REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Mohamed Khider – BISKRA

Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie

Département d'informatique

N° d'ordre: SIOD 2/M2/2020

Mémoire

Présenté pour obtenir le diplôme de master académique en

Informatique

Parcours : **système d'information d'optimisation et de décision (SIOD)**

Traitement parallèle pour la sélection et la classification des puces à ADN

Par:

DJEFFAL HIDAYET

Soutenu le XX/09/2020, devant le jury composé de :

XXXXXXXXX	M A A	Président
Djerou Leila	Professeur	Rapporteur
XXXXXXXXX	M A A	Examinateur

Déduction

Tout d'abord, je remercie le Dieu, notre créateur de m'avoir donné la force, la volonté et le courage afin d'accomplir ce travail modeste.

Je dédie ce travail à la mémoire de ma grande mère, j'aurais tant aimé que tu sois présente, que dieu ait ton âme dans sa sainte miséricorde.

A ma mère, la source de tendresse et la lumière qui guide mes routes et qui m'emmène aux chemins de la réussite, pour tous ses sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie.

A mon père que je le remercie énormément pour ses efforts, ses conseils et sa surveillance.

À mes chers frère et sœurs : Mahdi, Hadil et Asma

À mon meilleure amie : Nour

À tout ce que je connais sans exceptions.

A tous mes enseignants sans exception

Enfin, j'offre mes bénédictions à tous ceux qui m'ont soutenu dans l'accomplissement de ce travail.

Remerciement

La première et la dernière chose est pour Allah qui me donne la capacité suffisante pour terminer ce travail.

Je veux remercier mon superviseur madame DJEROU LEILA, pour ses conseils, ses encouragements et ses conseils qu'elle a prodigués au patient tout au long de mon séjour en tant qu'étudiant. J'ai été extrêmement chanceuse d'avoir un superviseur qui se souciait tellement de mon travail et qui a répondu à mes questions et mes requêtes si rapidement.

Je tiens également à remercier les membres du jury qui ont bien voulu lire et examiner notre travail.

Un merci spécial à tous ceux qui m'ont soutenu pour terminer ce travail.

'Cela semble toujours impossible jusqu'à ce qu'on le fasse'

-Nelson Mandela-

الملخص

تسمح رقائق الحمض النووي (الرقائق الدقيقة) بدراسة الترنسكريبتوم من خلال مراقبة تعبير عدة آلاف من الجينات في خلية أو نسيج في نفس الوقت. يلعبون دورًا رئيسيًا في فهم الأمراض الوراثية، مثل السرطان، وفي تطوير جزيئات علاجية جديدة. البيانات الخاصة بالتعبير الجيني من المصفوفات الدقيقة لها عيب رئيسي وهو الأبعاد، حيث يكون عدد الميزات ضخمًا مقارنة بعدد العينات. لا تختار طرق الانتقاء الجيني سوى الجينات التي تلعب دورًا رئيسيًا في اكتشاف المرض وعلاجه المناسب. ومع ذلك، فإن معظم هذه الأساليب تستغرق وقتًا طويلاً جدًا. بالإضافة إلى ذلك، فإن الأدوات التقليدية للتقييم المتسلسل لجودة الرقاقة والمعالجة المسبقة غير قادرة على التعامل مع كميات كبيرة من مجموعات البيانات..

في هذا العمل، اقترحنا طريقة ANOVA لاختيار الجينات بالتوازي من خلال استغلال صندوق أدوات الحساب المتوازي فيmatlab (Parallel Computing Toolbox PCT)، ثم طبقنا خوارزمية تصنيف kNN لتقييم دقة التصنيف.

الكلمات المفتاحية: المعلوماتية الحيوية، المصفوفة الدقيقة للحمض النووي، اختيار الجينات، الموازي.

الملخص

Résumé

Les puces à ADN (micropuces) permettent d'étudier le transcriptome par l'observation simultanée de l'expression de plusieurs milliers de gènes dans une cellule ou un tissu. Elles jouent unrôle majeur dans la compréhension des maladies génétiques, comme le cancer et développer de nouvelles molécules thérapeutiques. Les données sur l'expression des gènes de puces à ADN présentent l'inconvénient majeur d'une malédiction de dimensionnalié, où le nombre de caractéristiques est énorme par rapport à celui d'échantillons. Les méthodes de sélection des gènes ne sélectionnent que les gènes qui jouent un rôle de premier plan dans la détection d'une maladie et son traitement adéquat. Cependant, la plupart de ces méthodes prennent beaucoup de temps. En outre, les outils traditionnels d'évaluation et de prétraitement séquentiel de la qualité des micropuces ne sont pas en mesure de gérer une grande quantité de jeux de données.

Dans ce travail nous avons proposé une méthode d'ANOVA pour la sélection des gènes en parallèle en exploitant la boite outil de calcul parallèle en matlab (Parallel Computing Toolbox PCT), puis nous avons appliqué l'algorithme de classification kNN pour évaluer la précision de la classification.

Mots clés : *Bioinformatique, Puce à ADN, Sélection des gènes, ANOVA, calcul parallèle.*

Abstract

DNA chips (microarray) allow the study of the transcriptome by simultaneously observing the expression of several thousand genes in a cell or tissue. They play a major role in the understanding of genetic diseases, such as cancer and in developing new therapeutic molecules. The major downside to DNA microarray gene expression data is a dimensionality curse, where the number of features is huge compared to the number of samples. Gene selection methods select only those genes that play a major role in the detection of a disease and its adequate treatment. However, most of these methods are very time consuming. In addition, traditional microchip quality assessment and sequential preprocessing tools are not able to handle a large amount of data sets.

In this work, we proposed an ANOVA method for the selection of genes in parallel by exploiting the parallel computation toolbox in matlab (Parallel Computing Toolbox PCT), then we applied the kNN classification algorithm to assess the classification accuracy.

Keywords: *Bioinformatics, DNA microarray, Gene selection, ANOVA, Parallel computing.*

Liste des Tableaux

Liste des Figures

Table des matières

Introduction générale

Les microorganismes constituent la plus grande diversité du monde vivant. Ils jouent un rôle clef dans tous les processus biologiques grâce à leurs capacités d'adaptation et à la diversité de leurs capacités métaboliques.

Dans ce contexte, la technologie des puces à ADN est récemment devenue une technique populaire pour la bio-informatique, en particulier dans le diagnostic médical (qui est un élément très important dans le domaine de reconnaissance et traitement des maladies), la classification des maladies et la recherche de réglementations génétiques.

La technique observe les valeurs d'expression de milliers de gènes simultanément et analyse les niveaux d'expression pour le diagnostic médical et découvre des corrélations entre les gènes. Par exemple, les données sur les expressions génétiques des puces à ADN sont largement utilisées pour identifier les gènes candidats dans diverses études sur le cancer. Ces données contiennent généralement des milliers de gènes (parfois plus de 10000 gènes) et un petit nombre d'échantillons (généralement <100 échantillons).

Bien que de nombreux gènes soient exprimés dans une puce de microréseau, la plupart d'entre eux ne sont pas pertinents ou inutiles pour une analyse particulière car certains des gènes sont modulés de manière différentielle dans les tissus dans des conditions différentes et une quantité de bruit dans une micropuce. [1].

Dans une tâche de classification, ces conditions peuvent conduire à surapprentissage, ce qui signifie qu'un classificateur peut facilement montrer la performance d'une fonction de décision qui se comporte très bien avec les données d'apprentissage mais très mal dans les données de test. [2].

De plus, pour améliorer la validité du classificateur, il est nécessaire de réduire la dimensionnalité des données en sélectionnant un sous-ensemble de gènes pertinents pour

la classification. Ce problème consiste à rechercher un sous ensemble P, de variables (gènes) ayant la plus grande puissance de classification, parmi les N variables disponibles.

Dans ce travail, nous avons utilisé la technique d'Analyse de variances (ANOVA) qui se base sur le calcul du rapport statistique F, sa distribution (F-Distribution) et le p-value afin d'évaluer la potentialité des gènes et sélectionner les meilleurs qui peuvent mieux faire la distinction entre les différents classes. Nous avons exploité la boite à outils MATLAB concernant le calcul parallèle (PCT : Parallel Computing Toolbox) pour implémenter cette opération de sélection de gènes, dans un environnement parallèle multicœur, en réduisant le temps de calcul. La méthode proposée utilise l'algorithme de classification K plus proches voisins (kNN : K-Nearest Neighbor) pour évaluer la précision de la classification et en utilisant trois ensembles de données réels pour l'expérimentation.

Dans ce contexte, Ce manuscrit comporte cinq chapitres organisés comme suit :

- Dans le premier chapitre, nous allons présenter des notions élémentaires en biologie, la définition de la bio-informatique, ainsi que, les technologies biopuces, ses phases d'analyse, ses types et ses domaines d'application.
- Dans le second, nous allons exposer les approches de sélection des gènes, ses principes et processus de sélection, ainsi que les travaux existants sur leurs utilisations au domaine des puces à ADN.
- Ensuite dans le troisième nous allons présenter la méthode proposée pour la sélection des gènes, puis, la conception du système à réaliser et son architecture globale et détaillée.
- L'implémentation du système et les expérimentations et les résultats sont abordés dans le quatrième et le cinquième chapitre.

Le mémoire est terminé par une conclusion générale avec les perspectives envisagées.

Chapitre 1

Puces à ADN

1.1. Introduction

Les puces à ADN représentent un outil d'analyse de l'expression de gènes dans une cellule, elles permettent de visualiser simultanément le niveau d'expression de plusieurs milliers de gènes. Cette technologie offre des perspectives d'applications dans les domaines du diagnostic et pronostic médical. Elles présentent également l'avantage de pouvoir être de très haute densité et par conséquent sont susceptibles de recouvrir l'intégralité du génome d'un organisme.

Dans ce chapitre, nous allons définir cette technologie, ses phases d'analyse, ses types et ses domaines d'application.

1.2. Bio-informatique et puces à ADN

De nos jours, la technologie des puces à ADN a atteint une certaine maturité. Pour valider la qualité des cibles à hybrider plusieurs améliorations techniques et technologiques ont été faits. Ces améliorations permettent de travailler avec des quantités toujours plus faibles telles que des biopsies ou différents types de petites cellules.

A cause de la pléthore de données, les outils bio-informatiques ont été développés pour améliorer leur gestion, traitement, analyse et leur intégration.

1.2.1. Bio-informatique

La bio-informatique est la discipline de l'analyse de l'information biologique, en majorité sous la forme de séquences génétiques et de structures de protéines …C'est le décryptage de la « bio-information » « Computational Biology » [3].

Le terme bio-informatique est apparu pour la première fois dans une publication de Paulien Hogeweg et Ben Hesper, en référence à l'étude des processus d'information dans les systèmes biotiques.

Elle est constituée par l'ensemble des concepts et des techniques nécessaires à l'interprétation informatique de l'information biologique [4].

Ses principaux champs d'applications sont :

— la bio-informatique des réseaux,

— la bio-informatique structurale,

— la bio-informatique des séquences

Nous nous intéresserons particulièrement à la bio-informatique des séquences qui s'intéresse aux interactions entre gènes, protéines, cellules, organismes, en essayant d'analyser et de modéliser les comportements collectifs d'ensembles de briques élémentaires du Vivant.

1.2.2. Terminologie

a. La génomique

La génomique est une discipline récente de la biologie qui s'intéresse à l'étude exhaustive des génomes et en particulier de l'ensemble des gènes, de leur disposition sur les chromosomes, de leur séquence, de leur fonction et de leur rôle [5].

b. La cellule

C'est la plus petite unité structurale et fonctionnelle de tous les êtres vivants. Il existe des milliers de type de cellules différents par leur forme, leur taille, leur fonction et leur comportement, Chaque cellule d'un être humain comporte 23 paires de chromosomes.

Chez les organismes dits procaryotes tels que les bactéries, le matériel génétique n'est pas contenu dans un noyau mais est libre dans tout le cytoplasme de la cellule. Par contre, les organismes complexes comme les eucaryotes qui sont pluricellulaires, l'information génétique est localisée dans un noyau. L'homme, les animaux et les plantes sont des organismes eucaryotes. La Figure 1 présente la forme d'une cellule Eucaryote et une cellule Procaryote.

La plupart de leurs cellules sont capables de grossir et se diviser. Elles sont dotées d'un métabolisme, c'est à dire la capacité d'importer des nutriments et les convertir en molécules et en énergie [6].

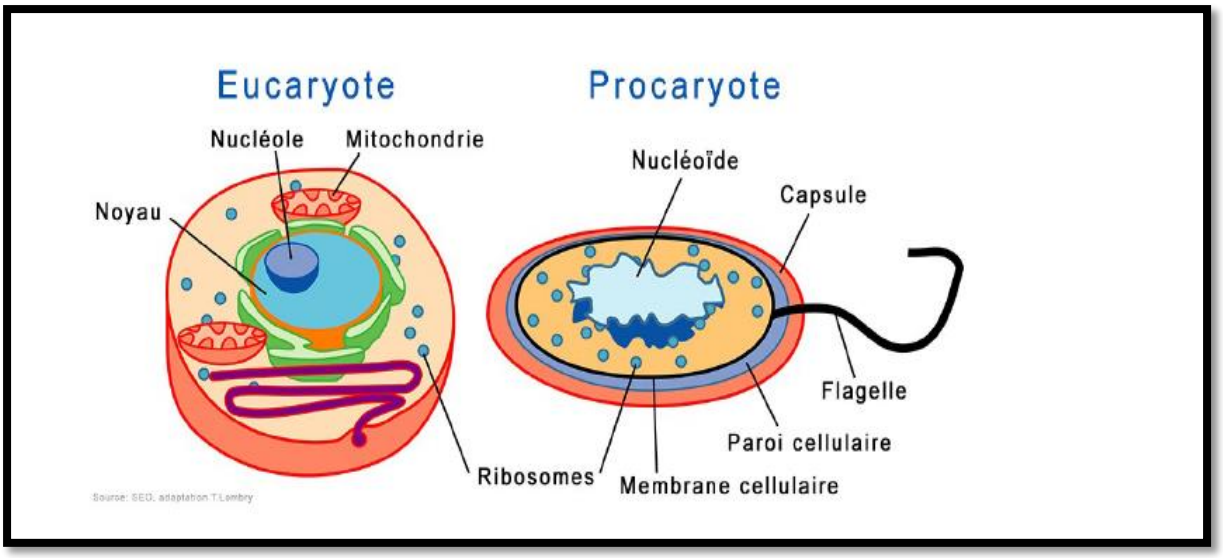


Figure 1: Schéma d'une cellule Eucaryote et une cellule Procaryote

c. Acide désoxyribonucléique ADN

L'acide désoxyribonucléique (ADN) est une molécule présente dans le noyau de la cellule qui joue un rôle central dans la vie cellulaire. Une molécule présente sous forme double hélice enroulée contient double brins (autour entre eux)

Macromolécule de millions/milliards d'atomes. C'est un motif identique tout le temps répète contenant :

- Des phosphates
- Des sucres (désoxyribose)
- Des bases azotées ; adénosine (A), cytosine (C), guanine (G), ou thymine (T).

Les couples A-T et G-C sont appelés bases complémentaires par lesquelles les deux brins vont s'associer, par des liaisons hydrogène (des sucres des oses, appelés désoxyriboses, et par des acides phosphoriques).

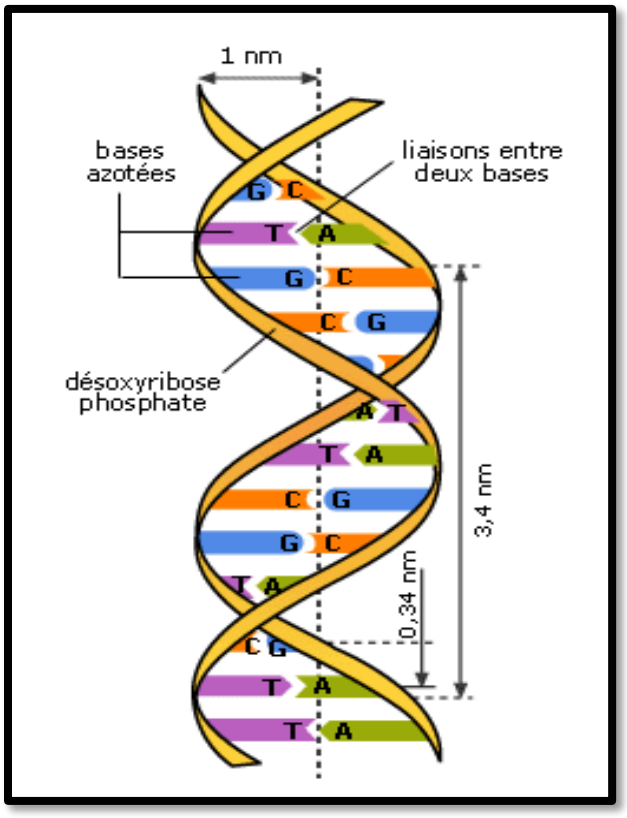


Figure 2: Structure d'une molécule d'ADN. [55]

d. Transcriptome

Un transcriptome est la gamme complète des molécules d'ARN (messagers, ribosomiques, de transfert et autres espèces d'ARN) issus de la transcription du génome.

En fonction de leurs besoins, les cellules utilisent à un instant donné une partie des gènes pour réaliser la synthèse des protéines nécessaires aux grandes fonctions cellulaires. Le passage du gène à la protéine se fait en deux grandes parties, la transcription et la traduction, à l'aide d'un agent essentiel l'ARNm, dit ARN messager (voir la Figure 3) [6].

Le gène est transcrit puis l'ARNm est conduit hors du noyau dans le cytoplasme ou il va servir de matrice pour la synthèse des protéines pour la traduction.

Les méthodes d'analyse du transcriptome les plus utilisées reposent sur la technologie des puces à ADN car elles permettent de visualiser simultanément le niveau d'expression

de plusieurs milliers de gènes dans un contexte physiologique ou pathologique particulier. [6]

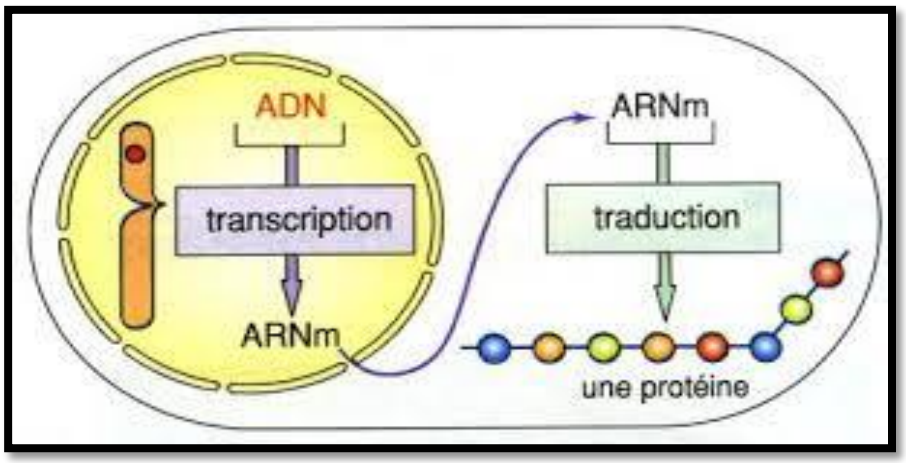


Figure 3: Synthèse de la protéine

1.3. Puces à ADN

1.3.1. Définition et principe

Les puces à ADN ont d'abord été conçues sur de grandes membranes poreuses en nylon ou macroarrays.

