UNIVERSITÉ DU QUÉBEC À MONTRÉAL

Analyse statistique des données d'une expérience de microréseau

MÉMOIRE

PRÉSENTÉ

COMME EXIGENCE PARTIELLE

DE LA MAÎTRISE EN MATHÉMATIQUES

PAR

MOURAD DAHHOU

MARS 2006

REMERCIEMENTS

Je tiens à remercier Madame Brenda MacGibbon, ma directrice de recherche, pour la grande disponibilité dont elle a fait preuve, ses conseils qui m'ont été très utiles pour réaliser cette étude, sa présence et sa grande patience. Je la remercie également pour le soutien financier de ses octrois CRSNG et FQRNT que j'ai reçus tout au long de mes études en maîtrise.

Je remercie aussi mon co-directeur M. Glenn Shorrock pour son aide et ses conseils qui m'ont été très utiles pour réaliser ce travail.

J'aimerais aussi remercier tous mes professeurs de statistique, pour ce qu'il m'ont appris ainsi que Manon Gauthier, Gisèle Legault et Bertrand Fournier pour leur disponibilité à répondre à mes nombreuses questions.

Mes remerciements à mes parents et à ma conjointe, pour leur support qui m'a encouragé à aller plus loin dans mes études.

Merci à toutes les personnes qui m'ont aidé et dirigé dans la réalisation de ce mémoire.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

RÉSUMÉ

Le but de ce mémoire est l'étude des méthodes d'analyse de données d'une expérience de microréseau. Nous nous intéressons au problème des tests d'hypothèses multiples, de contrôle de l'erreur de type 1 et de taux de faux positifs. Nous décrivons les techniques de Dudoit et *al.* (2003), de Hochberg (1988), de Benjamini et Hochberg (1995) et de Westfall et Young (1993) entre autres. Les problèmes des tests d'hypothèses multiples, les mesures différentes de l'erreur du type 1 et le contrôle de telles erreurs sont discutées ici d'après Dudoit et *al.* (2003). Nous nous intéressons également à la théorie des inégalités de probabilité dues à Dunn (1967), Jogdeo (1970, 1977) et Simes (1986) parce que ces inégalités forment la base de la méthodologie pour contrôler les différentes erreurs de type 1.

De plus, une description d'un progiciel informatique pour faire une telle analyse de données de microréseau est aussi présentée, une analyse utilisant une des méthodes de Dudoit et *al.* (2003) ainsi que des données de Golub et *al.* (1999) avec les gènes de deux types de leucémie. Nous avons introduit la méthode de ≪Jackknife≫ comme un autre contrôle du taux des faux positifs pour cet exemple. Le mémoire termine avec une étude de simulation afin d'illustrer les différences entre les mesures de l'erreur du type 1.

Mots clés : Micoréseaux, génétique, inférence statistique, taux d'erreur, jackknife, simulations.

INTRODUCTION

0.1 Description d'une expérience de microréseau

La description des expériences de microréseau d'ADN peut être trouvée dans plusieurs références. Nous citons Brown et Botstein (1999), Lee et *al.* (2000)et Dabrowski (2005). Dans ce qui suit nous allons donner une brève description des expériences de microréseau d'ADN selon la présentation de Lee et *al.* (2000) et Dabrowski(2005).

Le développement de l'expérience de microréseau afin de déterminer quels gènes sont ou ne sont pas impliqués dans certains processus biologiques a produit beaucoup de nouvelles et intéressantes techniques statistiques pour analyser ces résultats. Le but de ce mémoire est de décrire le problème en détail et de présenter certaines de ces techniques statistiques. Nous avons choisi d'illustrer une telle analyse en utilisant des procédures de Dudoit et *al.* (2003), les données de Golub et *al.* (2003) ainsi que quelques données concernant les tumeurs de Tibshirani et *al.* (2001). Afin de valider les expressions des gènes choisies nous employons la méthode de ≪Jackknife≫.

Une expérience typique illustrant cet article utilise des points de matière génétique connus sous le nom de ≪probe≫ sur une plaque en verre afin de déterminer la quantité de ce matériel présent dans une préparation donnée de test. Une glissière simple contient beaucoup (de 500 à 15000 ou plus) de points distincts, généralement disposés dans une rangée (array). Chaque point est composé de copies d'une seule pièce d'ADN complémentaire (cADN) qu'on appellera un gène, quoique la matière génétique dans ce point puisse ne pas correspondre exactement à celui du gène simple. Chaque gène peut être répété un nombre restreint de fois sur la glissière mais le nombre de gènes reste

relativement grand. Habituellement on prend le cas le plus typique où il y a 2 points sur la glissière pour chaque gène. D'autres arrangements existent et les méthodes utilisées peuvent être adaptées à ces cas. La préparation à appliquer au microréseau contient le cADN (ou le mARN) de tous les gènes de la cellule ou de l'organisme à l'étude. Ces gènes sont connus (provenant par exemple du projet humain de génome) mais le degré auquel chaque gène est exprimé en cellule vivante ne l'est pas. Par exemple, différents gènes peuvent être déclenchés dans l'activité à chaque moment pendant la vie d'une cellule, avec différents rôles à différents moments, ou en réponse à un simulant externe tel que la chaleur. Pour la simplicité, on peut penser à cette préparation du cADN de cellules comme purée du matériel génétique d'un groupe de cellules d'une ligne unicellulaire. Il se compose généralement en utilisant une certaine méthode d'inscription, telle les étiquettes radioactives ou de fluorescent, de sorte que le futur de cet échantillon puisse être commodément tracé. Quand cette préparation est appliquée à la glissière, elle sera enlevée à la fin d'une période de temps fixe. Ainsi, à la fin de l'expérience, on doit constater que certains points sur la projection de diapositives ont une signature radioactive (ou fluorescente) forte et que d'autres ne l'ont pas. De cette façon, on peut identifier quels gènes sont en activité dans la cellule originale. Le plus grand intérêt est le degré auquel chaque gène est exprimé qu'on mesure par l'intensité du signal au point approprié. Le degré d'expression indique le degré auquel le gène est en activité et ceci peut changer avec le type de la cellule, l'effort auquel il a été soumis et le temps.

Très souvent on utilise les types de microréseau suivants : les microréseaux Oligonucléotide qui sont des morceaux courts de ADN, environ 25 bases dont chaque 'oligo' est composé d'un appariement parfait et une disparité qui diffère de l'appariement parfait à un seul endroit (à une autre base au milieu, à la 13ème position).

Par contre, les cADN repérés sont des morceaux de cADN coloriés en rouge ou vert et longs (entre 200 et 500 bases). Dans le laboratoire, on crée beaucoup de fragments de cADN de gènes connus (les 'probes') et on forme des microréseaux sur une

plaque. Après, on prélève des mARN dans des cellules provenant de nos patients et on re-transcrit ce mARN en cADN (le 'target'), les molécules cADN-'target' et cADN-'probe' sont hybridisées.

Il faut se souvenir que les données analysées par les statisticiens subissent toujours un pré-processus et nous supposons que les données sont déjà prêtes à analyser avec les méthodes décrites dans ce mémoire.

Pour mieux comprendre la description du problème, permettons nous quelques définitions pertinentes. (Une bonne référence pour ces définitions est Lange (2002)).
Une cellule est une unité de la matière vivante capable de se reproduire, entourée d'un tissu de molécules de gras ; la membrane contient l'ADN, les robosomes, etc.
L'ADN (acide des oxy-ribo-nucléïque, molécule gêante de la cellule) est une molécule très stable qu contient l'information génétique qui reste inchangée à travers les générations. Elle est formée de deux chaînes parallèles et sur chaque chaîne les composantes chimiques sont liées entre elles de façon linéaire (sans branches). Cependant, ces fils sont très contorsionnés et l'ADN est beaucoup plus longue que la cellule qui le contient. L'ADN est le même dans toutes les cellules d'un même organisme vivant.

Le gène est une portion de l'ADN, ou une région spécifique de l'ADN ; chaque gène codifie une protéine spécifique.

L'expression du gène est le processus par lequel l'information contenue dans l'ADN est activée : d'abord elle est convertie en ARN et après en protéine. Ainsi on a Gène/ADN $\rightarrow$ ARN $\rightarrow$ Protéine. Ce processus a lieu en deux étapes : transcription et traduction. Même si l'ADN est le même dans toutes les cellules, il s'exprime différemment dans chaque cellule (ou groupe de cellules).

La transcription est la synthèse (création) d'une molécule d'ARN qui utilise une molécule d'ADN comme modèle ('template'). Elle se fait à l'aide d'un enzyme, ARN polymerase, qui reconnaît le début d'un gène à partir d'un site sur l'ADN qui s'appelle un promoteur. L'enzyme se place sur ce promoteur qui indique la bonne direction (sur l'ADN) et après il sépare l'ADN (sur une petite portion), ce qui permet à l'ARN de copier cette portion par le principe d'appariement des 4 bases. Le mARN (ARN messager) transporte l'information obtenue de l'ADN vers des ribosomes qui participent à la traduction.

La traduction est un message écrit avec l'alphabet à 4 lettres de l'ADN et du mARN et est traduit dans le langage à 20 lettres des protéines, la traduction a lieu sur les ribosomes.

La transcription à rebours ≪reverse transcription≫ est le processus par lequel le mARN d'un gène particulier est retranscrit en ADN complémentaire (d'où le nom de cADN).

Une hybridation est un appariement entre des morceaux de chaînes de ADN ou ARN. Afin d'obtenir cet appariement, on doit chauffer le ADN, séparer ses deux chaînes et ainsi permettre à une d'entre elles de s'apparier à un ARN (ou chaîne de ADN) disponible (si une compatibilité existe).

Une experience de microréseau en biotechnologie permet de tester plusieurs niveaux d'expression pour des millions de gènes simultanément. La question commune dans les expériences de microréseau d'ADN est l'identification des gènes dont les niveaux d'expression sont associés à une réponse ou covariable d'intérêt. Les covariables ou les réponses peuvent être continues ou censurées.

En tout cas, l'analyse des microréseaux nous amène au problème de test multiple d'hypothèse ou peut être traduit par des tests simultanés de l'hypothèse nulle pour chaque gène.

0.2 Organisation du mémoire

Le chapitre 1 donne des notions préalables en rappelant la base de l'inférence statistique comme les différents tests d'hypothèse.

Le chapitre 2 discute de l'inférence statistique pour les expériences de microréseau, y compris les tests multiples, les différentes mesures de l'erreur de type 1 et le contrôle de telles erreurs d'après Dudoit et *al.* (2003).

Dans le chapitre 3, nous discutons en détail un progiciel dû à Dudoit et *al.* (2004) pour l'analyse des données d'une expérience de microréseau. Nous faisons une analyse des données de Golub et *al.* (1999) qui concerne deux types de leucémie. De plus, nous introduisons la méthode de ≪jackknife≫ comme une aide à déterminer les gènes différentiellement exprimés. Nous utilisons cette méthode de ≪jackknife≫ dans l'analyse des données de Tibshirani et *al.* (2003) qui concerne des tumeurs traitées et non traitées. Nous incluons une étude de simulation pour illustrer les différences entre les mesures de l'erreur de type 1 décrites au chapitre 2.

CHAPITRE I

NOTIONS PRÉALABLES

1.1 Rappels des tests d'hypothèses

Rappelons les définitions nécessaires de l'inférence statistique comme le test d'hypothèse et les erreurs de type 1 et 2. L'exemple paramètrique le plus simple est le suivant : Soit θ un paramètre lié à un espace Ω. On suppose que Ω peut être divisé en deux sous ensembles disjoints Ω_0 et Ω_1, et que le statisticien doit décider si le paramètre inconnu θ est dans Ω_0 ou dans Ω_1. Notons l'hypothèse nulle par H_0, l'hypothèse que $\theta \in \Omega_0$ et l'hypothèse alternative par H_a, l'hypothèse que $\theta \in \Omega_1$. Puisque les sous-ensembles Ω_0 et Ω_1 sont disjoints et $\Omega_0 \bigcup \Omega_1 = \Omega$, alors une des hypothèses H_0 et H_a doit être vraie.

Un problème de ce type s'appelle un test d'hypothèse. En général et dans beaucoup de problèmes, on prend la décision d'accepter ou de rejeter l'hypothèse nulle en se basant sur des valeurs observées d'une statistique qui nous donnent l'information sur le paramètre inconnu θ. La procédure pour décider si on doit accepter ou rejeter l'hypothèse nulle H_0 s'appelle une méthode de test ou simplement un test.
Un test d'hypothèse simple est le suivant :

$$H_0 : \theta = \theta_0 \; vs \; H_1 : \theta \neq \theta_0.$$

Un exemple un peu plus complexe étudié ici est le suivant : l'hypothèse nulle qu'il n'y

a pas d'association entre le niveau d'expression du gène et la réponse ou la covariable.

Dans n'importe quelle situation de test d'hypothèse deux types d'erreur peuvent être commises, soit l'erreur de type 1 et l'erreur de type 2. On appelle erreur de type 1 ou faux positif, le cas où le test rejette l'hypothèse nulle H_0 quand elle est vraie. Dans notre cas, l'erreur de type 1 correspond à ce qu'un gène est différentiellement exprimé alors qu'il ne l'est pas. La probabilité d'une erreur de type 1 est notée par α. On fixe α à l'avance.

On appelle erreur de type 2 ou faux négatif, le cas où le test accepte l'hypothèse nulle H_0 quand elle est fausse. Dans notre cas, l'erreur de type 2 correspond à ce qu'aucun gène n'est différentiellement exprimé alors qu'il doit avoir des gènes différentiellement exprimés. La probabilité d'une erreur de type 2 est notée pas β. Si H_0 est fausse, on aimerait rejeter H_0 le plus souvent possible.
La puissance du test est définie comme suit : la probabilité qu'un test de niveau fixe α rejette H_0 lorsqu'une alternative particulière est vraie s'appelle la puissance du test envers cette alternative.

On dit qu'un test est plus conservateur qu'un autre, si la probabilité de rejeter l'hypothèse nulle de ce test est plus petite. Si on diminue l'erreur de type 1, l'erreur de type 2 augmente et vice versa.

Puisque une expérience de microréseau mesure des niveaux d'expression pour des milliers de gènes simultanément, alors de grands problèmes de multiplicité de tests sont produits.

Chaque test a une probabilité d'erreur de type 1 et cette probabilité augmente

tant que le nombre d'hypthèses nulles à tester augmente. Il en résulte qu'on va définir un taux d'erreur de type 1 et puis on va chercher à trouver des méthodes de tests multiples qui tiennent compte ou qui contrôlent ce taux d'erreur.

