisse de nombre de gènes au sein de la voie de signalisation.

La mise en place de GSEA nécessite deux éléments :

- ❖ Une liste (L) de gènes (d'intérêt) avec pour chaque gène, une valeur métrique de différence du niveau d'expression entre les deux conditions (Exemple, *fold change*). Cette liste doit être ordonnée suivant la valeur métrique.
- ❖ Un ensemble (S) de voies de signalisation (Gene Sets), qui peuvent provenir de différentes bases de données de voies de signalisation.

En effet, pour un ensemble S de gènes (voie de signalisation) donné, la méthode repose sur deux hypothèses :

- ❖ L'hypothèse nulle est que les gènes de S ayant conduit à son enrichissement sont distribués aléatoirement sur la liste (L)
- ❖ L'hypothèse alternative est que ces gènes sont situés au début ou en fin de la liste. Autrement dit, leur niveau d'expression a significativement augmenté ou diminué. (On parle de co-expression ou corégulation avec le phénotype).

Cela conduit à calculer une statistique de significativité, le score d'enrichissement, qui n'est autre que la statistique de Kolmogorov-Smirnov [6,7].

Soit  $L = \{g_1, g_2, \dots, g_N\}$  la liste de N gènes arrangés en ordre décroissant suivant une métrique de différence d'expression,  $r(g_i) = r_i$  (Exemple le *fold change*).

Soit S un *gene set* (pathway), contenant  $N_H$  gènes, le score d'enrichissement de S est calculé ainsi :

**$ES(S) = 0$  # Initialisation**

*Pour j allant de 1 à N faire:*

$$ES(S) = \begin{cases} ES(S) + \frac{(|r_j|)^p}{\sum_i^N (|r_i|)^p} & \text{si } g_j \in S \\ ES(S) - \frac{1}{N-N_H} & \text{sinon} \end{cases}, \text{ avec } p \text{ un poids de pondération (par défaut } p=1).$$

*Fin pour*

La significativité de  $ES(S)$  est évaluée en calculant une p-value basée sur une permutation des gènes. Et puis le taux des faux positifs est calculé [7].

### 3.2. Exemple de calcul du score d'enrichissement pour une voie de signalisation S

Pour donner une idée de la méthode au lecteur, nous donnons un exemple assez basique sur le calcul du score d'enrichissement. Nous avons choisi des valeurs fictives et simples afin de faciliter les calculs. On note :

**$\Sigma$**  : Somme des fold changes (FC) de tous les gènes dans L (exemple, 100).

**$N$**  : Nombre de gènes dans la liste (L) (exemple, 1020).

**$N_H$**  : Nombre de gènes (parmi ceux de la liste (L)) qui sont dans S (exemple, 20).

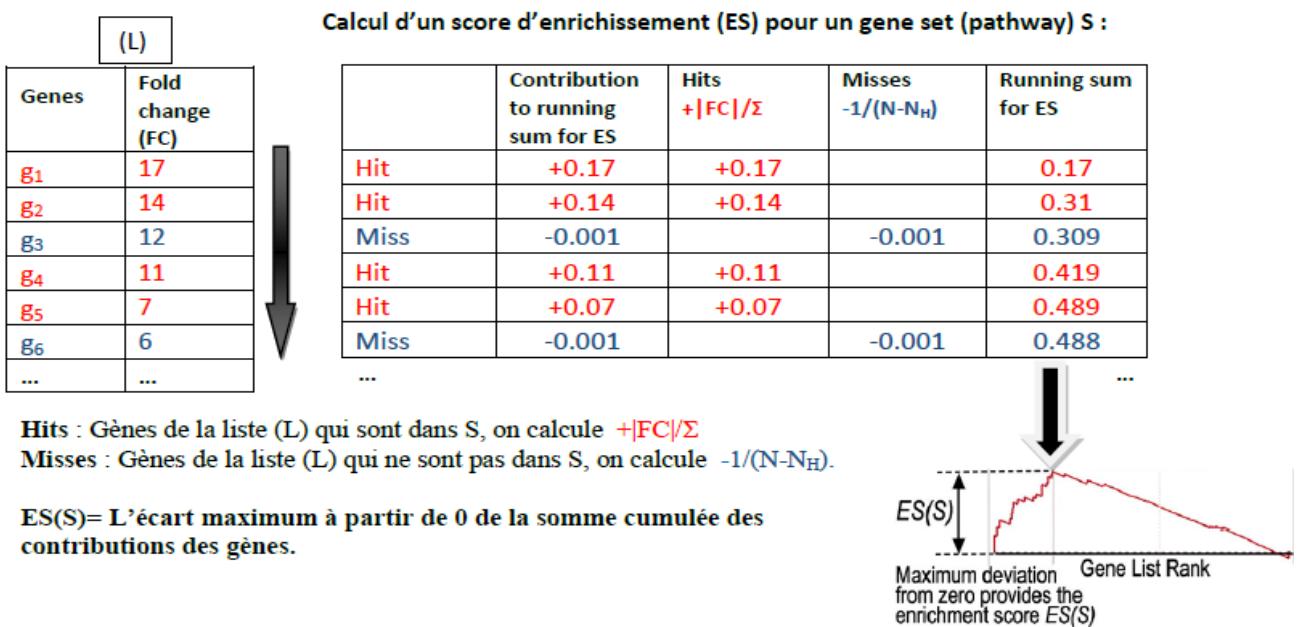


Figure 7: exemple de calcul du score d'enrichissement

Cette étape est répétée pour toutes les voies de signalisation pour avoir un score d'enrichissement. Puis vient l'étape de permutation des gènes pour déterminer la significativité du score. C'est-à-dire évaluer la p-value. Ensuite un score d'enrichissement normalisé est calculé qui servira au contrôle des faux positifs.

## 4. Apprentissage supervisé

L'apprentissage supervisé est la technique qui consiste à apprendre une fonction de prédiction à partir d'exemples du passé. Ces derniers sont des instances annotées et constituent la base d'apprentissage. La fonction de prédiction apprise est appelée "modèle". L'apprentissage supervisé est devenu un outil indispensable dans le domaine médical et en particulier en analyse des données à haut débit. Il est désormais possible de prédire l'apparition d'un phénomène biologique chez un patient rien qu'avec son profil du niveau d'expression de gènes.

Dans cette section, nous donnons quelques définitions des méthodes de classification utilisées durant nos projets.

### 4.1. Les machines à vecteurs supports(SVM) :

Les machines à vecteurs support ou séparateur à vaste marge est un ensemble de méthodes d'apprentissage automatique pour la classification et la régression introduit par Vapnik en 1998. Pour la classification binaire, l'objectif des SVM est de trouver un hyperplan de marge optimale, qui lorsque c'est possible, classe ou sépare au mieux les données en deux classes tout en étant le plus éloigné possible de toutes observations. Le principe est de trouver une fonction ou un classifieur dont la capacité de généralisation est la plus grande possible.

## 4.2. Le Random Forest (RF) :

Les forêts aléatoires ou forêts d'arbres décisionnels sont une méthode de classification supervisée qui prend son origine de la méthode d'arbres de décision combinée d'une technique appelée *bootstrapping*. Cette méthode consiste à générer aléatoirement plusieurs arbres de décision à partir des données d'apprentissage.

✚ Chaque arbre est construit en deux étapes :

- sur les observations, tirage avec remise d'un sous échantillon dans les données d'apprentissage, une technique connue sous le nom de bootstrapping.
- Sur les variables, tirage aléatoire d'un ensemble de variables parmi les variables prédictives.
- Construction d'un arbre de décision (version CART).

✚ Ce processus est répété autant de fois que le nombre d'arbres à créer.

✚ La classification est faite suite à un vote majoritaire des résultats de classification pour l'ensemble des arbres construits.

## 4.3. La méthode des k-plus proches voisins (k-pp) :

La méthode de k-pp pour *k-nearest neighbors* (*k-NN*) en anglais est une méthode d'apprentissage supervisé utilisée pour la classification et la régression. Pour estimer la classe d'un nouvel individu la méthode prend en compte les *k* individus dans l'échantillon d'apprentissage les plus proches du nouvel individu, selon une distance à définir. Contrairement aux autres méthodes qui construisent des règles d'apprentissage, le *k-NN* a la particularité d'être un "Lazy Learning" (apprentissage faible), ce qui signifie qu'il n'existe pratiquement pas une phase d'apprentissage. Pour prédire la valeur de sortie d'une nouvelle observation, la méthode *k-NN* se base sur le jeu de données entier.

## 4.4. PAM (Prediction Analysis of Microarrays):

Prediction Analysis Of Microarrays est une méthode de classification dédiée aux données des puces à ADN. La méthode PAM repose sur une combinaison de la méthode des plus proches centroides rétrécis pour "nearest shrunken cendroid" (NSC), lui permettant de sélectionner les gènes les plus importants pour la classification, et une probabilité à priori [30].

Soit une matrice d'expression composée de *n* lignes (les individus) et *p* colonnes (les gènes). Les individus sont répartis dans *K* classes.

Soit  $\mu_{ik}$  la moyenne du gène *i* pour les individus de la classe *k* et  $\mu_i$  la moyenne du gène *i* dans l'ensemble des individus.

Pour chaque gène *i*=1,..., *p*, PAM calcule une statistique définie par :

$$d_{ik} = \frac{\mu_{ik} - \mu_i}{w_i(s_i + s_0)} \quad (1)$$

avec  $w_i = (1/n_k - 1/n)^{1/2}$  ,  $s_i$  est l'écart-type et  $s_0$  est par défaut la médiane de  $s_i$ .

Ensuite PAM calcule à nouveau la valeur

$$d'_{ik} = \text{sign}(d_{ik})(|d_{ik}| - \Delta)_+ , (\Delta, \text{un paramètre de l'algorithme}).$$

Où le ‘+’ indique la partie positive (si( $|d_{ik}| - \Delta > 0$ , 0 sinon). Par conséquent toutes les valeurs  $d_{ik}$  inférieurs à  $\Delta$  sont rétrécies à zéro. (Si  $\Delta = 0$  , on utilise tous les gènes comme variables prédictives).

On peut redéfinir l'équation (1) :

$$\mu'_{ik} = w_i(s_i + s_0) * d'_{ik} + \mu_i$$

Cette nouvelle valeur représente le nouveau centroid rétréci du classe k.

Un nouvel individu ( $x_1, x_2, \dots, x_p$ ) est classifié en comparant son profil du niveau d'expression avec le nouveau centroid rétréci dans l'ensemble des  $q$  gènes restant.

$$\hat{y} = \arg \min_k \left\{ \sum_{i=1}^q \left( \frac{x_i - \mu'_{ik}}{s_i + s_0} \right)^2 - 2\log(\pi_k) \right\} ,$$

où  $\pi_k$  représente la probabilité que le nouveau individu à classifier soit de la classe k ( $\pi_k = n_k/n$ ) [29], (page 13,14).

## Résumé

Vue la nature complexe des données des puces à ADN, une première phase de prétraitement est requise pour garantir une meilleure qualité des données. Ensuite et selon les besoins, différentes analyses peuvent être réalisées :

- L'analyse différentielle d'expression des gènes : utilisation des méthodes statistiques telles que t-test, SAM, ou le fold change pour comparer la variation individuelle du niveau d'expression des gènes entre deux conditions.
- L'analyse d'enrichissement (GSEA) : analyse la variation du niveau d'expression au sein d'un groupe de gènes à priori. Elle est une solution pour les petites variations de l'expression de gènes et permet d'identifier un groupe de gènes co-exprimés et liés à une fonction biologique.
- L'analyse prédictive (classification supervisée) : utilisation des différents classifieurs pour classifier ou prédire des éventuels phénomènes biologiques à partir du profil d'expression des gènes.

## Chapitre III :

# Identification des gènes impliqués dans le diabète de type 2 chez un groupe d'individus

## Contexte

" PGC-1 $\alpha$ -responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes" [6] est un article apparu dans le journal **Nature genetics** en 2003. Dans cet article l'auteur a voulu montrer l'efficacité de la méthode GSEA par rapport aux méthodes standards d'analyse différentielle d'expression de gènes à travers des données expérimentales.

Le but de ce travail est de suivre et reproduire la démarche utilisée dans l'article et à la fin comparer les résultats.

### 1. Données

Les données sur lesquelles nous avons travaillé est un jeu de données constitué de 35 biopsies (échantillons) contenues dans des puces affymetrix dont chacune contient les valeurs d'intensités des 22283 probesets (gènes) pour un individu. Ce jeu de données est composé de trois groupes : 17 puces pour les individus classés comme *normal glucose tolerance diabète (NGT)* et 18 puces *diabète mellitus 2 (DM2)*. Chaque puce contient les valeurs d'expression de 22283 gènes mesurées. Ainsi notre jeu de données peut être vues comme une matrice de 22283 lignes et 35 colonnes.

### 2. Méthode

Tout comme dans l'article, l'enlèvement du bruit de fond puis la normalisation des puces affymetrix ont été réalisés en utilisant la méthode MAS5.0. Ensuite la méthode statistique SAM est utilisée pour identifier les GDE. En fin nous avons réalisé une analyse d'enrichissement en utilisant la méthode GSEA sur les *gene sets* de la base de données GO.

#### 2.1. Prétraitement des données

Toutes les données ont été normalisées avec la méthode MAS 5.0, ensuite nous avons filtré les données en éliminant les gènes ayant une variance très faible. Avec les méthodes et paramètres illustrés dans ce tableau, nous avons retenu 10984 gènes sur les 22283 gènes du départ. Ces gènes constituent notre jeu de données que nous analyserons par la suite.

Prétraitement des données	MAS5.0			Critère de filtrage
	Méthode pour l'enlèvement du bruit de fond	Méthode de normalisation	Méthode de sommarisation	
	MAS	Scaling	Tukey	IQR=0.144

Tableau 1: Résumé des méthodes utilisées pour le prétraitement des données des puces à ADN.

## 2.2. Contrôle de qualité des données

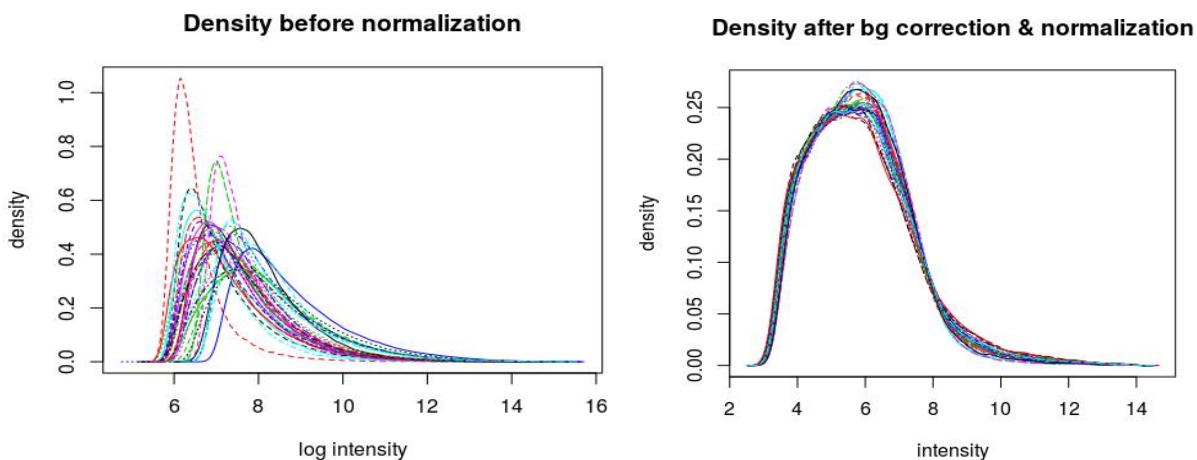


Figure 8: densité du niveau d'expression des 35 biopsies. (bg : background)

Sur la figure ci-dessus, l'image à gauche représente la densité du niveau d'expression des 35 microarrays avant le prétraitement. On voit bien qu'il y a une grande différence de leur distribution. Sur l'image à droite, après la suppression du bruit de fond et normalisation, les puces à ADN ont presque la même distribution.