Toutefois, dans une définition technique plus exacte, une puce ADN est constituée de fragments d'ADN appelées puces immobilisés de manière ordonnée sur un support solide généralement fait de verre, de silicium ou bien de membrane en nylon.

Chaque emplacement de séquence est soigneusement repéré : la position (x_i, y_i) correspond au gène i. Un emplacement est souvent appelé spot ou sonde [7].

Le fonctionnement de la technologie des puces à ADN ou biopuces repose sur le principe de complémentarité des brins de la double hélice d'ADN et la propriété d'hybridation entre deux séquences complémentaires d'acides nucléiques [6].

Les hybridations se font entre des sondes nucléotidiques ordonnées sur un support solide et des cibles marquées par une substance radioactive ou fluorescente permettent de

quantifier l'ensemble des cibles qu'il contient, présentes dans un mélange complexe [3]. Les sondes et les cibles représentent respectivement les gènes du transcriptome à analyser.

La fabrication d'une puce à ADN se décompose en trois étapes : la production des sondes (fragments courts d'une séquence d'ADN connus) et leur dépôt sur le support la production et le marquage des cibles (fragments inconnus d'ADN que l'on cherche à identifier), enfin l'hybridation des sondes avec les cibles. Ces différentes étapes constituent [8], les étapes de base pour la fabrication de toutes les puces, indépendamment de la technologie utilisée. Le schéma suivant donne une généralisation de cette technique de fabrication [8].

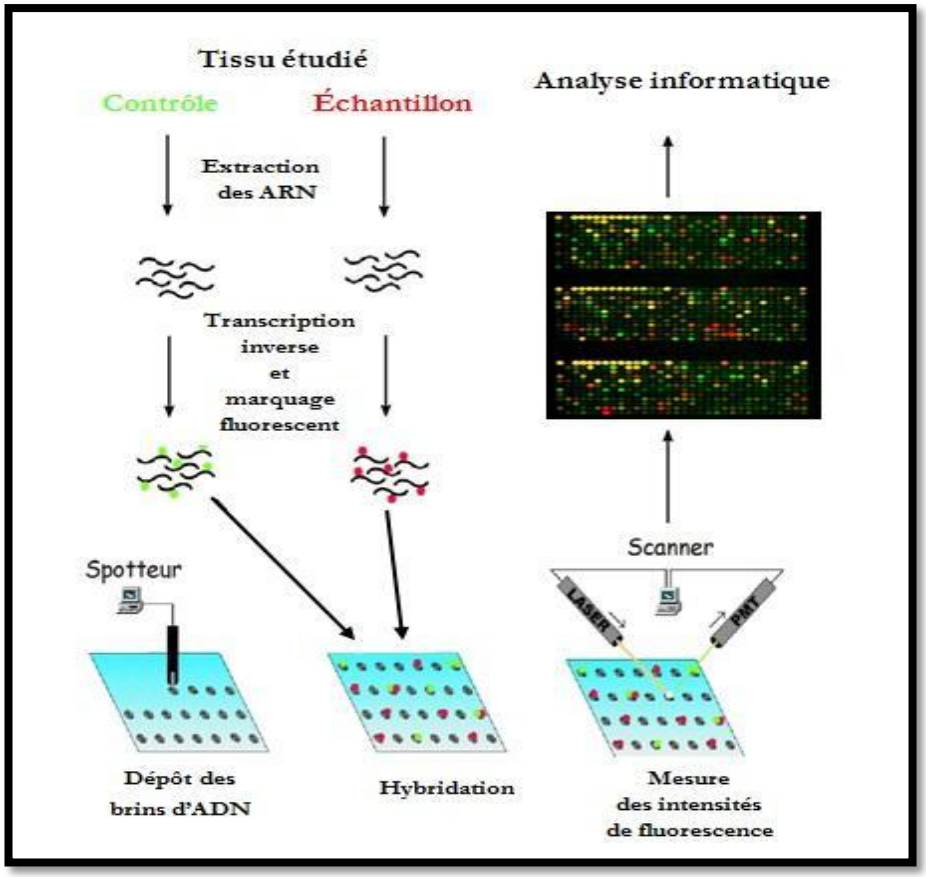


Figure 4: Processus de fabrication d'une puce à ADN [4]

1.3.2. Phase d'analyse par puces à ADN

Les différentes phases d'une analyse par puces ADN sont indiquées dans la figure ci-dessus :

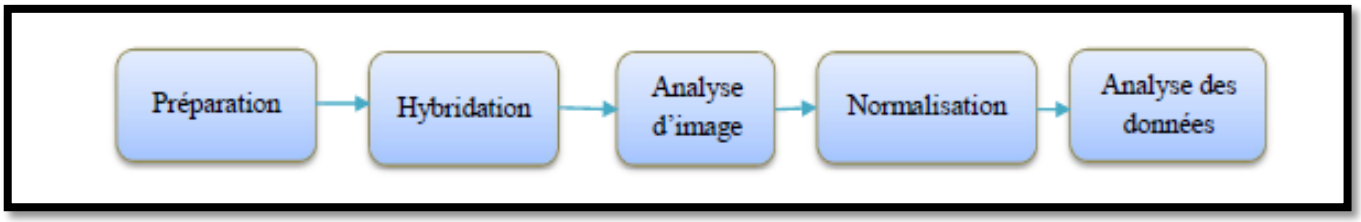


Figure 5: Les phases d'une analyse par puce à ADN [6]

1.3.2.1. La préparation des cibles et l'hybridation

Pour comparer les niveaux d'expression dans deux échantillons biologiques ou deux conditions (référence et pathologique), la première étape consiste en la préparation du génome exprimé dans ces deux échantillons. Il s'agit d'extraire les ARNm d'un échantillon biologique à analyser et la qualité de l'extraction est bien sûr primordiale pour la réussite de l'hybridation qui va suivre [6].

Les échantillons sont marqués par des substances fluorescentes (*Cy3* et *Cy5*), c'est-à-dire qu'une culture est marquée avec un fluorochrome vert, tandis que la seconde est marquée avec un fluorochrome rouge.

L'hybridation est ensuite réalisée sur une seule puce (simple marquage) ou sur deux puces (double marquage : un échantillon sur chaque puce). Les ADN marqués sont mélangés (cible) et placés sur la puce à ADN (sonde) [6].

Ce processus d'hybridation est réalisé dans une station fluidique (four) pour favoriser les liaisons entre séquences complémentaires [9].

La durée oscille entre 10 à 17 heures en milieu liquide à 60 degrés, en fait à cette température un fragment d'ADN simple brin ou d'ARN messager reconnaît son brin complémentaire (ADNc) parmi des milliers d'autres pour former un ADN de double brin

(duplex ou double hélice) [7].L'étape de nettoyage ou lavage des puces a pour but d'ôter de la puce des cibles non hybridées. La puce est lavée à plusieurs reprises afin qu'il ne reste sur la lame que les brins parfaitement appariés.

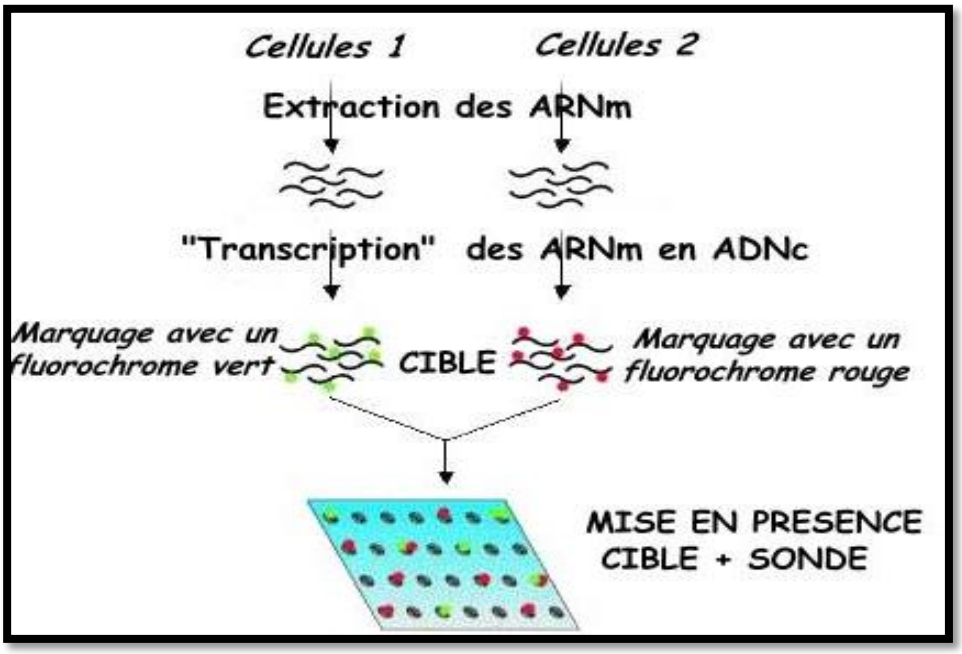


Figure 6: Schéma de la préparation de la cible et de son hybridation avec la sonde [4]

1.3.2.2. Acquisition et analyse des images

Suite à l'hybridation, une étape de lecture de la puce permet de repérer les sondes ayant réagi avec l'échantillon testé. Cette lecture est une étape clé [10] En effet, sa qualité conditionne de façon importante la précision des données et donc, la pertinence des interprétations.

L'obtention des images est réalisée par lecture des puces sur des scanners de haute précision. Dans le cas du marquage avec les deux fluorochromes, on obtient une image dont la couleur des spots va du rouge au vert :

- Un spot de couleur verte indique un gène dont le niveau d'expression est plus élevé dans l'échantillon marqué avec le *Cy3* que celui marqué avec le *Cy5*, et inversement pour la couleur rouge [4].
- La couleur jaune indique que le gène est exprimé de manière identique dans les deux échantillons tandis que la couleur noire indique l'absence de signal [4].

Après lavage et acquisition, l'image d'hybridation va être traitée par des logiciels d'analyse qui permettent de mesurer la fluorescence de chaque spot sur la lame (estimant les niveaux d'expression pour chacun des gènes présents sur la puce), mais aussi de relier chaque sonde à l'annotation correspondante (nom de gêne, numéro de l'ADNc utilisé, séquence de l'oligonucléotide, etc.) puis calculer l'intensité de chaque spot [7].

1.3.2.3. Transformation des données

Les données d'intensité sont rarement manipulées sans transformation et la transformation la plus couramment employée est celle qui utilise le logarithme à base deux. Il existe plusieurs raisons pour justifier cette transformation. D'une part, la variation du logarithme des intensités est moins dépendante de la grandeur des intensités, et d'autre part, cette transformation permet de se rapprocher d'une distribution symétrique et d'obtenir une meilleure dispersion avec moins de valeurs extrêmes. L'autre transformation qu'elle peut être appliquée sur ces intensités c'est la normalisation.

Elle consiste à éliminer les différences entre les différentes puces liées aux variations de quantité de départ, aux biais de marquage ou d'hybridation et aux variations du bruit de fond [7] afin de dresser pour chaque échantillon (une tumeur, par exemple) un véritable portrait moléculaire qui pourra alors être comparé à celui d'autres échantillons [11].

Présentation des données de puces à ADN : après les transformations décrites ci-dessus, les données recueillies pour l'étude d'un problème donné sont regroupées sous forme de matrice avec une ligne par gène et une colonne par échantillon.Chaque valeur de mij est la mesure du niveau d'expression du $i-ème$ gène dans le $j-ème$ échantillon, où $i = 1, \ldots, M$ et $j = 1, \ldots, N$ (11).

Gène id	Echantillon 1	Echantillon 2		…	Echantillon *M*
Gène 1				…	
Gène 2				…	
Gène 3				…	
⋮				⋮ ⋮ ⋮	
Gène *N*	m_{M1}	m_{M2}		…	m_{MN}

Tableau 1:Matrice d'expression des gènes

1.3.3. Types des puces à ADN

Le terme " puces à ADN " est un terme générique. Il existe actuellement 2 procédés majeurs de fabrications de puces à ADN ce qui permet de distinguer différents types de puces. Les deux types de puces à ADN qui dominent le marché sont :

a) Les puces à ADNc de la technologie Agilent

Ont été les premières puces à être développées, qui fonctionnent avec des micros points contenant des fragments d'ADN sur un support de verre.

En général, deux échantillons d'ARN (sous forme d'ADNc obtenus par transcription inverse) sont Co hybridés sur la puce à ADNc. Les deux échantillons marqués par un fluorochrome différent (*Cy-3* vert ou *Cy-5* rouge) s'hybrident simultanément avec les molécules complémentaires sur la puce. La puce est lue par un scanner afin de mesurer l'intensité du signal lumineux mesurée aux deux longueurs d'ondes correspondant aux différents fluorochromes [12].

Le rapport de fluorescence rouge/vert est ainsi déterminé. Il permet de comparer les taux d'expression relatifs de chacun des gènes pour les deux échantillons d 'ADNc. Un excès du gène X dans l'échantillon marqué en rouge produira un signal rouge au point représentant le gène, un excès du gène Y dans l'échantillon marqué en vert produira un signal vert ; enfin, une expression équivalente du gène Z dans les deux échantillons produira un signal jaune [12]. La société Agilent est l'une des plus grandes industries qui commercialise ce type [4].

b) Les puces à oligonucléotides

Reposent sur le principe de synthèse in situ de milliers de séquences distinctes d'oligonucléotides. Les sondes sont des oligonucléotides synthétisés par une technique de photolithographie. Cette technique consiste à diriger une lumière sur des sites spécifiques de la puce ce qui active la réaction d'oligosynthèse. On ajoute également des oligonucléotides dont la séquence varie pour une seule base pour confirmer que le signal obtenu pour chacun des gènes est bien spécifique. On hybride une seule expérience par puce et l'intensité de fluorescence mesurée par un scanner permet une mesure de l'abondance relative de chacun des ARNm présent dans l'échantillon biologique étudié [13]. La société Affymetrix est l'unique détenteur de cette technologie [14].

1.3.4. Domaine d'application

Les puces à ADN permettent des tests plus rapides, plus sensibles et plus spécifiques. Elles sont utiles dans divers domaines très important tels que l'environnement, pharmaceutique, les diagnostics médicaux les expertises médico-légales et bien d'autres domaines. [15]

1.3.4.1. Diagnostics médicaux

La puce à ADN a encore un grand rôle à jouer dans une autre application des polymorphismes et de la détection banalisée de ceux-ci. Cela pourrait prévenir les prédispositions qu'a un patient à diverses maladies génétiques [6].

La commercialisation de ces systèmes de petite taille, voire même portables pourrait être utilisée en hôpital et même par les médecins traitants. On attend que des labo-puces puissent faire en un temps réel et continu l'analyse de certains signes vitaux afin d'en prescrire immédiatement le traitement adéquat (par exemple le taux de glucose sanguin pour les diabétiques).

1.3.4.2. Expertise médico-légale

Le but est l'identification d'un corps humain dans le cadre d'enquêtes policières ou judiciaires. Les analyses sur le terrain étant très souvent complexes ainsi que la confidentialité et le respect de la procédure judiciaire assez lourdes, il sera souhaitable d'avoir sur les lieux d'enquêtes des systèmes portables d'analyse de l'ADN, permettant ainsi d'affiner la recherche d'échantillons.

1.3.4.3. L'environnement

Les secteurs de la défense et de l'environnement font partie des diverses applications des puces à ADN, notamment pour la détection rapide et à bas coût de substances organiques, principalement des agents pathogènes dilués dans l'environnement [6].

1.3.5. Banque de données d'expression des gènes

Aujourd'hui, Les banques de données sont devenues indispensables pour sauvegarder et structurer les informations issues des expériences de biologie et plus particulièrement des données générées par les différentes technologies de puces à ADN [3].

Parmi les banques de données publiques, les banques données d'expression de gènes sont particulièrement importantes et intéressantes en terme de partage les données d'expression de gènes au niveau de la communauté scientifique internationale.

La MGED (Microarray Gene Expression Data Society) a initié le développement et la promotion de standard pour le stockage et le partage des données de puces à ADN basées sur l'expression des gènes et du résultat des études effectuées sur ces données [4].

L'une de leur priorité est le respect par les biologistes du standard international MIAME (Minimum Information About a Microarray Experiment) [3]. MIAME est un standard conceptuel décrivant l'information minimum requise pour une interprétation et une vérification propre des expériences des puces à ADN [6].

Les deux principales banques de données généralistes pour le dépôt des données d'expression de gènes sont :

- **ArrayExpress** : Une base de données publique d'expérience de puce à ADN et de profils d'expression des gènes établie en 2002 à l'European Bioinformatics Institute (EBI).

- **GEO (Gene Expression Omnibus) :** Un entrepôt de données génomiques fonctionnelles publiques prenant en charge les soumissions de données conformes à MIAME. Il été établi en 2000 au National Center for Biotechnology Information (NCBI). Des outils sont fournis pour aider les utilisateurs à interroger et télécharger des expériences et des profils d'expressions génétiques organisés [16].

1.4. Conclusion

La technologie des puces à ADN connaît actuellement une croissance exceptionnelle et suscite un intérêt considérable dans la communauté scientifique en raison de son potentiel à mesurer simultanément le niveau d'expression d'un grand nombre de gènes dans des échantillons de tissus.

Dans une tâche de classification, ces conditions peuvent conduire à sur-apprentissage, pour améliorer la validité du classificateur, il est nécessaire de réduire la dimensionnalité des données en sélectionnant un sous-ensemble de gènes pertinents pour la classification. Dans le prochain chapitre, nous allons définir la sélection, en détaillant ses aspects fondamentaux

Chapitre 2

Sélection d’attributs

Chapitre 2. Sélection d'attributs

2.1. Introduction

Le problème de la sélection des caractéristiques ou bien réduire la dimensionnalité d'un ensemble de données, est depuis longtemps, un sujet de recherche actif. Plusieurs paramètres peuvent influer sur les performances d'un système de classification. Les caractéristiques sélectionnées à partir des entités peuvent être considérées parmi les paramètres les plus importants.

Dans ce chapitre nous nous intéressons au problème de la sélection d'attributs, son processus et montrer la nécessité de son utilisation dans le domaine Bio-puces et plus précisément, à l'utilisation d'approches bio-informatique pour effectuer la sélection de caractéristiques.

2.2. Sélection d'attributs

La réduction de la dimensionnalité via la sélection d'attributs est l'une des étapes les plus importantes dans le traitement de données. Son but est de sélectionner ou d'extraire un sous-ensemble optimal de caractéristiques pertinentes pour un critère fixé auparavant. Les principaux objectifs de la réduction de dimension sont [17] :

- Faciliter la visualisation et la compréhension des données.
- Réduire l'espace de stockage nécessaire.
- Réduire le temps d'apprentissage et d'utilisation.
- Identifier les facteurs pertinents.
- Améliorer la précision du module de classification.

2.2.1. Définition

La sélection d'attributs ou de caractéristiques (Feature Selection ou FS) est une procédure permettant de choisir un sous-ensemble minimum de caractéristiques à partir

d'un ensemble original de sorte que l'espace de caractéristiques soit réduit de façon optimale selon certains critères d'évaluation [17].

Dans la littérature, le problème de sélection de caractéristiques a été généralement défini comme suit :

Soit *E = {E1; E2 ...EN}* un ensemble de caractéristiques de taille *N* où *N* représente le nombre total de caractéristiques étudiées. Soit F une fonction qui permet d'évaluer un sous-ensemble de caractéristiques. Nous supposons que la plus grande valeur de F soit obtenue pour le meilleur sous-ensemble de caractéristiques. L'objectif de la sélection est de trouver un sous-ensemble $\acute{E}(\acute{E} \subseteq E)$ de taille $\acute{N}(\acute{N} \subseteq N)$ tel que :

$$F(\acute{E}) = max\, F(Z), Z \subseteq E$$

Où $|Z| = \acute{N}$, et $\acute{N}$ est, soit un nombre prédéfini par l'utilisateur ou soit contrôlé par une des méthodes de génération de sous-ensembles [18].