1.2 Exemple de différents tests d'hypothèse utilisés en analyse statistique de microréseau

L'exemple suivant illustre ce type de test : on suppose que $X \sim N(\theta, \sigma^2)$ et on veut tester l'hypothèse simple que $\theta = \theta_0$ quand on observe $X_1, \ldots, X_n$ i.i.d. $N(\theta, \sigma^2)$. On suppose que σ^2 est connu et on fixe α en avance.
La statistique de test est :

$$z = \frac{\bar{X} - \theta_0}{\sigma/\sqrt{n}}$$

On rejette $H_0 : \theta = \theta_0$ au niveau de signification α contre l'hypothèse alternative $H_a : \theta \neq \theta_0$ si $|z| \geq z^*$, où z^* est le point critique supérieur de niveau $\alpha/2$, noté Z_α.
La fonction de puissance du test est :

$$\beta(\theta) = P_\theta(\frac{\bar{X} - \theta_0}{\sigma/\sqrt{n}} > z^*) + P_\theta(\frac{\bar{X} - \theta_0}{\sigma/\sqrt{n}} < -z^*)$$

Donc :

$$\beta(\theta) = P_\theta(Z > z^* + \frac{\theta_0 - \theta}{\sigma/\sqrt{n}}) + P_\theta(Z < -z^* + \frac{\theta_0 - \theta}{\sigma/\sqrt{n}}),$$

où $Z = \frac{\bar{X} - \theta}{\sigma/\sqrt{n}}$ est une loi normale $N(0,1)$ et P_θ représente la probabilité sous la loi $N(\theta, \sigma^2)$. Il est évident de voir que $\beta(\theta)$ est une fonction croissante de θ. Et la puissance du test est $\beta = 1 - \alpha$ où $\alpha = Sup_{\theta \in H_0}\beta(\theta)$. Ce maximum est atteint pour $c = z_\alpha$ où $z_\alpha = \phi^{-1}(\alpha/2)$, avec $\phi(.)$ qui désigne la fonction cumulative d'un loi $N(0,1)$.

Dans le cas où la variance σ^2 est inconnue, on l'estime par :

$$S^2 = \sum_{i=1}^{n}(X_i - \bar{X})^2/n - 1$$

et on rejette l'hypothèse nulle $H_0 : \theta = \theta_0$ contre l'alternative $H_a : \theta \neq \theta_0$, si :

$$\bar{X} > \theta_0 + t_{n-1,\alpha}\sqrt{S^2/n} \; ou \; \bar{X} < \theta_0 - t_{n-1,\alpha}\sqrt{S^2/n}$$

où $t_{n-1,\alpha} = P_\theta(T \leq t)$ et $T = \frac{\bar{X}-\theta}{S/\sqrt{n}}$ suit une loi de student t de degré de liberté *n-1*.

CHAPITRE II

L'INFÉRENCE STATISTIQUE POUR LES EXPÉRIENCES DE MICRORÉSEAU

Deux importants problèmes sont associés à l'analyse statistique d'expérience de microréseau. Premièrement, on doit choisir pour chaque problème un test approprié. On a donné quelques exemples de tests utilisés dans le chapiter 2. Deuxièmement, il faut chercher à résoudre le problème de test d'hypothèses multiples. Une experience de microréseau produira autant de tests d'hypothèses que le nombre de gènes à étudier. Nous observerons probablement beaucoup de différences statistiquement significatives même s'il n'y a aucune expression différentielle des gènes en raison du grand nombre de tests réalisés, qui peut être parfois de l'ordre de 10.000. Il est évident que les valeurs de p doivent être ajustées d'une manière quelconque pour tenir compte du problème de multiplicité.

Dans cette partie, on va discuter de différentes approches aux problèmes d'inférence selon Dudoit et et *al.* (2003) afin de déterminer quels gènes se sont exprimés d'une manière significative dans une expérience de microréseau. On va également présenter des notions de base et des procédures pour des tests multiples et on va discuter de certaines propositions.

2.1 Tests multiples pour des expériences d'ADN de microréseau

Considérons une expérience de microréseau d'ADN : on a des données sur m gènes pour n échantillons mARN. Le tableau suivant illustre la présentation des données.

Tab. 2.1 Exemple de présentation de données

échantillon 1	échantillon 2	échantillon 3	échantillon 4	échantillon i	....
Gène 1	0.48	0.30	0.80	1.51	....
Gène 2	-0.10	0.59	0.24	0.06	....
Gène 3	0.15	0.74	0.06	0.10	....
Gène 4	-0.45	-1.03	-0.79	-0.56	....
Gène 5	-0.06	1.06	1.35	1.27	....

L'élément x_{ji} représente donc le niveau d'expression du gène j dans l'échantillon mARN i.

Ainsi, pour chaque échantillon, on a une réponse (exemple : type de tumeur, type de traitement, etc..) et le but dans ce cas sera d'identifier l'expression des différents gènes exprimés par rapport à chaque réponse. Les données pour un échantillon i sont composées d'une variable réponse y_i et de l'expression du gène $\mathbf{x}_i = (x_{1i}, \ldots, x_{mi})$ pour tout i = 1,...,n .

Ainsi, X_j est la variable aléatoire correspondant aux mesures pour le gène j, $j = 1, 2, \ldots, m$ et Y est la variable aléatoire réponse.

Les données de l'expression du gène peuvent être écrites sous la forme matricielle suivante :

X = (x_{ji}) de rang $m \times n$ où chaque ligne de la matrice représente les gènes et chaque colonne représente un échantillon mARN. En général n est compris entre 10 et 100 par contre m peut être égale à plusieurs milliers. Les paires $\{(x_i, y_i)\}_{i=1,\ldots,n}$ seront considérées comme l'échantillon aléatoire de la population d'intérêt.

Le but sera donc d'utiliser les données sur l'échantillon $\{\mathbf{x}_i, y_i)\}_{i=1,\ldots,n}$ pour faire l'inférence sur la population ou encore de tester l'hypothèse concernant la distribution jointe de $X = (X_1, \ldots, X_m)$ et de la covariable Y. En particulier, on s'intéresse généralement à voir quels gènes se sont exprimés d'une manière différente selon la réponse.

La question biologique peut être traduite par des tests multiples d'hypothèses : tester simultanément pour chaque gène j l'hypothèse H_j : pas d'association entre X_j et Y. L'approche standard du test est :

(1) Choisir un des tests univariés mentionnés dans le chapitre 2. En général, la statistique appropriée du test dépendra de la méthode expérimentale et du type de réponse ou de covariable.

(2) Calculer la statistique T_j pour chaque j qui est fonction de X_j et Y. Nous ne discuterons pas, pour le moment du choix de la statistique T_j, sauf pour dire que pour chaque gène j, l'hypothèse nulle H_j est évaluée en se basant sur la statistique T_j qui est fonction de X_j et de Y. Notons par t_j une réalisation de la variable aléatoire T_j et nous supposerons que l'hypothèse nulle H_j est rejetée pour des grandes valeurs de $|t_j|$, c'est-à-dire un test bilatéral.

(3) Appliquer une méthode de tests multiples pour déterminer quelles hypothèses à rejeter tout en tenant compte d'un taux d'erreur de type 1 convenablement défini.

La méthode la plus conservatrice pour résoudre le problème est celle de Bonferroni, qui sera définie prochainement.

Le problème (3) est le sujet de cet mémoire. Nous suivrons l'approche de Dudoit et *al.* (2004). Les expériences d'ADN de microréseau présentent un nouveau domaine d'application pour les méthodes de tests multiples. Dans le reste de ce chapitre, nous présenterons des notions de base des tests multiples comme Dudoit et *al.* (2003) ont fait et nous discuterons ensuite leurs propositions récentes pour traiter le problème de multiplicité dans des expériences de microréseau ainsi que les différentes méthodes de

contrôle des taux d'erreurs proposées par eux pour les tests multiples.

2.2 Mesures différentes du taux d'erreur de type1 et de type2

Ici on utilise les notations de Dudoit et *al.* (2003) et on suit leur approche. Considérons le problème où on teste simultanément m hypothèses nulles H_j *pour* $j = 1, \ldots m$ et notons par R le nombre de rejets. On peut récapituler le résultat dans le tableau suivant :

Tab. 2.2 Tableau récapitulatif des résultats

Nombre de	Non rejet	Rejet	Total
Hypothèse nulle vraie	U	V	m_0
Hypothèse nulle fausse	T	S	m_1
	m-R	R	m

On suppose comme étant connu d'avance le nombre d'hypothèses m à tester ; par contre, le nombre m_0 d'hypothèses nulles vraies et le nombre $m_1 = m - m_0$ d'hypothèses nulles fausses sont deux paramètres inconnus. R est une variable aléatoire observable ; par contre S,T,U et V sont des variables aléatoires non observables.

Dans le contexte de microréseaux, il y a une hypothèse nulle H_j pour chaque gène j et le rejet de H_j est équivalent à dire que le gène j s'est exprimé différentiellement dans les groupes considérés. Pour avoir un cas idéal, il faut penser à réduire le nombre V de faux positifs ou les erreurs de type 1 et le nombre T de faux négatifs ou les erreurs de type 2. Ceci revient à maximiser la puissance du test dans la classe des tests avec un taux d'erreurs de type 1 au plus égal à α. Habituellement c'est impossible de minimiser les erreurs de type 1 et de type 2 en même temps.

En testant une hypothèse simple H_0, la probabilité de commettre une erreur de type 1 ou de rejeter l'hypothèse nulle quand elle est vraie est généralement associée à un niveau de rejet α et ceci peut être calculé en choisissant une valeur critique C_α telle que :

$$Pr(\mid T_1 \mid \geq C_\alpha | H_0) \leq \alpha,$$

où T_1 est la statistique du test et on rejette H_0 quand $\mid T_1 \mid \geq C_\alpha$.
Une généralisation pour la situation des tests multiples est possible. Nous commençons par les définitions suivantes de Dudoit et *al.* (2003).

Définition 2.2.1 : (Dudoit et *al.* (2003))

Le taux d'erreur par-comparaison (≪ Per-comparison error rate ≫), PCER, est défini comme étant la valeur prévue du nombre d'erreurs de typel divisé par le nombre d'hypothèses m nulles à tester

$$PCER = E(V)/m.$$

Définition 2.2.2 : (Dudoit et *al.* (2003))

Le taux d'erreur espéré par famille (≪ Per-family error rate ≫) PFER est défini comme le nombre prévu d'erreurs de type 1

$$PFER = E(V).$$

Définition 2.2.3 :

Le taux d'erreur probabilité par famille (≪ Family-wise error rate ≫) FWER est

défini comme la probabilité d'avoir au moins une erreur de type 1

$$FWER = Pr(V \geq 1).$$

Dudoit et *al.* (2003) ont défini deux sortes de taux d'erreurs par famille (FWER), le taux d'erreur FWER calculé sous l'hypothèse nulle complète (c'est à dire que toutes les H_i sont vraies) et le FWEP calculé sous l'hypothèse nulle partielle (c'est à dire qu'un certain sous-ensemble d'hypothèses nulles est vrai noté $H_{j_1}, \ldots, H_{j_l}$). Le FWER est la probabilité de rejeter une hypothèse nulle vraie. Et le FWEC est défini par :

$$FWEC = Pr(Rejeter\ au\ moins\ un\ H_i | toutes\ les\ H_i\ sont\ vraies).$$

et

$$FWEP = Pr(rejeter\ au\ moins\ une\ H_i | H_{j_1}, \ldots, H_{j_l})\ sont\ vraies).$$

Clairement, le FWEP dépend des sous-ensembles particuliers $j_1, \ldots, j_l$ de vraies hypothèses nulles. On dit qu'une méthode de tests simultanés contrôle le FWER dans le sens faible si $FWER \leq \alpha$, indépendamment des sous-ensembles $j_1, \ldots, j_l$ d'hypothèses vraies. Il est clairement plus souhaitable que le test d'hypothèses simultanées contrôle fortement le FWER.

Selon Hochberg et de Tamhane (1995), des méthodes de tests simultanés sont généralement conçues pour contrôler le taux d'erreur de famille (Family-wise error rate (FWER)).

Le FWER a une autre propriété utile parce que l'erreur de type 1 est la probabilité de rejeter une hypothèse lorsqu'elle est vraie. L'erreur de type 2 est la probabilité d'accepter une fausse hypothèse. Kaiser (1960) a défini l'erreur de type 3 comme étant la probabilité de mal classer le signe de l'effet. Par exemple, si l'hypothèse H : $\mu = 0$ est rejetée en faveur de l'hypothèse alternative à un niveau de test $\alpha = 0.05$, est-il possible de dire que le signe de la vraie moyenne μ est le même que celui de la moyenne estimée $\bar{y}$?

Typiquement pour les statistiques de t et de Z, la réponse est oui à cause de la grande correspondance entre le test d'hypothèse et l'intervalle de confiance. Spécifiquement supposons que le test est basé sur la statistique de test $t = \bar{y}/(s/\sqrt{n})$ et qu'on rejette l'hypothèse nulle H si $|t| > t_{.975}$, où $t_{.975}$ est le 0.975 quantile de la distribution t avec n-1 degrés de liberté.

L'intervalle de confiance associé pour μ est $\bar{y} \pm t_{.975}s/\sqrt{n}$. Dans ce cas, les deux queus de rejet de H à un niveau de signification $\alpha = 0.05$ avec une moyenne d'échantillon $\bar{y} > 0$ sont équivalentes à avoir à 95% un intervalle de confiance entier au-dessus de la valeur présumée 0 :

$$t > t_{.975} \quad \Leftrightarrow \bar{y} - t_{.975}s/\sqrt{n} > 0$$

Réciproquement, les deux queues de rejet de H à un niveau de signification $\alpha = 0.05$ avec une moyenne d'échantillon $\bar{y} > 0$ est équivalent à avoir avec probabilité 0.95 un intervalle de confiance entier au-dessous de la valeur présumée 0 :

$$t < -t_{.975} \quad \Leftrightarrow \bar{y} - t_{.975}s/\sqrt{n} < 0$$

Pour le cas de tests multiples, le FWER de type 3 est la probabilité que le signe de n'importe quel effet testé est mal classifié. Quand un test simultané d'hypothèse contrôle le FWER de type 3, on peut dire que les vrais signes de tous les effets significatifs sont dans les mêmes directions que les signes estimés. Comme dans la situation de test univarié, cette propriété peut être démontrée quand il y a une correspondance directe entre la méthode de test simultané et le procédé simultané d'intervalle de confiance. C'est un avantage du contrôle du taux d'erreur FWER.

Définition 2.2.4 : Dudoit et *al.* (2003)

Le taux de fausse découverte (≪ False discovery rate ≫) FDR est défini comme

la proportion prévue d'erreurs de type 1 parmi les hypothèses rejetées

$$FDR = E(Q).$$

où $Q = V/R$ si $R > 0$ et égale à 0 si $R = 0$.

2.3 Contrôle d'erreur de type 1

On peut définir le contrôle du taux d'erreur de type 1 pour chaque définition de l'erreur de type 1 selon Dudoit et *al.* (2003). On dit qu'une méthode de test multiple contrôle le taux d'erreur de type 1 (PCER, PFER, FWER, FDR) à un niveau α, si ce taux d'erreur est inférieur ou égal à α, quand la procédure appliquée aux données produit R hypothèses nulles rejetées. Il est important de distinguer entre contrôle fort et contrôle faible.

Notons d'abord que les taux d'erreur PCER, PFER, FWER et le FDR sont définis pour une expérience avec des données vraies et typiquement de distribution $X = (X_1, \cdots, X_n)$ et de Y qui sont inconnues. En particulier, ils dépendent d'un sous-ensemble $\Lambda_0 \subseteq \{1, \cdots, m\}$, les hypothèses nulles vraies pour cette distribution. Par exemple pour le FWER :

$$\begin{aligned} FWER &= Pr(V \geq 1 | \bigcap_{j \in \Lambda_0} H_j) \\ &= Pr(\textit{Rejeter au moins un } H_j; j \in \Lambda_0 | \bigcap_{j \in \Lambda_0} H_j) \end{aligned}$$

où $\bigcap_{j \in \Lambda_0} H_j$ est l'intersection des sous-ensembles d'hypothèses vraies pour les données provenant de la distribution commune.