## 3. Résultats

### 3.1. Analyse différentielle avec SAM

Nous avons utilisé la méthode SAM pour évaluer la variation du niveau d'expression des gènes entre le groupe d'individus normaux (NGT) et le groupe d'individus diabétiques (DM2). Nous avons utilisé la même valeur de paramètre utilisée dans l'article, à savoir  $\Delta=0.5$ . Les résultats ont montré 3 GDE avec un FDR de 24%. Cette valeur de FDR est jugée trop élevée pour considérer que les gènes sont bien difféntiellement exprimés.

#### Gènes significatifs selon l'analyse avec SAM

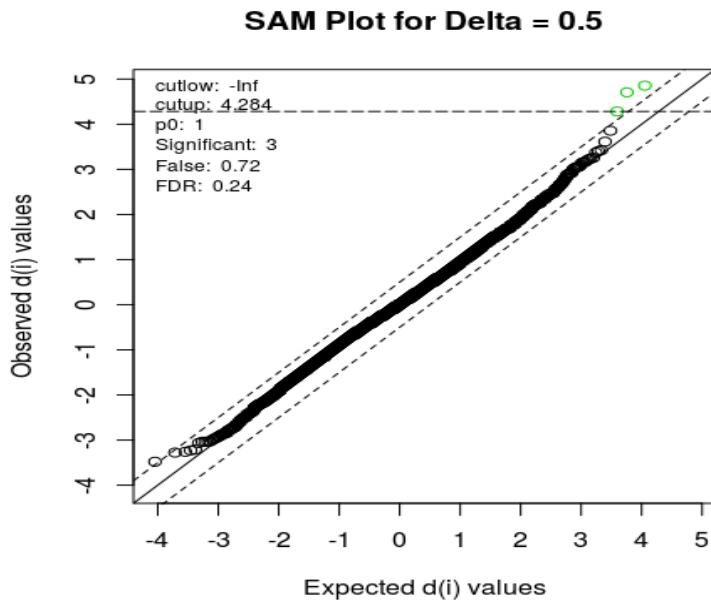


Figure 9: SAM plot pour les GDE entre les groupes NGT et DM2

#### Relation entre delta, gènes significatifs et FDR sur SAM

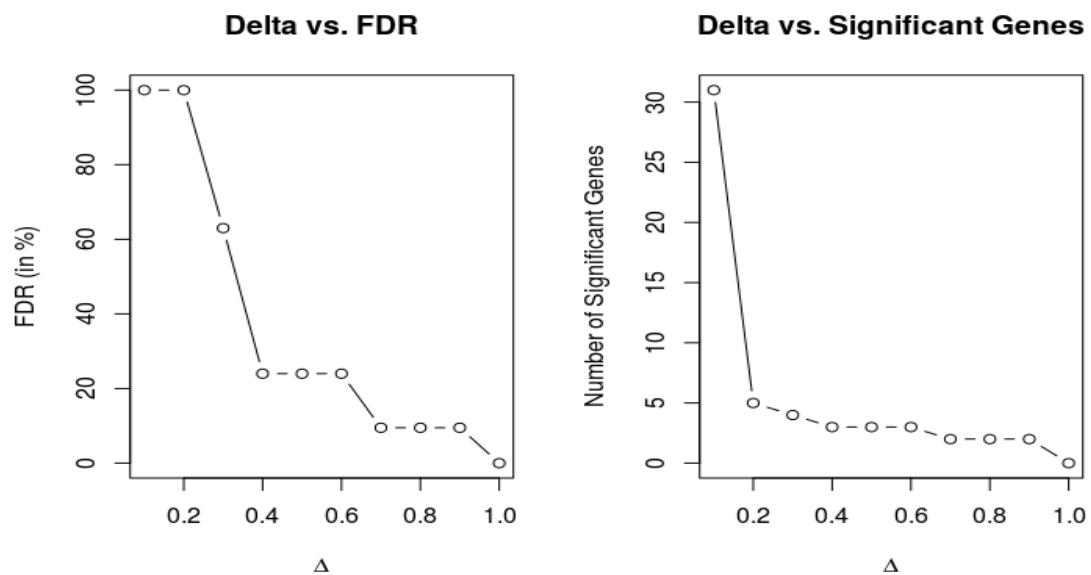


Figure 10: GDE et FDR en fonction du paramètre delta

Sur cette figure, on peut conclure deux choses :

- Plus la valeur de delta augmente, moins on a de gènes significatifs et des fausses découvertes.

- b) Vouloir augmenter le nombre de gènes significatifs en diminuant la valeur de delta, on augmente le FDR.

En fixant par exemple  $\delta=0.1$  nous aurons plus de 30 gènes significatifs mais avec un FDR de 100%. De même en fixant  $\delta=1$ , nous n'aurons aucun gène significatif avec un FDR de 0%. Ainsi, nous confirmons qu'avec la méthode SAM aucun gène n'a été identifié comme GDE comme il a été mentionné dans l'article de référence [6]. Cela est dû à la modeste variation d'expression des gènes entre les deux groupes d'individus, (figure suivante). D'où la nécessité d'utiliser une analyse d'enrichissement des groupes de gènes.

### 3.2. Différence entre les individus malades et les individus normaux

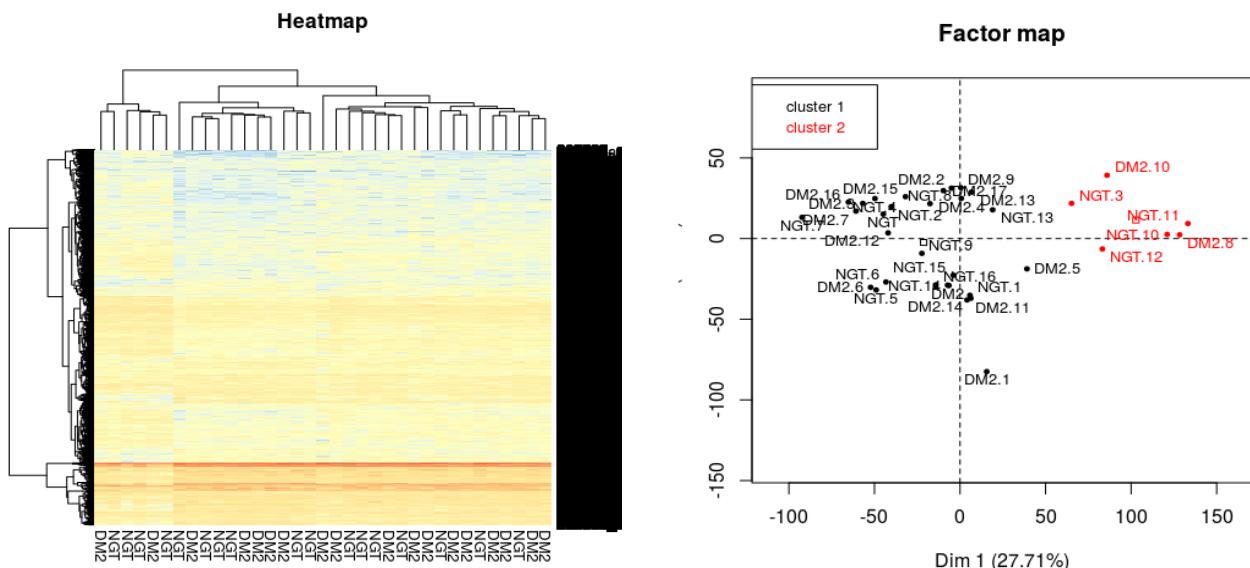


Figure 11: Différence du niveau d'expression entre les groupes NGT et DM2

L'image à gauche représente l'expression des gènes (en log2) dans les différents groupes. On voit qu'il est presque impossible de distinguer les NGT du DM2. On peut néanmoins voir une bande verticale (juste à gauche de l'image) formée de 4 NGT et 2 DM2. Sur l'image à droite, on a deux clusters. Ce sont en fait les mêmes clusters de l'image à gauche. Dans l'ensemble, les deux groupes sont loin d'être disjoints. Cela peut être dû à une faible variation du niveau d'expression entre les individus normaux et les malades.

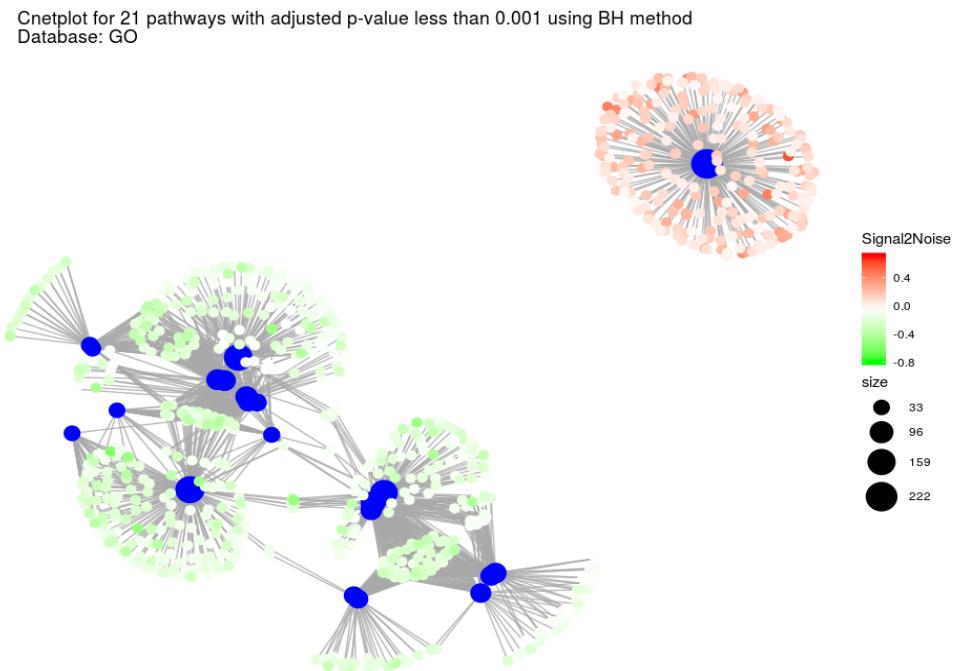
### 3.3. Analyse d'enrichissement

Nous avons utilisé la méthode GSEA sur ce même jeu de données avec 4436 « gene sets » provenant de la base de données des voies de signalisation GO (Gene Ontology). Nous avons utilisé le *signal to noise ratio* (Signal2Noise) comme métrique de différence d'expression entre les groupes NGT et DM2.

Au total vingt et une voies de signalisation sont enrichies avec une p-value ajustée  $<0.001$  que nous présentons ici.

#### Voies de signalisation enrichies :

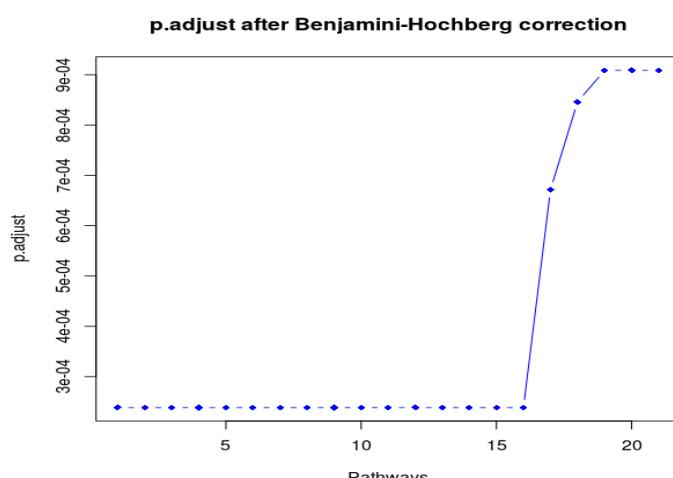
L'image ci-contre montre les 21 voies de signalisations régulées avec le diabète selon la méthode GSEA. Les points bleus représentent les *gene sets* (pathways) et les points verts et rouges sont les gènes. La couleur est donnée en fonction de la valeur du Signal2Nise. Un gène est relié à la voie de signalisation où il appartient. On voit un cluster formé des 20 *gene sets* sous-enrichies. Tous les gènes dans ces pathways ont un



Signal2Noise ration négatif.

Figure 12: Cenet plot des 21 gene sets

#### FDR des voies de signalisation :



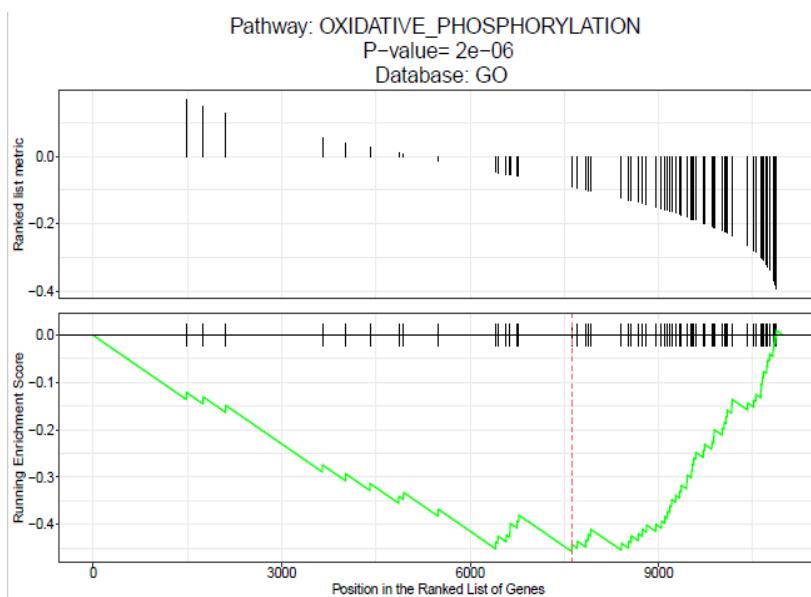
Cette image montre les p-values ajustées de 21 voies de signalisation enrichies. Chaque point représente une voie de signalisation.

Figure 13: FDR des 21 voies de signalisation enrichies

## **Sous-enrichissement de la voie de signalisation “Oxidative Phosphorylation”**

Comme ceux de l'article [6], nos résultats montrent que le pathway “**Oxidative phosphorylation**” est sous enrichie dans le groupe d'individus diagnostiqués du diabète de type 2. Cette voie de signalisation est liée à d'autres voies telles que “**Mitochondrial Respiratory**” et “**Electron transport**” qui sont dans les cinq tops pathways les plus significatives. Elles sont connues pour leur implication dans le processus de ce type de diabète [6].

### **Courbe du score d'enrichissement d'Oxidative phosphorylation**



*Figure 14: Courbe du score d'enrichissement d'Oxidative phosphorylation*

Sur cette image, les petits bars représentent les gènes dans les voies de signalisation. En haut de l'image, on voit le signal2Noise pour chaque gène. Les gènes situés après la ligne verticale rouge sont les plus significatifs pour GSEA. Ce sont les facteurs ayant contribué à l'enrichissement de la voie

bien qu'ils ont une valeur métrique (signal2Noise) très faible. La courbe verte est la courbe du score d'enrichissement. Le score d'enrichissement d'une voie est donné par la valeur minimale de cette courbe.