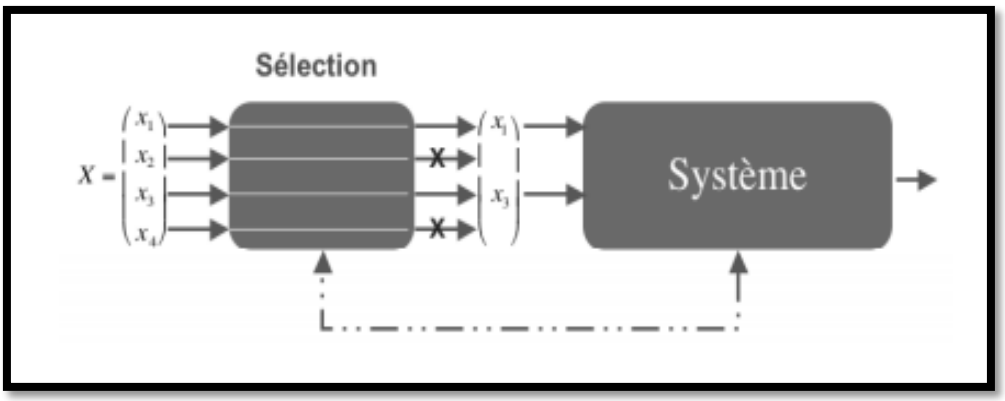


Figure 7: Principe de la sélection de variables (18)

2.2.2. Processus de la sélection d'attributs

Les différentes méthodes proposées dans la littérature pour la sélection d'attributs peuvent être décrites par un schéma général dans lequel on trouve les éléments clés suivants : [17]

1. Une procédure de génération de sous-ensembles candidats qui détermine l'exploration de l'espace de recherche.
2. Une fonction d'évaluation donnant la qualité des sous-ensembles candidats.
3. Une condition d'arrêt.
4. Un processus de validation pour vérifier si l'objectif souhaité est atteint.

Nous détaillons ci-dessous les étapes importantes de ce schéma. Ce processus peut être schématisé comme suit:

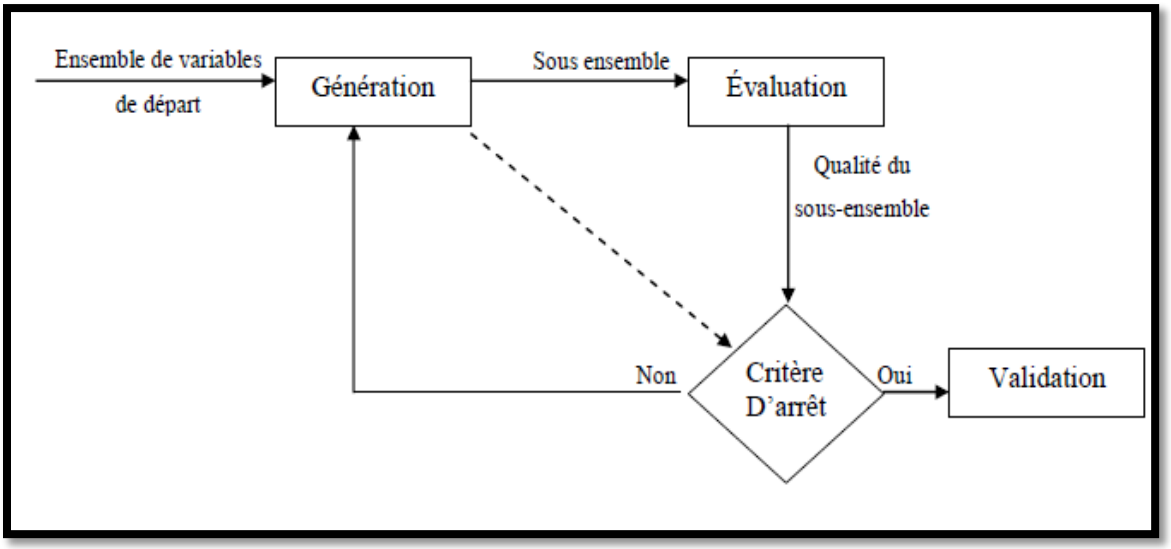


Figure 8. Processus de sélection de caractéristiques [17]

2.2.2.1. La procédure de génération

La procédure de génération n'est qu'une procédure de recherche. Elle génère un sous-ensemble d'attributs qui sera par la suite évalué selon un critère bien déterminé. La méthode de sélection procède par ajouts successifs (Forward Selection) ou élimination successives (Bakward Selection). Elle peut être initialisée soit par un sous-ensemble vide

ou tout l'ensemble initial des attributs ou simplement avec un sous-ensemble d'attributs aléatoire [17].

Les Méthodes de génération peuvent être classées en trois approches :

a) Génération complète

Elle effectue une recherche exhaustive pour trouver l'ensemble optimal d'attributs sur tout l'espace des solutions possibles, qui est de l'ordre O (2^N). Plusieurs procédures de recherche heuristique sont proposées afin de réduire le nombre de sous-espace à évaluer. [19]. Le problème majeur de cette approche est que le nombre de combinaisons croît exponentiellement en fonction du nombre de caractéristiques [18].

b) Génération séquentielle

Cette catégorie regroupe les algorithmes itératifs pour lesquels chaque itération permet de sélectionner ou rejeter une ou plusieurs caractéristiques. Les algorithmes avec une génération séquentielle sont simples à implémenter et rapides dans la production des résultats, l'espace de recherche utilisé est de l'ordre O(N2). Parmi les techniques utilisées, on trouve [17]:

- L'approche de type Forward ou Ascendante : La recherche peut commencer avec un ensemble vide de caractéristiques et successivement ajouter des caractéristiques.
- L'approche de type Backward ou Descendante : Inversement, la recherche peut commencer avec toutes les caractéristiques et successivement en supprimer.

Dans ces algorithmes, on abandonne l'exhaustivité et on risque ainsi de perdre les sous-ensembles optimaux.

c) Génération aléatoire

Bien que l'espace de la recherche est de l'ordre O (2^N), cette procédure n'évalue pas toutes les solutions possible dans cette espace. Un nombre maximal d'itération est imposé afin de limiter le temps de calcul.

2.2.2.2. Fonction d'évaluation

Les méthodes utilisées pour évaluer un sous-ensemble de caractéristiques dans les algorithmes de sélection peuvent être classées en trois catégories principales : "filter", "wrapper" et "embedded".

a) Filter Approch

Son principe consiste à évaluer chaque attribut pour lui assigner un score de pertinence. Ces méthodes reposent généralement sur le choix d'un seuil pour le critère de pertinence choisi ou d'un nombre d'attributs à sélectionner qui doit être fixé a priori. Le choix de ces paramètres n'est pas facile à réaliser [19].

Cette méthode est considérée, d'avantage comme une étape de prétraitement (filtrage) avant la phase d'apprentissage. En d'autres termes, l'évaluation se fait généralement indépendamment d'un classificateur. Les méthodes qui se basent sur ce modèle pour l'évaluation des caractéristiques, utilisent souvent une approche heuristique comme stratégie de recherche. La procédure du modèle "fiter" est illustrée par la figure 8.

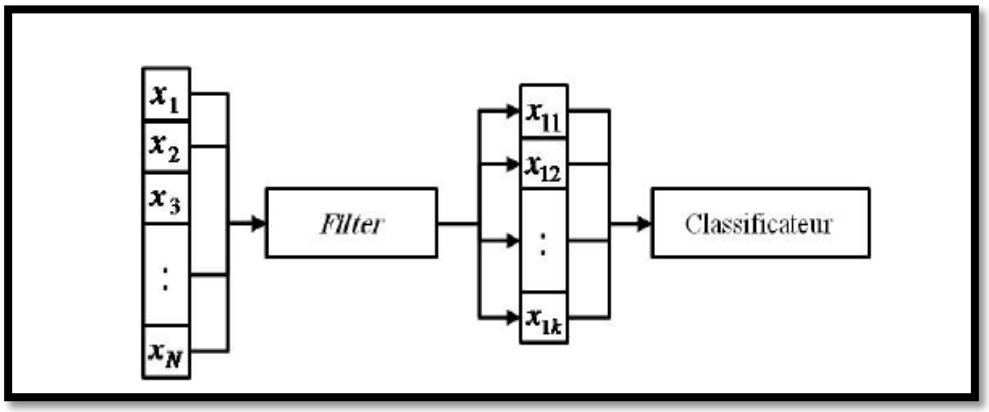


Figure 8: Processus du modèle Filtre [18]

Ces méthodes sont rapides, plus générales et moins coûteuses en temps de calcul, ce qui leur permet d'opérer plus facilement avec des bases de données de très grandes dimensions.

Cependant, comme elles sont indépendantes de l'étape de classification, elles ne permettent pas de garantir que le meilleur taux de classification soit obtenu dans l'espace retenu [20].

b) Wrapper Approch

Le principal inconvénient des approches "Filter" est le fait qu'elles ignorent l'influence des caractéristiques sélectionnées sur la performance du classificateur à utiliser par la suite.

Cette approche utilise l'algorithme d'apprentissage comme une fonction d'évaluation, elle définit donc la pertinence des attributs par l'intermédiaire d'une prédiction de la performance du système final [7]. La procédure du modèle "wrapper" est illustrée par la figure 9.

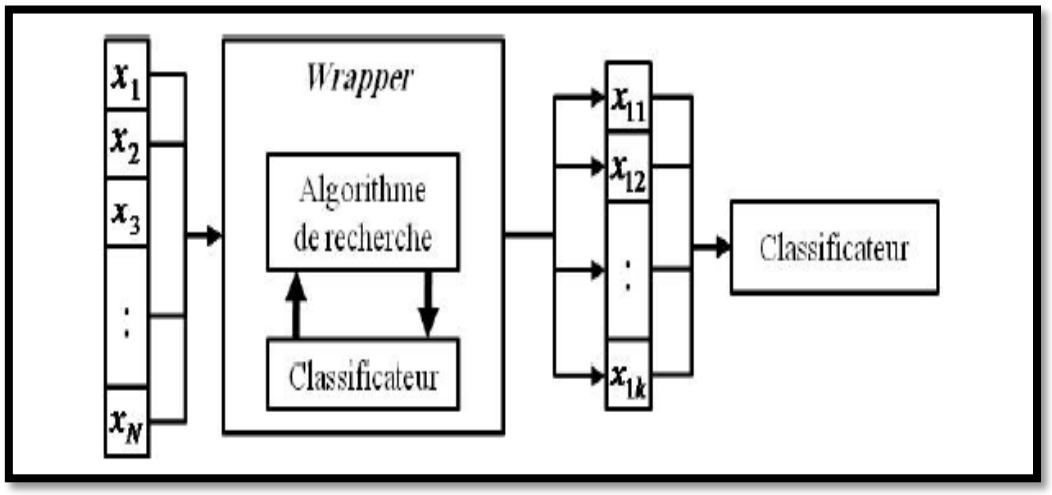


Figure 9: Processus du modèles Wrapper [18]

Les sous-ensembles de caractéristiques sélectionnées par cette méthode sont bien adaptés à l'algorithme de classification utilisé, mais ils ne sont pas forcément valides si on change le classificateur.

La complexité de l'algorithme d'apprentissage rend les méthodes "wrapper" très coûteuses en temps de calcul.

Le problème de la complexité de cette technique rend impossible l'utilisation d'une stratégie de recherche exhaustive. Par conséquent, des méthodes de recherche heuristiques ou aléatoires peuvent être utilisées [18].

En général, pour diminuer le temps de calcul et pour éviter les problèmes de sur apprentissage, le mécanisme de validation croisée est fréquemment utilisé.

c) Embedded Approch

A la différence des méthodes "wrapper" et "fiter", les méthodes "embedded" (appelées aussi méthodes intégrées) incorporent la sélection de variables lors du processus d'apprentissage [18].

Dans les méthodes de sélection de type "wrapper", la base d'apprentissage est divisée en deux parties : une base d'apprentissage et une base de validation pour valider le sous-ensemble de caractéristiques sélectionné. En revanche, les méthodes intégrées peuvent se servir de tous les exemples d'apprentissage pour établir le système. Cela constitue un avantage qui peut améliorer les résultats.

L'avantage de ces méthodes est que le processus de recherche est guidé par des informations intéressantes fournies par le classifieur, ce qui rend ces méthodes plus efficaces que les méthodes enveloppes [19]. Un autre avantage de ces méthodes est leur rapidité par rapport aux approches "Wrapper" parce qu'elles évitent que le classificateur recommence de zéro pour chaque sous-ensemble de caractéristiques [18].

Différentes mesures d'évaluation ont été proposées pour évaluer un attribut ou un sous-ensemble d'attributs dans un contexte de sélection. Elles peuvent être classées en cinq approches distinctes [18]:

1. Mesure de distance

Les mesures de distance sont aussi nommées mesures de séparabilité, divergence ou de discrimination, comme par exemple la distance Euclidienne [19]. Un attribut ou un sous-ensemble d'attributs est sélectionné s'il permet une meilleure séparabilité et cohérence des

classes [20]. Une variable *X* est préférée à *Y*, si *X* introduit une plus grande différence, entre les probabilités conditionnelles des deux classes, que *Y* [17].

2. Mesure d'information

Permet d'estimer le gain d'information d'une caractéristique. Le gain d'information de la caractéristique *X* est défini comme la différence entre l'incertitude a priori et celle a posteriori, c'est-à-dire avant et après la sélection de la caractéristique *X*. La caractéristique *X* est préférée à *Y*, si le gain d'information de *X* est plus grand de celui de *Y* [17]

3. Mesure de dépendance

C'est la mesure de corrélation qui peut qualifier la capacité de prédire la valeur d'une variable depuis une autre variable. Si la corrélation entre un attribut *X* et une classe *C* est supérieure à celle entre un attribut *Y* et la classe *C*, alors *X* est préféré à *Y* [19].

4. Mesure d'erreur de classification

L'attribut ou les sous-ensembles d'attributs considérés sont évalués en fonction de la qualité de la classification obtenue en utilisant ces attributs. Le sous-ensemble d'attributs le plus discriminant est celui pour lequel le taux d'erreur de classification est le plus faible [20].

5. Mesure de consistance

Les mesures de consistance cherchent à évaluer si l'attribut (ou le sous-ensemble d'attributs) étudié contient les informations nécessaires à la discrimination des classes. [20]

2.2.2.3. Critère d'arrêt

Le nombre optimal d'attributs n'étant pas connu a priori, il sera fixé grâce à un critère d'arrêt du processus de sélection.

C'est un choix souvent défini en fonction de la procédure de génération [20] et/ou de la fonction d'évaluation. Les critères d'arrêt les plus fréquents sont [17]:

- **Les critères d'arrêt basés sur la procédure de génération**
  - Le nombre de caractéristiques sélectionnées est égal à un nombre prédéfini.
  - Un nombre prédéfini d'itérations est atteint.
- **Les critères d'arrêt basés sur la fonction d'évaluation**
  - L'ajout (ou suppression) d'une caractéristique ne produit pas un meilleur sous ensemble.
  - Un sous-ensemble optimal de caractéristiques est obtenu à partir de certaines fonctions d'évaluation.

Par exemple, lorsque les méthodes par filtre sont utilisées, le critère d'arrêt couramment utilisé est basé sur l'ordre des caractéristiques, rangées selon certains scores de pertinence et une fois les caractéristiques ordonnées, celles qui ont les scores les plus élèves seront choisies et utilisées par un classificateur [18].

D'autre part, si on utilise l'approche "enveloppe" ou l'approche "hybride", les taux de bonne classification obtenus par les différents sous-espaces sont comparés pour mesurer le gain d'information. On peut ainsi décider d'arrêter la procédure de sélection dès que ce taux diminue ou alors dès qu'il atteint un certain seuil [17].

2.2.2.4. Procédure de validation

La validation ne fait pas partie de la procédure de sélection d'attributs mais elle permet de tester la validité du sous-ensemble d'attributs sélectionnés en effectuant plusieurs tests sur des exemples de données générées artificiellement et/ou sur des données réelles [20].

L'ensemble des données est généralement divisé en deux sous-ensembles distincts : le sous-ensemble d'apprentissage constitué des prototypes des classes (données avec leurs labels) et le sous-ensemble de test dont on ne connait pas les labels de classes de ses données. Selon la répartition des données entre ces deux sous-ensembles, il existe différentes approches de validation, nous citons [7]:

- **La méthode Holdout :** les données sont divisées en deux sous-ensembles : le sous-ensemble d'apprentissage et le sous-ensemble de test dans des proportions $\frac{1}{2}, \frac{1}{2}$ pour chacun de ses deux sous-ensembles ou $\frac{2}{3}$ pour l'ensemble d'apprentissage et $\frac{1}{3}$ pour l'ensemble de test.
- **La méthode de resubstitution :** l'ensemble d'apprentissage est utilisé comme ensemble de test.
- **La méthode V-validation croisée :** l'ensemble des données est partitionné en V parties de tailles à peu près égales. Nous réalisons ainsi V fois la procédure de validation et à chaque fois une des parties constitue l'ensemble test et les V-1 parties restantes sont réunies pour former l'ensemble d'apprentissage.

2.3. Sélection d'attributs biopuces

La motivation pour l'application des techniques de la sélection d'attributs dans le domaine de la bioinformatique a dépassé le fait d'être un exemple illustratif pour devenir une véritable condition préalable à la construction des modèles. Deux questions principales se posent comme des problèmes communs dans le domaine de la bioinformatique : la grande dimension des entrées, et la petite taille des échantillons.

Pour éviter cette « malédiction de la dimensionnalité », la sélection des gènes joue un rôle crucial dans l'analyse des puces à ADN.

La sélection de gènes pertinents pour la classification des échantillons est une tâche courante dans la plupart des études d'expression génique, où les chercheurs tentent d'identifier le plus petit ensemble possible de gènes capables d'atteindre de bonnes performances prédictives (par exemple, utilisation diagnostique en pratique clinique [9].

Il existe plusieurs raisons pour effectuer la sélection des gènes, nous citons quelques une selon les points de vue [21] :

a) **L'analyse discriminante :** parmi l'ensemble important de gènes, beaucoup pourraient être non pertinents, insignifiants ou redondants à un problème discriminant spécifique. Des études ont montré qu'un petit sous-ensemble de gènes pourrait être suffisant pour un problème biologique particulier. La sélection de gènes réduit le volume de données et facilite la manipulation et l'analyse des données de biopuces.
b) **Les biologistes :** l'importance de la sélection des gènes réside dans sa contribution à la compréhension des maladies et des fonctions de gènes particuliers, et à la conception d'expériences de biopuces à des fins de diagnostic clinique et de pronostic.

Dans le contexte de l'analyse des données d'expression génique, plusieurs approches de sélection de gènes ont été publiées.

— Mukesh Kumar, Nitish Kumar Rath, Amitav Swain and Santanu Kumar Rath ont proposés une méthode statistique ANOVA basée sur MapReduce pour la sélection des puces à ADN et le classificateur K-Nearest Neighbor (*K-NN*) basé sur MapReduce est également proposé pour classer les données de microréseau. Ces algorithmes sont mis en œuvre avec succès sur le framework Hadoop et une analyse comparative est effectuée à l'aide de divers ensembles de données [22].