Le contrôle fort est défini par Dudoit et *al.* (2003) pour contrôler le taux d'erreur de type 1 sous n'importe quelle combinaison de vraies et de fausses hypothèses nulles. C'est-à-dire pour n'importe quel sous ensemble d'hypothèses nulles vraies $\Lambda_0 \subseteq$

$\{1, \cdots, m\}$. Par contre, le contrôle faible est défini pour contrôler le taux d'erreur de type 1 dans le cas où toutes les hypothèses nulles sont vraies, c'est à dire dans le cas de l'hypothèse nulle complète : $H_0^c = \bigcap_{j=1}^{m_0} H_j$ *avec* $m_0 = m$.

En général l'hypothèse nulle complète H_0^c n'est pas réaliste et donc le contrôle faible sera insuffisant. En réalité, quelques hypothèses nulles peuvent être fausses mais le sous-ensemble Λ_0 est inconnu. Donc le contrôle fort assure que le taux d'erreur de type 1 soit contrôlé sous des données inconnues générées par la distribution. En particulier dans le cas des données de microréseau, où il est très peu probable qu'aucun gène ne soit exprimé, il semble important d'utiliser le contrôle fort du taux d'erreur de type 1. Notons que le concept de contrôle fort et faible s'applique à chacun des taux d'erreur de type 1 définis précédemment, le PCER, PFER, PWER et le FDR.

Dans tout ce qui suit et sauf indication contraire, les probabilités et les espérances seront calculées sous une combinaison de vraies et de fausses hypothèses nulles, c'est-à-dire sous l'hypothèse nulle composée : $\bigcap_{j \in \Lambda_0} H_j$ correspondant aux données provenant de la distribution, où $\Lambda_0 \subseteq \{1, \cdots, m\}$ est de taille m_0.

Dans la classe des méthodes de tests multiples qui contrôlent un taux d'erreur donné de type 1 à un certain niveau α acceptable, on cherche les procédures qui maximisent la puissance du test ou encore on cherche à réduire au minimum le taux d'erreur de type 2.

Comme avec des taux d'erreur de type 1, le concept de la puissance a été généralisé d'un test simple à un test multiple (cf. Dudoit et *al.* (2003)). Dans ce cas on définit trois concepts de puissance communs :

(1) La probabilité de rejeter au moins une hypothèse nulle fausse :

$$Pr(S \geq 1) = Pr(T \leq m_1 - 1).$$

(2) La probabilité moyenne de rejeter les hypothèses nulles fausses :

$$E(S)/m_1.$$

(3) La probabilité de rejeter toutes les hypothèses nulles fausses :

$$Pr(S = m_1) = Pr(T = 0).$$

Dans un esprit analogue au FDR, on peut aussi définir la puissance par :

$$E(S/R|R > 0)Pr(R > 0) = Pr(R > 0) - FDR.$$

Notons que ces probabilités dépendent du sous ensemble d'hypothèses nulles vraies $\Lambda_0 \subseteq \{1, \ldots, m\}$.

Lemme 2.3.1 (Dudoit et *al.* (2003)).
En général pour une méthode de test multiple donnée, on a :

$$PCER \leq FWER \leq PFER$$

Démonstration :
On a :

$$PCER = E(V)/m.$$

$$PFER = E(V).$$

$$FWER = Pr(V \geq 1).$$

où $V = \sum_{j=1}^{m_0} R_j$.

$$PCER = E(V)/m \quad = \quad 1/m[0P(V = 0) + 1P(V = 1) + \ldots + m_0 P(V = m_0)]$$

$$\begin{aligned} &\leq 1/m[mP(V=1)+\ldots+mP(V=m_0)] \\ &\leq m/m[P(V=1)+\ldots+P(V=m_0)] \\ &= P(V\geq 1)=FWER. \end{aligned}$$

Donc PCER $\leq$ FWER.

$$\begin{aligned} PFER=E(V) &= 0P(V=0)+1P(V=1)+\ldots+m_0P(V=m_0) \\ &\geq P(V=1)+\ldots+P(V=m_0) \\ &= P(V\geq 1)=FWER. \end{aligned}$$

■

Ainsi pour un certain critère α de contrôle des taux d'erreur de type 1, les procédures qui contrôlent le PFER sont généralement plus conservatrices ; c'est-à-dire qu'ells mènent à moins de rejets que celles qui contrôlent le FWER ou le PCER et les procédures qui contrôlent le FWER sont plus conservatrices que celles qui contrôlent le PCER. Pour illustrer les propriétés des différents taux d'erreur de type 1, on va supposer comme Dudoit et *al.* (2003) que chaque hypothèse H_j est testée individuellement à un niveau α_j et que la décision de rejeter ou non cette hypothèse nulle H_j est basée seulement sur ce test. On a vu que sous l'hypothèse nulle complète, le PFER est simplement la somme des α_j. Quant au FWER, il est fonction des niveaux de test α_j et de la distribution conjointe des statistiques de test T_j. Anisi on a :

$$\begin{aligned} PCER &= \frac{\alpha_1+\cdots+\alpha_m}{m} \\ &\leq FWER \\ &\leq PFER=\alpha_1+\cdots+\alpha_m. \end{aligned}$$

Le FDR dépend également de la distribution commune des statistiques de test et Benjamini et Hochberg (1995) ont démontré que : $FDR \leq FWER$, pour une procédure fixe avec $FDR = FWER$ sous l'hypothèse nulle complète.

■

Lemme 2.3.2 (Dudoit et *al.* (2003))

Sous l'hypothèse nulle complète on a : FWER $\leq$ FDR .

Démonstration :

Sous l'hypothèse nulle complète, on a supposé que toutes les hypothèses nulles sont vraies, ce qui est équivalent à dire que $m = m_0$ ou encore que $m_1 = 0$.

Si $m_1 = 0$ alors $T + S = 0$ et puisque T et S sont positives alors $T = S = 0$.

On aussi $V + S = R$ et puisque $S = 0$ alors $V = R$, donc :

$$\begin{aligned} FDR = E(V/R) &= E(1) \\ &= 1 \\ &\geq P(V \geq 1) = FWER \end{aligned}$$

Ce qui démontre que FWER $\leq$ FDR, d'où l'égalité.

■

L'approche classique pour mener un test multiple fait appel au contrôle fort du FWER par la méthode de Bonferroni définie dans la section 2.4. Les propositions récentes de Benjamini et Hochberg (1995) pour le contrôle du FWER dans le sens faible (à partir de FDR = FWER sous l'hypothèse nulle complète) peut être moins conservatrices que les autres FWER. Les procédures qui contrôlent le PCER sont généralement moins conservatrices que celles qui contrôlent le FDR ou le FWER, mais tendent à ignorer le problème de multiplicité. Nous démontrons que $FDR \leq FWER$ dans un cas simple présenté dans l'exemple suivant de Dudoit et *al.* (2003).

Exemple 2.3.1 (Dudoit et *al.* (2003)) : Dans l'exemple suivant de Dudoit et *al.* (2003), on décrit le comportement des divers taux d'erreur de type 1 quand le nombre d'hypothèses nulles m et la proportion d'hypothèses vraies $\frac{m_0}{m}$ changent.

Considérons les m vecteurs aléatoires gaussiens de moyenne $\mu = (\mu_1, \cdots, \mu_m)$ et de matrice de covariances égale à la matrice identité. Supposons qu'on veut tester simul-

tanément les hypothèses nulles :

$$H_j : \mu_j = 0 \; versus \; H_j^{'} : \mu_j \neq 0.$$

Considérons un échantillon aléatoire de n m-vecteurs. La procédure simple de test multiple consiste à rejeter H_j si

$$\mid \bar{X}_j \mid \geq Z_{\alpha/2}/\sqrt{n}$$

où $\bar{X}_j$ est la moyenne de la $j$ième coordonnée du n m-vecteur et $Z_{\alpha/2}$ est choisi tel que $\Phi(Z_{\alpha/2}) = 1 - \alpha/2$ où $\Phi(.)$ est la fonction de répartition gaussienne standardisée.

Soit R_j la probabilité de rejeter H_j. On a alors : $R_j = 1(\mid \overline{X_j} \mid \geq Z_{\alpha/2}/\sqrt{n})$ où $1(.)$ est la fonction indicatrice qui égale 1 quand la condition entre parenthèses est vraie, 0 si non. Notons que R_j ne peut prendre que les valeurs 0 ou 1.
Supposons et sans perte de généralité, que les m_0 vraies hypothèses nulles sont $H_1, \cdots, H_{m_0}$, alors $\Lambda_0 = \{1, \cdots, m_0\}$ et

$$V = \sum_{j=0}^{m_0} R_j \; et \; R = \sum_{j=0}^{m} R_j.$$

Lemme 2.3.3

Dans l'exemple décrit plus haut, on a que

$$PCER \leq FWER \leq PFER,$$

et de plus on a :

$$FDR \leq FWER \; si \; \mu_j = d/\sqrt{n}, \; j = m_0 + 1, \ldots m.$$

Démonstration :

Ici on suit la démonstration de Dudoit et *al.* (2003).

Si on suppose que :

$$\lambda_j = E(R) = Pr(R_j = 1) \quad = \quad 1 - \Phi(Z_{\alpha/2} - \mu_j\sqrt{n})) + \Phi(-Z_{\alpha/2} - \mu_j\sqrt{n}))$$

alors les formules analytiques pour les taux d'erreur de type 1 peuvent être déduites comme suit :

$$PFER = \sum_{j=0}^{m_0} \lambda_j$$

C'est évident parce que :

$$\begin{aligned} PFER = E(V) &= E(\sum_{j=1}^{m_0} R_j) \\ &= \sum_{j=1}^{m_0} E(R_j) \\ &= \sum_{j=1}^{m_0} \lambda_j \quad car\ E(R_j) = \lambda_j \end{aligned}$$

Et clairement :

$$PCER = (\sum_{j=0}^{m_0} \lambda_j)/m.$$

Maintenant selon Dudoit et *al.* (2003), on a :

$$FWER = 1 - \prod_{j=1}^{m_0}(1 - \lambda_j)$$

On donne notre démonstration pour cette égalité de la manière suivante :

$$\begin{aligned} FWER = Pr(V \geq 1) &= Pr(\sum_{j=1}^{m_0} R_j \geq 1) \quad car\ V = \sum_{j=1}^{m_0} R_j \\ &= 1 - Pr(\sum_{j=1}^{m_0} R_j < 1) \\ &= 1 - Pr(R_j = 0, j = 1, \ldots, m_0) \quad car\ R_j = 0\ ou\ 1 \\ &= 1 - \prod_{j=1}^{m_0} Pr(R_j = 0) \\ &= 1 - \prod_{j=1}^{m_0}(1 - Pr(R_j = 1)) \quad car\ R_j = 0\ ou\ 1 \\ &= 1 - \prod_{j=1}^{m_0}(1 - \lambda_j) \quad car\ Pr(R_j = 1) = E(R_j) = \lambda_j. \end{aligned}$$

$$FDR = \sum_{r_1=0}^{1} \cdots \sum_{r_m=0}^{1} \frac{\sum_{j=1}^{m_0} r_j}{\sum_{j=1}^{m} r_j} \prod_{j=1}^{m} \lambda_j^{r_j}(1 - \lambda_j)^{1-r_j}$$

où :

$$\lambda_j = E(R) = Pr(R_j = 1) \quad = \quad 1 - \Phi(Z_{\alpha/2} - \mu_j\sqrt{n})) + \Phi(-Z_{\alpha/2} - \mu_j\sqrt{n}))$$

Ici, si on suppose que la probabilité de rejeter H_j est $\lambda_j = \alpha$ *pour* $j = 1, \ldots, m_o$ et que $\mu_j = d/\sqrt{n}$ *pour*$j = m_0 + 1, \ldots, m$, alors les expressions des taux d'erreur s'écrivent comme suit :

$$PFER = m_0\alpha$$

$$PCER = m_0\alpha/m$$

$$FWER = 1 - (1-\alpha)^{m_0}$$

$$FDR = \sum_{s=0}^{m_1}\sum_{v=1}^{m_0} \frac{v}{v+s}\binom{m_0}{v}\alpha^v(1-\alpha)^{m_0-v} \times \binom{m_1}{s}\beta^s(1-\beta)^{m_1-s}$$

où $\beta = 1 - \Phi(Z_{\alpha/2} - d) + \Phi(-Z_{\alpha/2} - d)$.

Dudoit et *al.* donnent l'inégalité suivante dans ce cas :

$$FDR \leq FWER$$

.

où :

$$FWER = 1 - (1-\alpha)^{m_0}$$

et

$$FDR = \sum_{s=0}^{m_1}\sum_{v=1}^{m_0} \frac{v}{v+s}\binom{m_0}{v}\alpha^v(1-\alpha)^{m_0-v} \times \binom{m_1}{s}\beta^s(1-\beta)^{m_1-s}.$$

Nous démontrons l'inégalité de la manière suivante :

$$\sum_{s=0}^{m_1}\sum_{v=1}^{m_0} \frac{v}{v+s}\binom{m_0}{v}\alpha^v(1-\alpha)^{m_0-v} \times \binom{m_1}{s}\beta^s(1-\beta)^{m_1-s} \leq 1 - (1-\alpha)^{m_0}$$

On sait que dans le cas d'une loi binomiale que si $X \sim Bin(n,p)$ alors :

$$Pr(X = x) = \binom{n}{x}p^x(1-p)^{n-x} \ \leq 1, \quad x = 0, 1, \ldots, n.$$

et on a :

$$\sum_{x=0}^{n} P(X = x) = 1.$$

Alors pour une loi binomiale de paramètres m_1 et β on aura :

$$\sum_{s=0}^{m_1} \beta^s (1-\beta)^{m_1-s} = 1$$

et puisque $\frac{v}{v+s} \geq 1$, alors :

$$\sum_{s=0}^{m_1} \beta^s (1-\beta)^{m_1-s} \frac{v}{v+s} \leq 1.$$

Donc :

$$\sum_{s=0}^{m_1} \sum_{v=1}^{m_0} \frac{v}{v+s} \binom{m_0}{v} \alpha^v (1-\alpha)^{m_0-v} \times \binom{m_1}{s} \beta^s (1-\beta)^{m_1-s} \leq \sum_{v=1}^{m_0} \binom{m_0}{v} \alpha^v (1-\alpha)^{m_0-v}.$$

Il reste à démontrer que :

$$\sum_{v=1}^{m_0} \binom{m_0}{v} \alpha^v (1-\alpha)^{m_0-v} = 1 - (1-\alpha)^{m_0-v}.$$

On a :

$$\begin{aligned} 1 = [\alpha + (1-\alpha)]^{m_0} &= \sum_{v=0}^{m_0} \binom{m_0}{v} \alpha^v (1-\alpha)^{m_0-v} \\ &= \binom{m_0}{0} \alpha^0 (1-\alpha)^{m_0} + \sum_{v=1}^{m_0} \binom{m_0}{v} \alpha^v (1-\alpha)^{m_0-v} \\ &= (1-\alpha)^{m_0} + \sum_{v=1}^{m_0} \binom{m_0}{v} \alpha^v (1-\alpha)^{m_0-v} \end{aligned}$$

Donc on a :

$$\sum_{v=1}^{m_0} \binom{m_0}{v} \alpha^v (1-\alpha)^{m_0-v} = 1 - (1-\alpha)^{m_0}.$$

Ce qui démontre que $FDR \leq FWER$.