### **Facteurs impliqués à l'enrichissement d'Oxidative phosphorylation**

Cnetplot for top 3 pathways with adjusted p-value less than 0.001 using BH method  
Database: GO

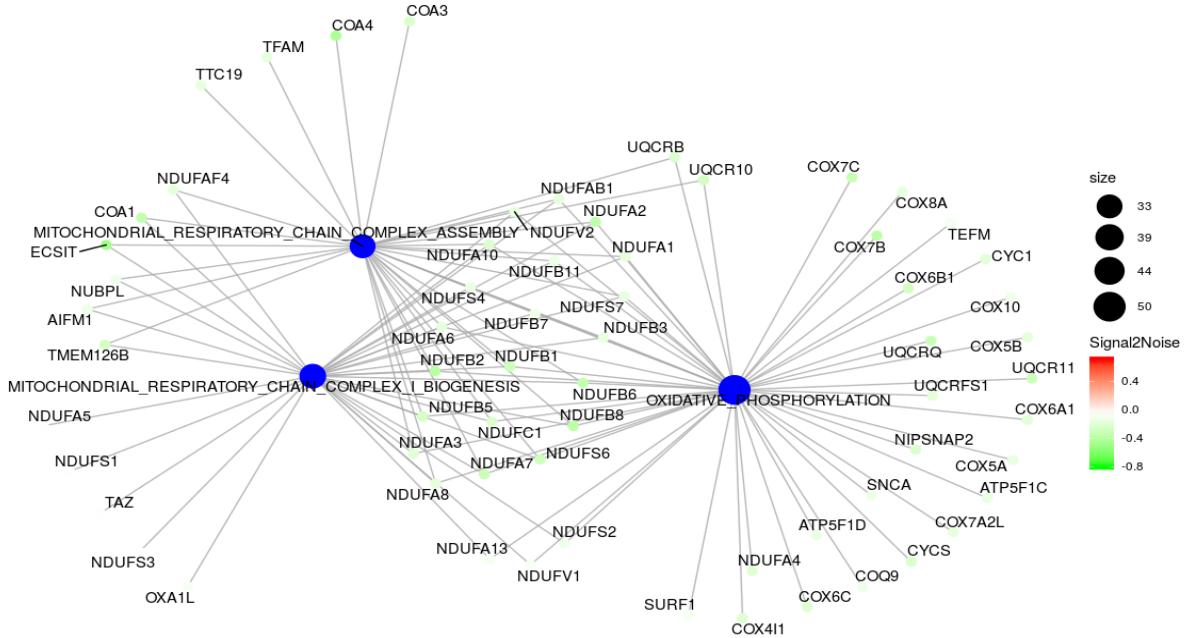


Figure 15: Gènes dans Oxidative phosphorylation

## Conclusion

L'objectif de ce projet est de mettre en évidence les gènes impliqués dans la maladie de diabète de type 2 en se référant à l'article qui a traité le même projet. Après une phase de prétraitement des données, on a utilisé la méthode SAM basée sur t-test modifié qui n'a pas pu identifier un seul gène significatif vue la modeste variation du niveau d'expression des gènes dans les deux groupes d'individus NGT et DM2. La méthode GSEA, qui est une méthode destinée à l'analyse des groupes de gènes, a révélé la sous-régulation de la voie de signalisation "Oxidative phosphorylation". Une voie de signalisation connue pour son rôle dans le diabète de type 2[6].

## Chapitre IV :

# Etude de l'effet de la molécule Ganoderma sur une lignée cellulaire

## Introduction

Ganoderma est une molécule extraite de la plante de champignon. Cette molécule est testée *in vitro* sur une lignée cellulaire humaine. L'expérience a été réalisée avec 1,5 µM (micro molaire) de Ganoderma. On a évalué le niveau d'expression de ces gènes au bout de 12h, 24h, puis 36h par rapport à des gènes de référence. La même expérience est répétée avec 3 µM. Pour chaque gène, la moyenne du niveau d'expression (normalisée par rapport à la moyenne des gènes de références) pour le traitement et le contrôle a été relevée.

L'objectif est d'analyser l'effet de la molécule sur cette lignée cellulaire et identifier les voies de signalisation (groupe de gènes) ciblées par cette molécule et plus en détails les facteurs impliqués.

Pour arriver à ce but, nous avons réalisé une analyse d'enrichissement des groupes de gènes pour évaluer les gènes co-exprimés en réponse au traitement par la molécule. D'abord nous avons réalisé une analyse sur les « *gene sets* » issus de différentes bases de données telles que Reactome, KEEG, WikiPathways, GO avant de mettre en place un workflow pour l'analyse d'enrichissement avec les « *gene sets* » de AnyGenes.

### 1. Données

Notre jeu de donnée est une matrice composée de 237 lignes qui représentent les gènes de l'expérience et 9 colonnes qui représentent le niveau d'expression des gènes. Trois colonnes pour les gènes contrôles (non traités), trois colonnes pour les gènes traités 1.5 µM après 12h, 24h et 36h et enfin trois autres colonnes après traitement avec 3 µM.

#### Notation des variables :

**CTR\_12h** : représente l'expression des gènes contrôles (non traités) 12 heures après l'expérience.

**V1.5\_12h** : représente l'expression des gènes traités avec une dose de 1,5 µM après 12 heures de l'expérience.

**V3\_36h** : représente l'expression des gènes traités avec une dose de 3 µM après 36 heures de l'expérience.

```

> mydata[1:20,]
      CTR_12H     V1.5_12H     V3_12H     CTR_24H     V1.5_24H     V3_24H     CTR_36H     V1.5_36H     V3_36H
TNFSF18  2.178066e-04 1.689476e-04 0.0010806333 0.0016852842 0.0009741049 0.0015212521 1.415180e-04 6.929129e-05 0.0010073115
IL1A     5.267507e-03 4.049416e-03 0.0034663854 0.0025093163 0.0023939085 0.0040392575 1.119735e-03 2.075024e-03 0.0063968350
PROK2    1.603840e-04 1.426345e-04 0.0010974675 0.0009791795 0.0006181362 0.0008106628 1.230985e-04 6.790773e-05 0.0007235706
BNIP3L   1.965504e-01 2.267288e-01 0.3402596253 0.1954033065 0.3840012470 0.5364280968 2.403765e-01 5.359242e-01 1.1249519530
BCL2L10  1.100317e-04 9.696827e-05 0.0005500661 0.0002096618 0.0003252371 0.0003409466 8.747114e-05 6.920716e-05 0.0003379080
CEPB2    3.081304e-01 3.738177e-01 0.5972546792 0.2029376908 0.2383207185 0.7472653975 2.445094e-01 2.535849e-01 0.7250048552
SERPINB2 2.921896e-04 3.140193e-04 0.0008135002 0.0008971168 0.0005570814 0.0007299757 2.245790e-04 3.462958e-04 0.0006789733
MCL1     9.125432e-01 1.037174e+00 1.2868120986 0.6746536684 0.9360776771 1.4219778169 6.528138e-01 8.587558e-01 1.8168566020
SOC52    6.921661e-02 7.707825e-02 0.1532934714 0.0602576463 0.0609893080 0.1810302908 6.274232e-02 4.950072e-02 0.1710940372
CBX4     1.520801e-01 1.763230e-01 0.2506676024 0.1242944436 0.1302486080 0.2933138659 1.546471e-01 1.669213e-01 0.4217235019
MPO      8.801648e-05 1.298574e-04 0.0007091600 0.0002643674 0.0002310748 0.0002858488 1.932564e-04 1.039354e-04 0.0004659206
FOXO1    6.677445e-03 6.951441e-03 0.0079832822 0.0058440030 0.0066283337 0.0103335012 6.233286e-03 6.326131e-03 0.0146970412
BCL2L2   3.659322e-02 3.479051e-02 0.0474946839 0.0217433433 0.0270835086 0.0484468336 2.569750e-02 4.527262e-02 0.0481378262
SPHK1    5.844058e-03 5.497273e-03 0.0081236356 0.0035205741 0.0047828319 0.0151526193 5.059033e-03 4.555103e-03 0.0078153399
HSPA1B   1.576598e+00 1.166493e+00 0.7122561345 1.3145288435 1.1428417254 1.1514806590 1.661499e+00 1.425437e+00 1.0343925211
ATF5     8.561243e-03 6.325871e-03 0.0053177484 0.0069784873 0.0052920664 0.0029173878 6.977939e-03 6.073649e-03 0.0044824165
BDNF    1.102044e-01 8.517815e-02 0.0770434572 0.1169689848 0.1034814627 0.0578296018 1.469124e-01 1.151526e-01 0.0841823934
SERPINB9 8.868133e-02 6.586308e-02 0.0554089624 0.0862131155 0.0620427284 0.0302093991 9.267836e-02 7.726470e-02 0.0441828085
BCL2A1   9.981031e-02 6.300689e-02 0.0587396092 0.0879374324 0.0547153888 0.0264180608 7.198779e-02 4.405290e-02 0.0318944212
HMGB1    4.290270e+00 3.969245e+00 3.4913076710 4.2500085297 3.7821746307 1.9149718525 4.536392e+00 3.810718e+00 1.9452767231
> |

```

Figure 16: extrait de la matrice de données expérimentales

En ligne nous avons les gènes utilisés lors de l'expérience et en colonne on a le label de l'expérience.

A partir de ces données nous voulons répondre à la question suivante : ***"Quels sont les groupes de gènes sensibles à la molécule de Ganoderma, pour quelle quantité et au bout de combien de temps ?"***.

## Analyse exploratoire des données expérimentales

### Hierarchical clustering on the factor map

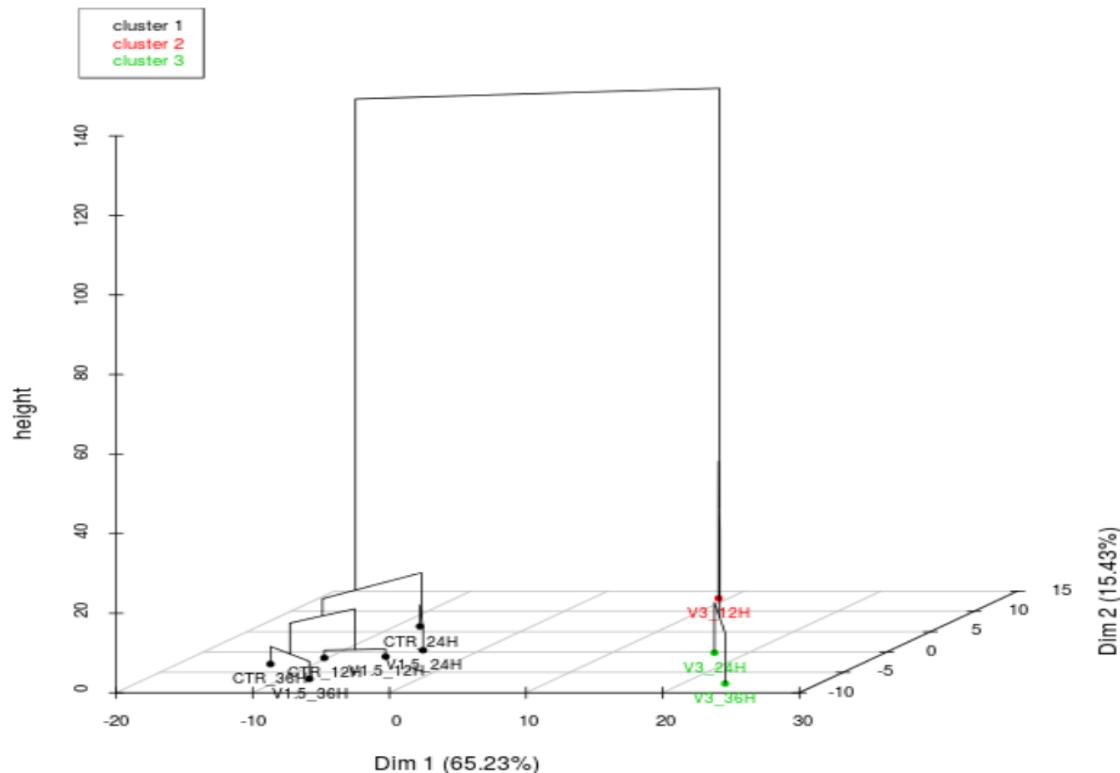


Figure 17: clustering hierarchique sur les données expérimentales

Sur cette image on voit qu'avec une quantité de 1.5  $\mu\text{M}$ , le niveau d'expression des gènes traités n'a pas connu une grande variation par rapport aux gènes de référence (les contrôles, notés CTR\_\_). Par contre avec 3  $\mu\text{M}$  les gènes ont des profils très similaires après 24H et 36H et s'éloignent des gènes contrôles. Cette première analyse nous a permis d'orienter et approfondir l'analyse sur les données issue de l'expérience avec 3 micros molaires.

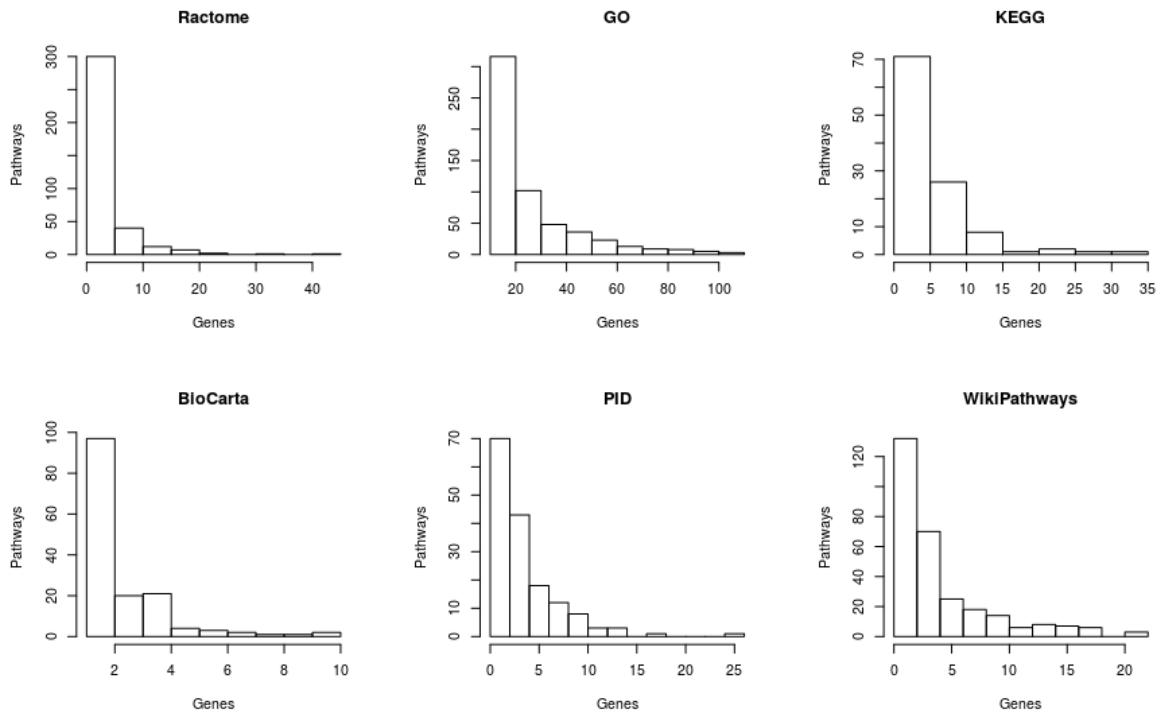
## 2. Méthode

Notre travail est réalisé en deux phases. La première phase consiste à réaliser une analyse d'enrichissement à l'aider des gènes sets (pathways) provenant des bases de données publiques. Après avoir validé les résultats de cette première phase, une deuxième phase consistait à mettre en place un workflow pour l'analyse d'enrichissement en utilisant les *gene sets* internes de la société AnyGenes.