— A. K. M. Tauhidul Islam et al.14 ont proposé une méthode de sélection de gêne parallèle basée sur MapReduce, qui utilise techniques d'échantillonnage pour réduire les gènes non pertinents en utilisant le rapport somme des carrés (BW) entre groupes et intra-groupes. Le BWratio indique les variances entre les valeurs d'expression génique. Après la sélection du gène, il applique la technique MRkNN pour exécuter plusieurs kNN en parallèle à l'aide du modèle de programmation MapReduce. Enfin, l'efficacité de la méthode est vérifiée par des expériences approfondies utilisant plusieurs ensembles de données réels et synthétiques [1].

— Li et al. [23] ont proposé une méthode hybride de l'algorithme génétique et *K* plus proche voisin, pour l'évaluation des gènes et la classification des échantillons. L'idée principale de GA / KNN est de trouver un grand nombre de sous-ensembles optimaux ou quasi-optimaux et d'évaluer l'importance des gènes pour la classification en examinant la fréquence des appartenances aux gènes dans ces sous-ensembles.

— Parmi les méthodes statistiques, *SVST* [24] a introduit le concept d'élagage des échantillons pour éliminer les échantillons moins pertinents et aberrants et a appliqué *SVM* pour trouver des gènes biologiquement pertinents. Afin d'améliorer la précision de la technique de classification, la méthode supprime les échantillons moins pertinents qui ne sont pas localisés sur des vecteurs supports.

2.4. Conclusion

Dans ce chapitre, nous avons présenté les aspects fondamentaux du domaine de sélection d'attributs qu'il devient de plus en plus un sujet de recherche plus actif et indispensable en raison de la multiplication des données. Dans la section suivante, on va introduire notre méthode utilisée pour la réduction de dimensionnalité des données des puces à ADN basée sur la sélection des gènes les plus pertinents.

Chapitre 3
Solution envisagée et conception

Chapitre 3. Solution envisagée et conception

3.1. Introduction

Dans le chapitre précédent, nous avons présentés les différentes méthodes de sélection des gènes, et notre problème posé de sorte que le nombre de variables (gènes) est très élevé.

En outre, en raison de l'évolution récente de la technologie des puces de puces à ADN, dont ces expériences peuvent traiter plus des milliers des gènes simultanément dans une seule puce et peut générer grande quantité de données de puces à ADN à faible coût. Le calcul de haute performance est devenu extrêmement important pour analyser la grande quantité de ces données biologiques.

Nous avons concentré notre étude sur la sélection parallèle des variables pour la construction d'un bon prédicteur avec un nombre minimal des variables les plus pertinents.

Dans ce chapitre nous présentons notre méthode que nous avons choisie pour résoudre notre problème de sélection des gènes, puis la conception de notre travail. D'abord la conception générale puis la conception détaillée en spécifiant les différents éléments composant notre travail et précisant son fonctionnement.

3.2. Analyse de la variance ANOVA

ANOVA (ANalysis Of VAriance) est une méthode statistique qui signifie l'analyse de la variance. Elle est l'extension du t-test et z-test. Avant l'utilisation de l'ANOVA, le t-test et le z-test étaient couramment utilisés. Mais le problème avec le t-test est qu'il ne peut pas être appliqué à plus de deux groupes. En 1918, Ronald Fisher a développé un test appelé l'analyse de la variance. Ce test est également appelé analyse de la variance de Fisher, qui est utilisée pour faire l'analyse de la variance entre et au sein des groupes lorsque les groupes sont plus de deux [25].

Les techniques dites d'analyse de variance sont des outils entrant dans le cadre général du modèle linéaire et où une variable quantitative est expliquée par une ou plusieurs variables qualitatives [26]. L'analyse de la variance (ANOVA : ANalysis Of VAriance) a pour objectif d'étudier l'influence d'un ou plusieurs facteurs sur une variable quantitative. C'est la comparaison des moyennes empiriques de la variable quantitative observées pour différentes catégories d'unités statistiques. Ces catégories sont définies par l'observation des variables qualitatives ou *facteurs* prenant différentes modalités ou encore de variables quantitatives découpées en classes ou niveaux [26]. Une combinaison de niveaux définit une cellule, groupe ou traitement.

Il s'agit de comparer la variance intergroupe (entre les différents groupes : écart des moyennes des groupes à la moyenne totale) à la variance intragroupe (somme des fluctuations dans chaque groupe). S'il n'y a pas de différence entre les groupes, ces deux variances sont (à peu près) égales. Sinon, la variance intergroupe est nécessairement la plus grande [27].

L'ANOVA est basée mathématiquement sur une régression linéaire et des modèles linéaires généraux qui quantifient la relation entre la variable dépendante et la ou les variables indépendantes. Il existe trois modèles linéaires généraux différents pour l'ANOVA :

- **Le modèle à effets fixes**: fait des inférences spécifiques et valables uniquement pour les populations et les traitements de l'étude. Par exemple, si trois traitements impliquent trois doses différentes d'un médicament, des conclusions inférentielles ne peuvent être tirées que pour ces doses de médicament spécifiques. Les niveaux de chaque facteur sont fixés comme défini par le plan expérimental.
- **Le modèle à effets aléatoires**: fait des inférences sur les niveaux du facteur qui ne sont pas utilisés dans l'étude, comme un continuum de doses de médicaments lorsque l'étude n'utilisait que trois doses. Ce modèle se rapporte aux effets aléatoires à l'intérieur des niveaux et fait des inférences sur la variation aléatoire d'une population.

- **Le modèle à effets mixtes** : contient à la fois des effets fixes et aléatoires.

3.2.1. Hypothèse de l'ANOVA

Comme beaucoup de nos procédures d'inférence, l'ANOVA a des hypothèses sous-jacentes qui devraient être en place afin de rendre les résultats des calculs complètement fiables. Ils comprennent [28] :

- Les sujets sont choisis sur un simple échantillon aléatoire.
- Les données sont normalement distribuées au sein de chaque groupe: l'ANOVA peut être considérée comme un moyen de déduire si les courbes de distribution normales de différents ensembles de données sont mieux considérées comme provenant de la même population ou de populations différentes [29].
- Homogénéité de la variance au sein de chaque groupe : en se référant à nouveau à la notion selon laquelle l'ANOVA compare les courbes de distribution normale des ensembles de données, ces courbes doivent être similaires en forme et en largeur pour que la comparaison soit valide. En d'autres termes, la quantité de dispersion des données (variance) doit être similaire entre les groupes [29].
- Observations indépendantes : Une hypothèse générale de l'analyse paramétrique est que la valeur de chaque observation pour chaque sujet est indépendante de la valeur de toute autre observation [29].

3.2.2. Modèles d'ANOVA

Ces jours-ci, les chercheurs utilisent l'ANOVA de plusieurs manières. L'utilisation de l'ANOVA dépend de la conception de la recherche. Généralement, les chercheurs utilisent l'ANOVA de trois manières :

a. Modèle à un facteur (One Way ANOVA)

Dans ce modèle, il n'y a qu'un seul facteur à plusieurs niveaux (deux, trois, quatre, etc.). Chaque observation doit provenir d'un individu qui n'est pas réutilisé dans la même expérience, c'est-à-dire que chaque observation doit être indépendante [30]. Par exemple,

il est utilisé si une entreprise manufacturière souhaite comparer la productivité de trois employés ou plus en fonction des heures de travail. C'est ce qu'on appelle un modèle ANOVA à un facteur [25].

En bref, l'ANOVA à un facteur est basée sur l'idée qu'on peut partitionner la variabilité d'un ensemble de données en différentes sources, par exemple en variabilité aléatoire entre les individus au sein des groupes (parfois appelée variabilité résiduelle ou inexpliquée) et en variabilité due à différence systématique entre les groupes [31].

Sous l'hypothèse nulle (notée H_o) que les moyennes des groupes sont les mêmes, les variances intra et inter sont censées être les mêmes. Par exemple, s'il y a k groupes :

$$H_o : \mu_1 = \mu_2 = \mu_3 = ... = \mu_k$$

Cependant, s'il y a des différences systématiques entre les groupes (*Ha* : Au moins deux des moyennes des groupes $\mu_1, \mu_2, \mu_3, ..., \mu_k$ ne sont pas les mêmes), il serait s'attend à ce que la variance entre les groupes soit supérieure à celle au sein des groupes et un test peut être construit sur la base du rapport de ces deux variances.

Ce rapport est connu sous le nom de statistique *F* (*F*-Static) et les valeurs critiques pour un test de signification peuvent être obtenues à partir des tableaux de la distribution F (F-Distribution), mais pour ce faire, on doit connaître les degrés de liberté (*df*), dont il existe deux types. Il y a ceux dus à la variabilité entre les groupes (*df* = nombre de groupes -1) et ceux dus à la variabilité au sein des groupes (*df* = nombre total d'observations - nombre de groupes).

b. Modèle à deux facteurs (Two Way ANOVA)

Dans le modèle ANOVA à un facteur, nous classons les populations selon une seules variable catégorielle ou un seul facteur. Dans le modèle ANOVA à deux facteurs, il existe deux facteurs, chacun avec plusieurs niveaux. Lorsque nous nous intéressons aux effets de deux facteurs, une conception bidirectionnelle offre de grands avantages par rapport à deux études à facteur unique [32].

Ce type comprend des nombreux concepts clés similaires à ceux de l'ANOVA à un facteur, mais la présence de plus d'un facteur introduit également de nouvelles idées. Nous supposons une fois de plus que les données sont approximativement normales et que les groupes peuvent avoir des moyennes différentes mais avoir le même écart type; nous

regroupons à nouveau pour estimer la variance; et nous utilisons à nouveau les statistiques F pour les tests de signification [32].

Par exemple, lorsqu'une entreprise souhaite comparer la productivité des employés en fonction de deux facteurs (deux variables indépendantes), elle dit qu'il s'agit d'un modèle ANOVA à deux facteurs (factorielle). Par exemple, en fonction des heures de travail et des conditions de travail, si une entreprise souhaite comparer la productivité de ses employés, elle peut le faire via ce type d'analyse de variance [25].

Le modèle ANOVA à deux facteurs peut être utilisé pour voir l'effet de l'un des facteurs après avoir contrôlé l'autre, ou elle peut être utilisée pour voir l'interaction entre les deux facteurs.

c. Modèle à N facteurs (N Way ANOVA)

Lorsque la comparaison des facteurs est prise, on dit qu'il s'agit d'un modèle ANOVA à N facteurs. Par exemple, dans la mesure de la productivité, si une entreprise prend tous les facteurs pour la mesure de la productivité, alors on dit qu'il s'agit d'un modèle ANOVA à N facteurs.

3.2.3. F-test statistique

Le processus d'évaluation des hypothèses concernant les moyennes de groupe de populations multiples est appelé analyse de variance (ANOVA).comme nous ne considérons qu'un seul facteur, cette méthode est spécifiquement appelée ANOVA one way. La statistique de test pour examiner l'hypothèse nulle est appelée F Statistic (plus spécifiquement, ANOVA F statistic) et est définie comme :

$$F=\frac{SSB/(k-1)}{SSW/(n-k)}$$

Où : ***n***= nombre d'échantillons.

k= nombre des groupes.

(k-1) = le degré de liberté au numérateur *dfn.*

(n-k) = le degré de liberté dans le dénominateur *dfd*.

1. **SSB (Variance entre les groupes):** Une estimation de σ2 qui est la variance des moyennes de l'échantillon. Si les échantillons sont de tailles différentes, la variance entre les échantillons est pondérée pour tenir compte des différentes tailles d'échantillon. La variance est également appelée variation due au traitement ou variation expliquée [33] :

$$\text{SSB}=\sum_{i=1}^{k} n_i(\bar{y_i} - \bar{y})^{\,2}$$

SSB= la somme des carrés qui représente la variation entre les différents échantillons.

2. **SSW (Variance au sein des groupes):** Une estimation de σ2 qui est la moyenne des variances de l'échantillon (également appelée variance groupée). Lorsque la taille des échantillons est différente, la variance au sein des échantillons est pondérée. La variance est également appelée variation due à une erreur ou à une variation inexpliquée [33].

$$\text{SSW}=\sum_{i=1}^{k} \sum_{j=1}^{n_i}(y_{ij} - \bar{y_i})^2$$

SSW= la somme des carrés qui représente la variation au sein des échantillons due au hasard

Afin de calculer ces rapports, pour chaque groupe, on doit compter le nombre d'observations (***n***), la moyenne de la variable d'intérêt, ***y***, la somme des observations (***T***) et la somme des observations au carré (*S*). Additionnez ensuite chacune de ces quantités dans les groupes [31].

Supposons, par exemple, qu'on a un nombre ***k*** de groupes et voyons les calculs indiqués dans la table suivante.

Group	1	2…	k	Tous les groups combinés
Nombre d'observation	n_1	$n_{2\ldots}$	n_k	$N=\sum_{i=1}^{k} n_i$
Somme d'observation	T_1	$T_{2\ldots}$	T_k	$T=\sum_{i=1}^{k} T_i$
Moyenne d'observation	$\bar{y}_1$	$\bar{y}_{2\ldots}$	$\bar{y}_k$	$\bar{y} = T/N$
Somme des carrés d'observation	S_1	$S_{2\ldots}$	S_k	$S=\sum_{i=1}^{k} S_i$

Tableau 2: Calculs de rapport statique F (*F-stat*) [31].

3.2.4. Table d'ANOVA

Une fois que les quantités ci-dessus ont été calculées, vous pouvez ensuite construire une table d'analyse de la variance pour obtenir le rapport statique F (Table 3)

	Sum of squares (SS)	Degre of freedom (df)	Mean squares (MS)	F statistic (variance ratio)
Between groups	SS_B	k-1	SS_B/df	MS_B/MS_W
Within groups	SS_W	N-k	SS_W/df	
Total	SS_B+SS_W	N-1		

Tableau 3: Table d'analyse de variance [31].

Le test ANOVA dépend du fait que la $MS_{between}$ peut être influencée par les différences de population entre les moyennes des différents groupes. Puisque MS_{within} compare les valeurs de chaque groupe à sa propre moyenne de groupe, le fait que les moyennes de groupe puissent être différentes n'affecte pas MS_{within}.

L'hypothèse nulle dit que tous les groupes sont des échantillons de populations ayant la même distribution normale. L'hypothèse alternative dit qu'au moins deux des groupes d'échantillons proviennent de populations ayant des distributions normales différentes. Si l'hypothèse nulle est vraie, $MS_{between}$ et MS_{within} doivent tous deux estimé la même valeur [33].

Si $MS_{between}$ et MS_{within} estiment la même valeur (suivant la croyance que *Ho* est vrai), alors le rapport F devrait être approximativement égal à 1. Seules les erreurs d'échantillonnage contribueraient à des variations loin de 1. Il s'avère que $MS_{between}$ est constitué de la variance de la population plus une variance produite à partir des différences entre les échantillons. MS_{within} est une estimation de la variance de la population. Puisque les variances sont toujours positives, si l'hypothèse nulle est fausse, $MS_{between}$ sera plus grand que MS_{within}. Le rapport F sera supérieur à 1 [33].

Nous notons la valeur observée de la statistique F par f. Si l'hypothèse nulle est vraie, alors la statistique de test F a une F distribution.

3.2.5. F-Distribution

La distribution F est une distribution de probabilité de la statistique F. En d'autres termes, il s'agit d'une distribution de toutes les valeurs possibles de la statistique f.

La distribution est une distribution asymétrique habituellement utilisée pour l'ANOVA. Elle a une valeur minimale de zéro; il n'y a pas de valeur maximale. Le pic de la distribution se situe juste à droite de zéro et plus la valeur f est élevée après ce point, plus la courbe est basse [34].

La distribution F est en fait une collection de courbes de distribution. Elle est liée au chi carré, car la distribution f est le rapport de deux distributions du chi carré avec des degrés de liberté *v1* et *v2* (note: chaque chi carré est d'abord divisé par ses degrés de liberté).

Chaque courbe dépend des degrés de liberté du numérateur (*dfn*) et du dénominateur (*dfd*). Celles-ci dépendent des caractéristiques des échantillons.

On note F (*dfn, dfd*) [34]. La distribution F est également appelée distribution F de Snedecor, F de Fisher ou distribution de Fisher – Snedecor.

3.2.6. P-value

Une valeur p est utilisée dans les tests d'hypothèse pour l'aide de supporter ou rejeter l'hypothèse nulle, Plus la valeur p est petite, plus l'évidence de rejeter l'hypothèse nulle est forte.

Les valeurs P sont exprimées sous forme de décimales, bien qu'il puisse être plus facile de comprendre ce qu'elles sont si vous les convertissez en pourcentage. Par exemple, une valeur p de 0,0254 est 2,54%. Cela signifie qu'il y a 2,54% de chances que vos résultats soient aléatoires (c'est-à-dire qu'ils sont arrivés par hasard). C'est assez minuscule. D'un autre côté, une valeur p élevée de 0,9 (90%) signifie que les résultats ont une probabilité de 90% d'être complètement aléatoires et non dus à quoi que ce soit dans l'expérience. Par conséquent, plus la valeur p est petite, plus les résultats sont importants (significatifs) [35].

3.2.7. P-value et niveau alpha

Les niveaux alpha sont contrôlés par les chercheurs et sont liés aux niveaux de confiance. On obtient un niveau alpha en soustrayant le niveau de confiance de 100%. Par exemple, si on veut être confiant à 98% dans la recherche, le niveau *alpha* serait de 2% (100% - 98%). Lorsque on exécute le test d'hypothèse, le test donne une valeur pour p. on compare cette valeur au niveau alpha choisi. Par exemple, disons que nous avons choisi un niveau alpha de 5% (0,05). Si les résultats du test nous donnent [36]:

- Un petit p ($\leq$ 0,05), rejette l'hypothèse nulle. C'est une preuve forte que l'hypothèse nulle n'est pas valide.
- Un grand p ($>$ 0,05) signifie que l'hypothèse alternative est faible, alors on accepte l'hypothèse nulle.

P-value=1-F-disribution

Dans un monde idéal, on aura un niveau alpha. Mais si on ne le faites pas, on peut toujours utiliser les directives approximatives suivantes pour décider de soutenir ou de rejeter l'hypothèse nulle [36]:

- Si p> 0,10 → non significatif
- Si p ≤ .10 → marginalement significatif
- Si p ≤ 0,05 → significatif
- Si p ≤ 0,01 → hautement significatif

3.3. Calcul parallèle via Matlab Parallel Computing Toolbox

3.3.1. Le calcul parallèle

Généralement, La plupart des logiciels ont été écrits pour le calcul en série. Cela signifie qu'il est exécuté sur un seul ordinateur ayant une seule unité centrale de traitement (CPU). Par conséquent, le problème sera divisé en un nombre d'instructions en série. Où l'exécution des instructions sera séquentielle [37]. Le calcul parallèle est l'une des méthodes de calcul qui exécutent de nombreux calculs (processus) simultanément. Là où le principe du calcul parallèle est souvent on peut diviser le gros problème en plus petits morceaux, puis le résoudre simultanément ("en parallèle") [37].

En d'autres termes, le calcul parallèle consiste à utiliser les multiples ressources de calcul pour résoudre simultanément un problème de calcul. Qui doit être exécuté sur plusieurs processeurs. La figure 1 (a) et (b) montre comment diviser le problème en séquence et en parallèle.

Le matériel prenant en charge le calcul parallèle se compose d'un ordinateur multicœur, d'un multiprocesseur symétrique, d'un ordinateur distribué tel que des postes de travail en cluster et de processeurs parallèles spécialisés tels que FPGA, GPU et circuit intégré spécifique à l'application (AISC). Avec le développement du matériel parallèle de support, en particulier le développement de l'ordinateur multicœur, l'architecture de programmation parallèle devient plus importante qu'avant [38].