■

Dudoit et *al.* (2003) ont noté que seul le FDR dépend de la distribution de la statistique de test sous l'hypothèse alternative H_j' *pour* $j = m_0 + 1, \ldots, m$. En général, il est plus difficile de travailler avec le FDR que de calculer les trois autres taux d'erreur de type 1 décrits précédemment et c'est pourquoi on préfère travailler avec le FWER, PCER ou le PFER. Nous illustrons ces concepts avec l'exemple un peu plus complexe de Dudoit et *al.* (2003), mais nous avons changé les valeurs des paramètres.
On a calculé les différentes valeurs des taux d'erreur de type 1 PCER, FWER et FDR pour différentes valeurs de m et de $\frac{m_0}{m}$, pour une valeur de α fixe égale à 0.05 et pour différentes valeurs de α et de $\frac{m_0}{m}$, pour un m fixe égal à 100.

On a représenté les graphiques du FWER, PCER et du FDR pour différentes valeurs de $\frac{m_0}{m}$ (1.0, 0.9, 0.5 et 0.1), avec $\alpha = 0.05$ *et* $d = 1$. On a remarqué que le FWER et le PFER augmentent brusquement avec m tandis que le PCER reste pratiquement constant. Et si $\frac{m_0}{m}$ décroît, le FDR reste relativement stable et s'approche du PCER. Puis dans un autre graphique on a représenté le FWER, PCER et le PFER en fonction de α pour différentes valeurs de $\frac{m_0}{m}$ (1.0, 0.9, 0.5 et 0.1), avec $1 \leq m \leq 100$ *et* $d = 1$. On a constaté que pour des valeurs grandes de m (en particulier pour un cas d'expérience de microréseau) ces taux d'erreur tendent à atteindre un seuil. Le FWER est généralement beaucoup plus grand que le PCER. Les taux d'erreur tendent à avoir un comportement semblable pour des valeur assez grandes du nombre d'hypothèses m, avec une forte hausse du FWER à mesure que α augmente.

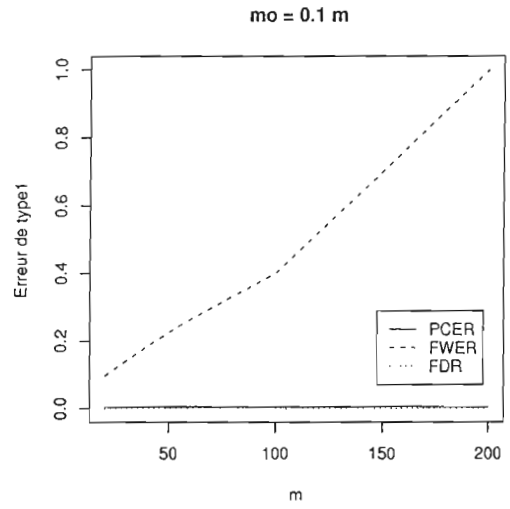

Fig. 2.1 Taux d'erreur de type 1 versus m

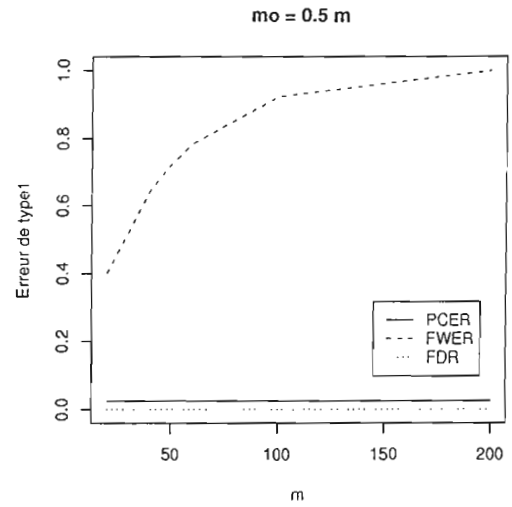

Fig. 2.2 Taux d'erreur de type 1 versus m

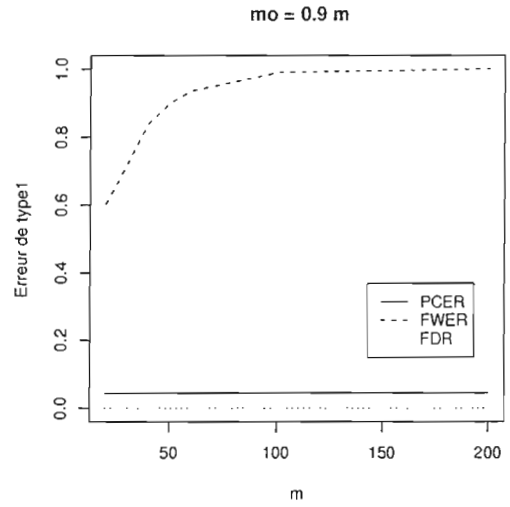

Fig. 2.3 Taux d'erreur de type 1 versus m

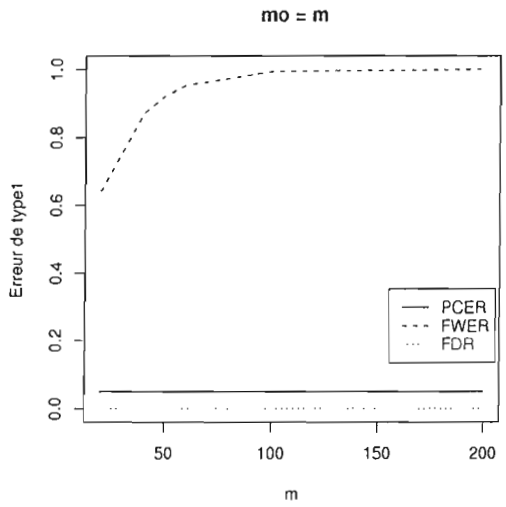

Fig. 2.4 Taux d'erreur de type 1 versus m

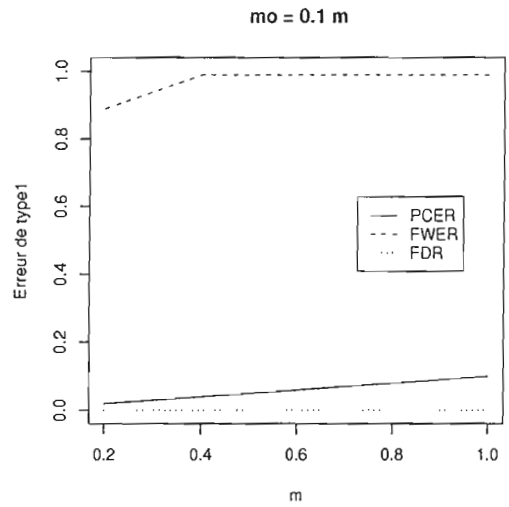

Fig. 2.5 Taux d'erreur de type 1 versus α

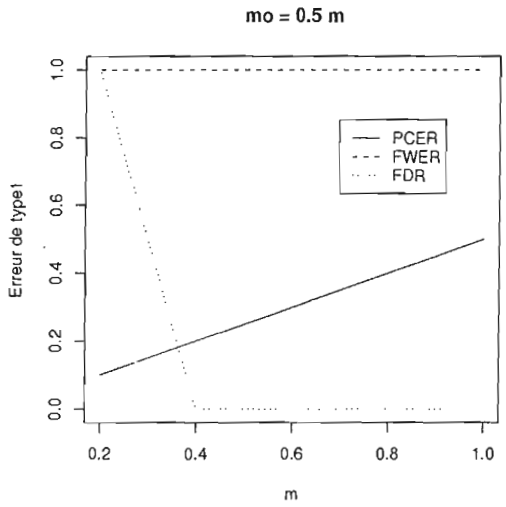

Fig. 2.6 Taux d'erreur de type 1 versus α

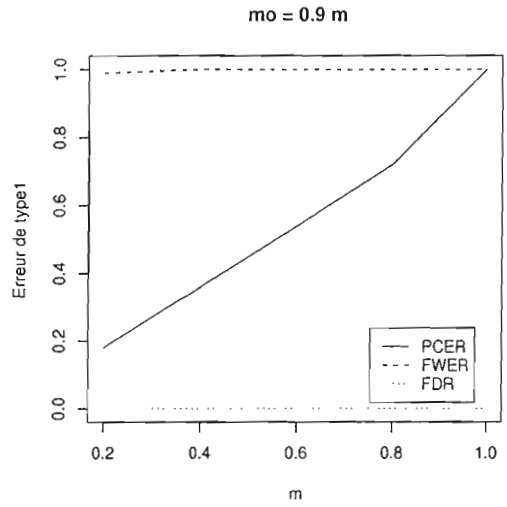

Fig. 2.7 Taux d'erreur de type 1 versus α

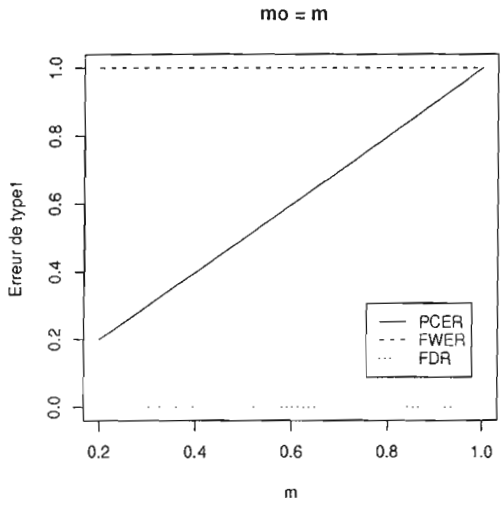

Fig. 2.8 Taux d'erreur de type 1 versus α

2.4 Contrôle du taux d'erreur de type 1

Après la définition des différents taux d'erreur de type 1, on doit définir le contrôle de telles erreurs.

Définition 2.4.1

Considérons d'abord une seule hypothèse nulle H_1 et une statistique de test T_1 pour tester H_1 à un niveau α. On suppose qu'on a une famille de tests de H_1 telle que : $S_{\alpha'} \subseteq S_{\alpha'}$ *si* $\alpha' \leq \alpha$. La région de rejet S_α, pour tout $\alpha \in [0,1]$ est telle que :
(1) $Pr(T_1 \in S_\alpha | H_1) = \alpha$.
(2) $S_{\alpha'} = \bigcap_{\alpha \geq \alpha'} S_\alpha$ pour tout α et α' pour lesquelles les régions de rejet sont définies par (1).
La valeur de p du test est définie comme étant la plus petite valeur de α qui nous permet quand même de rejeter H_1 :

$$p_1 = inf(\alpha : t_1 \in S_\alpha))$$

où t_1 est une réalisation de T_1.

Rejeter l'hypothèse H_1 quand $p_1 \leq \alpha$, mène à un contrôle du taux d'erreur de type 1 à un niveau α. Dans notre contexte, la valeur de p considérée comme étant la probabilité p_1 telle que : $p_1 = Pr(|T_1| \geq |t_1||H_1)$, soit la probabilité d'observer la statistique de test extrême dans le même sens de rejet que la valeur observée. Une généralisation du concept de la valeur de p au cas de tests multiples mène à la définition de la valeur de p ajustée.

La méthode de Bonferroni se retrouve parmi les méthodes d'ajustement de la valeur de p pour contrôler les effets des tests multiples. Pour un test simple d'hy-

pothèse, on rejette l'hypothèse nulle si la valeur de p est inférieure à un seuil α. Dans le cas où plusieurs hypothèses sont testées simultanément, la probabilité d'observer une différence considérable, ou de rejeter l'hypothèse nulle, pour un des tests, augmente avec leur nombre. Si on décide de faire m tests d'hypothèses simultanées, on doit décider de rejeter l'hypothèse nulle à partir d'une valeur ajustée de α. Il y a plusieurs méthodes d'ajustement de la valeur de alpha et parmi ces méthodes il y a la méthode de Bonferroni. C'est une méthode de tests multiples standards et elle est la plus conservatrice des méthodes. La valeur de α utilisée pour rejeter l'hypothèse nulle est divisée par le nombre m de tests d'hypothèses à faire. Explicitement, étant donné m tests T_j pour les hypothèses H_j, $1 \leq j \leq m$ et étant donné que H_0 indique que toutes les hypothèses $H_j, 1 \leq j \leq m$ sont fausses si la valeur critique de chaque test individuel est inférieure à α/m alors : Pr(acceptée $T_j|H_0) \leq \alpha/m$, pour $1 \leq j \leq m$, donc : P(quelques T_j soient acceptées $|H_0) \leq \alpha$. Cet ajustement est dans le but de contrôler très étroitement les faux positifs et la conséquence de ceci est la maximisation du nombre de faux négatifs.

La méthode de Bonferroni est utilisée pour évaluer simultanément plusieurs statistiques de test dépendantes ou indépendantes. Quand une valeur donnée α est le niveau de test pour chaque hypothèse H_j, $j = 1, \ldots, m$, elle ne le sera pas alors pour tester toutes ces hypothèses simultanément. Afin d'éviter d'avoir beaucoup de faux positifs dans le cas de tests multiples simultanés, la valeur du niveau de rejet α doit être ajustée. L'approche simple et la plus conservatrice est celle de Bonferroni. Cette méthode utilise un niveau de test corrigé α/m au lieu de α pour tester les m hypothèses nulles.

Maintenant on va définir la valeur de p ajustée d'après Dudoit et *al.* (2003) pour les tests multiples. Soit T_j la statistique de test pour tester l'hypothèse H_j et $p_j = Pr(|T_j| \geq |t_j||H_j)$ pour $j = 1, \ldots, m$. La valeur de p non ajustée correspond au test de l'hypothèse H_j (gène j) pour $j = 1, \ldots, m$.
Comme dans le cas d'un test d'une seule hypothèse simple, la procédure de tests multiples peut être définie en termes des valeurs critiques des statistiques de test ou en

termes de valeurs de p de chaque hypothèse H_j, $j = 1, \ldots, m$. Par exemple, on rejette H_j si $|t_j| \geq C_j$ ou si $p_j \leq \alpha_j$, où les valeurs critiques C_j et les α_j sont choisies pour contrôler un taux d'erreur de type 1 (FWER, PCER, PFER ou FDR) à un niveau α pré-défini.

Définition 2.4.2 (Wright (1992)) :

Pour chaque méthode de test, la valeur de p ajustée qui correspond au test d'une hypothèse H_j peut être définie comme le niveau nominal de la méthode de test auquel H_j sera juste rejetée, étant donnée la statistique de test. Dans le cas où on s'interesse au contrôle du FWER par exemple, Dudoit et *al.* (2003) montrent que la valeur de p ajustée $\tilde{p}$ pour tester l'hypothèse H_j, étant donnée une méthode de test multiple est : $\tilde{p} = inf\{\alpha \in [0,1] : H_j \text{ est rejetée à un niveau nominal du FWER} = \alpha\}$ où la valeur nominale du FWER est le niveau α auquel la méthode du test est produite.