### 2.1. Collection des *gene sets* dans les bases de données publiques

Nous avons collecté manuellement les ensembles de gènes (voies de signalisation) provenant de Reactom, WikiPathway, GO, BioCarta, PID et KEEG.

**Répartition des 237 gènes dans les différents *gene sets*.**



*Figure 18:proportion des gènes dans les gene sets*

Sur cette image, on voit que le nombre de gènes dans les gene sets dépend des bases de données. Dans Reactome on voit qu'il y a 300 gene set qui contiennent moins de 5 gènes de notre liste de gènes. Et il y a plus 300 qui contiennent plus 20 gènes dans GO.

#### a) Workflow d'analyse d'enrichissement avec les données de AnyGenes

Tout comme les autres sociétés travaillant dans le domaine de la génomique, la société **AnyGenes** dispose elle aussi de sa base de données de voies de signalisation. Au lieu de faire recours aux « *gene sets* » issus des bases de données externes, la société souhaite pouvoir réaliser une analyse d'enrichissement avec ses propres voies de signalisation au quotidien.

La première tâche est d'automatiser la récupération, le nettoyage des données non structurées. Puis les transformer en un format utilisable pour la méthode GSEA.

#### Automatisation du processus de traitement des données

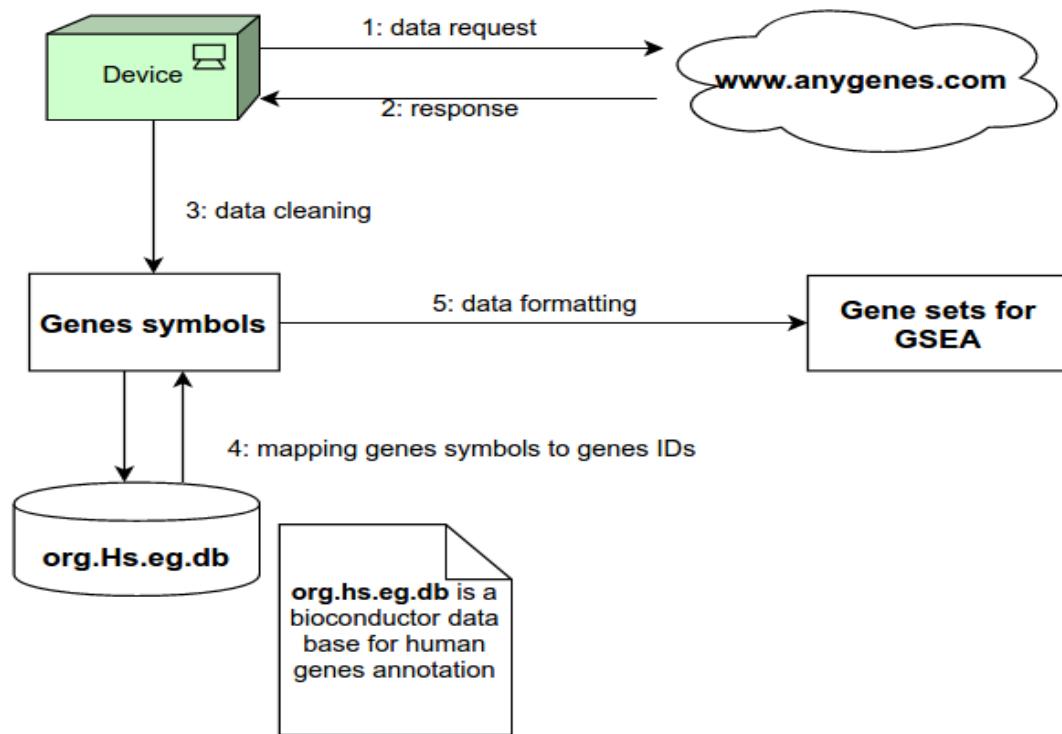


Figure 19: Processus de collection et traitement des données de AnyGenes.

## 2.2. Analyse d'enrichissement des gene sets

Pour chaque expérience nous avons calculé le *fold change* en logarithme à base 2 des gènes traités par rapport aux contrôles. Le *fold change* est ensuite trié en ordre décroissant. Les gènes les plus exprimés sont au début de la liste et les moins exprimés enfin de la liste. Ensuite, pour chaque expérience nous avons utilisé les *gene sets* des six bases de données pour réaliser une analyse d'enrichissement. Pour chaque *gene set*, un score d'enrichissement est calculé. Nous avons utilisé 10000 permutations pour calculer la p-value que nous avons ensuite corrigé avec la méthode de Benjamini-Hochberg.

## 3. Résultats

La méthode GSEA est appliquée sur les groupes de gènes provenant des différentes bases de données notamment Reactome, GO, BioCarta, KEEG, WikiPathways, PID et en fin AnyGenes.

### a) Voies de signalisation régulièrement significatives

Vue le nombre élevé de voies de signalisation enrichies, nous avons défini la notion de "*voie de signalisation régulière*" pour désigner une voie de signalisation ayant une p-value <0.05 au moins 3 fois (au moins 3 cas sur 6 expériences) pour la même base de données. Ainsi, on pourra identifier les groupes de gènes sensibles à la molécule qu'avec une quantité de 1.5 µM

ou 3  $\mu$ M. Pour raison de simplicité nous présentons ici uniquement les voies de signalisation provenant de la base de données **Reactome** qui sont régulièrement enrichies.

Base de données	Nombre de voies de signalisation enrichies
KEGG	7
Reactome	18
GO	66
WikiPathways	15
BioCarta	5
PID	2

Ce tableau regroupe les 113 voies de signalisation ayant un score d'enrichissement significatif au moins trois fois pour une p-value <0.05. Ces résultats sont obtenus en croisant tous les résultats des différentes études. C'est-à-dire avec 1.5  $\mu$ M et 3  $\mu$ M à 12h, 24h et 36h pour toutes les bases de données. Un grand nombre des voies de signalisation sont enrichies soit avec 1.5  $\mu$ M soit avec 3  $\mu$ M. Nous précisons que ces 113 voies sont comptées dans l'ensemble des bases de données. C'est-à-dire qu'une voie peut être comptée deux fois ou plus s'elle apparaît significative dans plusieurs bases de données. Nous avons identifié '**Cell Cycle**' et certaines voies dérivées de '**Cell Cycle**' qui ont répondu à ce critère pour toutes les bases de

**Tableau 2:** Ce tableau résume le nombre des voies de signalisation significativement enrichies au moins 3 fois (parmi six) avec une p-value<0.05 Au total nous avons noté 113 voies de signalisation répondant à ce critère dans l'ensemble des bases de données.

données. Nous rappelons que '**Cell Cycle**' n'est pas la seule à avoir répondu à ce critère mais plutôt celle qui a été régulière dans toutes les bases de données. Cela n'est pas toujours le cas pour les autres voies de signalisation comme '**Target Of Rapamycin (TOR) Signaling**' qui a satisfait ce critère dans WikiPathways seulement, ou encore '**Defense response**' qui n'existe que dans GO.

### Courbes du score d'enrichissement des voies de signalisation

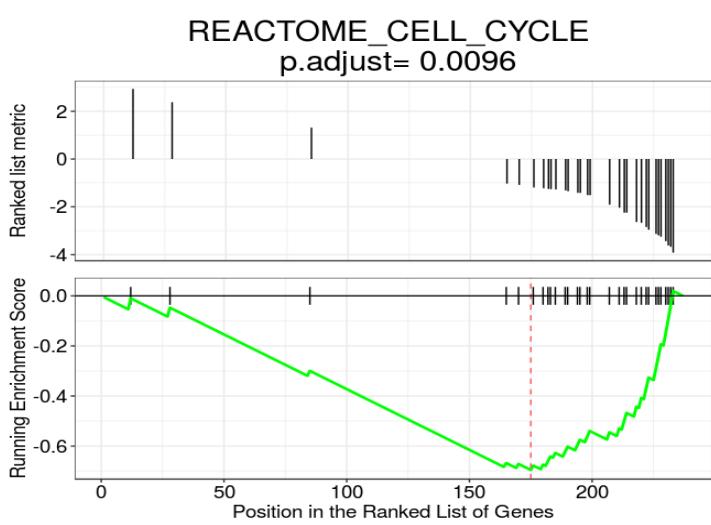


Figure 21:Courbe du score d'enrichissement du cycle cellulaire. Base de données : Reactome

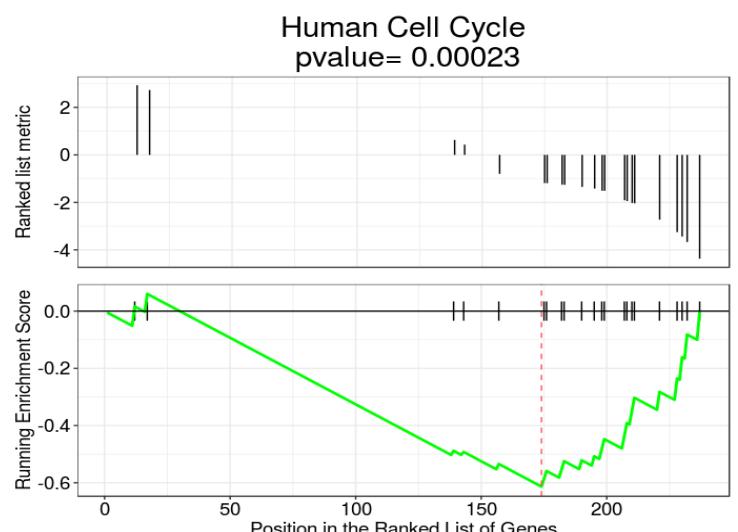


Figure 20:Courbe du score d'enrichissement du cycle cellulaire. Base de données : AnyGenes

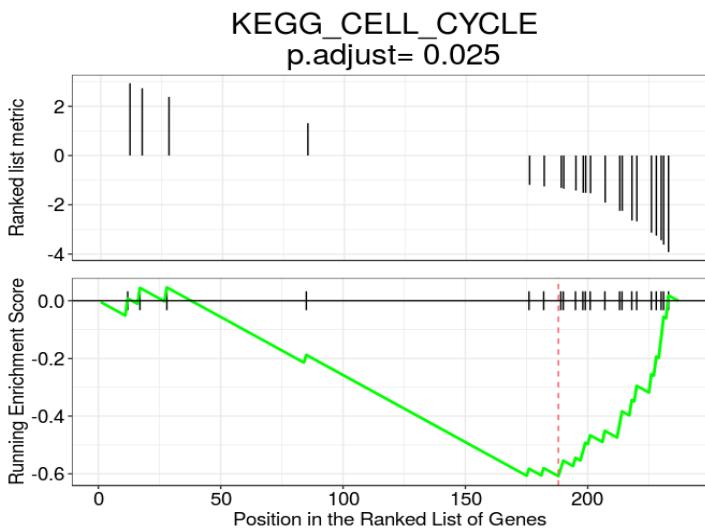


Figure 25: Courbe du score d'enrichissement du cycle cellulaire. Base de données : KEEG

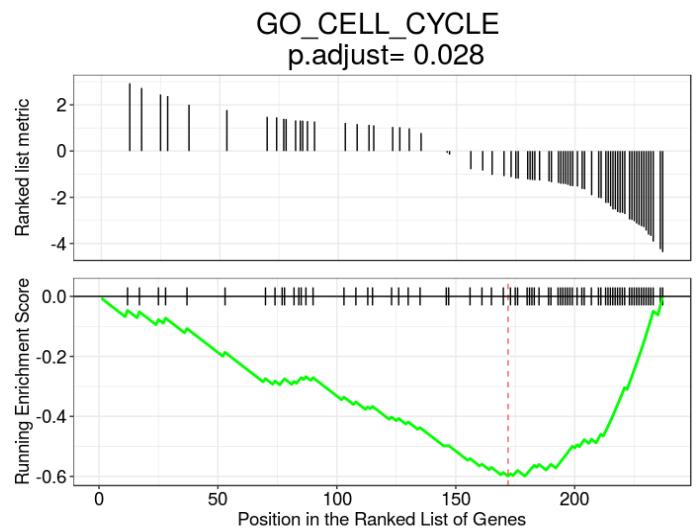


Figure 24: Courbe du score d'enrichissement du cycle cellulaire. Base de données : GO

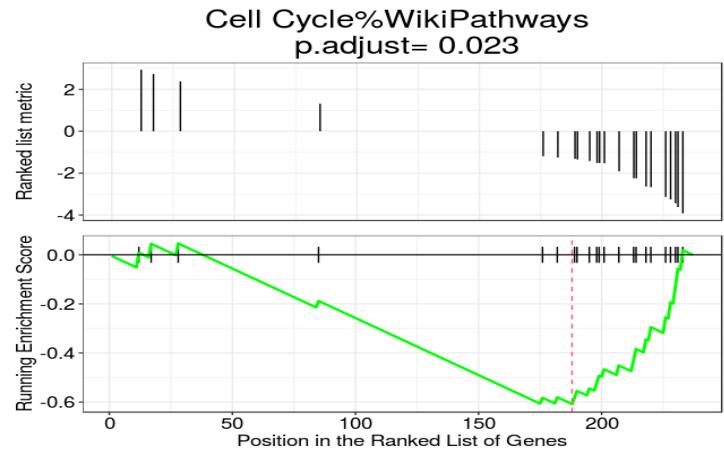


Figure 23: Courbe du score d'enrichissement du cycle cellulaire. Base de données : WikiPathways

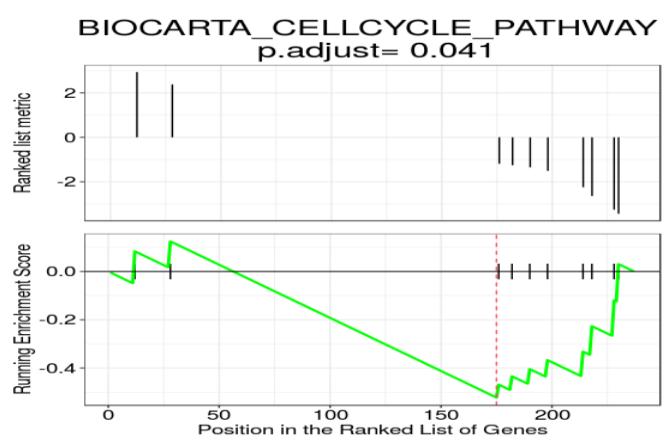


Figure 22: Courbe du score d'enrichissement du cycle cellulaire. Base de données : BioCarta

## Interprétation de la courbe du score d'enrichissement

Les courbes précédentes représentent les scores d'enrichissement pour les "voies de signalisation régulière" du cycle cellulaire issue de différentes base de données pour 3 µM à 36h. Sur ces images, les petites barres représentent les gènes dans les voies de signalisation. Le "Ranked List Metric" représente le *fold change* utilisé comme métrique.

Les gènes situés après la ligne verticale rouge sont les plus significatifs selon GSEA. Ce sont les facteurs ayant contribué à l'enrichissement de la voie. La courbe verte est celle du score d'enrichissement. Le score d'enrichissement d'une voie est donné par la valeur minimale de cette courbe. En effet la courbe prend une autre allure quand cette valeur est atteinte.

### b) Voies de signalisation régulièrement enrichies dans Reactome.

Dans cette partie nous présentons en détail les résultats de l'analyse d'enrichissement avec les *gene sets* de Reactome. Nous avons fait le choix sur Reactome car c'est l'une des bases de données de voies de signalisation de référence pour l'analyse d'enrichissement.