En fait, les principaux avantages du calcul parallèle sont :

- Un gain de temps et / ou d'argent;
- Résoudre des problèmes plus importants;
- Fournir la concurrence;
- Utilisation de ressources non locales;
- Limites du calcul en série

3.3.2. Types de parallélisme

Les moyens les plus courants de parallélisme incluent le parallélisme de tâches, le parallélisme de pipeline et le parallélisme de données [38] :

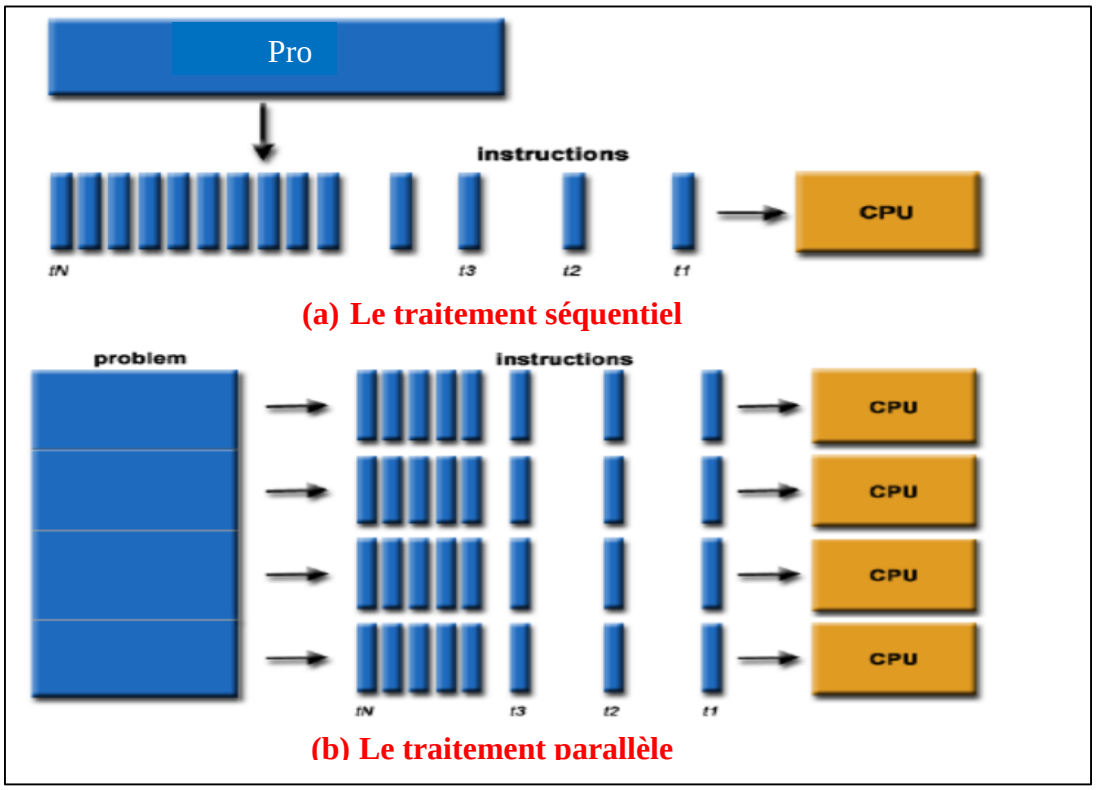


Figure 10: La différence entre le traitement séquentiel et parallèle d'un problème [37]

a. Le parallélisme des tâches

Egalement connu sous le nom de parallélisme fonctionnel, est une structure de développement parallèle dans laquelle des parties de calcul indépendantes d'une méthode peuvent être effectuées simultanément dans différents processeurs [38].

Ce type peut être utilisé lorsque les calculs dans une grande boucle sont indépendants les uns des autres et peuvent être exécutés dans n'importe quel commande sans affecter les résultats. Dans de tels cas, plusieurs processeurs peuvent analyser les sous-ensembles de données simultanément, sans nécessiter de communication entre processeurs [39].

b. Le parallélisme de pipeline

Le problème est divisé en une série de tâches. Chaque tâche sera exécutée par un processus ou un processeur distinct. Chaque processus parallèle est généralement appelé une étape de pipeline. La sortie d'un étage de pipeline sert d'entrée d'un autre de sorte qu'à tout moment, chaque étape de pipeline travaille sur un ensemble de données différent [38].

c. Le parallélisme des données

Se concentre principalement sur le même processus appliqué simultanément à différentes parties d'un ensemble de données. C'est-à-dire que des séquences d'opérations ou des fonctions similaires sont exécutées en parallèle sur des éléments d'une grande structure de données [38].

Dans tels algorithmes, les données sont généralement trop volumineuses pour être analysées sur un seul processeur. Par conséquent, des paradigmes de calcul parallèle sont utilisés pour distribuer les données entre les processeurs, et chaque processeur travaille sur une plus petite partie des mêmes données. Dans de tels cas, certaines communications peuvent être nécessaires entre différents processeurs qui impliquent l'échange de données pour traiter les conditions aux limites. En fonction de la façon dont les données sont distribuées, chaque processeur a besoin d'une petite quantité de données de son voisin pour effectuer les calculs [39].

L'une des méthodes de parallélisme mentionnées précédemment ou une combinaison de celles-ci peut être utilisée dans les applications de parallélisme.

3.3.3. Outils de calcul parallèle sous Matlab

Parallel Computing Toolbox (PCT) est le logiciel introduit à partir de Mathworks en novembre 2004 (initialement nommé Distributed Computing Toolbox ™ et Matlab Distributed Computing Engine ™, respectivement mais ensuite divisé en Parallel Computing Toolbox et Distributed Computing Toolbox).

Le PCT nous permet de résoudre des problèmes informatiques et gourmands en données à l'aide de processeurs multicœurs, de GPU et de clusters d'ordinateurs [40].Il utilise des ordinateurs multicœurs ou connectés au bus pour partager la charge de calcul par traitement parallèle.

Lancée pour la première fois en septembre 2006 dans le cadre de MATLABs 2006b, cette boîte à outils de calcul parallèle a été développée vers la version 4.1 prenant en charge jusqu'à huit *travailleurs* (appelés *workers ou clusters* : Moteurs de calcul MATLAB fonctionnant indépendamment des clients MATLAB) [38]. Ces travailleurs s'exécutent sur le bureau (la boîte à outils exécute jusqu'à quatre travailleurs localement sur le bureau) ou sur un cluster (à l'aide de MATLAB Distributed Computing Server). Les constructions simplifient le développement de code parallèle en supprimant la complexité de la gestion de la coordination et de la distribution des calculs et des données entre le client MATLAB et les travailleurs, ainsi qu'entre les travailleurs [41].

Les algorithmes peuvent être parallélisés dans MATLAB sans codage supplémentaire pour des architectures matérielles et réseau spécifiques. Par conséquent, la conversion d'anciens codes MATLAB séquentiels en applications parallèles ne nécessite que quelques modifications de code et aucune programmation dans un pilote de communication de bas niveau.

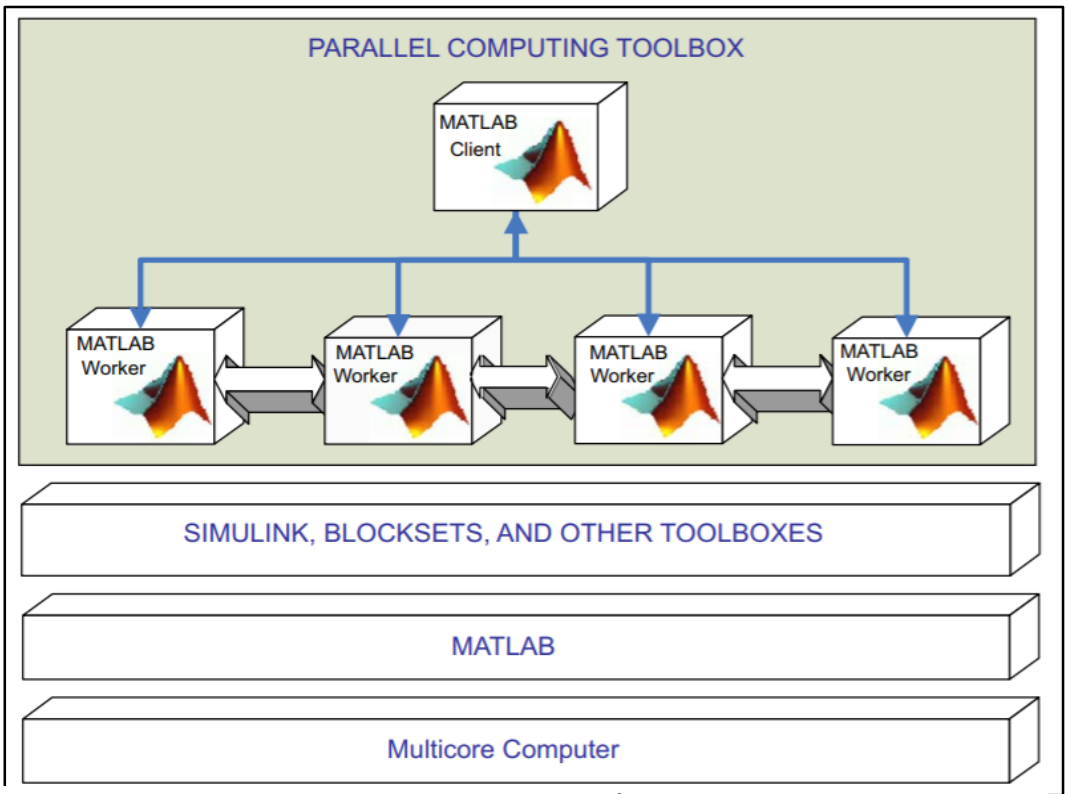


Figure 11: Structure de la boite à outil PCT [38]

Généralement, PCT prend en charge le développement d'applications parallèles aux tâches et aux données. Pour le parallélisme des tâches, PCT est limitée aux problèmes embarrassants dans lesquels les «itérations» d'une boucle parallèle n'ont aucune dépendance ou communication entre elles. En remplaçant simplement «for» dans l'ancien code MATLAB par «*parfor*», la boîte à outils de calcul parallèle gère automatiquement le transfert de données et de code entre la session client MATLAB et les sessions de travail. En outre, le parallélisme des données dans la boîte à outils de calcul parallèle de MATLAB (PCT) cible principalement les algorithmes qui nécessitent le traitement d'une grande quantité de données [38].

PCT fournit des tableaux distribués, des fonctions parallèles et utilise le mot-clé «*spmd*» pour une exécution parallèle sur plusieurs sessions de travail. Lorsque le jeu de données est divisé en éléments plus petits et envoyé à différents cœurs, certains pixels non

bordés du jeu de données d'origine deviennent des pixels de bords de ces jeux de données plus petits. Par conséquent, des effets de bord seront introduits [38].

Si la boîte à outils de calcul parallèle ne parvient pas à détecter la présence des sessions de travail, elle revient automatiquement à une exécution séquentielle [pap2].

La figure montre la structure de la boîte à outils de calcul parallèle PCT comme une superstructure sur d'autres boîtes à outils dans l'environnement MATLAB.

Par exemple, si quatre sessions de travail (*workers*) ont été ouvertes, la boîte à outils de calcul parallèle réservera automatiquement la ressource et les cœurs matériels sous-jacents. Dans le gestionnaire de tâches Windows, cinq processus MATLAB seront visibles, dont une session client et quatre sessions de travail. Les sessions de travail sont à l'état inactif sauf si des instructions parallèles sont appelées.

3.3.3.1. Approches de calcul parallèle sous PCT

Le Matlab PCT prend en charge les modèles de programmation parallèle suivants: parallèlisme de données, mémoire distribuée (passage de messages), données multiples à programme unique (SPMD) et données multiples à programme multiple (MPMD). La commande ***parfor*** implémente la programmation parallèle des données, la fonction ***spmd*** implémente le SPMD et le programmateur technique ***scheduler*** implémente à la fois la programmation de la mémoire distribuée et le SPMD [42].

a. La commande *Parfor*

La commande ***parfor*** (boucle parallèle) est la commande de programmation parallèle la plus simple de MATLAB. Elle remplace la commande ***for*** dans les cas où une boucle peut être parallélisée [42].

Dans la boucle *parfor*, chaque itération est traitée comme indépendante de toutes les autres, et le planificateur intégré de MATLAB répartit chaque itération vers un travailleur pour le calcul. Les résultats sont ensuite collectés et renvoyés de manière appropriée par le planificateur [43].

b. La commande *Spmd*

Une autre méthode pour effectuer des tâches en parallèle consiste à utiliser la programmation de «données multiples à programme unique» (*spmd* dans MATLAB). La programmation *spmd* permet un contrôle plus fin sur certains aspects du processus en permettant de choisir quel travailleur (appelé «laboratoire») exécute le code de quelle manière [43].

Bien que *spmd* autorise différents laboratoires à exécuter simultanément des différents codes, le nom provient de la forme la plus simple et la plus simple d'utilisation de *spmd*: exécuter le même code sur différents ensembles de données [43].

Dans *spmd*, la même ligne de code est exécutée sur tous les laboratoires exactement comme elle apparaît. La seule différence est que les variables appelées dans le code peuvent être différentes selon les laboratoires. L'exécution de **spmd** comprend : la configuration et la préparation du code parallèle, qui est la rotation du système, l'exécution des processeurs et la collecte des résultats d'exécution [44].

En raison de l'utilisation facile de *parfor* et de *spmd*, la plupart des nouveaux venus les utiliseront pour réaliser afin d'améliorer l'accélération de l'exécution du code. Cependant, une grande limite existe dans *parfor*, qui est que chaque itération étant totalement indépendante des autres itérations est nécessaire, ce qui est si strict en fait. Alors que *spmd* a l'avantage de permettre la programmation interactive et parallèle de manière transparente, ce qui rend plus faisable la mise en œuvre de la programmation parallèle et une plus large gamme d'applications [44].

a. La technique *Scheduler*

Le planificateur ***Scheduler*** avec un gestionnaire de travail à gérer un gros travail « ***job*** » qui est divisé en travaux indépendants varient. Supposons que chaque travail sera exécuté par la même fonction MATLAB. Chaque travail s'exécute sur un seul processeur (même s'il n'est pas nécessaire d'exclure le parallélisme) et possède sa propre mémoire. Les travaux ne communiquent pas pendant l'exécution ; ils commencent par l'entrée, ils renvoient les résultats à la fin. Lorsque tous les travaux sont terminés, il est possible de rassembler, d'analyser et de tracer les résultats combinés [42].

La méthode la plus générale pour effectuer des calculs parallèles, ou des calculs du tout d'ailleurs, sur un cluster est de soumettre des travaux au planificateur. Étant donné que MATLAB peut s'interfacer avec de nombreux planificateurs en plus du sien, l'option permettant d'ouvrir un pool MATLAB interactif n'est pas toujours disponible. Ainsi, la soumission d'une tâche soigneusement emballée au cluster permettra au planificateur actif de confier votre programme à un ensemble isolé de travailleurs MATLAB selon son protocole [43].

La soumission d'un travail à un cluster peut être accomplie en 3 étapes, avec une 4ème étape facultative de récupération des données du travail « *Job* » terminé. La première étape consiste à créer un travail. Ce dernier a des descriptions de base telles que le nom du travail (nécessaire pour suivre l'état et la récupération des résultats), le type du travail (régulier, parallèle, etc.), et des informations sur l'endroit où enregistrer les données dans le travail si nécessaire [43].

Il existe trois types des travaux de base : les travaux « *jobs* », les tâches parallèles « parallel *job* » et les travaux de pool MATLAB « Matlab pool *jobs* » [43] :

- Les travaux « *jobs* » sont l'unité de base, pour ainsi dire, et exécutent le code sur un travailleur pendant la durée du travail.
- Les travaux parallèles « parallel *job* » utilisent plusieurs cœurs, mais exécutent essentiellement le même code avec différentes valeurs aléatoires. Cela peut être utile pour simuler le même code avec plusieurs entrées aléatoires, par exemple pour tester des algorithmes de débruitage ou des schémas de filtres adaptatifs sur des entrées aléatoires.
- Les travaux de pool MATLAB « Matlab pool *jobs* » sont les plus polyvalents, car ils exécutent n'importe quel code distribué sur un nombre spécifié de nœuds de calcul. Ainsi, le code utilisant des boucles *spmd* ou *parfor* doit être exécuté sur les travaux de pool MATLAB.

Une fois le descripteur de travail créé, l'étape suivante consiste à spécifier toutes les tâches à effectuer dans le travail. N'importe quel nombre de tâches peut être effectué dans

le travail. L'avantage de l'enchaînement de plusieurs tâches dans un même travail est que pour la plupart des planificateurs, un pool de nœuds de calcul MATLAB est démarré pour chaque travail, mais maintenu ouvert jusqu'à la fin du travail. Ainsi, attribuer plusieurs tâches au lieu de soumettre plus de travaux permet de gagner du temps sur les frais généraux liés au démarrage de MATLAB.

Une fois qu'un travail a été créé avec toutes ses tâches, la soumission du travail au cluster peut être effectuée en passant le descripteur de job à la commande *submit.*

3.4. Conception générale et détaillée

3.4.1. Conception globale

Globalement, notre système est composé de trois modules principaux :

1. **Module Prétraitement :**

Les données utilisées dans notre travail sont des données brutes résultant de l'analyse de l'image des puces (fichiers CEL). Une phase de prétraitement est alors nécessaire pour qu'on peut les utiliser dans notre problème.

2. **Module de sélection parallèle des gènes**

Ce module représente le cœur de notre travail, dont, il consiste à résoudre le problème de la grande dimensionnalité des données biopuces par la sélection des gènes pertinents, en exploitant les capacités des tests d'analyse de variance et les outils de calcul parallèle sous Matlab.

3. **Module de classification**

Afin de prouver leur signification, les gènes jugés pertinents résultants de module de sélection sont utilisés pour classifier les données biopuce par la méthode de classification KNN.

4. Module de visualisation des résultats

Ce module nous permet de présenter les résultats d'application de l'analyse de variance sous un outil de calcul parallèle.

L'architecture générale de notre système est présentée dans la figure suivante

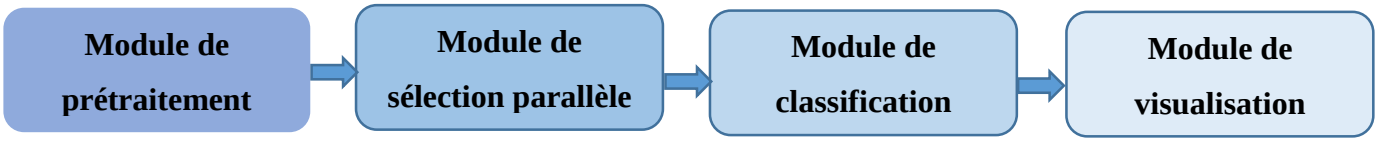


Figure 12: Architecture générale de notre travail

3.4.2.1. Prétraitement

i. Fichier CEL

Les données brutes de puce à ADN sont stockées dans un fichier à l'extension CEL. On obtient dans ces fichiers une quantité énorme d'informations. On a pour chaque gène : la moyenne des intensités de tous les pixels sur la zone correspondante au gène, la médiane. De ces intensités, l'écart-type de ces intensités et le nombre de pixels dans la zone considérée. Certaines informations additionnelles telles que l'identifiant associant une sonde est stocké dans un fichier CDF.

ii. Étape de prétraitement (transformation des données)

L'étape de transformation des données passe par 2 étapes :

a. Correction du bruit de fond (Background Correction)

- Estimation de bruit global (Constant Background Correction) par l'utilisation de la moyenne ou la médiane d'intensité des gènes.
- Estimation du bruit local (Local Background Correction) par l'utilisation les pixels se trouvant près du spot.