Les variables aléatoires correspondantes aux valeurs de p ajustées et non ajustées sont notées respectivement $\tilde{p_j}$ et p_j. L'hypothèse nulle H_j est alors rejetée (le gène j est différentiellement exprimé) à une valeur du FWER α si $\tilde{p_j} \leq \alpha$. Notons que pour plusieurs méthodes, telle que la méthode de Bonferroni décrite précédemment, le niveau nominal α est habituellement plus grand que le niveau réel, ce qui en fait un test plus conservateur.

Les valeurs de p ajustées pour des procédures contrôlant d'autres types de taux d'erreur sont définies de la même manière. C'est à dire, pour des procédures de contrôle du FDR, $\tilde{p_j} = inf\{\alpha \in [0,1] : H_j \text{ est rejetée à un niveau nominal FDR} = \alpha\}$ (Yekutieli et Benjamini, 1999). Comme dans le cas simple de test d'hypothèse, un avantage de rapporter des valeurs de p ajustées, comme de décider de rejeter ou pas des hypothèses, est qu'on n'a pas besoin de déterminer à l'avance le niveau du test. Quelques méthodes

de tests multiples sont décrites en termes de leurs valeurs de p ajustées et ces dernières peuvent alternativement être estimées par des méthodes de ré-échantillonnage (Westfall et Young, 1993).

Maintenant on va décrire les méthodes de tests multiples. Dudoit et *al.* (2003) ont distingué trois types de méthodes de tests multiples : Les méthodes de type pas à pas et deux méthodes en plusieurs étapes, les méthodes descendantes et les méthodes ascendantes.

Définition 2.4.3 : (Dudoit et *al.* (2003))

Dans les méthodes de type pas à pas, chaque hypothèse est évaluée en utilisant une valeur critique qui est indépendante des résultats de test des autres hypothèses. Selon Dudoit et *al.* (2003), les tests séquentiels peuvent avoir une meilleure puissance, tout en préservant le contrôle du taux d'erreur de type 1. Avec ces méthodes, le rejet d'une hypothèse particulière est basé non seulement sur le nombre total d'hypothèses mais également sur les résultats de test des autres hypothèses.

Définition 2.4.4 : (Dudoit et *al.* (2003))

Dans les méthodes descendantes, les hypothèses qui correspondent aux statistiques de test les plus significatives (c'est-à-dire, les plus petites valeurs de p non ajustées ou les plus grandes statistiques de test en valeur absolue) sont considérées successivement. Dès qu'on ne rejette pas une hypothèse nulle, aucune autre hypothèse restante n'est rejetée.

Définition 2.4.5 : (Dudoit et *al.* (2003))

Dans les méthodes ascendantes, les hypothèses qui correspondent aux statistiques

de test les moins significatives sont considérées successivement. Dès qu'une hypothèse nulle est rejetée, toutes les autres hypothèses nulles restantes seront rejetées.

2.4.1 Contrôle du FWER

Nous utilisons l'exemple du contrôle de FWER comme Dudoit et *al.* (2003) pour illustrer ces méthodes.

Méthode pas à pas

Dans le cas du contrôle fort du FWER à un niveau α, la méthode de Bonferroni rejette H_j, $j = 1, \ldots, m$ pour un valeur de p non ajustée p_j telle que : $p_j \leq \alpha/m$. Les valeurs de p ajustées de Bonferroni correspondantes sont :

$$\tilde{p_j} = min(mp_j, 1) \qquad (1)$$

Lemme 2.4.1

La méthode de Bonferroni contrôle fortement le FWER.

Démonstration (Dudoit et *al.* (2000, 2003))

Supposons, sans perte de généralité, que les hypothèses nulles vraies sont H_j pour $j =, \ldots, m_0$, alors :

$$\begin{aligned} FWER &= Pr(V \geq 1) \\ &= Pr(\bigcup_{j=1}^{m_0} (\tilde{p_j} \leq \alpha)) \\ &\leq \sum_{j=1}^{m_0} Pr(\tilde{p_j} \leq \alpha) \\ &\leq \sum_{j=1}^{m_0} Pr(p_j \leq \alpha/m) \\ &\leq m_0 \frac{\alpha}{m}. \end{aligned}$$

La dernière inégalité découle du fait que :

$$Pr(p_j \leq x | H_j) \leq x \quad \forall x \in [0,1].$$

Une autre méthode liée à celle de Bonferroni est la méthode de Sidak (1967).

Lemme 2.4.2

Les valeurs de p ajustées de Sidak pour la méthode pas à pas sont définies par :

$$\tilde{p_j} = 1 - (1 - p_j)^m \qquad (2)$$

La méthode de Sidak est exacte dans le sens du contrôle faible du FWER (sous l'hypothèse nulle complète) quand les valeurs de p sont distribuées selon une loi uniforme $U(0,1)$.

Nous donnons notre démonstration de ce lemme ici.

Démonstration

Sous l'hypothèse complète :

$$\begin{aligned} 1 - FWER &= Pr(V = 0) \\ &= Pr(\tilde{p_1} > 1 - \alpha, \ldots, \tilde{p_m} > 1 - \alpha) \\ &= \prod_{j=1}^{m} Pr(\tilde{p_j} > 1 - \alpha) \end{aligned}$$

La dernière égalité découle de l'indépendance, donc :

$$\begin{aligned} 1 - FWER &= \prod_{j=1}^{m} ((1 - p_j)^m > 1 - \alpha) \quad par\ (2) \\ &= \prod_{j=1}^{m} Pr(U(0,1) > (1 - \alpha)^{1/m}) \\ &= 1 - \alpha. \end{aligned}$$

Cependant dans beaucoup de situations, les statistiques de test et les valeurs de p non ajustées sont dépendantes. C'est le cas des expériences de microréseau d'ADN

où les mesures d'expressions des gènes tendent à avoir une forte corrélation. Westfall et Young (1993) ont proposé des valeur de p ajustées pour les méthodes de tests multiples qui sont moins conservatrices mais qui tiennent compte de la structure de dépendance des statistiques de test. Soit pour la méthode pas à pas, ces valeurs de p, dites valeurs de p ajustées par *MinP* sont définies par :

$$\tilde{p_j} = Pr(\min_{1 \leq l \leq m} P_l \leq p_j | H_0^c) \qquad (3)$$

où H_0^c est l'hypothèse nulle complète (aucun gène ne s'est différentiellement exprimé) et P_l est la variable aléatoire pour la valeur de p non ajustée correspondante au test de la ième hypothèse.

Comme alternative, on peut considérer des méthodes pas à pas basées sur les valeurs de p ajustées par Max T, qui sont définies en terme des statistiques T_j par :

$$\tilde{p_j} = Pr(\max_{1 \leq l \leq m} |T_l| \geq |t_j| | H_0^c). \qquad (4)$$

Notons les points suivants concernant les quatres procédures citées ci-dessus :

La méthode de Sidak ne garantit pas le contrôle du FWER pour des distributions quelconques des statistiques de test, par contre on a le résultat suivant pour des statistiques de test satisfaisant l'inégalité de Sidak :

$$Pr(|T_1| \leq C_1, \ldots, |T_m| \leq C_m) \geq \prod_{j=1}^{m} Pr(|T_j| \leq C_j). \quad (5)$$

La méthode de Sidak contrôle le FWER, si l'inégalité est vraie.

Quand cette inégalité est satisfaite, les valeurs de p ajustées de la méthode de *MinP* sont inférieures ou égales aux valeurs de p ajustées de Sidak. Cette inégalité est également connue sous le nom de propriété de la dépendance positive d'orthant. On a ce résultat important de Sidak (1967) :

Théorème 2.4.1

Supposons que $(T_1, \ldots, T_m)$ a une distribution normale multivariée avec moyenne zéro,

et variances $\sigma_1^2, \ldots, \sigma_m^2$ et une corrélation arbitraire $\{\rho_{ij}\}$, donc l'inégalité (5) est vraie.

Démonstration

Dunn (1967) était la première à démontrer ce résultat sous quelques conditions sur la matrice de covariance. Malgré le fait que Jogdeo (1970) a donné une démonstration plus simple du résultat de Sidak, nous suivons le raisonnement de Sidak (1967) ici.
L'inégalité est une application du résultat suivant de Anderson (1955) : si T est un vecteur aléatoire, ayant la densité g(t) telle que $g(t) = g(-t)$, et l'ensemble $\{t : g(t) \geq v\}$ est convexe pour tout $v \geq 0$, et si U est un ensemble convexe symétrique autour de l'origine, et si x est un vecteur et β tel que $0 \leq \beta \leq 1$ alors :

$$Pr(T + \beta x \in U) \geq Pr(T + x \in U).$$

La première étape est de démontrer l'inégalité suivante :

$$Pr(|T_1| \leq c_1, |T_2| \leq c_2, \ldots, |T_m| \leq c_m) \geq Pr(|T_1| \leq c_1).Pr(|T_2| \leq c_2), \ldots, |T_m| \leq c_m) \quad (6)$$

Maintenant pour démontrer le résultat, notons que si $\rho_{12} = \ldots = \rho_{12m} = 0$ l'inégalité (6) est alors vrai et on peut alors supposer qu'au moins une de ces corrélations est différente de 0.

Premièrement, soient les variables aléatoires $T_1, T_2, \ldots, T_m$ de densité $f(t_1, t_2, \ldots, t_m)$. Et soient $f(t_1)$ et $f(t_2, \ldots, t_m$ les distributions marginales correspondantes et $f(t_2, \ldots, t_m|t_1)$ la distribution conditionnelle pour $T = t_1$. Alors :

$$\begin{aligned} F(c_1) &= \int_{-c_1}^{c_1} \int_{-c_2}^{c_2} \cdots \int_{-c_k}^{c_k} f(t_1, t_2, \ldots, t_m) dt_1 dt_2 \ldots dt_m \\ &- \int_{-c_1}^{c_1} \int_{-c_2}^{c_2} \cdots \int_{-c_k}^{c_k} f(t_1) f(t_2, \ldots, t_m) dt_1 dt_2 \ldots dt_m \end{aligned}$$

Clairement :

$$F(c_1) = 2 \int_0^{c_1} \int_{-c_2}^{c_2} \cdots \int_{-c_k}^{c_k} [f(t_1, t_2, \ldots, t_m) - f(t_1) f(t_2, \ldots, t_m)] dt_1 dt_2 \ldots dt_m$$

$$= 2\int_0^{c_1} f(t_1)\{\int_{-c_2}^{c_2} \dots \int_{-c_k}^{c_k} [f(t_2,\dots,t_m|t_1) - f(t_2,\dots,t_m)]dt_2 \dots dt_m\}dt_1$$

Donc :

$$\frac{\partial F(c_1)}{\partial c_1} = 2f(c_1)\int_{-c_2}^{c_2} \dots \int_{-c_k}^{c_k} [f(t_2,\dots,t_k|c_1) - f(t_2,\dots,t_k)]dt_1 dt_2 \dots dt_k$$

Maintenant, nous allons étudier le comportement de la fonction :

$$G(c_1) = \int_{-c_2}^{c_2} \dots \int_{-c_k}^{c_k} [f(t_2,\dots,t_k|c_1) - f(t_2,\dots,t_k)]dt_1 dt_2 \dots dt_k$$

Soit

$$H(c_1) = \int_{-c_2}^{c_2} \dots \int_{-c_k}^{c_k} f(t_2,\dots,t_k|c_1)dt_1 dt_2 \dots dt_k \qquad (7)$$

Notons que $f(t_2,\dots,t_k|c_1)$ est la densité d'une distribution normale avec les valeurs moyennes $\rho_{12}\sigma_2\sigma_1^{-1}c_1,\dots,\rho_{1k}\sigma_k\sigma_1^{-1}c_1$ et avec certains coefficients de variance et de corrélation ne dépendant pas de c_1. Ainsi la densité $f(t_2,\dots,t_k|c_1)$ est obtenue par une translation de la densité $f(t_2,\dots,t_k|0)$ et la région d'intégration dans (8) satisfait les hypothèses du résultat de Anderson qui a montré que $H(c_1)$ est une fonction décroissante en $H(c_1)$ (sauf dans le cas où $\rho_{12} = \dots = \rho_{1k} = 0$; dans ce cas $H(c_1)$ est constante et ce cas est déjà omi).

De ce fait, $H(c_1)$ est décroissante et $lim_{c_1\to\infty}H(c_1) = 0$, si $lim_{c_1\to\infty}G(c_1) < 0$ et alors les deux cas suivants peuvent se produire : soit que $G(c_1) \leq 0$ pour tout c_1, $0 \leq c_1 \leq \infty$ où il existe un c^* tel que $G(c_1) > 0$ pour tout $0 \leq c_1 \leq c^*$, et $G(c_1) < 0$ pour tout $c^* < c_1 < \infty$. Les mêmes inégalités sont alors vraies pour la dérivée : $\partial F(c_1)/\partial c_1$. Ainsi, dans le deuxième cas, $F(c_1)$ est croissante pour $0 \leq c_1 < c^*$ et décroissante pour $c^* < c_1 < \infty$. Et puisque

$$F(0) = 0, \quad lim_{c_1\to\infty}F(c_1) = 0, \qquad (8)$$

alors, $F(c_1) \geq 0$ pour tout c_1 telle que : $0 \leq c_1 \leq \infty$.

Dans le deuxième cas, $F(c_1)$ est décroissante. Mais c'est clairement impossible en raison de (9). L'inégalité (7) ainsi prouvée fournie la distribution non-singulière $T_1, T_2, \dots, T_m$

si sa distribution est singulière, elle peut être rapprochée par une séquence de distributions non-singulières; par conséquent, et par un passage à la limite, la validité de (7) peut être établie en général.
Par induction nous pouvons immédiatement prouver (6).

■

Jogdeo (1977) a prouvé que l'inégalité de Sidak est vraie pour une plus grande classe de distributions, y compris quelques distributions multivariées de t et de F.
Dudoit et *al.* (2003) ont noté que si (5) est vraie, les procédures basées sur les valeurs de p ajustées par la méthode de *MinP* sont moins conservatrices que celles basées sur les procédures de Bonferroni ou de Sidak. Dans le cas où les statistiques du test sont indépendantes, les valeurs de p ajustées de Sidak et les valeurs de p ajustées par *MinP* sont équivalentes.

Les méthodes basées sur le *MaxT* et celles basées sur *MinP* mènent à un contrôle faible du FWER sous l'hypothèse de pivotage de sous-ensemble selon Westfall et Young (1993). On dit que la distribution des valeurs de p non-ajustées $(p_1, \ldots, p_m)$ a la propriété de pivotage de sous-ensemble, si la distribution conjointe des vecteurs aléatoires $\{p_j, j \in \Lambda_0\}$ est identique aux distributions satisfaisant l'hypothèse nulle composée : $\bigcap_{j\in\Lambda_0} H_j$ et $H_0^c = \bigcap_{j=1}^m H_j$, pour tout sous-ensemble $\Lambda_0 \in \{1, \ldots, m\}$.

Ici, l'hypothèse composée $\bigcap_{j\in\Lambda_0} H_j$ se réfère à la distribution commune des statistiques de test T_j ou des valeurs de p, p_j, pour le test de l'hypothèse j. Sous pivotage de sous-ensemble, l'ajustement de la multiplicité sera plus complexe car on devrait considérer la distribution des statistiques de test sous les hypothèses nulles partielles $\bigcap_{j\in\Lambda_0} H_j$, plutôt que sans l'hypothèse nulle complète H_0^c.