Normal Enrichment Score (NES) des pathways activés au moins 3 fois avec une p.value<0.05

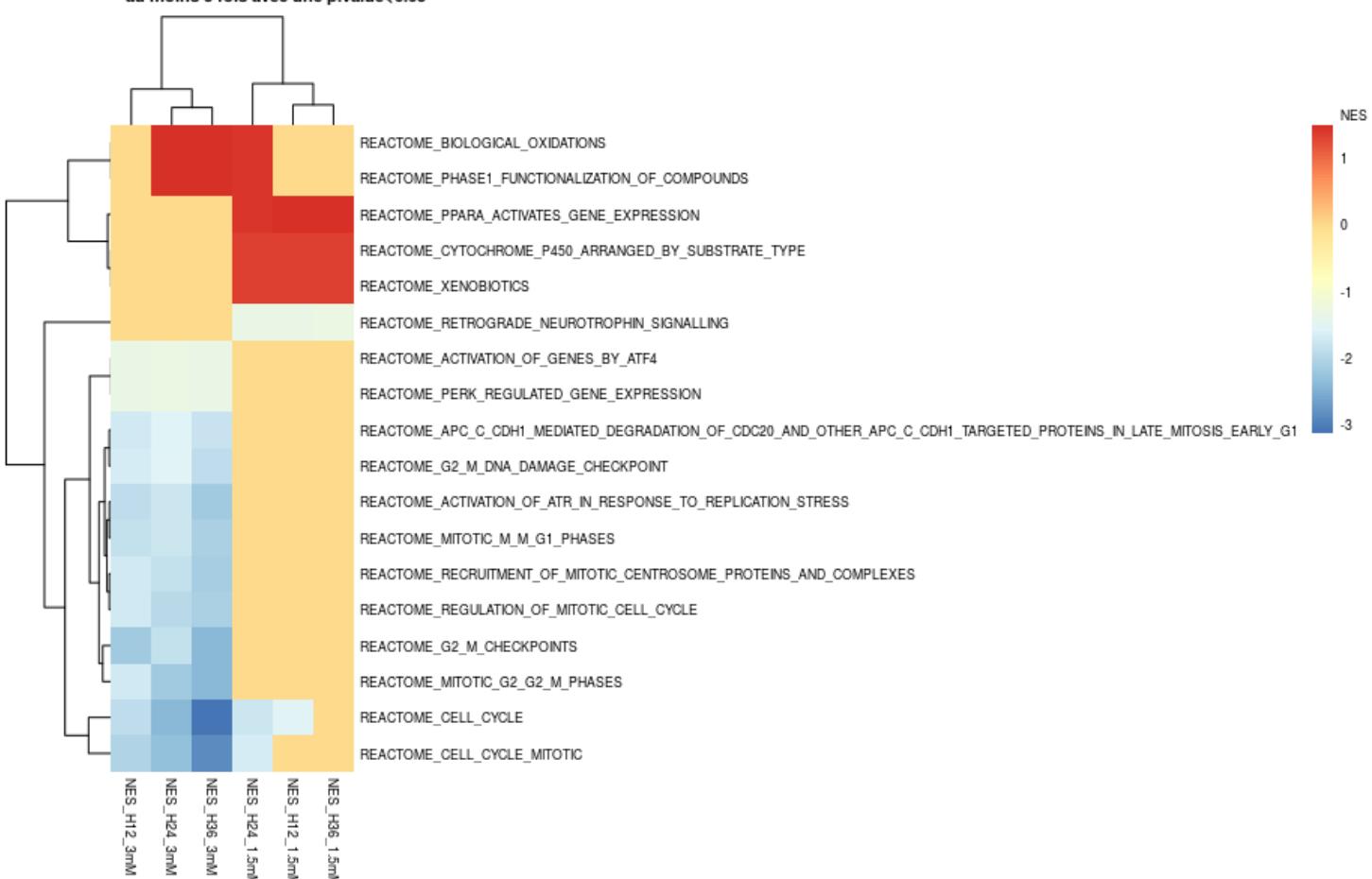
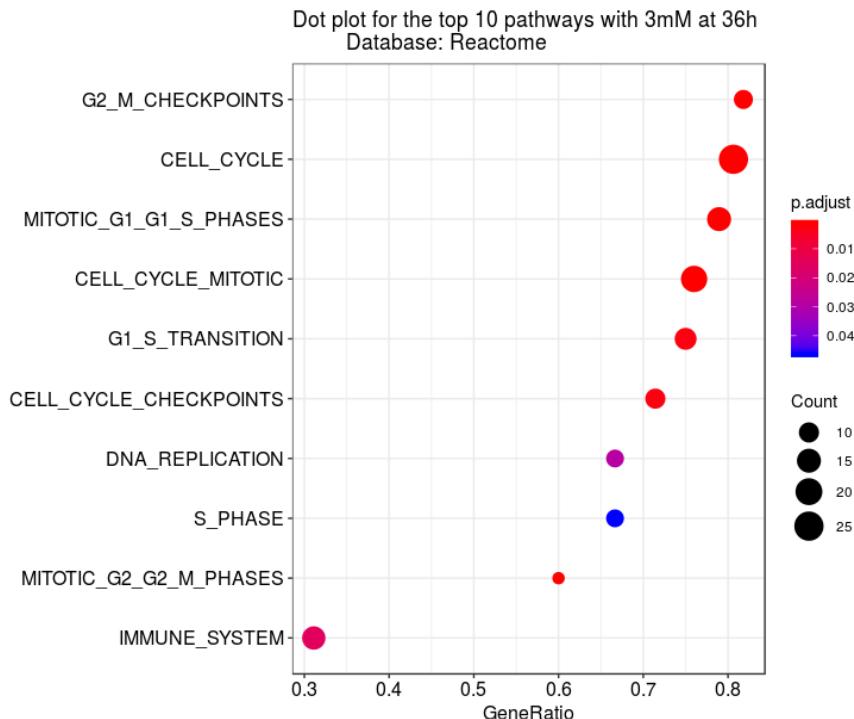


Figure 26: Heatmap des voies de signalisation en réponse de la molécule de Ganoderma

Sur cette figure, la couleur jaune indique que la voie de signalisation n'est pas significative ( $p\text{-value}>0.05$ ) et dans ce cas son score d'enrichissement normalisé est mis à 0 sur le heatmap. Les couleurs bleu et rouge signifient que la voie est respectivement sous-enrichie ou sur-enrichie. On voit que les voies de signalisation du cycle cellulaires sont sous-enrichies avec 3mM. On voit le cluster bleu formé des voies de signalisation du cycle cellulaire sous-enrichies avec une quantité de 3 micros molaire. La molécule inhibe alors les voies du cycle cellulaire.

## Statistiques sur les *gene sets* de Reactome



La figure ci-contre montre les voies de signalisation enrichies pour Reactome. A l'instar de la voie "*Immune System*", ces voies sont liées au cycle cellulaire. La taille de chaque point (*Count*) reflète le nombre des gènes significatifs parmi les gènes impliqués dans l'activation du pathways. *GeneRatio* représente le ratio entre le nombre total des gènes dans le pathway et le nombre de gènes ayant contribué à l'enrichissement.

Figure 27: Dotplot des voies de signalisations enrichies.

## Réseaux de voies de signalisation

L'image ci-dessous représente le réseau des voies de signalisation enrichies en réponse de la molécule. Les connexions entre les voies de signalisation montrent que les voies de signalisation partagent des gènes en communs. La largeur du lien qui connecte deux pathways détermine à quel niveau les deux ont des gènes en communs. On peut voir d'un côté le cluster en haut formé de neuf pathways appartenant tous au cycle cellulaire et d'autre côté la voie de signalisation "Immune System" ici isolée des autres.

Enrichmap plot for the top 10 pathways with 3mM at 36h  
Database: Reactome

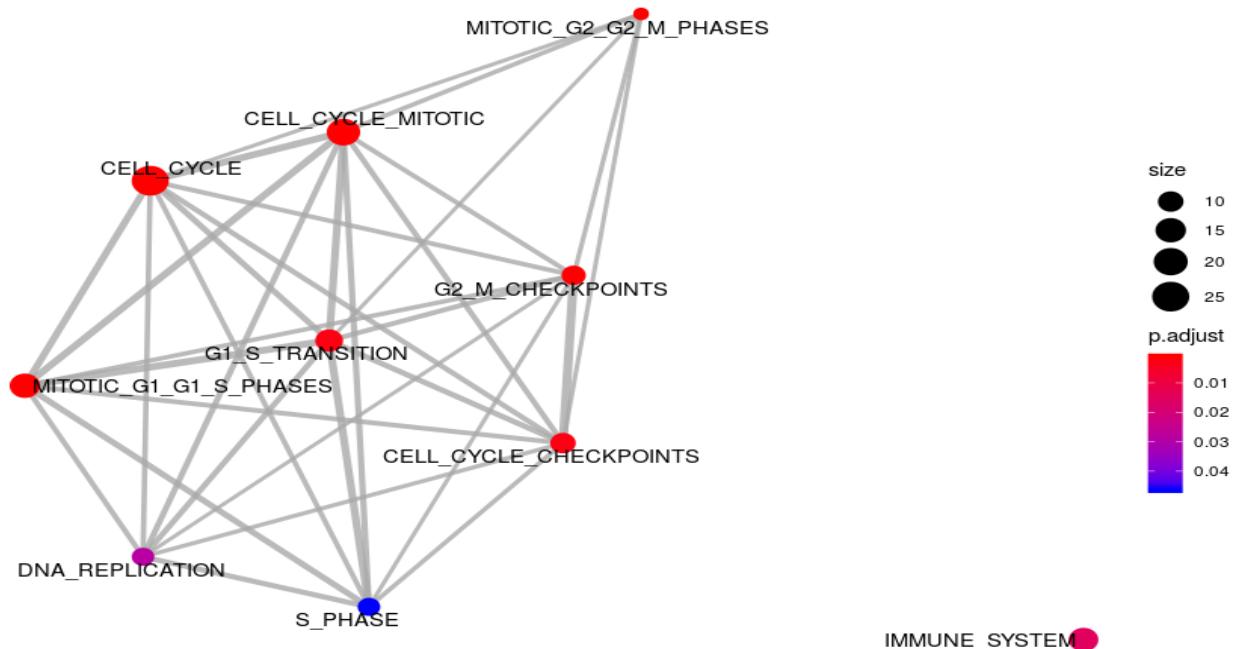


Figure 28: EnrichMap des pathways de Reactome

### Gènes co-exprimés en réponse de la molécule Ganoderma

Cnetplot for the first 10 pathways with adjusted p-value less than 0.05 with 3mM at 36h  
Database: Reactome

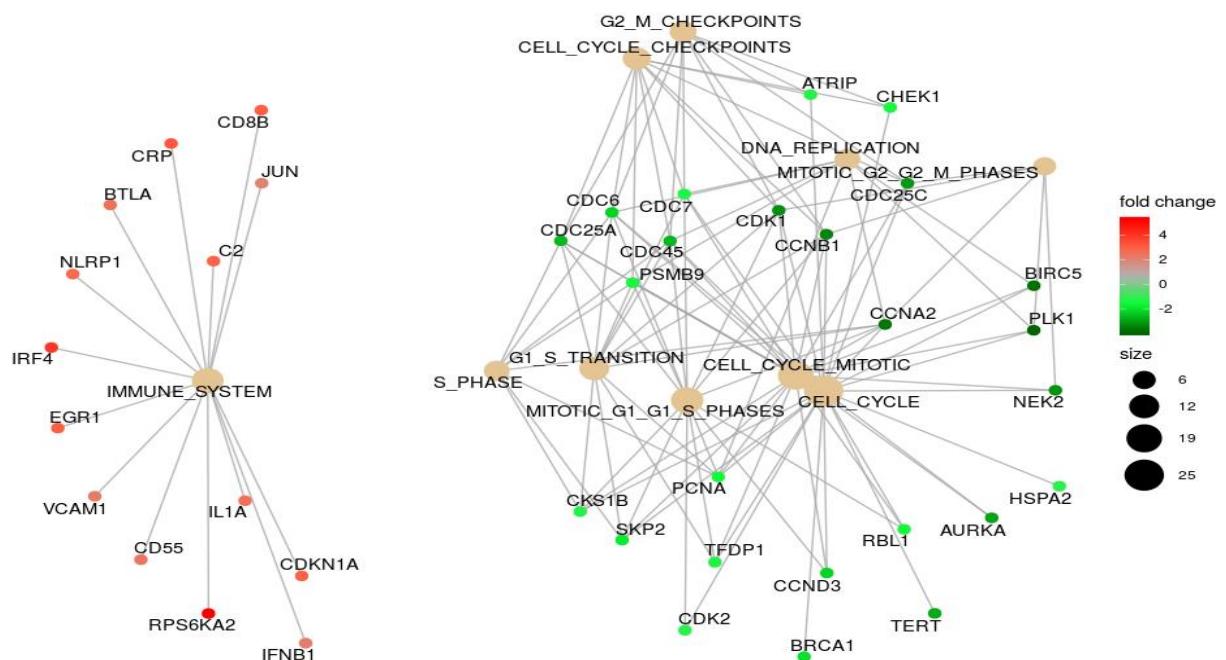


Figure 29: Cnet plot des facteurs exprimés en réponse de la molécule.

Cette image représente les gènes dans les pathways enrichies pour 3 µM à 36h. La couleur d'un gène reflète la valeur de son *fold change*.

### c) GSEA sur les données de AnyGenes

Nous avons ensuite réalisé l'analyse d'enrichissement avec 143 *gene sets* de la société AnyGenes pour 3 µM. Nous avons observé six sur 143 voies de signalisation qui sont significativement enrichies. Les résultats sur le tableau ci-dessous montrent encore une fois que les ensembles de gènes du cycle cellulaire sont sous régulés en réponse de la molécule Ganoderma. Sur ce tableau, *Down* et *Up* signifient que le pathway est respectivement sous-enrichi ou sur-enrichi.