– Estimation du bruit de fond (Morphological Opening) par l'utilisation des méthodes non-linéaire. Nous avons utilisé la 1ère méthode dans notre travail qui est la plus utilisée.

b. Normalisation

Pour recentrer la distribution des données et la rendre symétrique, Nous avons appliqué une transformation logarithmique à base deux. Le résultat de prétraitement est une matrice d'expression des gènes filtrés et normalisés, nous permet par la suite de sélectionner les gènes les plus pertinents.

Gène id	**Echantillon 1**	**Echantillon 2**	**….**	**Echantillon N**
Gène 1	m_{11}	m_{12}		m_{1N}
Gène 2	m_{21}	m_{22}		m_{2N}
…..				
Gène M	m_{M1}	m_{M2}		m_{MN}

Tableau 4:Matrice d'expression des gènes normalisée et filtré

3.4.2.2. Données d'apprentissage

Dans notre travail, nous nous somme intéressée aux jeux de données de puces à ADN décrivant le niveau d'expression de gènes mesuré sur des tissus réparties en deux classes (tissus normaux et tissus cancéreux).

Les échantillons normaux de la classe1 possèdent un étiquette y=1, et ceux de la classe 2 (échantillons cancéreux) possèdent un étiquette y=-1.

3.4.2.3. Sélection des gènes en parallèle

Dans ce module, nous avons basé sur l'utilisation d'analyse de variance, plus précisément le rapport statistique F, sa distribution (F-Distribuation) et le p-value afin

d'évaluer l'importance des gènes et sélectionner les meilleurs qui peuvent mieux différencier les différents classes.

Cette méthode est implémentée dans un environnement parallèle multicœur pour réduire le temps de calcul. Le parallélisme au niveau des données et au niveau des tâches est utilisé pour développer la méthode sur l'outil MATLAB Parallel computing toolbox (PCT).

Le processus de sélection parallèle des gènes par ANOVA sous PCT avec un nombre **X** des laboratoires (Labs ou Workers) est décrit selon le schéma présenté dans la figure 4 et il est passé par les étapes suivantes :

1. Une fois la base des expressions des gènes est normalisée, elle va être considérer comme une entrée à l'outil de parallélisme sous Matlab (Parallel Computing Toolbox). Ce dernier est composé d'un client et un ensemble *X* des laboratoires ou travailleurs (Labs ou Workers) fixé par l'utilisateur.
2. La base va être par la suite partitionnée en *X* parties selon le nombre des laboratoires par le client de PCT.
3. Ensuite, le client va diffuser les *X* partition sur les *X* laboratoires.
4. Chaque Laboratoire *j* tel que *j* : 1 à *X* répète un ensemble des opérations pour chaque $gène_i$ de l'ensemble des gènes qu'il reçoit :
    - Calculer le moyen carré intergroupe $MS_{between}$ en utilisant $SS_{between}$ et $df_{between}$
    - Calculer le moyen carré intragroupe MS_{withi} en utilisant SS_{within} et df_{within}
    - Calculer le rapport statistique F_i
    - Calculer $p\text{-}value_i$ basant sur le rapport F_i et la table de distribution de *F*
    - Si $p\text{-}value_i < \alpha$ alors gène $_i$ est jugé comme une gêne significatif, sinon on l'élimine.
    - Ajouter le $gène_i$ à l'ensemble S_j des gènes pertinents du lab_j
5. Une fois tous laboratoires finis la boucle précédente, ils vont envoyer leurs sous-ensembles Sj (*j : 1 X*) au client de PCT.
6. Le client PCT accumule les sous-ensembles et renvois un ensemble final des gènes pertinents.

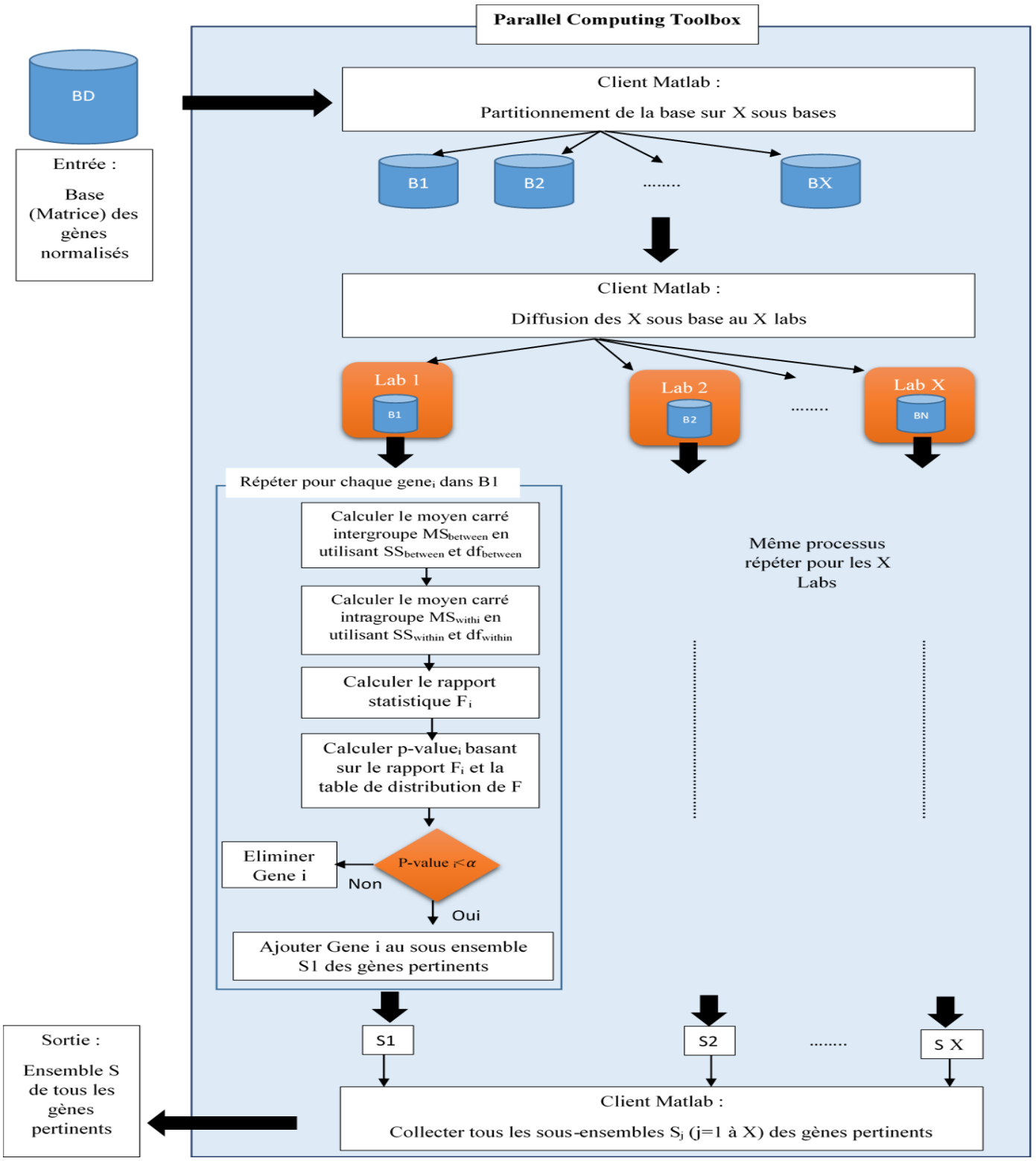


Figure 13: Processus de sélection parallèle des gènes

L'évaluation des performances et de la qualité de modèle de résultat d'ANOVA, est jugée par sa capacité de réduire l'erreur de test RMSE et maximise le coefficient de détermination (corrélation) R^2, tel que :

- **RMSE :** L'erreur quadratique moyenne est l'écart type des résidus (erreurs de prédiction). Les résidus mesurent la distance des points de données de la ligne de régression; RMSE est une mesure de la répartition de ces résidus. En d'autres termes, il indique la concentration des données autour de la ligne de meilleur ajustement. L'erreur quadratique moyenne est couramment utilisée en climatologie, en prévision et en analyse de régression pour vérifier les résultats expérimentaux.

- **R^2 :** Le coefficient de détermination peut être considéré comme un pourcentage. Cela nous donne une idée du nombre de points de données compris dans les résultats de la ligne formée par l'équation de régression.

3.4.2.4. Classification par KNN

1. **KNN**

La classification est un processus d'apprentissage automatique supervisé qui mappe les données d'entrée en groupes ou classes prédéfinis [45]. La condition principale pour appliquer une technique de classification est que tous les objets de données doivent être affectés à des classes, et que chacun des objets de données doit être affecté à une seule classe.

Les algorithmes de classification basés sur la distance sont des techniques utilisées pour classer des objets de données en calculant la distance entre la base de test et la base d'apprentissage à l'aide d'une fonction de distance.

Une des techniques de classification basée sur l'utilisation de mesures de distance est celle des k-plus proches voisins (k-NN) [46].

k-NN a été proposé en 1951 par Fix et Hodges [47] et modifié par Cover et Hart [46]. La technique peut être utilisée à la fois pour la classification et la régression [48].

Le concept principal de k-NN dépend du calcul des distances entre les échantillons de données testés et d'apprentissage afin d'identifier ses voisins les plus proches. L'échantillon testé est alors simplement affecté à la classe de son plus proche voisin [49].

Dans k-NN, la valeur k représente le nombre de voisins les plus proches. Cette valeur est le principal facteur décisif pour ce classificateur en raison de la valeur k déterminant le nombre de voisins qui influencent la classification [50].

Les voisins sont extraits d'un ensemble d'objets de données d'apprentissage pour lesquels la classification correcte est déjà connue. k-NN fonctionne naturellement avec des données numériques. Diverses mesures numériques ont été utilisées telles que les distances euclidiennes, Manhattan, Minkowsky, City-blocket Chebyshev. Parmi celles-ci, l'Euclidienne est la fonction de distance la plus utilisée avec k-NN [51].
Les principales étapes de l'algorithme k-NN sont :

1. Déterminez le nombre de voisins les plus proches (valeurs K).
2. Calculez la distance entre l'échantillon de test et tous les échantillons de formation.
3. Triez la distance et déterminez les voisins les plus proches basé sur la K-ième distance minimale.
4. Rassemblez les catégories des voisins les plus proches.
5. Utiliser la majorité simple de la catégorie des voisins les plus proches comme valeur de prédiction du nouvel objet de données.

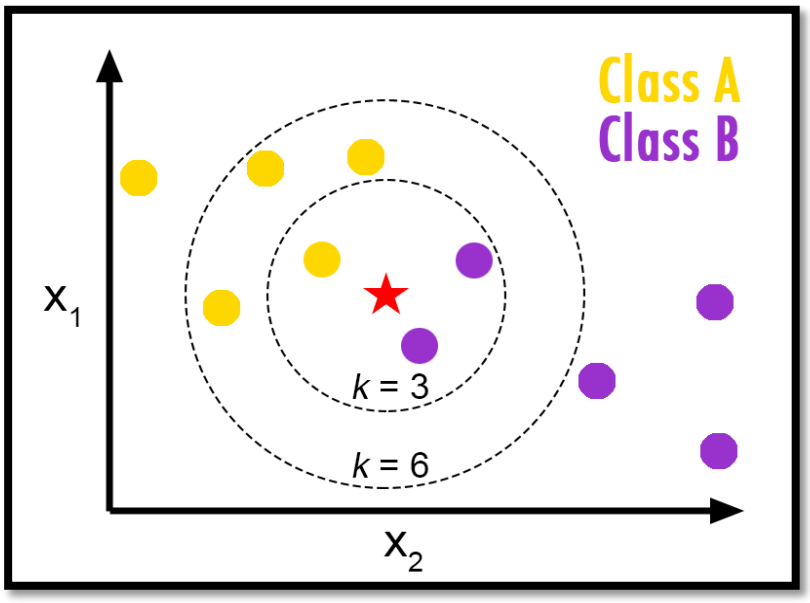


Figure 14. kPPV(kNN) classification [50]

2. Mesures de performance pour la classification

La technique la plus largement utilisée pour résumer les performances d'un algorithme de classification est la matrice de confusion.

Le tableau suivant montre la matrice de confusion pour le cas de la classification binaire avec les éléments suivants:

		Prédiction	
		-	+
Réelle	'	VP	FN
	+	FP	VN

Tableau 5. Matrice de confusion

1. Les vrais positifs (VP) sont définis par le nombre total de sorties précises lorsque la classe réelle de l'objet de données était Vrai et que la prédiction était également la valeur Vrai.
2. Les vrais négatifs (VN) sont définis par le nombre total de sorties précises lorsque la classe réelle de l'objet de données était Faux et que la valeur prédite est également la valeur Faux.
3. Faux positifs (FP) lorsque la classe réelle de l'objet de données était Faux et la valeur de sortie était la valeur Vrai
4. Faux négatifs (FN) lorsque la classe réelle de l'objet de données était Vrai et que la valeur de sortie était la valeur Faux.

Mesures calculées à partir d'une matrice de confusion :

Une matrice de confusion donne des informations utiles sur les performances du modèle. Cependant, ses éléments peuvent être utilisés pour calculer de nombreuses mesures de performances afin d'obtenir encore plus d'informations [50].

1. **Accuracy :** est la mesure de performance la plus intuitive et définie comme le rapport entre le nombre d'objets correctement classés et le nombre total d'objets évalués.
2. **Précision :** est simplement un rapport entre les objets de données positifs correctement prédits et le total des objets de données positifs prédits.
3. **Racall :** qu'il est défini par le nombre de résultats positifs corrects divisé par le nombre total d'échantillons pertinents (tous les échantillons qui auraient dû être identifiés comme positifs).
4. **F-score :** il peut être défini comme une moyenne pondérée de la précision et du rappel. Un score F est considéré comme parfait lorsqu'il atteint sa meilleure valeur à 1, tandis que le modèle est un échec total lorsqu'il atteint la valeur 0.

Mesure	Formule
Accuracy	$\frac{VP + VN}{VP + FP + VN + FN}$
Précision	$\frac{VP}{VP + FP}$
Recall	$\frac{VP}{VP + FN}$
F-Score	$2*\frac{(Recall*Precision)}{(Recall+Precision)}$

Figure 6. Mesures d'évaluation pour l'ensemble de données de classe binaire

3.4.2.5. Résultats et visualisation

Apres la phase de sélection des gènes, nous avons présenté notre résultat obtenu tels que les noms de ses gênes, les propriétés statistiques d'ANOVA test avec une visualisation de ses propriétés statistiques permettant une bonne analyse de nos résultats.

3.5. Conclusion

Dans ce chapitre, nous avons décrit le principe et le fonctionnement de la méthode d'ANOVA sur laquelle notre projet se base, puis nous avons présenté le processus de sélection des gènes par cette méthode avec une description détaillée de ses composants.

Dans le chapitre suivant, nous passerons à l'implémentation du processus de sélection de gènes, en présentant les différents détails de sa réalisation.

Chapitre 4

Implémentation

Chapitre 4. Implémentation

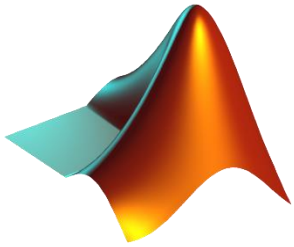
4.1. Introduction

Dans ce chapitre, nous allons présenter l'environnement de travail, le langage de programmation et les outils que nous avons utilisé pour construire le système. Puis nous allons présenter le jeu de donnée utilisé dans notre travail, un jeu de donnée publique qui est utilisé dans de nombreux travaux concernant l'analyse des données de puces à ADN. Enfin nous allons présenter interface de notre application.

4.2. Environnement et outils de développement

Pour développer notre application et valider notre proposition, nous avons utilisé le langage Matlab et l'environnement Matlab. Pour la transformation et la normalisation des données nous avons utilisé la boîte à outils Bioinformatics Toolbox et particulièrement la catégorie Microarray Analysis.Et Parallel Computing Toolbox Pour la sélection des gènes en parallèle.

4.2.1. Environnement et développement



Matlab est un logiciel de calcul numérique commercialisé par la société MathWorks1. Il a été initialement développé à la fin des années 70 par Cleve Moler. Matlab est abréviation de MATrix LABoratory. Il est avant tout un programme de calcul matriciel. Il est un langage pour le calcul scientifique, l'analyse de données, leur visualisation, le développement d'algorithmes. Son interface propose, un environnement de développement intégré (IDE) pour la programmation d'applications. Le logiciel peut être complété par de multiples toolboxes, c'est-à-dire des boîtes à outils. Celles-ci sont des bibliothèques de fonctions dédiées à des domaines particuliers. [21]

4.2.2. Outils utilisés

4.2.2.1. Bioinformatics Toolbox

Bioinformatics Toolbox est une boite à outils intégrer dans le Matlab, elle fournit des algorithmes et des applications pour le séquençage de nouvelle génération (NGS), l'analyse de puces à ADN, la spectrométrie de masse et l'ontologie génique. En utilisant les fonctions de la boîte à outils, vous pouvez lire des données génomiques et protéomiques à partir de formats de fichiers standards tels que SAM, FASTA, CEL et CDF, ainsi que de bases de données en ligne telles que NCBI Gene Expression Omnibus et GenBank. La boîte à outils fournit également des techniques statistiques pour détecter les pics, imputer des valeurs pour les données manquantes et sélectionner des caractéristiques. [52]

Nous nous sommes intéressés à l'analyse de puces à ADN, ç'est pour cela nous avons choisis la catégorie Microarray Analysis de Bioinformatics Toolbox. Microarray Analysis est une catégorie de Bioinformatics Toolbox, elle est largement utilisée pour l'analyse de données sur microarray, y compris la lecture, le filtrage, la normalisation et la visualisation des données de microarray. [52]

Nous avons utilisé la fonction d'affyrma de Microarray Analysis. Affyrma est un algorithme utilisé pour créer une matrice d'expression à partir de données CEL. Les valeurs d'intensité brutes sont corrigées en arrière-plan, transformées et normalisées en log2.

Paramètres d'entrée affyrma

CELFile : c'est le nom de fichier CEL peut être l'une des valeurs suivantes : vecteur des caractères spécifiant un seul nom de fichier CEL, qui lit tous les fichiers CEL dans le dossier en cours, " qui ouvre la boîte de dialogue Sélectionner les fichiers CEL à partir de laquelle on sélectionne les fichiers CEL

CDFFile : c'est le nom de fichier CDF peut être l'une des valeurs suivantes : vecteur des caractères spécifiant un seul nom de fichier CDF; qui ouvre la boîte de dialogue Sélectionner le fichier CDF à partir de laquelle on sélectionne le fichier CDF.
CELPathValue : c'est le chemin et le dossier où les fichiers spécifiés dans CELFiles sont stockés.
MedianValue : on Spécifie l'utilisation de la médiane des valeurs classées au lieu de la moyenne pour la correction du bruit de fond, le choix est TRUE ou False (par défaut).
OutputValue : on Spécifie le choix de fonction de normalisation : *log*2, *log*, *log*10, linear, @functionname.