Dans le contexte de microréseau, chaque hypothèse nulle H_j est liée à un gène

j et chaque statistique de test T_j est fonction de la réponse Y et de la mesure d'expression X_j. L'hypothèse composée $\bigcap_{j\in\Lambda_0} H_j$ se rapporte à la distribution commune des variables Y et $\{X_j, j \in \Lambda_0\}$ et spécifie que le sous-vecteur aléatoire des mesures d'expression $\{X_j, j \in \Lambda_0\}$ est indépendant de la réponse Y.

Selon Dudoit et *al.* (2003) les valeurs de p ajustées par $MaxT$ sont plus faciles à calculer que les valeurs de p ajustées par $MinP$ et sont égales au valeurs de p ajustées par $MinP$ quand les statistiques de test T_j sont indépendantes. Cependant, les deux méthodes donnent deux valeurs de p ajustées différentes sans indépendance. Les méthodes basées sur les valeurs de p de $MinP$ exigent plus de calculs que celles basées sur les valeurs de p de $MaxT$ car les valeurs de p non ajustées doivent être estimées avant de considérer la distribution de leurs minimums (voir Ge, Dudoit et Speed, 2003).

2.4.2 Méthodes descendantes

Les procédures de pas à pas sont simples à mettre en application, mais elle tendent à être conservatrices pour le contrôle du FWER. L'amélioration de la puissance, tout en préservant le contrôle fort du FWER peut être réalisé par les méthodes descendantes. Soient $p_{r_1} \leq p_{r_2} \leq \ldots \leq p_{r_m}$ les valeurs de p observées, non ajustées et ordonnées et soient $H_{r_1}, H_{r_2}, \ldots, H_{r_m}$ les hypothèses nulles correspondantes.

Définition 2.4.6

Pour le contrôle fort du FWER à un niveau α, la procédure de Holm (1979) est définie comme suit :

soit $j^* = min\{j : p_{r_j} > \alpha/(m-j+1)\}$, et on rejette l'hypothèse H_{r_j}, pour tout $j = 1, \ldots, j^* - 1$. Si un tel j^* n'existe pas alors on rejette toutes les hypothèses nulles.

Les valeur de p ajustées descendantes de Holm sont données par :

$$\tilde{p_i} = p_i,$$

$$\tilde{p}_{r_j} = \max_{k=1,\ldots,j}\{\min((m-k+1)p_{r_k}, 1)\}. \qquad (10)$$

Lemme 2.4.3

D'après Westfall and Young (1993), la procédure de Holm est moins conservatrice que la procédure standard de Bonferroni qui multiplie les valeurs de p non ajustées par m à chaque étape.

Démonstration :

Nous donnons la démonstration suivante .

Il faut démontrer que :

$$\tilde{p}_{r_j} \leq \alpha$$

Supposons qu'on rejette toutes les hypothèses nulles, donc :

$$\forall j = 1, \ldots, k \; p_{r_j} \leq \frac{\alpha}{m-j+1}$$

alors :

$$(m-j+1)p_{r_j} \leq \alpha \;\; \forall j = 1, \ldots, k$$

et $\tilde{p}_{r_j} \leq \alpha$.

■

Notons qu'en considérant successivement le maximum des quantités $min((m-k+1)p_{r_k}, 1)$, la monotonicité des valeurs de p ajustées est imposée, c'est à dire $\tilde{p}_{r_1} \leq \tilde{p}_{r_2} \leq \ldots \leq \tilde{p}_{r_m}$. On peut rejeter une hypothèse particulière seulement si toutes les hypothèses avec de plus petites valeurs de p non ajustées sont rejetées à l'avance.

Définition 2.4.7

De même on définit les valeurs de p ajustées descendantes de Sidak par :

$$\tilde{p}_{(i)} = 1 - (1 - p_{(i)})^m$$

$$\tilde{p}_{r_j} = \max_{k=1,\ldots,j} \{1 - (1 - p_{r_k})^{(m-k+1)}\}. \qquad (11)$$

Les méthodes descendantes de Westfall et Young (1993) sont définies pour le *MinP* par :

$$\tilde{p}_{r_j} = \max_{k=1,\ldots,j} \{Pr(\min_{l \in \{r_k,\ldots,r_m\}} p_l \leq p_{r_k} | H_0^c)\}; \qquad (12)$$

et pour le *MaxT* par :

$$\tilde{p}_{r_j} = \max_{k=1,\ldots,j} \{Pr(\max_{l \in \{r_k,\ldots,r_m\}} |T_l| \geq |t_{r_k}| | H_0^c)\} \qquad (13)$$

où $|t_{r_1}| \leq |t_{r_2}| \leq \ldots \leq |t_{r_m}|$ sont les statistiques de test ordonnées.
Westfall et Young (1993) ont montré que les procédures basées sur les valeurs de p ajustées par *MinP* descendantes sont moins conservatrices que celles de Holm.

2.4.3 Méthodes ascendantes

Contrairement aux procédures descendantes, les procédures ascendantes commencent par la valeur de p la moins significative p_{r_m} et sont habituellement basées sur le résultat de Simes (1986) suivant :

$$Pr(p_{(j)} > \alpha j/m; \forall j = 1, \ldots, m | H_0^c) \geq 1 - \alpha. \qquad (14)$$

Afin d'aider les lecteurs de ce mémoire, nous illustrons la démonstration de ce résultat de Simes (1986). Sous l'hypothèse nulle complète H_0^c et pour des statistiques de test indépendantes, les valeurs de p non ajustées ordonnées $p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(m)}$ satisfont l'inégalité de Simes (14), et on a l'égalité dans le cas continu.

Au lieu d'utiliser *MinP* ou *MaxT*, on peut utiliser le test de Simes (1986) qui considère la distribution de toutes les valeurs de p. Wright (1992) a donné une version abrégée de ce test. En effet, Simes (1986) a modifié et amélioré la procédure de

Bonferroni de la manière suivante : soient $p_{(1)}, \ldots, p_{(n)}$ les valeurs de p ordonnées pour tester l'hypothèse $H_0 = \{H_{(1)}, \ldots, H_{(n)}\}$. Puis, on rejette H_0 si $p_{(j)} \leq j\alpha/n$ pour tout $j = 1, \ldots, n$. Cette méthode de test a comme probabilité d'erreur de type 1 égale à α pour les tests indépendants comme le montre le résultat suivant :

Lemme 2.4.4 : (Simes (1986)

Soient $p_{(1)}, \ldots, p_{(n)}$ les valeurs de p ordonnées pour n variables aléatoires uniformes indépendantes $(0,1)$ et $A_n(\alpha) = Pr\{p_{(j)} > j\alpha/n; j = 1, \ldots, n\}$ $(0 \leq \alpha \leq n)$. Alors $A_n(\alpha) = 1 - \alpha$.

Démonstration

On suit la démonstration donnée par Simes (1986).

Il est clair que le résultat est vrai pour $n = 1$.

Maintenant, $\{p_{(1)}/p_{(n)}, \ldots, p_{(n-1)}/p_{(n)}\}$ sont les statistiques d'ordre de $(n-1)$ variables aléatoires uniformes indépendantes $(0,1)$ et elles sont indépendantes de $p_{(n)}$. De plus, la distribution de $p_{(n)}$ est la fonction p^n $(0 < p < 1)$. Alors :

$$A_n(\alpha) = \int_{\alpha}^{1} A_{n-1}\{\frac{\alpha(n-1)}{pn}\} np^{n-1} dp.$$

Si $A_{n-1}(\alpha) = 1 - \alpha$ alors $A_n(\alpha) = 1 - \alpha$. Par conséquent le résultat est prouvé par induction.

■

Même si l'inégalité n'est pas vraie en général parce qu'il existe les contre exemples pathologiques, les études de simulation de Simes (1986) ont montré qu'elle peut être vraie pour beaucoup de distributions multivariées.

Hochberg (1988) a utilisé l'inégalité de Simes (1986) pour déduire la procédure suivante pour contrôler le FWER : pour le contrôle du FWER à un niveau α, soit $j^* = \max\{j : p_{r_j} \leq \alpha/(m-j+1)\}$ et il faut rejeter les hypothèses H_{r_j}, pour $j = 1, \ldots, j^*$; si un tel j^* n'existe pas, on ne rejette aucune hypothèse nulle. Les valeurs de p ajustées ascendantes de Hochberg sont définies par :

$$\tilde{p}_{(m)} = p_{(m)}$$

$$\tilde{p}_{r_j} = \min_{k=j,\ldots,m}\{\min((m-k+1)p_{r_k}, 1)\} \qquad (15)$$

Les valeurs de p de Hochberg peuvent être considérées comme étant des procédures ascendantes analogues aux procédures descendantes de Holm puisque les valeurs de p non ajustées de ces deux procédures sont comparées aux valeurs critiques, à savoir, $\alpha/(m-j+1)$ et donc la procédure de Hochberg est plus puissante que celle de Holm. Les procédures ascendantes se sont avérées, par Dudoit et *al.* (2003), souvent plus puissantes que les procédures descendantes, alors il est important de savoir que toutes les procédures basées sur l'inégalité de Simes comptent sur l'hypothèse qui suppose que le résultat prouvé sous la condition de l'indépendance donne un test plus conservateur. Selon Dudoit et *al.* (2003), plus de recherches sont nécessaires pour déterminer les circonstances dans lesquelles de telles méthodes sont applicables et en particulier, si elles sont applicables pour des types de structures de corrélation produites dans des expériences de microréseau d'ADN.

2.4.4 Contrôle de FDR (false discovery rate)

Le fait que le contrôle du FWER peut mener à des procédures très conservatrices a conduit Benjamini et Hochberg (1995) à chercher une approche moins conservatrice et qui contrôle l'espérance de la proportion d'erreur de type 1 prévue parmi les hypothèses rejetées, soit le FDR (false discovery rate), qui est défini comme suit :

$FDR = E(Q)$ où $Q = V/R$ *si* $R > 0$ et $Q = 0$ *si* $R = 0$, alors :

$$FDR = E(V/R|R > 0)Pr(R > 0).$$

Sous l'hypothèse nulle complète et étant donnée la définition de $0/0 = 0$ quand $R = 0$, le FDR est égal au FWER et les procédures qui contrôlent le FDR contrôlent également le FWER dans le sens faible. Benjamini et Hochberg (1995) décrivent une procédure ascendante pour le contrôle fort du FDR pour des statistiques de test indépendantes. Soient $p_{r_1} \leq p_{r_2} \leq \ldots \leq p_{r_m}$, les valeur de p non ajustées observées. Pour le contrôle du FDR à un niveau α, on définit :
$j^* = \max\{j : p_{r_j} \leq (j/m)\alpha\}$ et on rejette H_{r_j} pour $j = 1, \ldots, j^*$ et si une telle j^* n'existe pas on ne rejette aucune hypothèse nulle.
Les valeurs de p correspondantes sont définies par :

$$\tilde{p}_{r_j} = \min_{k=j,\ldots,m}\{\min(\frac{m}{k}p_{r_k}; 1)\} \qquad (16)$$

Benjamini et Yekulieti (2001) ont prouvé que cette procédure contrôle le FDR sous certaine structures dépendantes (par exemple la dépendance positive de régression), et ils ont proposé une modification conservatrice simple de la procédure qui contrôle le FDR pour des structures de dépendances arbitraires. Les valeurs de p pour la procédure ascendante modifiée sont :

$$\tilde{p}_{r_j} = \min_{k=j,\ldots,m}\{\min(\frac{m\sum_{j=1}^{m}(1/j)}{k}p_{r_k}; 1)\} \qquad (17)$$

Selon Dudoit et *al.* (2003) les deux procédures ascendantes citées ci-dessus diffèrent seulement dans le multiplicateur appliqué aux valeurs de p non ajustées. Pour la première procédure standard (16), la pénalité est m/k alors que pour la deuxième procédure conservatrice (17), la pénalité est $(m\sum_{j=1}^{m}1/j)/k$. Pour un grand nombre m d'hypothèses nulles, les pénalités diffèrent par un facteur d'environ $\log m$. Dudoit et *al.* (2003) ont noté que la première procédure définie ci-dessus peut être conservatrice même dans le cas d'indépendance car pour cette procédure ascendante, on avait montré que :

$E(Q) \leq (\frac{m_0}{m})\alpha \leq \alpha.$

La plupart des procédures de contrôle du FDR ont été conçues pour des statistiques de test indépendantes ou encore on ne s'est pas servi des structures de dépendance des statistiques de test. Selon Dudoit et *al.* (2003), de nouvelles études ont proposé des procédures de contrôle du FDR en utilisant les structures de dépendance parmi les statistiques de test (ces procédures supposent que les valeurs de p non ajustées pour les hypothèses nulles vraies sont indépendantes des valeurs de p non ajustées pour les hypothèses nulles fausses). Dans le cas des expériences de microréseau, où des milliers de tests peuvent être faits simultanément, la procédure contrôlant le FDR peut être une alternative à celle contrôlant le FWER selon Dudoit et *al.* (2003).

2.5 Ré-échantillonnage

Dans beaucoup de situations, les distributions jointes et marginales des statistiques de test sont inconnues. Des méthodes de ré-échantillonnage (bootstrap, permutation) peuvent être utilisées pour estimer des valeurs de p non ajustées et ajustées tout en évitant de trouver la distribution jointe des statistiques de test. Nous considérons les hypothèses nulles H_j : il n'y a pas d'association entre la variable X_j et la réponse Y, $j = 1, \ldots, m$. Dans le cas de microréseaux et pour de telle hypothèses nulles, la distribution jointe des statistiques de test $(T_1, \ldots, T_m)$ sous l'hypothèse nulle complète peut être estimée en permutant les colonnes de la matrice X des données de l'expression du gène.

Nous décrivons ici deux algorithmes de permutation pour les valeurs de p ajustées et non ajustées de Dudoit et *al.* (2003). Golub et *al.* (1999) et Tusher et *al.* (2001) ont aussi proposé des algorithmes de ré-échantillonnage pour les tests multiples avec les données d'une expérience de microréseau pour les valeurs de p non ajustées.

Algorithme 1 Dudoit et *al.* (2003)

Pour la bième permutation, $b = 1, \ldots, B$:

1- Permutez les n colonnes de la matrice X de données.

2- Calculez les statistiques de test $t_{1,b}, \ldots, t_{m,b}$ pour chaque hypothèse (chaque gène).

La distribution de permutation de la statistique de test T_j pour l'hypothèse $H_j, j = 1, \ldots, m$ est donnée par la distribution empirique de $t_{j,1}, \ldots, t_{j,B}$. La valeur de p pour la distribution pour l'hypothèse H_j est donnée par :

$$p_j^* = \frac{\sum_{b=1}^{B} I(|t_{j,b}| \geq |t_j|)}{B}$$

où $I(.)$ est la fonction indicatrice, qui est égale à 1 si la condition entre parenthèses est satisfaite et à 0 sinon.

La permutation des colonnes entières de cette matrice nous mène à la situation où la réponse ou la covariable Y sera indépendante des mesures d'expression du gène, tout en essayant de préserver les structures de corrélation et les caractéristiques distributionnelles des mesures d'expression du gène pour les grandes tailles d'échantillon n. Il peut être impossible de considérer tous les B choix de sous ensembles aléatoires de permutations. (voir Dudoit et *al.* (2003)).