#### ***Voies de signalisation régulées en réponse de la molécule de Ganoderma***

Base de données	Voies de signalisation	3mM		
		12h	24h	36h
AnyGenes	Human G1- S Regulators1		Down	Down
	Human Cell Cycle			Down
	Human Cell Cycle& Cancer			Down
	Human Stress& Toxicity Signaling Pathway		Up	Down
	Human Drug& Toxin Response1		Down	Down
	Human Vitiligo		Up	Down

Tableau 3:*Gene sets de AnyGenes enrichies*

#### **Réseau de voies de signalisation**

Cnetplot for the first 10 pathways with adjusted p-value less than 0.05 with 3mM at 36h  
Database: AnyGenes

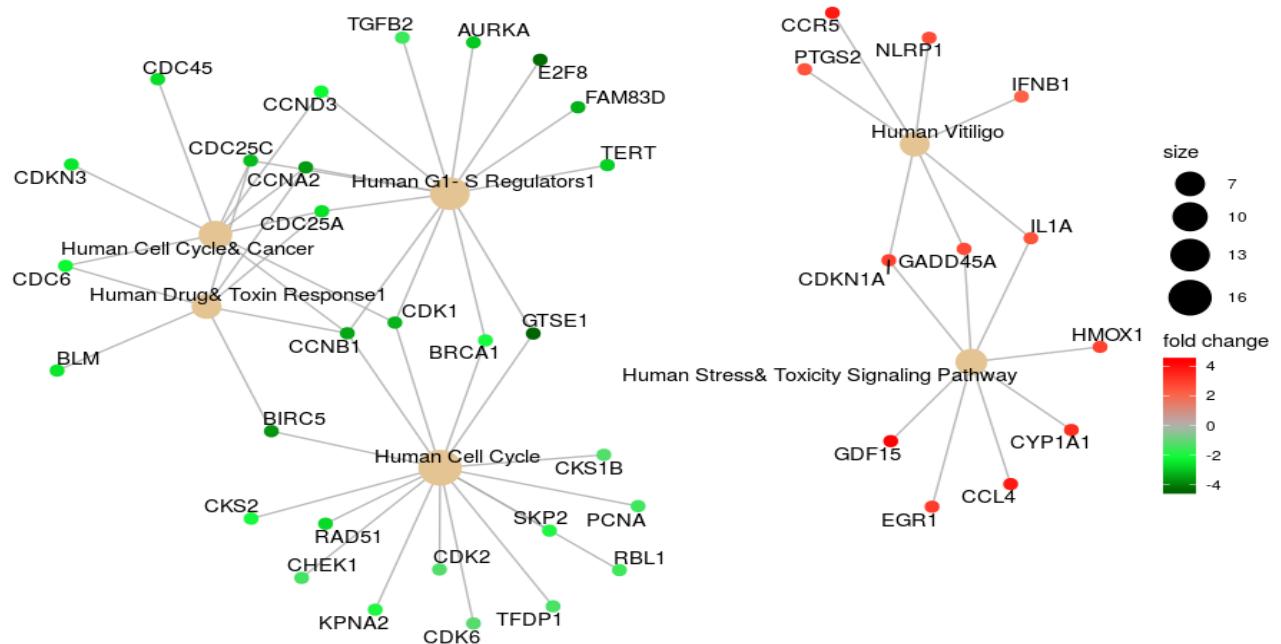


Figure 30: Cenet plot pour les gene sets de AnyGenes

## Conclusion et perspectives

L'objectif de ce projet était d'analyser l'effet de la molécule Ganoderma sur une lignée cellulaire, et identifier les biomarqueurs sur lesquels agit la molécule. Dans cet objectif, nous avons réalisé une analyse d'enrichissement à l'aide de la méthode GSEA en utilisant les voies de signalisation provenant des différentes sources. Au long de cette analyse, les voies de signalisation du cycle cellulaire sont toujours sous-enrichies avec une quantité de 3  $\mu$ M au bout de 36 heures, voir 24 heures. On peut donc conclure qu'une quantité de 3  $\mu$ M de la molécule Ganoderma permet d'inhiber le cycle cellulaire et empêcher le développement des cellules cancéreuses après 36 heures. Nos résultats doivent être approuvés par d'autres analyses, des études cliniques et biologiques avancées pour qu'enfin cette molécule puisse être classée ou non comme un possible remède pour ce type de cancer.

## **Chapitre V :**

# **Mise en place d'un modèle de prédition de l'échec tardif de la greffe rénale basé sur le profil d'expression des gènes**

## **Introduction**

L'échec tardif des greffes de reins reste un problème clinique préoccupant. Aux États-Unis, 5469 cas de greffes de rein ont développé une insuffisance rénale au stade terminal en 2008, faisant de l'insuffisance rénale la quatrième cause de néphropathie au stade terminal [21]. En France, 9861 malades attendaient une greffe de rein en janvier 2013, contre 9056 en janvier 2012 [22]. Ces chiffres montrent l'ampleur du problème rénal. Le suivi des patients ayant reçu une transplantation rénale pour s'assurer de la réussite ou l'échec reste une nécessité. L'un des moyens de suivre ses patients est l'analyse du niveau d'expression des gènes impliqués dans l'échec de la transplantation avec la technologie des puces à ADN (biopsies).

Dans ce projet nous voulons, à partir du profil d'expression des gènes des patients ayant reçu une transplantation rénale, mettre en place un modèle prédictif permettant de prédire l'échec de la greffe chez un nouveau patient qui recevra une greffe rénal. La nature des données sur lesquelles nous travaillons exige une attention particulière sur les méthodes de traitement et de classification à utiliser. Des méthodes de sélection des gènes les plus informatifs telles que SAM (Significance Analysis Of Microarrays)[1] et SVM-RFE (Support Vectors Machine Recursive Features Elimination), et d'enrichissement des données notamment la méthode SMOTE (Synthetic Minority Over-sampling Technique) ont été utilisées pendant la phases de prétraitement des données. Nous avons construit différents modèles basés sur les algorithmes k-NN, SVM, PAM et Random Forest et évalué sur un jeu de données de test. L'objectif est de comparer ces modèles et garder celui qui aura une meilleure capacité de généralisation afin de prédire l'état de santé sur de nouveaux échantillons issus du laboratoire.

## **1. Données**

Les données sur lesquelles nous avons travaillé est un jeu de données composé de 226 puces affymetrix provenant de 226 échantillons. Chaque puce (biopsie) contient les mesures du niveau d'expression de 54675 gènes prélevés chez un patient ayant reçu une greffe rénale. Chaque biopsie porte un label "Censored" ou "Failure" indiquant le résultat observé après la transplantation. Ce label constitue la variable classe à prédire. Les 183 puces proviennent des patients de la classe "Censored" et les 43 proviennent des patients de la catégorie "Failure".

## 2. Méthode

Les puces affymetrix ont été normalisées avec la méthode RMA (Robust Microarrays Analysis). Aucun filtre n'a été appliqué. Un "random split" est réalisé sur l'ensemble des puces pour séparer les données en deux parties. Une partie 70 % (Train set) pour l'entraînement des modèles et le 30 % (Test set) pour le test. Sur le jeu de données d'apprentissage nous avons utilisé la méthode SAM, qui est l'une des méthodes les plus utilisées pour la sélection des variables [15], en premier lieu pour sélectionner les gènes différentiellement exprimés (GDE), c'est-à-dire en corrélation avec l'échec du greffage. Avec une valeur seuil delta de 3.1 on a retenu 493 gènes significatifs avec un FDR de 0. Ensuite la méthode de sélection des variables SVM-RFE [19] est appliquée sur les 493 gènes pour retenir les tops 30 gènes qui constitueront les variables prédictives.

Par ailleurs nous avons utilisé l'algorithme SMOTE (Synthetic Minority Over-sampling Technique) pour équilibrer la répartition des individus dans les classes "Failure" et "Censored".

Différentes méthodes de classification notamment le k-NN, PAM, SVM et Random Forest sont utilisées sur le jeu de données d'apprentissage. Les modèles résultant sont évalués sur le jeu de données de test pour mesurer leurs capacités de généralisation sur des nouveaux individus.

Ce schéma présente le processus global suivi de l'acquisition des données jusqu'au choix d'un modèle final.

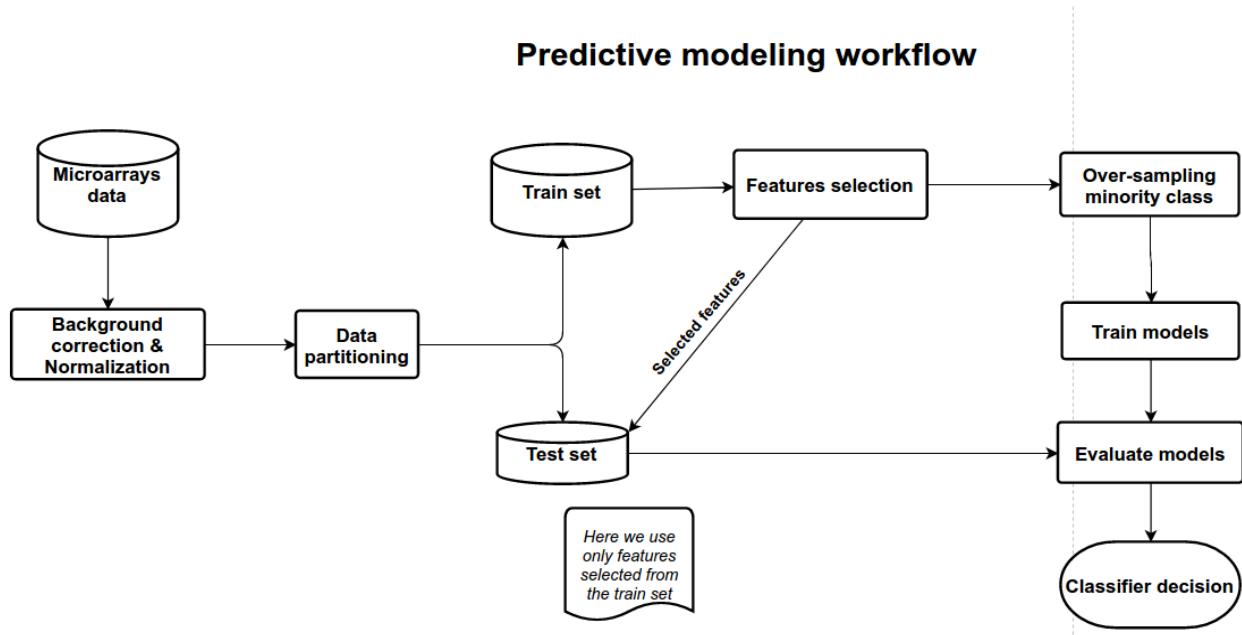


Figure 31: Workflow du modèle prédictif

## 2.1. Méthode de sélection des gènes (Features sélection)

Sélectionner les variables les plus discriminantes sur un jeu de données de grande dimension telles que les données des puces ADN est un sujet complexe. De nombreux chercheurs penchent sur le sujet en proposant différentes méthodes et en améliorant les méthodes existantes. Cependant aucune de ces méthodes n'est parfaite. Leurs efficacités dépendent de la nature des données, de l'utilisateur, etc. Dans la littérature on parle généralement de trois types de méthodes pour la sélection des variables.

Les méthodes de type "*filter*", les méthodes de type "*wrapper*" et les méthodes de type "*embedded*".

### a) Les méthodes *filter*

Les méthodes de filtrage consistent à sélectionner une sous partie des variables en se basant sur la corrélation avec la variable à prédire sans tenir compte de la performance d'un modèle d'apprentissage. Ces méthodes éliminent les variables non significatives vis-à-vis de la variable d'intérêt. SAM, t-test, filtrage par variance, gain d'information et test de chi-2 sont des exemples parmi tant des méthodes de filtrage [15,17].

### b) Les méthodes *wrapper*

Pour les méthodes de type "*wrapper*" telle que SVM-RFE (SVM Recursive Feature Elimination), l'ensemble des variables est utilisé pour entraîner le modèle (SVM avec un Kernel linéaire), puis chaque variable est évaluée sur son importance pour le modèle. L'importance d'une variable est basée sur un poids qui lui est attribué. En effet, le SVM-RFE utilise le fait que l'hyperplan du séparateur optimal est associé à un vecteur  $\mathbf{w}$  des poids  $w_k$ ,  $k=1, \dots, p$  avec  $p$  le nombre de variables,  $\mathbf{w} = (w_1, w_2, \dots, w_p)^T$ . Après l'entraînement du modèle, le vecteur de poids du séparateur optimal est utilisé comme critère pour ranger les variables. La variable la moins importante, c'est-à-dire ayant le poids le plus faible, est éliminée, [4, 32, 33]. Le même processus est répété jusqu'à ce que toutes les variables soient évaluées. Ce qui rend les méthodes de type "*wrapper*" plus coûteuses en calcul.

### c) Les méthodes *embedded*

Tout comme les méthodes "*wrapper*", Les méthodes de type "*embedded*" sélectionnent les variables importantes en entraînant un modèle d'apprentissage. La différence est qu'ici les variables moins importantes sont éliminées lors de l'apprentissage. Les méthodes basées sur les arbres de décision telles que C4.5, CART, Random Forest, sont les plus connues comme étant des méthodes "*embedded*" [18].

### d) Autres méthodes

- Méthodes hybrides : Combiner une méthode de type "*filter*" et une méthode de type "*wrapper*" ou "*embedded*" [8].
- Utilisation des connaissances biologiques, exemple GSEA [17].

## 2.2. Méthode adoptée pour la sélection des gènes

Nous avons opté pour la méthode hybride qui consiste à utiliser une méthode de filtrage pour sélectionner des gènes qui sont en corrélation avec la variable d'intérêt (*Graft failure*) à l'aide de la méthode SAM puis raffiner cette liste de gènes avec la méthode SVM-RFE qui est une méthode de type *wrapper*. La combinaison SVM-REF et SVM pour la classification aboutit généralement à un meilleur taux de classification [16] (page 11).

### a) Étape 1 : Sélection des GDE avec la méthode SAM

La première étape consiste à utiliser SAM sur le jeu de données d'entraînement pour sélectionner les GDE avec une FDR de 0.0.

#### Gènes significatifs avec une valeur delta=3.1

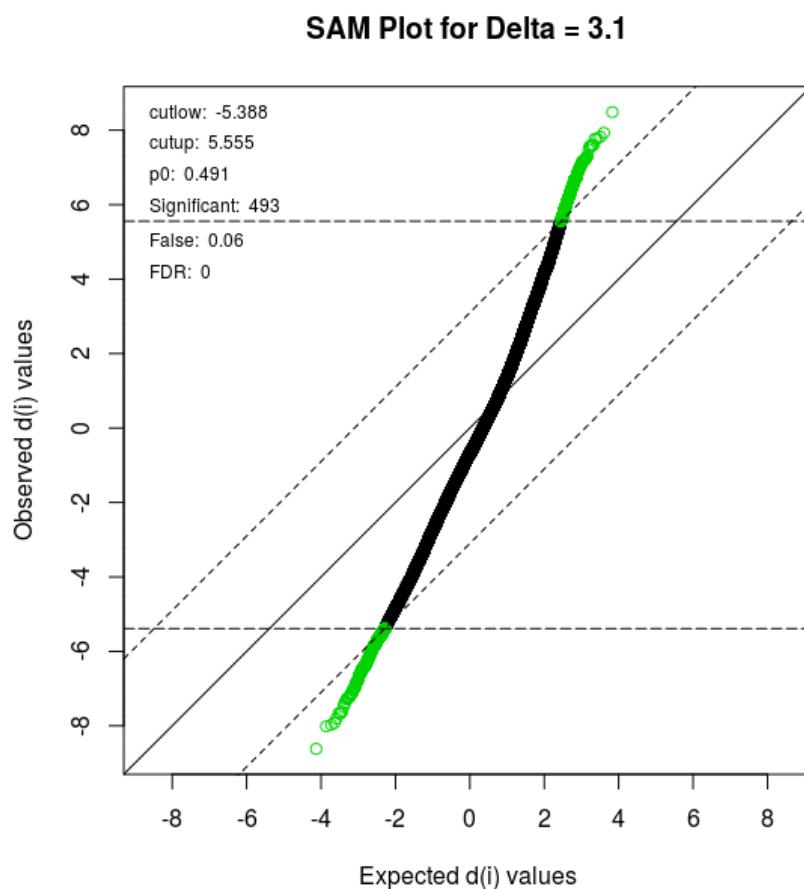


Figure 32: SAM Plots

**Sur cette figure, les points verts représentent les GDE. Au total on a 493 gènes identifiés comme significativement exprimés avec un FDR de 0. Les points verts en haut sont les gènes surexprimés et en bas les sous-exprimés.**

### b) Étape 2: Recursive feature elimination

Le SVM-RFE est une méthode de type *wrapper* pour la sélection des variables. Elle consiste à entraîner un modèle basé sur l'algorithme SVM (avec kernel linéaire) puis évaluer la qualité

de chaque variable. La variable la moins importante est éliminée de l'ensemble des variables et le modèle est entraîné à nouveau avec les autres variables. Le processus se répète jusqu'à ce que toutes les variables soient évaluées et à la fin toutes les variables sont arrangeées en ordre d'importance.