4.2.2.2. Parallel Computing Toolbox

La boîte à outils de calcul parallèle d'ATLAB permet aux utilisateurs de résoudre des problèmes de calcul et de données gourmandes en utilisant des ordinateurs multicœurs et multiprocesseurs, des clusters d'ordinateurs et des GPU. Les utilisateurs peuvent utiliser des fonctions MATLAB de haut niveau pour paralléliser des applications sans programmation OpenMP, MPI et CUDA. L'une des caractéristiques importantes de Parallel Computing Toolbox de MATLAB est que la même application peut être exécutée sur un cœur simple, un processeur multicœur ou un cluster d'ordinateurs sans changer le code. En combinant Parallel Computing Toolbox avec Distributed Computing Server, les utilisateurs peuvent exécuter leurs programmes MATLAB sur des clusters d'ordinateurs, des grilles et des nuages. [53]

Les principales raisons d'envisager le calcul parallèle sont [54]:

- Gagnez du temps en répartissant les tâches et en les exécutant simultanément
- Résolvez les problèmes de Big Data en distribuant des données
- Tirez parti des ressources de votre ordinateur de bureau et évoluez vers les clusters et le cloud computing

Avec Parallel Computing Toolbox, on peut :

- Accélérez votre code à l'aide d'outils de calcul parallèle interactif, tels que parfor et parfeval
- Faites évoluer votre calcul à l'aide d'outils de traitement Big Data interactifs.

a) Fonctions de Parallel Computing Toolbox

Parallel Computing Toolbox fournit des outils et des fonctions pour un cluster local de travailleurs (workers) sur un ordinateur client. Une liste des fonctions est dans la table6

Fonction	**Description**
Parpool	Démarrer un pool parallèle de nœuds de calcul à l'aide du profil de cluster par défaut. Avec les préférences par défaut.
Parpool(poolsize)	crée et renvoie un pool avec le nombre spécifié de nœuds de calcul. poolsize peut être un entier positif ou une plage spécifiée comme un vecteur à 2 éléments d'entiers.
Distributed	Créer un tableau distribué à partir des données de l'espace de travail client ou d'une banque de données.
Spmd	Execute code in parallel on workers of parallel pool.
Composite	Accéder à des variables non distribuées sur plusieurs workers à partir du client.
getLocalPart	Partie locale du tableau codistribué.
tic	Enregistre l'heure actuelle.
toc	Utilise la valeur enregistrée pour calculer le temps écoulé.
Delete(gcp)	arrête le pool parallèle associé à l'objet, par défaut gcp.

Tableau 6. Fonctions de PCT

b) SPMD 'Single Program Multiple Data'

La construction de langage à SPMD permet un entrelacement transparent de et programmation parallèle. L'instruction spmd permet de définir un bloc de code à exécuter simultanément sur plusieurs travailleurs (workers). Les variables affectées à l'intérieur de l'instruction spmd sur les workers permettent un accès direct à leurs valeurs du client par référence via des objets composites.

L'aspect «**Single Program**» de spmd signifie que le code identique s'exécute sur plusieurs nœuds de calcul. Vous exécutez un programme dans le client MATLAB et les parties de celui-ci étiquetées comme blocs spmd s'exécutent sur les nœuds de calcul. Lorsque le bloc spmd est terminé, votre programme continue de s'exécuter dans le client.

L'aspect «**Multiple Data**» signifie que même si l'instruction spmd exécute un code identique sur tous les nœuds de calcul, chaque collaborateur peut avoir des données différentes et uniques pour ce code. Ainsi, plusieurs ensembles de données peuvent être hébergés par plusieurs travailleurs.

Les applications typiques appropriées pour spmd sont celles qui nécessitent l'exécution simultanée d'un programme sur plusieurs ensembles de données, lorsqu'une communication ou une synchronisation est requise entre les travailleurs. Certains cas courants sont:

- Programmes qui prennent beaucoup de temps à exécuter - spmd permet à plusieurs travailleurs de calculer des solutions simultanément.
- Programmes fonctionnant sur de grands ensembles de données - spmd permet de distribuer les données à plusieurs travailleurs.

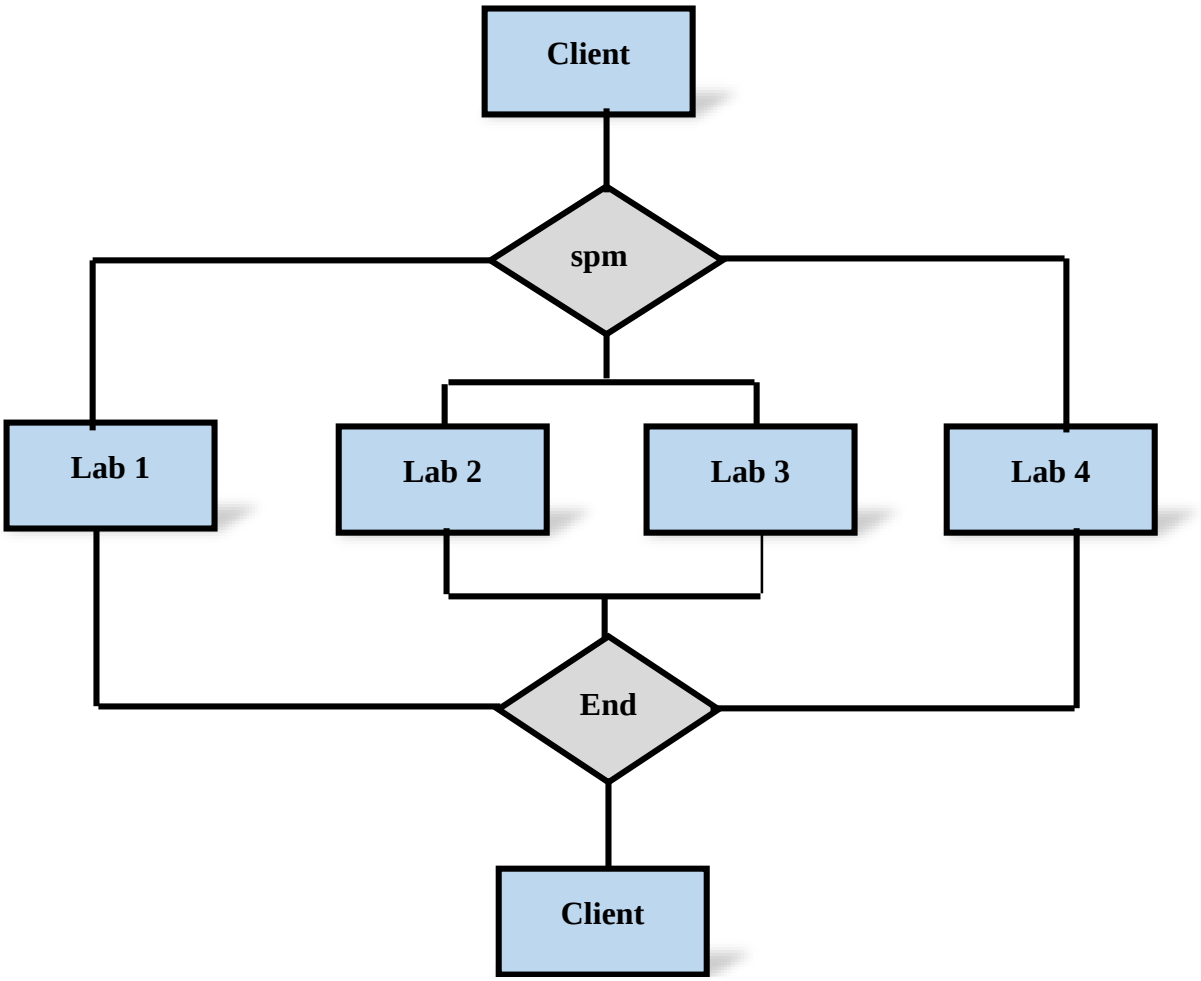


Figure 15: Mécanisme de Fonctionnement SPMD

4.3. Système de sélection proposée

Nous avons développé une application implémentant la proposition présenté dans le chapitre trois. L'application est composée à trois parties :

— Prétraitement (transformation des données).

— Sélection des gènes.

— Classification.

4.3.1. Ensemble des données utilisées

D'après nos recherches, nous avons utilisé des jeux de données publics, accessibles et utilisés dans des nombreux travaux concernant l'analyse des données de puces à ADN.

- **Cancer du poumon**

Dans ce jeu de données, le niveau d'expression de 12625 gènes est mesuré sur 207 Tissus (18 tissus normaux et 189 tissus cancéreux).

- **Leukimia**

Ce jeu de données présente le niveau d'expression de 12626 gènes mesuré sur 52 Tissus (24 tissus normaux et 28 tissus cancéreux).

- **Sarcome**

Dans ce jeu de données, le niveau d'expression de 22283 gènes est mesuré sur 23 Tissus (15 tissus normaux et 8 tissus cancéreux).

Base de données	Gênes	Échantillon	Classe	Normale	Tumeur
Cancer du poumon	12625	207	2	18	189
Leukeimia	12626	52	2	24	28
Sarcome	22283	23	2	8	15

Tableau 7. Ensemble des données utilisées

Pour une description complète de ces jeux de données consulter l'adresse :

http://www.broad.mit.edu/cgi-bin/cancer/datasets.cgi

Les fichiers CEL de ces jeux de données sont organisés comme il est présenté dans la figure 16.

Figure 16: Exemple des données utilisé (CEL).

4.3.2. Prétraitement des données

Le module de prétraitement présenté dans la figure 17, nous permettons de transformer et normaliser les données utilisés, par l'utilisation de la fonction Affyrma de la boite outils Bioinformatics. Le résultat de cette fonction est montré dans la figure 18.

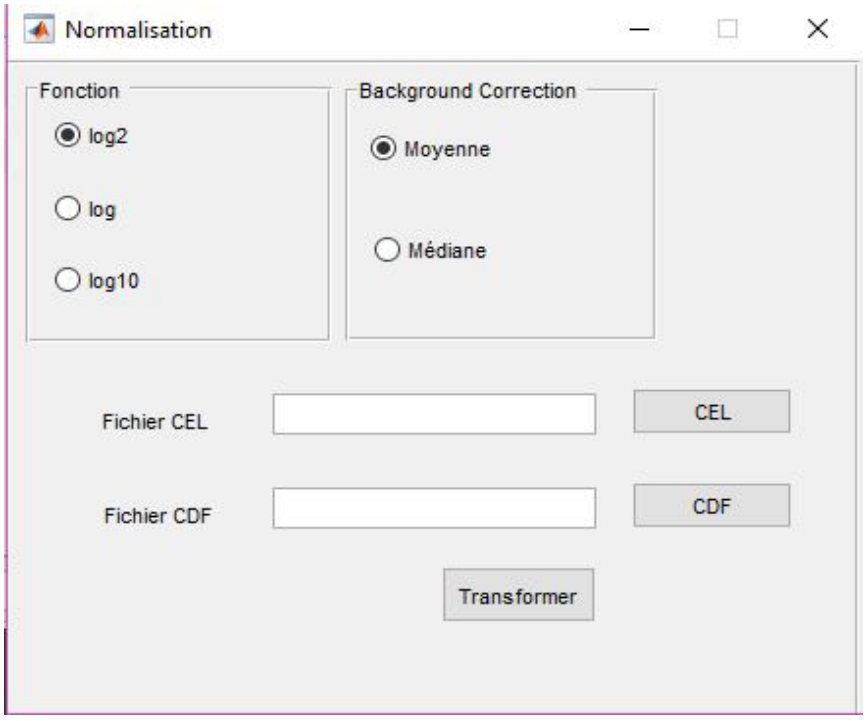


Figure 17: Interface du module de prétraitement.

	CL2001032212AA	CL2001032213AA	CL2001032214AA	CL2001032216AA	CL2001032227AA
AFFX-MurIL2_at	5.6147	5.8557	5.2494	5.4717	5.4212
AFFX-MurIL10_at	4.7569	4.9489	4.3067	4.4734	4.3944
AFFX-MurIL4_at	3.2623	4.1402	3.126	3.3234	3.1907
AFFX-MurFAS_at	4.1078	4.5543	3.8287	3.9093	4.0356
AFFX-BioB-5_at	4.2063	4.5537	4.0612	4.8949	4.7577
AFFX-BioB-M_at	4.1087	4.2101	3.9226	4.2386	3.9414
AFFX-BioB-3_at	3.8553	3.7783	3.415	3.7452	3.4373
AFFX-BioC-5_at	5.2633	5.2736	4.7337	5.1339	4.7238
AFFX-BioC-3_at	4.6224	4.7358	4.245	4.6798	4.2669
AFFX-BioDn-5_at	3.4355	4.2308	3.1433	3.4659	3.1632
AFFX-BioDn-3_at	9.0635	9.1214	8.6672	9.1451	9.0533
AFFX-CreX-5_at	3.3428	3.3686	3.1755	3.3716	3.2586
AFFX-CreX-3_at	5.3794	5.5425	4.9774	5.3002	4.9339
AFFX-BioB-5_st	4.8504	4.9349	4.52	5.1838	4.8635
AFFX-BioB-M_st	4.5776	4.7643	4.2107	4.587	4.216
AFFX-BioB-3_st	5.9324	6.1391	5.5797	5.8331	5.6235
AFFX-BioC-5_st	5.4393	5.964	5.3511	5.6492	5.3823
AFFX-BioC-3_st	3.7289	4.1509	3.6168	4.0574	3.6382
AFFX-BioDn-5_st	4.966	5.6188	4.8518	5.2761	4.8839
AFFX-BioDn-3_st	6.2993	6.1746	5.7476	6.1401	5.879

Figure 18: Résultat de prétraitement.

4.3.3. Sélection des gènes

Une fois les données sont chargés, l'interface présentée dans la figure nous permettons de :

- Choisir le mode de test (utilisation la base d'entrainement, charger une base de test ou le mode Holdout).
- Faire l'opération de sélection des gènes.
- Afficher le résultat de sélection (le meilleur modèle, les noms des gènes sélectionnés).
- Présentation des mesures d'évaluation et le Temps d'exécution.

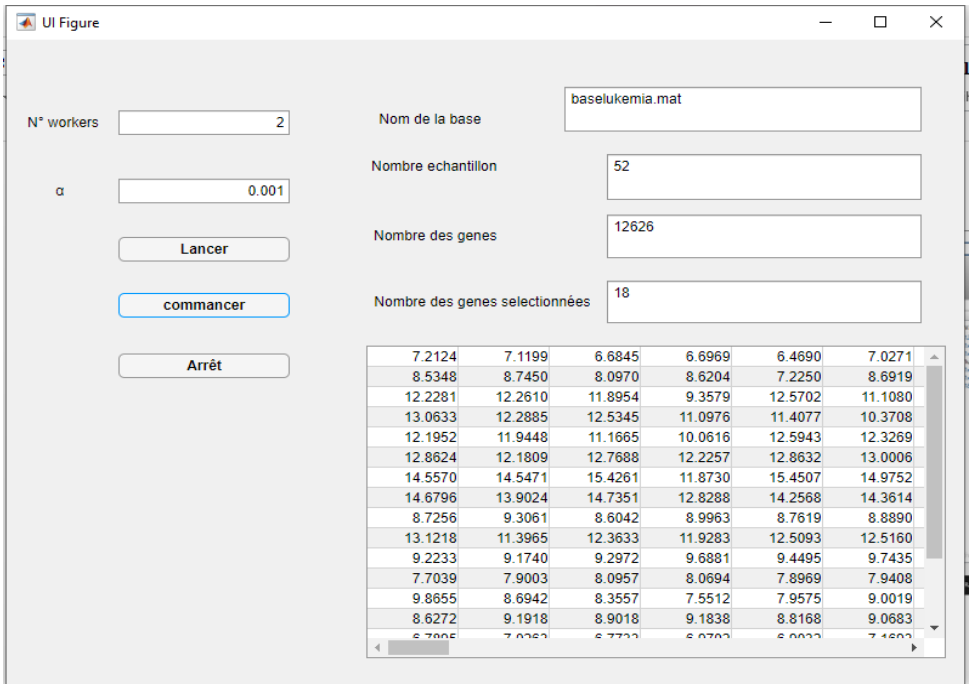


Figure 19: Interface principale de la sélection.

Les trois boutons représentés dans cette interface sont :

- **Lancer :** pour lancer le parallélisme avec un nombre donné de travailleurs (workers)
- **Commencer :** pour charger la base de données et appliquer la méthode de sélection proposé.
- **Arrêt :** pour arrêter le parallélisme.

4.3.4. Visualisation des résultats

Cette interface montre la signification statistique de chacun des gènes de modèle développé (corrélation entre les gênes) et des informations sur les performances du modèle tel que la précision.

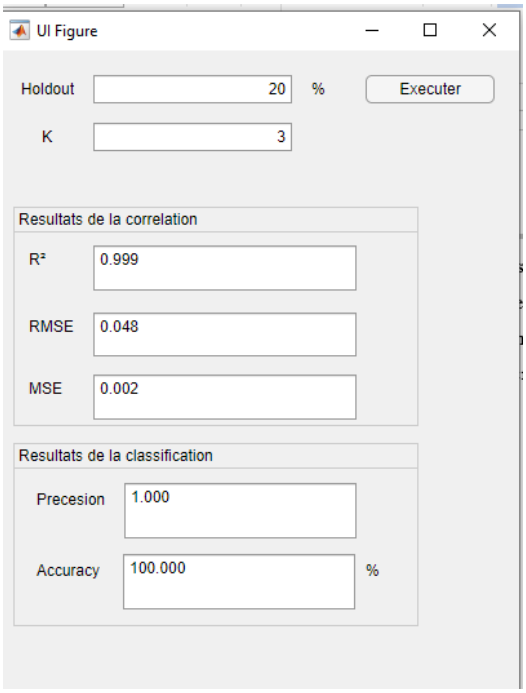


Figure 20:Interface de résultats des performances

4.4. Conclusion

Dans ce chapitre, Nous avons représenté l'implémentation de notre système : L'environnement et les outils de développement. Dans le chapitre suivant, nous avons présenté les expérimentations effectuées sur les trois bases utilisées, ainsi que les résultats obtenus en termes d'erreur quadratique moyenne RMSE, R^2 et les mesures de performance pour la classification.

Chapitre 5
Expérimentation et résultats

Chapitre 5. Expérimentation et résultats

5.1. Introduction

Dans ce chapitre, nous allons expliquer les expérimentations que nous avons appliquées sur la méthode proposée et les résultats obtenus, en utilisant un jeu de donnée publique qui est utilisé dans de nombreux travaux concernant l'analyse des données de puces à ADN.

5.2. Expérimentations et résultats

5.2.1. Détermination des meilleurs Paramètres

Afin de déterminer les meilleurs paramètres d'ANOVA, plusieurs expérimentations sont faites. Les paramètres présentés dans le tableau (8) sont jugés comme les meilleurs après son application sur les trois bases : (a)le cancer du poumon, (b) du leukeimia et (c) du sarcome.

Paramètres	Valeur		
	(a)	(b)	(c)
Seuil de signification Alpha	0.001	0.001	0.001
Nombre de travailleurs	Optionnel	Optionnel	Optionnel
Données d'apprentissage	80%	80%	80%
Données de test	20%	20%	20%

Tableau 8: Valeurs optimales des paramètres de sélection par ANOVA.