Les valeurs de p ajustées par permutation pour les procédures de Bonferroni, Sidak, Holm et Hochberg peuvent être obtenues en remplaçant p_j par p_j^* dans les équations (1), (2), (11), (12) et (15). Les valeurs de p non ajustées de permutation peuvent également être utilisées pour les procédures de contrôle du FDR décrites précédemment.

Algorithme 2 : On a l'algorithme suivant pour la valeur de p ajustée de *MaxT* descendante de Westfall et Young (1993).

Pour la b ième permutation, $b = 1, \ldots, B$:

1- Permutez les n colonnes de la matrice de données X.

2- Calculez les statistiques de test $t_{1,b}, \ldots t_{m,b}$ pour chaque hypothèse (chaque gène).

3- Calculez les maximums successifs des statistiques de test : $u_{m,b} = |t_{r_m,b}|$.

$u_{j,b} = \max(u_{j+1,b}, |t_{r_j,b}|$ pour $j = m-1, \ldots, 1$, où r_j est telle que $|t_{r_1}| \geq |t_{r_2}| \geq \ldots \geq |t_{r_m}|$ pour les données originales.

Les valeurs de p ajustées de permutation sont données par :

$$\tilde{p}^*_{r_j} = \frac{\sum_{b=1}^{B} I(u_{j,b} \geq |t_{r_j}|)}{B}$$

Avec les contraintes de monotonicité imposées par :

$\tilde{p}^*_r \leftarrow \tilde{p}^*_{r_1}$, $\tilde{p}^*_{r_j} \leftarrow \max(\tilde{p}^*_{r_j}, \tilde{p}^*_{r_{j-1}})$

pour $j = 2, \ldots, m$.

Algorithme 3 : Golub et *al.* (1999) utilisent le terme de voisinage pour se référer à des ensembles de gènes avec des statistiques de test t_j plus grandes en valeur absolue qu'une valeur critique $c > 0$, c'est à dire, l'ensemble d'hypothèses rejetées : $\{j : t_j \geq c\}$ ou $\{j : t_j \leq -c\}$.

Les étiquettes des deux conditions ont été permutées $B = 400$ fois pour estimer la distribution sous l'hypothèse nulle complète, des nombres $R(c) = V(c) = \sum_{j=1}^{m} I(T_j \geq c)$ des faux positifs pour différentes valeurs critiques c (et on fait de même pour le cas où $t_j \leq -c$).

Mais selon Dudoit et *al.* (2003) Golub et *al.* (1999) n'ont fourni aucune autre procédure pour choisir la valeur critique c et ils n'ont pas discuté du contrôle de taux d'erreur de type 1 dans leur procédure. Comme quelques procédures de contrôle de PFER, de PCER ou de FWER, l'analyse de voisinage considère la distribution nulle complète du nombre d'erreurs de type 1 $V(c) = R(c)$. Cependant, au lieu de contrôler $E(V(c)), E(V(c))/m$ ou $Pr(V(c) \geq 1)$, ils cherchent à contrôler une quantité différente, soit, $G(c) = Pr(R(c) \geq r(c)|H_0^c)$ où G(c) peut être considérée comme étant la variable aléatoire valeur de p sous l'hypothèse nulle complète H_0^c pour le nombre d'hypothèses rejetées.

Dudoit et *al.* (2002) ont montré sous la condition que les statistiques de test observées soient ordonnées en valeur absolue : $|t|_{(1)} \geq |t|_{(2)} \geq \ldots \geq |t|_{(m)}$, que la fonction $G(c)$ est continue à gauche et discontinue aux points $|t|_{(j)}, j = 1, \ldots, m$.
Bien que la fonction $G(c)$ est décroissante en c dans des intervalles $(|t|_{(j+1)}, |t|_{(j)})$, elle n'est pas globalement décroissante et il peut y avoir plusieurs valeurs de c telle que $G(c) = \alpha$. Par conséquent, on doit décider d'une procédure appropriée pour choisir la valeur critique c. Dudoit et *al.* (2002) ont proposé une méthode qui contrôle le FWER faiblement.

Algorithme 4 : Analyse de signification des microréseaux (SAM) de Tusher, Tibshirani et Chu (2001) Une procédure très prometteuse est l'analyse de signification de la méthode de test multiple de microréseau de Tibshirani et *al.* ≪ SAM ≫. Cette méthode permet d'avoir un choix approprié des statistiques de test pour différents types de réponses ou covariables. On donne une brève description de la méthode de Tibshirani, Tusher er Chu (2001).

L'algorithme utilise les Q-Q graphiques, donc il faut d'abord décrire le Q-Q graphique.

Définition 2.5.1

Le Q-Q graphique <<Q-Q plot>> est un graphique des quantiles d'un premier ensemble de données versus les quantiles d'un deuxième ensemble de données. Un quantile représente des fractions (ou des pourcentages) de points au-dessous d'une valeur donnée. C'est-à-dire les 0,3 (ou 30%) quantiles sont le point auquel 30% des données sont au dessous et 70% des données sont au-dessus de cette valeur.
Une ligne de référence de 45° est également tracée. Si les deux ensembles viennent d'une population avec la même distribution, les points devraient tomber approximativement le long de cette ligne de référence. Plus la distance des points à cette ligne de référence

est grande, plus on peut conclure avec certitude que les deux ensembles de données proviennent de deux populations avec distributions différentes. Les avantages d'un tel Q-Q graphique sont :

1- On n'aura pas besoin d'avoir les mêmes tailles d'échantillon pour les deux bases de données.

2- Beaucoup d'aspects distributionnels peuvent être simultanément examinés. Par exemple, des variations dans la localisation, les variations dans l'échelle, les changements de la symétrie et la présence des valeurs aberrantes peuvent tous être détectés de ce Q-Q graphique. Par exemple, si les deux ensembles de données viennent de populations dont les distributions diffèrent seulement par une variation dans la localisation, les points devraient se trouver suivant une ligne droite qui est déplacée vers le haut ou vers le bas de la ligne de référence de 45°. Le Q-Q graphique est semblable à un graphique de probabilité. Pour un graphique de probabilité, les quantiles pour un des échantillons de données sont remplacés par les quantiles d'une distribution théorique. En général, un Q-Q graphique est constitué de :

un axe vertical : les quantiles estimés du premier ensemble de données.

un axe horizontal : les quantiles estimés du deuxième ensemble de données.

En générale, on peut utiliser un Q-Q graphique pour répondre aux questions suivantes :

1- deux ensembles de données proviennent-ils des populations avec une distribution commune ?

2- deux ensembles de données ont-ils des formes distributionnelles semblables ?

3- deux ensembles de données ont-ils un comportement semblable ?

Exemple de Q-Q graphique.

Par exemple le Q-Q graphique pour les données TIB de Tibshirani et *al.* (2003) décrit dans la figure 2.9 ci-dessous nous montre que :

1- ces 2 groupes ne semblent pas être venus de populations avec une distribution commune.

2- les valeurs du premier groupe sont sensiblement plus petites que les valeurs corres-

pondantes du deuxième groupe.

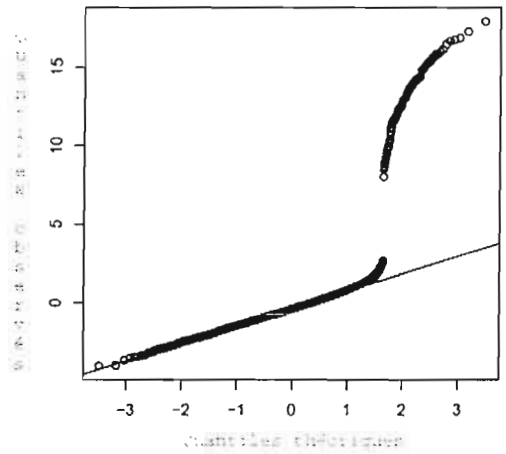

Fig. 2.9 Q-Q graphique pour les données TIB de Tibshirani

Maintenant on décrit la procédure SAM de Tusher, Tibshirani et Chu (2001).
1- On suppose qu'une statistique de test appropriée est calculée pour chaque gène et on définit les statistiques d'ordre $t_{(j)}$ telles que :

$$t_{(1)} \geq t_{(2)} \geq \ldots \geq t_{(m)}.$$

2- On fait les B permutations des réponses ou covariables $y_1, \ldots, y_n$. Pour chaque permutation b, on calcule la statistique de test $t_{j,b}$ et les statistiques d'ordre correspondantes $t_{(1),b} \geq t_{(2),b} \geq \ldots \geq t_{(m),b}$. Notons que les $t_{j,b}$ peuvent correspondre à un gène différent que celui qui correspond à $t_{i,j}$.

3- À partir des B permutations, on estime l'espérance (sous l'hypothèse nulle complète) des statistiques d'ordre par : $\hat{t}_j = (1/B) \sum_b t_{(j),b}$.

4- On fait un graphique Q-Q (Quantile-Quantile) pour la valeur observée $t_{(j)}$ versus la valeur estimée $\hat{t}_{(j)}$.

5- Pour un seuil fixé Δ, soit : $j_0 = \max\{j : \hat{t}_{(j)} \geq 0\}$, $j_1 = \max\{j \leq j_0 : t_{(j)} - \hat{t}_{(j)} \geq \Delta\}$ et $j_2 = \min\{j > j_0 : t_{(j)} - \hat{t}_{(j)} \leq -\Delta\}$.
Pour un seuil fixe Δ, tous les gènes avec $j \leq j_1$ seront positivement significatifs et tous les gènes avec $j \geq j_2$ seront négativement significatifs.
On définit le point $\ll$ Upper Cut $\gg$ par : $Cut_{Up}(\Delta) = \min\{t_{(j)} : j \leq j_1\} = t_{(j_1)}$.
Le point ≪Lower Cut≫ est défini par : $Cut_{Low}(\Delta) = \max\{t_{(j)} : j \geq j_2\} = t_{(j_2)}$.
Et si de tels j_1 (j_2) n'existent pas on note : $Cut_{Up}(\Delta) = \infty$ et $Cut_{Low}(\Delta) = -\infty$

6- Pour un seuil donné Δ, le nombre prévu de faux positifs, PFER, est estimé en calculant premièrement pour chacune des B permutations, le nombre de gènes avec un $t_{j,b}$ plus grand que le $Cut_{Up}(\Delta)$ ou inférieur au $Cut_{Low}(\Delta)$, puis en faisant deuxièmement la moyenne de cette quantité pour toutes les permutations.

7- Un seuil Δ sera choisi pour contrôler le nombre prévu de faux positifs, PFER, sous l'hypothèse nulle complète, à un niveau nominal acceptable.

Ainsi, la procédure SAM de Tusher, Tibshirani et Chu (2001) utilise des statistiques de test ordonnées des données originales seulement afin d'obtenir des seuils (≪cutoffs≫) globaux pour les statistiques de test. Dans les permutation, les seuils (≪cutoffs≫) sont maintenus fixes et le PFER est estimé en comptant le nombre de gènes avec des statistiques de test plus grandes ou plus petites que le ≪cutoff≫ global. Notons que ces ≪cutoffs≫ sont des variables aléatoires car elles dépendent des statistiques de test observées.

Dudoit, Shaffer et Boldrick (2002) discutent une comparaison plus détaillée des propriétés des statistiques de test de SAM décrites par Efron et *al.* (2000) et Tusher, Tibshirani et Chu (2001). Ils démontrent que les deux procédures de SAM visent à contrôler le PFER (ou le PCER) mais les procédures de Efron et *al.* (2000) contrôlent ces taux d'erreur seulement dans le sens faible. La seule différence entre le version SAM de Tusher, Tibshirani et Chu (2001) et les procédures standards pour lesquelles on rejette le H_j pour $|t_j| \geq c$ est dans l'utilisation des valeurs critiques asymétriques choisies à partir du Q-Q graphique. En résumé, la procédure SAM de Tusher, Tibshirani et Chu rejette l'hypothèse H_j si $t_j \geq Cut_{Up}(\Delta)$ ou si $t_j \leq Cut_{Low}(\Delta)$ où les $Cut_{Up}(\Delta)$ et $Cut_{Low}(\Delta)$ sont choisis à partir du Q-Q graphique de permutation (Quantile-Quantile) et tel que le PFER est contrôlé au sens fort à un niveau donné.

Dudoit et *al.* (2002) ont remarqué que la définition de Tusher et *al.* (2001) du FDR est différente de la définition standard de Benjamini et Hochberg (1995) déjà discutée : le FDR de SAM estime $E(V|H_0^c|R)$ et non le $E(V|R)$ comme dans le cas de la méthode de Benjamini et Hochberg. En outre, le FDR de Tusher et *al.* peut être plus grand que 1.

Dudoit et *al.* (2003) ont décrit un certain nombre de méthodes de tests multiples pour contrôler les différents taux d'erreur de type 1, y compris le FWER et FDR. Ils ont décrit ces méthodes en termes de leurs propriétés principales : la définition du taux d'erreur de type 1, le type de contrôle de ce taux d'erreur (fort versus faible), la nature séquentielle de la procédure et les hypothèses distributionnelles. Pour chaque procédure, ils ont obtenu des valeurs de p ajustées. Ils ont introduit les graphiques des valeurs de p ajustées qui sont particulièrement utiles en récapitulant les résultats de différentes méthodes de test multiples appliquées à un grand nombre de gènes. Les graphiques permettent à des chercheurs d'examiner les divers taux de faux positifs (FWER, FDR, ou PCER) liés à différentes listes de gènes. Ils n'exigent pas des chercheurs de présélectionner une définition particulière du taux ou du niveau d'erreur du type 1 mais leur fournissent plutôt des outils pour décider d'une combinaison appropriée du nombre

de gènes et du taux de faux positifs acceptable pour une expérience particulière et des ressources disponibles.

Les deux types de graphiques proposés par Dudoit et *al.* (2003) sont les suivants : 1- Graphique des valeurs de p ajustées ordonnées ($\tilde{p}_j$ versus j). Pour un nombre donné de gènes r, par exemple, cette représentation fournit le taux d'erreur de type I nominal pour une procédure donnée quand les r gènes avec les plus petites valeurs de p ajustées pour ce procédé sont déclarés être différentiellement exprimés. Par conséquent, plutôt que de choisir un type d'erreur spécifique de contrôle et d'un niveau α, les chercheurs pourraient d'abord choisir un nombre r de gènes avec lesquels ils se sentent confortables par la suite. Les taux de faux positifs nominaux (ou les valeurs de p, $\tilde{p}_r$) correspondant à ce nombre sous divers types de contrôle d'erreur et de procédures peuvent alors être lus à partir du graphique. Par exemple, selon Dudoit et *al.* (2003), pour $r = 10$ gènes, le FWER nominal de la procédure descendante de Holm pourrait être 0.1 et le FDR nominal de la procédure ascendante de Benjamini et Hochberg (1995) pourrait être égale à 0.07.

2- Graphique du nombre de gènes déclarés différentiellement exprimés versus le taux d'erreur de type 1 (r versus α). Ce type graphique est la transposée du graphique précédent et peut être utilisé comme suit :

Pour un niveau nominal donné α, on cherche le nombre r de gènes qui seraient déclarés être différentiellement exprimés sous une procédure, puis on lit le niveau exigé pour l'utiliser dans d'autres méthodes. Alternativement, on cherche le nombre de gènes qui seraient identifiés en utilisant une procédure contrôlant le FWER à un niveau nominal fixe α, puis on cherche combien d'autres gènes seraient identifiés en utilisant des procédures contrôlant le FDR et le PCER à ce niveau.