Nous présentons un pseudo code de l'algorithme SVM-RFE.

```

Data : Dataset with  $p^*$  variables and binary outcome.
Output: Ranked list of variables according to their relevance.

Find the optimal values for the tuning parameters of the SVM model;
Train the SVM model;
 $p \leftarrow p^*$ ;
while  $p \geq 2$  do
     $SVM_p \leftarrow$  SVM with the optimized tuning parameters for the  $p$  variables and
    observations in Data;
     $w_p \leftarrow$  calculate weight vector of the  $SVM_p$  ( $w_{p1}, \dots, w_{pp}$ );
     $rank.criteria \leftarrow (w_{p1}^2, \dots, w_{pp}^2)$ ;
     $min.rank.criteria \leftarrow$  variable with lowest value in  $rank.criteria$  vector;
    Remove  $min.rank.criteria$  from Data;
     $Rank_p \leftarrow min.rank.criteria$ ;
     $p \leftarrow p - 1$  ;
end
 $Rank_1 \leftarrow$  variable in Data  $\notin (Rank_2, \dots, Rank_{p^*})$ ;
return ( $Rank_1, \dots, Rank_{p^*}$ )

```

**Fig. 1** Pseudo-code of the SVM-RFE algorithm using the linear kernel in a model for binary classification

*Figure 33: Pseudo code de SVM-RFE (source [19], page 3)*

Dans les 493 gènes sélectionnés par SAM, nous avons procédé à une élimination récursive des variables. Cette technique utilise la méthode SVM-REF (SVM Recursive Features Elimination) pour éliminer les variables les moins significatives d'une manière récursive en entraînant à chaque fois un modèle basé sur la méthode SVM. Ensuite nous avons retenu les 30 tops variables les plus distinctives selon cette méthode. Ces gènes constitueront les entrées de nos modèles d'apprentissage.

Les gènes sélectionnés comme meilleures variables sont utilisés pour les données d'entraînement et pour les données de test.

### 2.3. SMOTE (Synthetic Minority Over-sampling Technique)

Les individus de notre jeu de données d'apprentissage sont inégalement répartis dans les deux classes. Cela peut conduire à des mauvais résultats de classification surtout pour la classe minoritaire. Pour espérer avoir un taux de classification assez significatif, nous avons proposé une méthode dite SMOTE. Cette technique utilise l'algorithme de k-NN pour synthétiser de nouvelles instances de la classe minoritaire pour équilibrer les deux classes. Cette technique est beaucoup plus utilisée pour les données non balancées [20,24]. D'ailleurs vue la

complexité des données (très grande dimension), V. Bolon-Canedo et A. Alonso-Betanzos suggèrent d'utiliser SMOTE après avoir sélectionné les variables [23], (page 293). Dans leur conclusion, R. Blagus et L. Lusa reportent dans leur article [25] que SMOTE est bénéfique pour les méthodes de k-NNs si une sélection de variables a eu lieu avant son application, et qu'elle n'est pas du tout adaptée à l'analyse discriminante (LDA). Un autre article de Li Ma et Suohai Fan montre qu'il est possible d'améliorer le taux de classification de la classe minoritaire pour le Random Forest en utilisant des techniques de prétraitement comme SMOTE [26].

**a) Algorithme SMOTE :**

$O$  is the original data set

$P$  is the set of positive instances (minority class instances)

For each instance  $x$  in  $P$

    Find the  $k$ -nearest neighbors (minority class instances) to  $x$  in  $P$

    Obtain  $y$  by randomizing one from  $k$  instances

$difference = x - y$

$gap = \text{random number between } 0 \text{ and } 1$

$n = x + difference * gap$

    Add  $n$  to  $O$

End for

Figure 34: SMOTE algorithme, source [27].

**b) SMOTE appliquée à notre jeu de données d'apprentissage après sélection des variables**

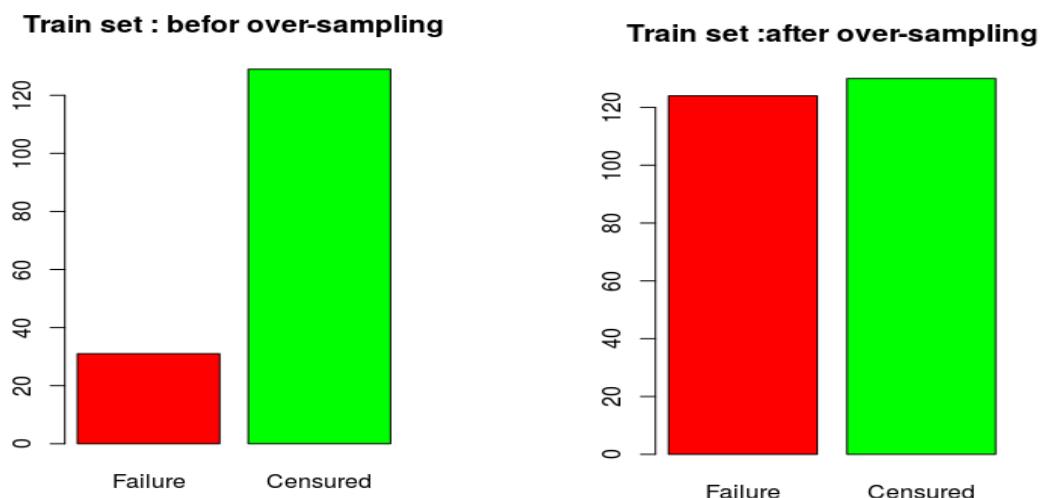


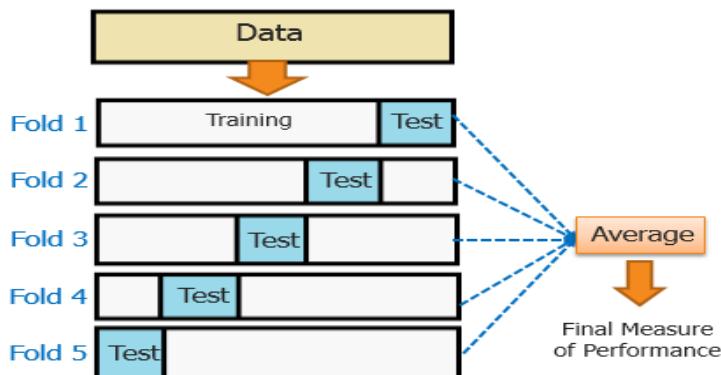
Figure 35: Données avant et après SMOTE

Bien évidemment cette technique des données synthétisées peut présenter des inconvénients notamment le sur-apprentissage. De nombreux articles sont dédiés à cette technique sur ces avantages et inconvénients. Ce qui est sûr est que cette méthode reste un mal nécessaire dans le cas où nous n'avons pas des données à notre disposition pour enrichir la classe minoritaire.

## 2.4. Méthode de k-fold cross validation

Le jeu de données "Train set " (70%) est utilisé pour entraîner différents modèles d'apprentissage en utilisant la technique de k-fold cross validation. Cette technique est une étape interne lors de l'entraînement d'un modèle. Elle permet en général d'avoir de bons résultats en évitant le phénomène de sur-apprentissage. Le processus de k-fold cross validation peut se résumer sur ce schéma :

### Processus de k-fold cross validation



*Figure 36: Illustration de la méthode de k-fold cross validation*

Sur cette figure les données d'entraînement (Data) sont séparées en 5 parties (5 folds). Les 4/5 sont utilisées pour entraîner le modèle et le 5ème (Test) pour évaluer le modèle. Ce processus est répété 5 fois et à chaque fois, les mesures de performance du modèle sont obtenues en faisant la moyenne des 5 mesures internes.

## 3. Résultats

### 3.1. Comparaison des modèles

## Measuring performance: ROC, Sensitivity and Specificity

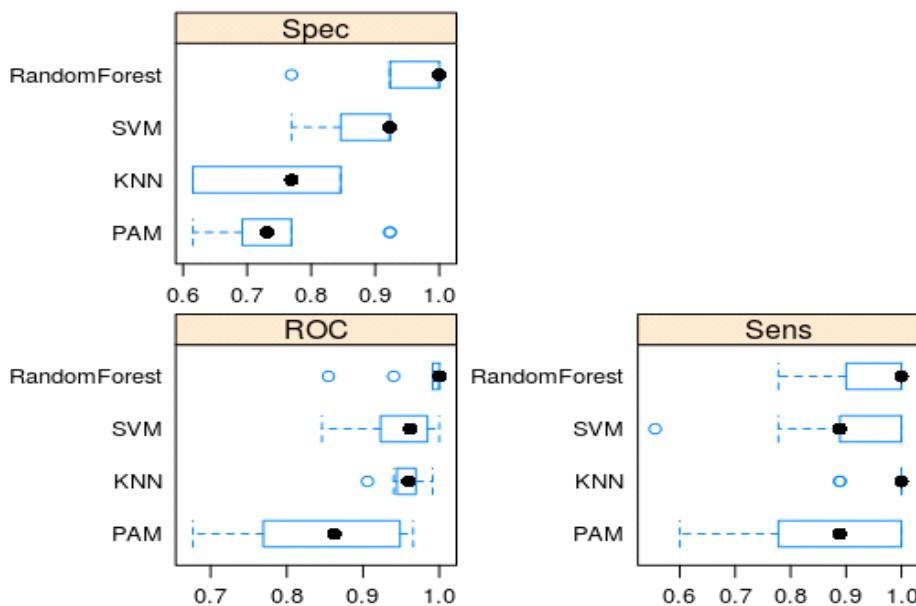


Figure 37: Mesures de performances des modèles

Les modèles Random Forest et SVM achèvent avec un bon score en sensibilité et en spécificité contrairement à KNN et PAM qui ont un très bon score en sensibilité uniquement. Cela veut dire que les méthodes KNN et PAM vont bien détecter les individus de la classe "Failure" et commettre plusieurs erreurs sur les individus de la classe "Censored" ce qui diminuera la précision en augmentant le taux des faux positifs.

Les modèles finaux à retenir sont donc le SVM et le Random Forest que nous présentons leurs matrices de confusion et statistiques sur le jeu de données de test.

### Evaluation des modèles de Random Forest et SVM sur les données de test

On note:

TP : True Positif (Individus de la classe "Failure" classés comme "Failure").

TN : True Negative (Individus de la classe "Censored" classés comme "Censored").

FP : False Positif (Individus de la classe "Censored" classés comme "Failure").

FN : False Negative (Individus de la classe "Failure" classés comme "Censored").

#### a) Rappelle pour l'interprétation de la matrice de confusion

$$\text{Accuracy} = (TP+TN)/(TP+TN+FP+FN)$$

$$\text{Sensitivity or Recall} = TP/(TP+FN)$$

$$\text{Specificity} = TN/(TN+FP)$$

$$\text{Precision} = TP / (TP+FP)$$

#### b) Matrices de confusion des modèles de SVM et Random Forest sur les données de test

```
> confusionMatrix(as.factor(pred_rf), as.factor(testSet$outcome))
Confusion Matrix and Statistics
```

Reference		
Prediction	Failure	Censured
Failure	11	4
Censured	1	50

Accuracy : 0.9242  
95% CI : (0.832, 0.9749)  
No Information Rate : 0.8182  
P-Value [Acc > NIR] : 0.01279  
Kappa : 0.7679  
McNemar's Test P-Value : 0.37109  
Sensitivity : 0.9167  
Specificity : 0.9259  
Pos Pred Value : 0.7333  
Neg Pred Value : 0.9804  
Prevalence : 0.1818  
Detection Rate : 0.1667  
Detection Prevalence : 0.2273  
Balanced Accuracy : 0.9213  
'Positive' Class : Failure

```
> confusionMatrix(as.factor(svm_pred), as.factor(testSet$outcome))
Confusion Matrix and Statistics
```

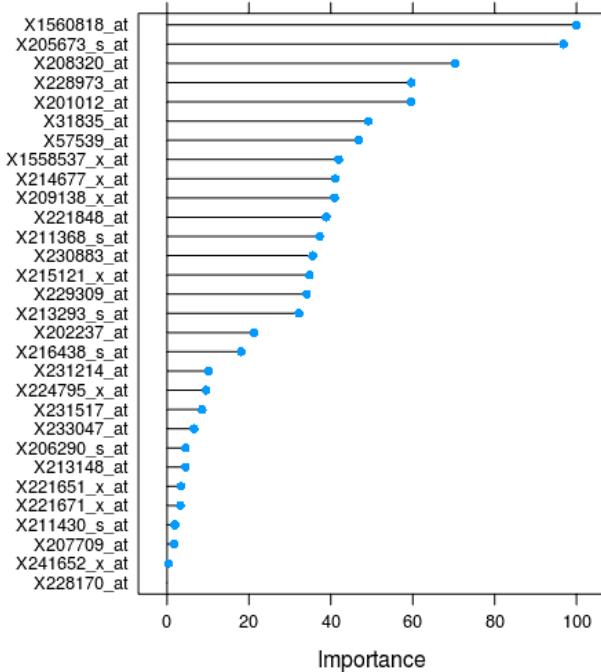
Reference		
Prediction	Failure	Censured
Failure	11	3
Censured	1	51

Accuracy : 0.9394  
95% CI : (0.852, 0.9832)  
No Information Rate : 0.8182  
P-Value [Acc > NIR] : 0.004216  
Kappa : 0.8087  
McNemar's Test P-Value : 0.617075  
Sensitivity : 0.9167  
Specificity : 0.9444  
Pos Pred Value : 0.7857  
Neg Pred Value : 0.9808  
Prevalence : 0.1818  
Detection Rate : 0.1667  
Detection Prevalence : 0.2121  
Balanced Accuracy : 0.9306  
'Positive' Class : Failure

Figure 38: Matrices de confusion et statistiques des Random Forest (gauche) et SVM (droite)

## Importance des variables

Variables Importance for Random Forest



Variables Importance SVM

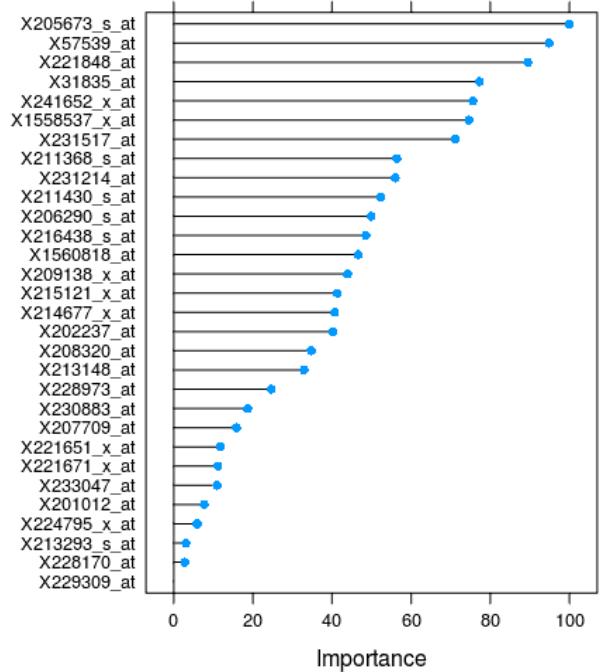


Figure 39: Importance des variables pour Random Forest

Les 30 gènes utilisés ne sont pas tous importants surtout pour le Random Forest (Fig.38). Nous avons décidé de choisir les 20 premiers gènes et construire à nouveau le modèle basé sur le Random Forest.

### 3.2. Utilisation de 20 tops gènes avec Random Forest

Nous avons sélectionné uniquement les tops 20 gènes de SVM-RFE et repris à nouveau le modèle de Random Forest (RF).