Le parallélisme est effectué par un certain nombre de workers (optionnel), car plus le nombre est élevé, plus le temps d'exécution est court.

```
Lab 1:
  Elapsed time is 0.304870 seconds.
Lab 2:
  Elapsed time is 0.255852 seconds.
Lab 3:
  Elapsed time is 0.287378 seconds.
fx >>
```

Figure 21:interface montre le temps d'éxecution de 3 workers

```
Lab 1:
  Elapsed time is 0.065419 seconds.
Lab 2:
  Elapsed time is 0.084085 seconds.
Lab 3:
  Elapsed time is 0.101653 seconds.
Lab 4:
  Elapsed time is 0.109790 seconds.
Lab 5:
  Elapsed time is 0.103612 seconds.
fx >>
```

Figure 22:Interface montre le temps d'exécution de 5 workers

```
Lab 1:
  Elapsed time is 0.076332 seconds.
Lab 2:
  Elapsed time is 0.063469 seconds.
Lab 3:
  Elapsed time is 0.077286 seconds.
Lab 4:
  Elapsed time is 0.081912 seconds.
Lab 5:
  Elapsed time is 0.066295 seconds.
Lab 6:
  Elapsed time is 0.061089 seconds.
Lab 7:
  Elapsed time is 0.063076 seconds.
fx >>
```

Figure 23: Interface montre le temps d'exécution de 7 workers

5.2.2. Analyse des résultats

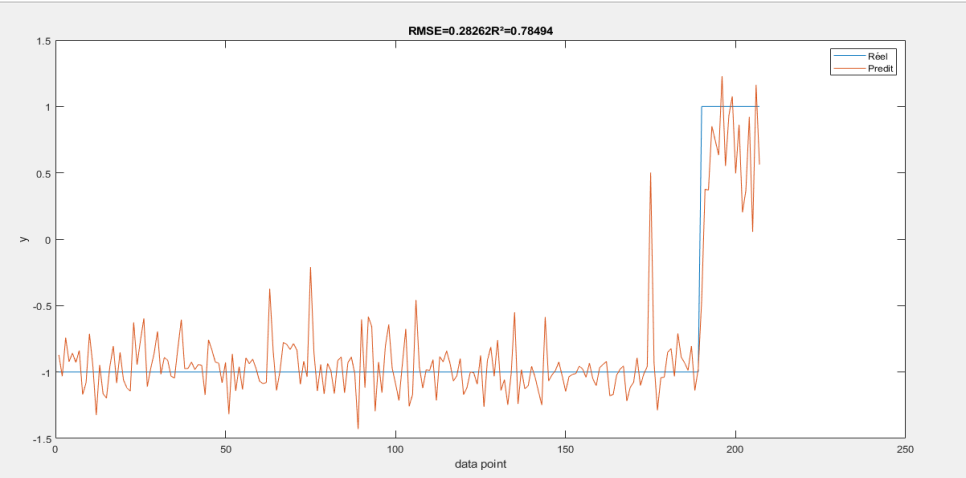
5.2.2.1. Evaluation des modèles de prédiction

Les résultats présentés dans la table(9), montrent l'efficacité de notre méthode pour résoudre le problème de sélection des gènes, dont, pour chacun des modèles de prédiction, nous avons calculé les métriques d'évaluation suivants : R^2, RMSE, MSE, SSE sur le jeu de utilisé.

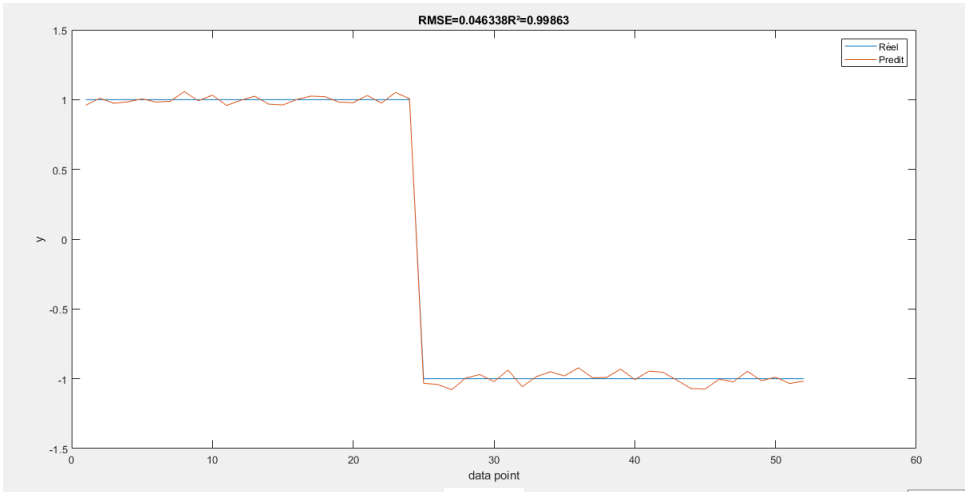
	Mesures d'évaluation			
	R^2	RMSE	MSE	SSE
(a)	0.784	0.282	0.079	14.13
(b)	0.998	0.046	0.0021	0.070
(c)	0.994	0.117	0.0137	0.109

Tableau 9: Valeurs des mesures d'évaluation obtenue

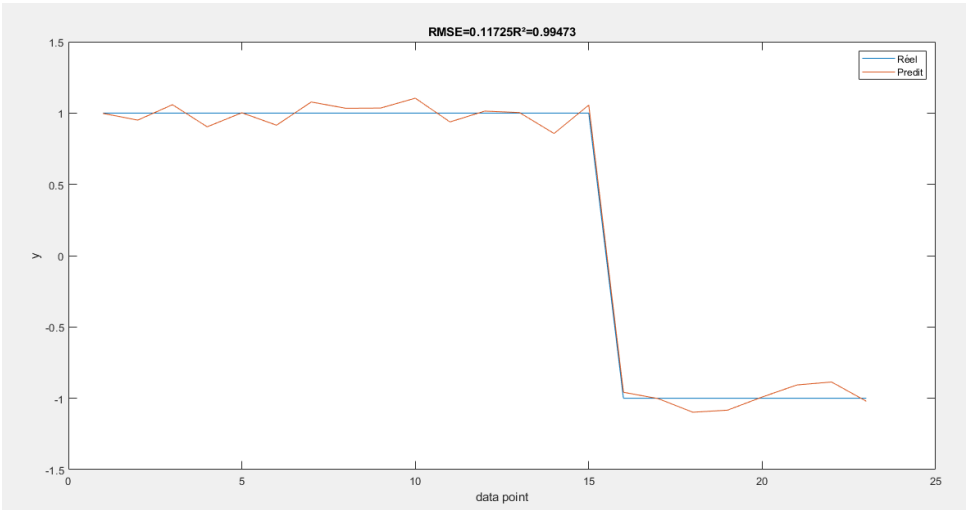
Nous remarquons que les valeurs des métriques d'évaluation montrées dans le tableau 9 sont très intéressantes pour les trois bases utilisées, car les valeurs R^2 sont proche de 1 et les valeurs de RMSE et MSE proche de 0 et les valeurs de SSE(somme d'erreurs au carré) dans le pire des cas 14.13 (la base du poumon et dans le meilleur des cas 0.070 (la base de leukeimia) le même chose, nous avons obtenu une bonne corrélation entre les classes réelles et prédites comme il est montré la figure suivante :



(a)



(b)



(c)

Figure 24:Corrélation entre les valeurs prédites ($\hat{y}$) et réelles (y).

5.2.2.2. Gènes sélectionnées

La table (10) représente les noms des gènes sélectionnés et utilisés par les Modèles ANOVA développés pour diagnostiquer les maladies de cancérologie, de sorte que le nombre des gènes sélectionnés était 29 gènes pour le cancer du poumon, 18 gènes pour le cancer du leukeimia et 14 gènes pour le sarcome.

	Nombre des gènes sélectionnés
Cancer du poumon (29 gènes sélectionnées)	'35868_at' '37777_at' '38177_at' '38183_at' '39634_at' '40739_at' '33756_at' '33766_at' '34194_at' '34708_at' '35152_at' '35730_at' '36065_at' '36569_at' '37196_at' '37247_at' '38995_at' '36155_at' '37397_at' '38044_at' '38430_at' '39541_at' '40994_at' '41837_at' '32542_at' '1897_at' '1596_g_at' '407_at' '268_at'
Leukeimia (18 gènes sélectionnées)	'AFFX-CreX-5_at' 'AFFX-CreX-3_at' '34168_at' '36239_at' '38242_at' '38521_at' '39318_at' '38017_at' '1864_at' '1373_at' '1302_s_at' '1215_at' '1216_at' '439_at' '426_at' '383_at' '266_s_at' '125_r_at'
Sarcome (14 gènes sélectionnées)	'200780_x_at' '200944_s_at' '200981_x_at' '204571_x_at' '204818_at' '207186_s_at' '208581_x_at' '211285_s_at' '211456_x_at' '211825_s_at' '211858_x_at' '212273_x_at' '212599_at' '219489_s_at'

Tableau 10: Noms des gènes sélectionnés

5.2.3. Evaluation des résultats

H Pour garantir la qualité des sous-ensembles trouvés par ANOVA test, nous avons fait une classification par KNN, la table(11) présente le résultat de la classification et la validité des sous-ensembles.

	kNN k=5	
	i	ii
(a)	80%	100 %
(b)	100 %	100 %
(c)	75%	100 %

Tableau 11: Taux de classification par kNN i= gènes avant la sélection, ii=gènes sélectionnés

5.3. Conclusion

Les résultats obtenus dans les différentes bases montrent l'efficacité des paramètres de contrôle d'ANOVA que nous avons déterminé par expérimentations.

Dans le cadre de la sélection des gènes, L'erreur quadratique moyenne RMSE, R^2, les précisions montrent clairement l'efficacité des modelés déterminés par ANOVA

Pour cela, nous pouvons conclure que le modèle ANOVA permettant la réduction du nombre d'attributs et la sélection du sous-ensemble optimal des gènes ce qui conduira à diagnostiquer les maladies de cancérologie efficacement et surmonter le problème de sur-apprentissage des données.

La même analyse peut être étendue pour prendre en charge d'autres maladies de cancer tel que cancer de sein, cancer du cerveau, cancer du côlon . . . etc

Conclusion générale

Dans ce mémoire, nous nous somme intéressées au problème de la réduction de la dimensionnalité des données de puce à ADN, qui est une tâche primordiale pour plusieurs applications de bio-informatique, de reconnaissance et du traitement des maladies.

La réduction de la dimensionnalité via la sélection de gènes consiste à extraire un sous-ensemble optimal de gènes pertinents permettant de réduire le volume de données et faciliter la manipulation et l'analyse des données de puce à ADN.

Pour traiter ce problème, nous avons utilisé une méthode statistique d'ANOVA pour la sélection des gènes en exploitant la boite à outils MATLAB concernant le calcul parallèle (PCT : Parallel Computing Toolbox) pour l'implémenter. Ensuite l'algorithme de classification kNN a été appliqué sur les données représentées par les gènes sélectionnés, pour évaluer la précision de la classification.

Pour valider notre proposition et pour tester la possibilité de l'utilisation de notre méthode et ses propriétés de réduire les attributs (gènes) et trouver le sous-ensemble optimal, nous avons utilisé trois bases de plus de 12600 gènes. La qualité des modèles et l'influence des gènes sélectionnés sur la qualité des modèles est déterminé par le RMSE et R^2.

Pour les perspectives et les travaux futurs, nous proposons des idées qui peuvent améliorer et généraliser notre système de sélection des gènes telles que :

- L'utilisation d'autres types de prétraitement des données pour adapter les données au type d'analyse souhaité.
- Le traitement des données classées en multi-classes (multi-label).
- L'utilisation d'autres bases des données concernent d'autres problèmes.

Références

[1] Akm Tauhidul Islam, Byeong-Soo Jeong, A.T.M Golam Bari et Chae-Gyun Lim, «MapReduce based parallel gene selection method,» *Springer,* 2014.

[2] Edmundo Bonilla Huerta, Béatrice Duval, et Jin-Kao Hao., «A hybrid ga/svm approach for gene selection and classification of microarray data. In Workshops on Applications of Evolutionary Computation,,» *Springer,* p. pages 34–44., 2006.

[3] LE MEUR Nolwenn, *De l'Acquisition des Données de Puces à ADN vers leur Interprétation : Importance du Traitement des Données Primaires,* Université de Nantes: PhD thesis, 2005.

[4] Abdelillah HASSAM,, Ismael Abraham OUATTARA, Lynda ZAOUI, Khadija HENNI et Rahal SD, Construction d'un workflow d'analyse de données issues de puces à ADN, Gene Expression, 2014, p. 1:6.

[5] Maude Pupin, La génomique., Université de Lille: CRISTAL, 1994..

[6] MOUSSATI Omar, Classification des données de biopuces, Université des Sciences et de la Technologie d'Oran Mohamed Boudiaf: PhD thesis, 2016.

[7] J. C. H. Hernandez, Algorithmes Métaheuristiques hybrides pour la sélection de gênes et la classification de données de biopuces., Université d'Angers: PhD thesis, 2008.

[8] Nguyen Hoai Tuong, Puces à adn., Université de Lille: Ecole Polytechnique, 2008.

[9] Ramón Díaz-Uriarte et Sara Alvarez De Andres, «Gene selection and classification of microarray data using random forest,» *BMC bioinformatics,* 2006.

Références

[10] E. M. Southern., «DNA Arrays methods and protocols, chapter DNA Microarrays,,» vol. pages 1–15, 2001..

[11] F. Bertucci., «Profils d'expression génique et puces à adn dans le cancer du sein : choix du patient, choix du protocole. In Cancer du sein,,» *Springer,* p. pages 267–276. , 2006.

[12] M. Khabzaoui., «Modélisation et résolution multi-objectifs des règles d'association: application à l'analyse de données biopuces.,» 2006.

[13] Moussa A. et Vannier B., Workflow d'analyse de données des puces à ADN, Spectra Analyse, Mai 2013.

[14] «Genome Resource Facility GRF, Microarray section,» 2006. [En ligne]. Available: http://grf.lshtm.ac.uk/index.htm.

[15] Bernard R, Puces à ADN, Cours de biologie, Université d'Aix enProvence, 2010.

[16] «National Center for Biotechnology Information,» [En ligne]. Available: https://www.ncbi.nlm.nih.gov/geo/.

[17] M. Kamilia., « Approches Bio-inspirées pour la Sélection d'Attributs.,» 1945.

[18] CHOUAIB Hassan, Sélection de caractéristiques : méthodes et applications., Université Paris Descartes: PhD thesis, 2011.

[19] Sabra EL FERCHICHI, Sélection et extraction pour les problèmes de classification, Université LILE1: PhD thesis, 2011.

[20] M. Kalakech., «Sélection semi-supervisé d'attributs : Application à la classification de textures couleur.,» 2011.

[21] Xin Zhou et KZ Mao, Ls bound based gene selection for dna microarray data., vol. 21, Bioinformatics, 2004, p. 1559–1564.

[22] Mukesh Kumara, Nitish Kumar Rath, Amitav Swain et Santanu Kumar Rath, Feature Selection and Classification of Microarray Data using MapReduce based ANOVA and K-Nearest Neighbor, Procedia Computer Science, December 2015.

Références

[23] Leping Li, Clarice R Weinberg, Thomas A Darden et Lee G Pedersen, Gene selection for sample classification based on gene expression data : study of sensitivity to choice of parameters of the ga/knn method., vol. 17(12), Bioinformatics, 2001, p. 1131–1142.

[24] Chen AH et Lin CH, «A novel support vector sampling technique to improve classification accuracy and to identify key genes of leukaemia and prostate cancers.,» vol. 38(4):3209–3219, 2011.

[25] «ANOVA (Analysis of Variance),» [En ligne]. Available: http://www.statisticssolutions.com.

[26] «Analyses de variance et covariance,» Wiki Stat.

[27] Robert L. Miller, John Maltby, Jo Campling, Deirdre A. Fullerton et Ciaran Acton, «Analyse de la variance (ANOVA),» chez *SPSS for Social Scientists*, 2002, pp. 145-154.

[28] *Basic ANOVA concepts*, Math 143 – ANOVA.

[29] Steven F. Sawyer, «Analysis of Variance: The Fundamental Concepts,» chez *Journal of Manual & Manipulative Therapy*, vol. volume 17, The Journal of Manual & Manipulative Therapy, 2013.

[30] Bodo Winter, «The F distribution and the basic principle behind ANOVAs,» 2015. [En ligne]. Available: www.bodowinter.com.

[31] Jenny V Freeman et Michael J Campbell, ONE-WAY ANALYSIS, Scope, 2007.

[32] Richard M et HeibergerBurt Holland, «Two-Way Analysis of Variance,» chez *Statistical Analysis and Data Display*, Springer, pp. 377-426.

[33] «F Distribution and ANOVA,» [En ligne]. Available: http://cnx.org/content/m17065/.

[34] T. Archdeacon, «Correlation and Regression Analysis: A Historian's Guide.,» 1994.

[35] Y. Dodge, «The Concise Encyclopedia of Statistics.,» (2008).

[36] L. Gonick, « The Cartoon Guide to Statistics.,» 1993.

Références

[37] Z. A. A. Alyasseri, «Survey of Parallel Computing with MATLAB,» July 2014.

[38] Wenjing Gao, Qian Kemao, Haixia Wang, Feng Lin et Hock Soon Seah, «Parallel computing for fringe pattern processing :Amulticore CPU approachin MATLAB environment,» 6June2009.

[39] Le Thi My Hanh, Nguyen Thanh Binh et Khuat Thanh, Parallel Mutant Execution Techniques in Mutation Testing Process for Simulink Models, Vietnam: Journal of Telecommunications and Information Technology, 2017.

[40] «Parallel Computing Toolbox :User's Guide,» Mathworks.

[41] «Perform Parallel Computation onMulticore computers and computer clusters,» Mathworks.

[42] Dimitris N. Varsamis, Christos Talagkozis, Paris A. Mastorocostas, Evangelos Outsios et Nicholas P. Karampetakis, «The performance of the MATLAB Parallel Computing Toolbox in specific problems,» *Advances in Information Science and Applications* , vol. 1, 2014.

[43] A. Charles, «Parallel MATLAB,» August 3, 2010.

[44] CAO Jun-jun, FAN Shan-shan et YANG Xuan, «Spmd Performance Analysis With Parallel Computing of Matlab,» 2012 .

[45] Jiawei Han, Micheline Kamber et Jian Pei, «Data mining: concepts and techniques,» chez *The Morgan Kaufmann Series in Data Management Systems*, Elsevier, Amsterdam, 2011.

[46] H. P. Cover TM, «Nearest neighbor pattern classification.,» *IEEE Transactions on Information Theory,* p. 21–27, 1967.

[47] Evelyn F et Hodges JL Jr , Discriminatory analysis-nonparametric discrimination: consistency properties., Technical report, California University, Berkeley : International Statistical Institute (ISI), 1951.

[48] Mohssen Mohammed, Muhammad Badruddin Khan et Eihab Bashier Mohammed Bashier, Machine learning:algorithms and applications, CRC Press, Boca Raton, 2016.

Références

[49] Danial T.Larose, Data mining and predictive analytics, New York: Wiley, 2015.

[50] Najat Ali, Daniel Neagu et Paul Trundle, «Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets,» 24 September 2019.

[51] Larose DT et Larose CD , «Discovering knowledge in data: an introduction to data mining.,» 2014.

[52] «Bioinformatics toolbox.,» [En ligne]. Available: https://www.mathworks.com/help/bioinfo/index.html.. [Accès le 10 07 2020].

[53] Nikolaos Ploskas et Nikolaos Samaras, in GPU Programming in MATLAB, sciencedirect, 2016.

[54] «What Is Parallel Computing?,» mathworks, [En ligne]. Available: https://www.mathworks.com/help/parallel-computing/what-is-parallel-computing.html.

[55] «Universalité de l'ADN,» [En ligne]. Available: https://www.maxicours.com/. [Accès le 02 03 2020].