#### Matrice de confusion du RF sur les données d'apprentissage

```
> rf_model

Call:
randomForest(formula = outcome ~ ., data = trainSet, importance = T,      corr.bias = T)
  Type of random forest: classification
                    Number of trees: 500
  No. of variables tried at each split: 4

    OOB estimate of  error rate: 3.54%
Confusion matrix:
             Failure Censured class.error
Failure       122        2  0.01612903
Censured        7       123  0.05384615
> |
```

Figure 40: Matrice de confusion de Random Forest pour 20 variables

#### Estimation de l'erreur pour le modèle de Random Forest

Sur cette image l'axe horizontale représente le nombre d'arbres créés et l'axe verticale montre l'erreur de classification en fonction du nombre d'arbres de décision construits. On

voit qu'à moins de 150 arbres, l'erreur n'est pas encore stable. Nous avons gardé 500 arbres comme paramètre par défaut. On constate qu'avec 20 variables on a réussi à améliorer le taux de classification avec une erreur d'environ 3.54 %.

Pour confirmer de telles mesures, nous devons évaluer le modèle sur le jeu de données non appris lors de l'entraînement du modèle.

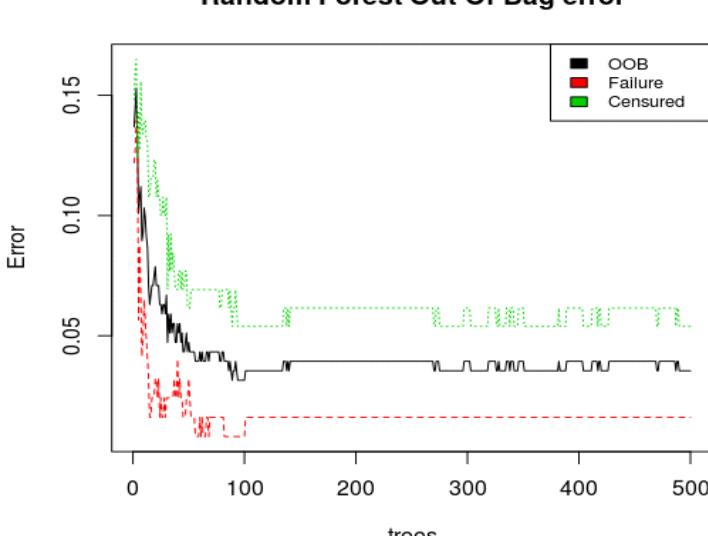


Figure 41: Out-Of-Bag Error

## Évaluation du modèle sur le jeu de données de test

Sur l'image ci-dessous on constate un taux d'erreur de  $2/(12+54)=3\%$  ce qui est bien proche de l'erreur estimée lors de l'apprentissage ( $\text{OOB}=0.354$ ). En plus on observe un excellent taux de classification pour les individus de la classe "Failure", (Sensitivité). En effet sur la Fig.40 c'est bien cette classe qui a une très faible erreur (presque 0).

Les mesures obtenues pour les données de test confirment bien la capacité de généralisation du modèle sur des nouveaux cas.

```
Console ~/Desktop/Aboubakar/Classifiers/New Approche/ ↵
> confusionMatrix(as.factor(rf_pred), as.factor(testSet$Outcome))
Confusion Matrix and Statistics

Reference
Prediction Failure Censored
Failure      12      2
Censored       0     52

Accuracy : 0.9697
95% CI : (0.8948, 0.9963)
No Information Rate : 0.8182
P-Value [Acc > NIR] : 0.0002153

Kappa : 0.9043

McNemar's Test P-Value : 0.4795001

Sensitivity : 1.0000
Specificity : 0.9630
Pos Pred Value : 0.8571
Neg Pred Value : 1.0000
Prevalence : 0.1818
Detection Rate : 0.1818
Detection Prevalence : 0.2121
Balanced Accuracy : 0.9815

'Positive' Class : Failure
> |
```

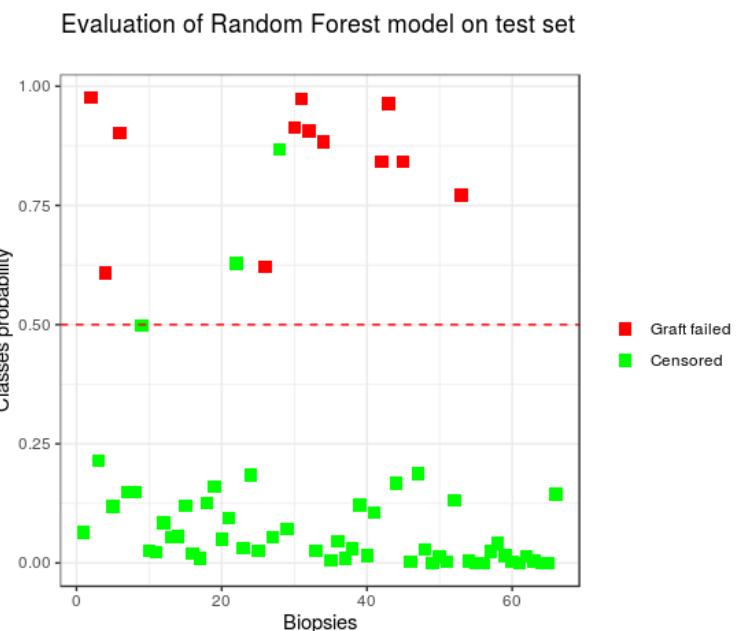


Figure 42: Matrice de confusion et visualisation pour le jeu de données de test

## Conclusion et perspectives

L'objectif de ce projet est la mise en place d'un modèle prédictif pour la prédiction de l'échec tardif chez les patients qui ont reçu une transplantation rénale à partir du profile d'expression des gènes. Après une phase de prétraitement des données incluant la sélection des variables pertinentes et un enrichissement de la classe minoritaire, différentes méthodes de classification ont été utilisées. Les résultats montrent que le SVM et le Random Forest achèvent avec les scores plus élevés par rapport à KNN et PAM. Les résultats ont montré aussi qu'un nombre réduit de variables bien sélectionnées peut suffire pour avoir un bon taux de classification avec le Random Forest. En perspective, nous pensons qu'il serait curieux de combiner GSEA pour la sélection des gènes et Random Forest et enfin comparer les résultats pour les deux démarches. En revanche le modèle mis en place peut être utilisé pour prédire des données réelles. Une autre perspective serait de déployer ce modèle dans un environnement de production pour l'utilisation quotidienne. Cela demande la mise en œuvre d'une architecture logicielle.

## Conclusion générale

Qu'elles proviennent des puces à ADN ou d'autres technologies, les données génomiques constituent une source d'informations qui préoccupent les acteurs de la biologie.

L'objectif de ce stage était l'analyse des données génomiques à haut débit. Plusieurs approches ont été mises en œuvre notamment l'analyse d'enrichissement et les méthodes de classification. La méthode GSEA nous a permis de révéler les gènes impliqués dans le diabète de type 2 quand la méthode classique d'analyse d'expression différentielle, SAM n'a pas pu identifier aucun gène vue la modeste variation du niveau d'expression entre les individus normaux et les individus diabétiques. Encore, grâce à la méthode GSEA et en explorant les bases de données publiques des voies de signalisation, nous avons montré qu'avec une quantité de 3 µM la molécule Ganoderma est capable d'inhiber le cycle cellulaire au bout de 36 heures. Les méthodes de classification ont pris de l'ampleur sur les données des puces à ADN. Malgré sa grande dimension et le nombre souvent réduit d'échantillons, les données des puces à ADN ont une importance capitale pour la détection et à la prédiction des maladies chez les êtres vivants. Nous avons construit quatre modèles de classification pour prédire l'échec tardif du greffage rénal en utilisant le profil d'expression de gènes. Les modèles de Random Forest et de SVM ont d'une part montré un meilleur taux de réussite comparé aux modèles de k-NN et PAM. D'autre part le Random Forest a montré un excellent taux de prédiction une fois qu'on a réduit le nombre de variables en seulement 20 gènes.

Durant ce stage, nous avons eu l'occasion de travailler sur différents projets, allant du traitement des données volumineuses à la diffusion des résultats de l'analyse, ce qui nous a permis de maîtriser le processus complet d'analyse de données génomiques à haut débit. En outre, ce stage nous a permis de comprendre le monde de l'entreprise, appliquer nos compétences académiques et en acquérir d'autres. Nous avons appris à travailler en collaboration avec des personnes d'un domaine différent du nôtre et à répondre aux exigences d'un environnement qui nécessite de la rigueur. Sur ce, j'ai enfin découvert un métier, le « data mining » qui me passionne.

## Références

- [1] François L., *Comparaison des méthodes d'analyse de l'expression différentielle basée sur la dépendance des niveaux d'expression*, Montréal, Université de Montréal, Faculté de médecine département de biochimie, 2011, p121.
- [2] Affymetrix, *The Structure, Function, and Applications of GeneChip® Microarrays*, [http://www.affymetrix.com/about\\_affymetrix/outreach/educator/downloads/chip\\_function\\_teacher\\_notes.pdf](http://www.affymetrix.com/about_affymetrix/outreach/educator/downloads/chip_function_teacher_notes.pdf)
- [3] Voie de signalisation, <https://www.eupati.eu/fr/glossary/voie-de-signalisation/>
- [4] Darius M. D., *DATA MINING FOR GENOMICS AND PROTEOMICS Analysis of Gene and Protein Expression Data*, New Jersey, Willey, 2010, 349p.
- [5] Claudio M. S., *Exploration transcriptomique et logique de la voie TLR4 dans le contexte physiopathologique du sepsis*.
- [6] Mootha, V. K., Lindgren, C. M., Eriksson, K.-F., Subramanian, A., Sihag, S., Lehar, J., ... Groop, L. C. *PGC-1 $\alpha$ -responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes*. *Nature Genetics*, 2003, 34(3), 267–273.
- [7] Shi, J., & Walker, M. *Gene Set Enrichment Analysis (GSEA) for Interpreting Gene Expression Profiles*. *Current Bioinformatics*, 2007, 2(2), 133–137.
- [8] Alberoto P., *Preprocess and Data Analysis Techniques for Affymetrix DNA Microarrays Using Bioconductor: A Case Study in Alzheimer disease*, Department of Computer Science and Artificial Intelligence, Univeristy of Basque Country, 2013,p79.
- [9] Gil C., Michael S., Jun L., Balasubramanian N. , Robert T., Virginia T. , *SAM “Significance Analysis of Microarrays” Users guide and technical document*,p26.
- [10] Ariel E. B., Monica G. L., Pablo M. G., Juan C. G., and Elizabeth T., *Gene Set Enrichment Analysis Using Non-parametric Scores*.
- [11] Jing S. and Michael G. W., *Gene Set Enrichment Analysis (GSEA) for Interpreting Gene Expression Profiles*, Stanford University, Stanford, USA, Biomedical Informatics Program, MC 5429, CA 94305.
- [12] *Analysis of DNA Chips and Gene Networks Spring Semester, 2009 Lecture 14a: January 21, 2010 Lecturer: Ron Shamir*
- [13] *Gene Set Enrichment Analysis*, [https://www.pathwaycommons.org/guide/primers/data\\_analysis/gsea/](https://www.pathwaycommons.org/guide/primers/data_analysis/gsea/)
- [14] *Interpreting GSEA Results*, <https://software.broadinstitute.org/gsea/doc/GSEAUserGuideFrame.html>

- [15] Shahjaman, M., Kumar, N., Ahmed, M. S., Begum, A., ... Islam, S. M. S. *Robust Feature Selection Approach for Patient Classification using Gene Expression Data Bioinformation*, 2017, 13(10), 327–332.
- [16] Pirooznia, M., Yang, J. Y., Yang, M. Q., & Deng, Y. *A comparative study of different machine learning methods on microarray gene expression data*. *BMC Genomics*, 2008, 9(Suppl 1), S13.
- [17] Bair, E. *Identification of significant features in DNA microarray data*. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2013, 5(4), 309–325.
- [18] Jovic, A., Brkic, K., & Bogunovic, N. *A review of feature selection methods with applications*. *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2015.
- [19] Sanz, H., Valim, C., Vegas, E., Oller, J. M., & Reverter, F. *SVM-RFE: selection and visualization of the most relevant features through non-linear kernels*. *BMC Bioinformatics*, 2018, 19(1).
- [20] Prasad Potharaju, S., & Sreedevi, M. *An Improved Prediction of Kidney Disease using SMOTE*. *Indian Journal of Science and Technology*, 2016, 9(31).
- [21] Sellares, J., de Freitas, D. G., Mengel, M., Reeve, J., Einecke, G., Sis, B., ... Halloran, P. F. *Understanding the Causes of Kidney Transplant Failure: The Dominant Role of Antibody-Mediated Rejection and Nonadherence*. *American Journal of Transplantation*, 2011, 12(2), 388–399.
- [22] *La greffe à partir d'un donneur vivant peut être une solution*, [http://www.fondation-du-rein.org/assets/sites/www.fondation-du-rein.org/uploaded/Brochure\\_ABM - Greffe de rein de donneur vivant 2015.pdf](http://www.fondation-du-rein.org/assets/sites/www.fondation-du-rein.org/uploaded/Brochure_ABM - Greffe de rein de donneur vivant 2015.pdf).
- [23] V. Bolon-Canedo, A. Alonso-Betanzos , *Data complexity measures for analyzing the effect of SMOTE over microarrays* , <https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2016-134.pdf> .
- [24] Ajinkya More, *Survey of resampling techniques for improving classification performance in unbalanced datasets*, <https://arxiv.org/pdf/1608.06048.pdf>
- [25] Blagus, R., & Lusa, L. *SMOTE for high-dimensional class-imbalanced data*. *BMC Bioinformatics*, 2013, 14(1), 106.
- [26] Ma, L., & Fan, S. *CURE-SMOTE algorithm and hybrid algorithm for feature selection and parameter optimization based on random forests*. *BMC Bioinformatics*. 2017, 18(1).
- [27] Jeatrakul, P., Wong, K. W., & Fung, C. C. *Classification of Imbalanced Data by Combining the Complementary Neural Network and SMOTE Algorithm*. *Neural Information Processing. Models and Applications*, 2010, 152–159.
- [28] Affymetrix, *Statistical Algorithms Description Document*, 2002.

- [29] Mohammad O. S., *Improving the performance of the prediction analysis of microarrays algorithm via different thresholding methods and heteroscedastic modeling*, Department of Statistics College of Arts and Sciences KANSAS STATE UNIVERSITY Manhattan, 2014, p134.
- [30] Robert T., Trevor H., Balasubramanian N. and Gilbert C., *Class Prediction by Nearest Shrunken Centroids, with Applications to DNA Microarrays*, Institute of Mathematical Statistics, 2003, p.14.
- [31] Marko, P., *qPCR, Microarrays or RNA-sequencing: When To Choose One Over the Other?*, National Institute of Biology, Ljubljana, Slovenia, 2017. <https://biosistemika.com/blog/qpcr-microarrays-rna-sequencing-choose-one/>
- [32] Huang, M.-L., Hung, Y.-H., Lee, W. M., Li, R. K., & Jiang, B.-R. *SVM-RFE Based Feature Selection and Taguchi Parameters Optimization for Multiclass SVM Classifier*. *The Scientific World Journal*, 2014, 1–10.
- [33] Mouhamadou L., Fodé C., Samba N., Yahya S., & Mohamed A. E., *A Novel RFE-SVM-based Feature Selection Approach for Classification*. *International Journal of Advanced Science and Technology*, 2012, p10.