Dudoit et *al.* (2003) ont remarqué que les méthodes de test multiple considérées dans leur étude peuvent être divisées en deux catégories larges : celles pour lesquelles les valeurs de p ajustées sont monotones dans les statistiques de test t_j et celles pour lesquelles les valeurs de p ajustées sont monotones dans les valeurs de p non ajustées p_j. En général, selon Dudoit et *al.* (2003) le contrôle des gènes basés sur les statistiques de test t_j différera de celui basé sur la valeur de p non ajustée p_j. Ils ont donné l'exemple suivant : pour un niveau nominal donné α, une procédure de contrôle de FWER telle que celle de Bonferroni identifie seulement les 20 premiers gènes avec les plus petites valeurs de p non ajustées alors qu'une procédure de contrôle de FDR, telle que celle de Benjamini et Hochberg (1995) maintient les 15 gènes additionnels avec les 15 suivantes plus petites valeurs de p non ajustées. Il y a beaucoup de recherches à faire dans ce domaine.

CHAPITRE III

ANALYSE DES DONNÉES DES EXPÉRIENCES DE MICRORÉSEAU ET UNE ÉTUDE DE SIMULATION

Nous illustrons les méthodes d'analyse des données des expériences de microréseau de Dudoit et *al.* avec le progiciel Bioconductor (http ://www.bioconductor.org/packages/bioc/stable/src/contrib/html/multtest.html) en utilisant des données d'expression de gènes dans l'étude de la leucémie ALL/AML de Golub et *al.* (1999) et des données de Tibshirani (http ://www-stat.stanford.edu/˜ tibs/clickwra p/sam/academic/index.html). Nous concluons ce chapitre avec une étude de simulation qui démontre la différence entre les erreurs de type 1 décrites par Dudoit et *al.* (2003).

3.1 Introduction

Le logiciel ≪multtest≫ de Dudoit et Ge (2004) dans le progiciel Bioconductor contient une collection de fonctions pour les tests multiples d'hypothèses. Ces fonctions peuvent être employées pour identifier les gènes différentiellement exprimés dans des expériences de microréseau, c'est-à-dire, les gènes dont les niveaux d'expression sont associés à une réponse ou à une covariable d'intérêt. Il met en application des méthodes de test multiples pour contrôler des taux d'erreur de type 1 et de type 2 incluant des procédures pour contrôler le taux d'erreur du type 1 de type FWER : les procédures de Bonferroni, Hochberg, Holm, Sidak, Westfall et Young, $MinP$ et $MaxT$. Il inclut également des procédures pour contrôler le taux de fausses découvertes (FDR) : les

procédures ascendantes de Benjamini et Hochberg et celles de Benjamini et Yekutieli. Ces procédures sont mises en application pour des tests basés sur la statistique de student t, la statistique de Ficher F, la statistique de student t appareillée et la statistique de Wilcoxon. Les résultats des procédures sont résumés en utilisant les valeurs de p ajustées. Des valeurs de p ajustées peuvent être obtenues à partir de la distribution nominale des statistiques de test ou par permutation. Nous utiliserons toutes ces méthodes pour calculer les valeurs de p et les taux d'erreurs qu'on a déjà définis.

3.2 Méthodes de tests multiples mises en application dans ≪multtest≫.

Ici nous allons utiliser le progiciel <<multtest>> appliqué à un ensemble de données de leucémie.

Golub et *al.* (1999) se sont intéressés à identifier les gènes qui sont différentiellement exprimés pour des patients qui présentent deux types de leucémie, la leucémie lymphoblastique aiguë (ALL, classe 0) et la leucémie myéloïde aiguë (AML, classe 1). L'ensemble de données de Golub contient les données du niveau d'expression de gène pour les 38 échantillons de mARN de tumeur et 3051 gènes après pré-traitement. Des niveaux d'expression de gène ont été mesurés en utilisant des morceaux de ≪l'affymetrix≫ de haute densité de l'oligonucléotide contenant $p = 6817$ gènes humains. L'ensemble d'étude comporte n = 38 échantillons, 27 cas d'ALL et 10 cas d'AML. D'après Golub et *al.* (1999) trois étapes de pré-traitement ont été appliquées à la matrice normale des valeurs d'intensité :

(i) seuillage : plancher de 100 et plafond de 16.000.

(ii) filtrage : exclusion des gènes avec max/min ≤ 5 ou (max - min) ≤ 500, où le max et le min se réfèrent respectivement aux intensités maximum et minimum pour un gène particulier à travers des échantillons de mARN.

(iii) transformation logarithmique de base 10. Les niveaux d'expression pour chacun des 38 échantillons a indiqué la nécessité de normaliser les niveaux d'expression dans des rangées avant de combiner des données à travers des échantillons. Ils ont maintenu les

données qui ont été alors récapitulées par une matrice $X = (x_{ji})$ 3051 $\times$38, où le x_{ji} dénote le niveau d'expression pour le gène j dans l'échantillon i de mARN de tumeur. Nous avons enlevé un individu d'une manière aléatoire pour avoir une base de données 3051×37. Nous avons fait ce changement pour distinguer nos résultats de ceux de Dudoit et Ge (2004).

Dans l'analyse des données de Golub et *al.* (1999), un but est de distinguer entre les groupes connus ALL et AML en utilisant les expressions observées de gène. La méthode classique serait de faire une analyse discriminante qui exige une matrice non singulière de covariance. Ce n'est pas le cas ici cependant, puisque la matrice est 6817×6817 et a un rang approximativement de $10 + 27 = 37$. Il est nécessaire de ne pas utiliser cette méthode classique et d'adopter la méthode de tests multiples.

Une approche intéressante est celle de la classification ou de la prévision de classe proposée par Tibshirani et *al.* (2003) (voir aussi Storey et Tibshirani (2001)). Ces méthodes sont très prometteuses, mais ici nous nous concentrerons sur les méthodes proposées par Dudoit et *al.* (2002, 2003) déjà décrites dans le chapitre 2. Nous proposons ici d'utiliser le *MaxT* de Westfall et Young (1993) qui a d'excellentes propriétés théoriques. Les fonctions de <<mt.teststat>> et de <<mt.teststat.num.denum>> nous permettent de calculer d'une manière commode des statistiques de test pour chaque rang d'un ensemble de données. Ici nous allons calculer la statistique de t comparant l'expression, pour chaque gène, dans les cas ALL à l'expression dans les cas AML.

Le Q-Q graphique est un outil utile pour l'analyse d'une expérience de microréseau. L'utilisation du Q-Q graphique nous aide à identifier les gènes avec des statistiques de test qui sont peu communs. Le Q-Q graphique est utilisé pour un grand nombre de comparaisons et les points qui dévient nettement de la droite linéaire correspondent aux gènes dont les niveaux d'expression diffèrent entre les deux groupes.

Le Q-Q graphique obtenu pour les données de leucémie est présenté dans la figure 3.1 suivante :

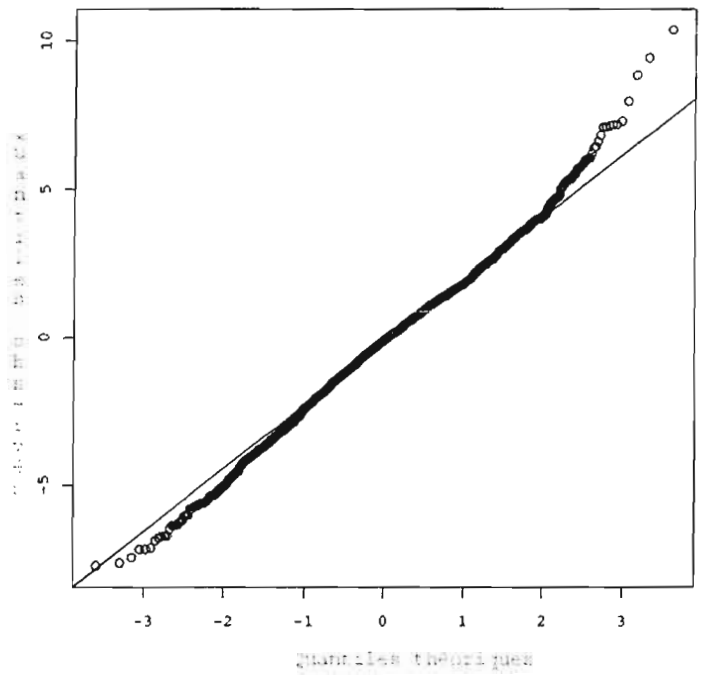

Fig. 3.1 Exemple de Q-Q graphique pour les données de leucémie.

Nous remarquons qu'il y a des gènes qui sont différentiellement exprimés pour ALL et AML aux deux extrêmes. L'expression asymétrique est à noter aussi.

Nous présentons dans la figure 3.2 le graphique des numérateurs et des dénominateurs des statistiques de test :

Le graphique du numérateur versus la racine carrée du dénominateur des statistiques de test nous indique qu'il y a plus de gènes sur-exprimés (associés aux valeurs positives des numérateurs) que de sous-exprimés (associés aux valeurs négatives des

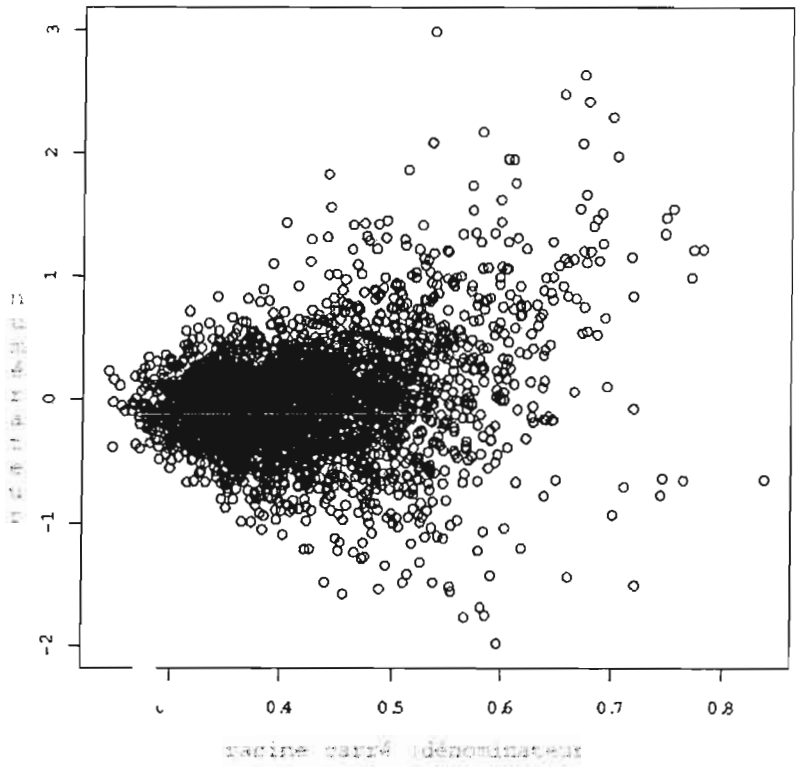

Fig. 3.2 Numérateur versus la racine carrée du dénominateur de statistique de Student t pour les données de leucémie.

numérateurs), à cause du fait que le numérateur varie de -2 à $+3$. Il faut remarquer qu'on a vu les mêmes types d'assymétrie dans le Q-Q graphique. De plus le graphique du numérateur versus la racine carrée du dénominateur montre qu'il y a plus de grandes valeurs positives du numérateur associé avec les grandes valeurs de la variance.

Dans le tableau 3.1 nous présentons les valeurs de p ajustées pour les procédures incluant celles de Bonferroni, Holm, Hochberg et Sidak pour le contrôle fort du taux d'erreur de type 1 (FWER) et celles de Benjamini et Hochberg et de Benjamini et Yekutieli pour le contrôle (fort) du taux de fausses découvertes (FDR).

Comme première approximation, on calcule les valeurs de p nominales brutes, ou non ajustées, pour les 3051 statistiques de test en utilisant la distribution gaussienne standard.

Les valeurs de p ajustées pour ces 7 méthodes de test multiples peuvent être calculées et stockés dans l'ordre original de gène, où SidakSS désigne la procédure pas à pas de Sidak ≪single-step≫ et SidakSD désigne la procédure descendante de Sidak ≪step-down≫.

Les résultats de ces valeurs de p ajustées sont présentés dans le tableau ci-dessous.

Tab. 3.1 Les 10 premières différentes valeurs de p

Bonferroni	rawp	Holm	Hochberg	Sidak SS	Sidak SD	BH	BY
0.08	1	1	1	1	1	0.18	1
0.36	1	1	1	1	1	0.54	1
0.92	1	1	1	1	1	0.96	1
0.73	1	1	1	1	1	0.84	1
0.17	1	1	1	1	1	0.32	1
0.21	1	1	1	1	1	0.36	1
0.47	1	1	1	1	1	0.64	1
0.59	1	1	1	1	1	0.74	1
0.99	1	1	1	1	1	0.99	1
0.89	1	1	1	1	1	0.94	1

Les valeurs présentées dans ce tableau représentent, les 10 premières valeurs de p, parmi les 3051 valeurs de p calculées, pour les 7 différentes méthodes. Clairement, le contrôle de Benjamini et Hochberg est le meilleur. Les valeurs de p pour cette méthode est légèrement plus grande que celles de la méthode de Bonferroni.

On peut obtenir le nombre d'hypothèses rejetées pour plusieurs méthodes de test multiples et différents taux d'erreur nominaux du type 1 avec ≪multtest≫ de Dudoit et Ge (2004). Le nombre d'hypothèses à rejeter en utilisant les valeurs de p non ajustées et les valeurs de p de $MaxT$ pour différentes valeurs du taux d'erreur du type 1 ($\alpha = 0; 0.1; 0.2; \ldots 1$) sont données dans le tableau 3.2.

Tab. 3.2 Le nombre d'hypothèses rejetées pour les méthodes de p non-ajustée et ≪maxT≫ pour différentes valeurs de α

p	p non-ajustée	maxT
0	0	0
0.1	1213	74
0.2	1581	108
0.3	1854	125
0.4	2058	154
0.5	2237	170
0.6	2408	187
0.7	2570	215
0.8	2728	256
0.9	2895	311
1	3051	3051

Le tableau montre qu'il y a beaucoup moins d'hypothèses rejetées avec la méthode de ≪$MaxT$ ≫.

Les gènes avec des valeurs de p de $MaxT$ inférieures ou égales à 0,01 pour l'ensemble de données de leucémie avec 37 individus sont :

Tab. 3.3 Les gènes avec des valeurs de p de $MaxT$ inférieures ou égales à 0,01

Rang	Nom du gène
[1]	"CYSTATIN A"
[2]	"Macmarcks"
[3]	"IEF SSP 9502 mARN"
[4]	"Inducible protein mARN"
[5]	"LYN V-yes-1 Yamaguchi sarcoma viral related oncogene homolog"
[6]	"CD33 CD33 antigen (differentiation antigen)"
[7]	"CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)"
[8]	"FAH Fumarylacetoacetate"
[9]	"ACADM Acyl-Coenzyme A dehydrogenase, C-4 to C-12 straight chain"
[10]	"Azurocidin gene"
[11]	"GLUTATHIONE S-TRANSFERASE, MICROSOMAL"
[12]	"Leukotriene C4 synthase (LTC4S) gene"
[13]	"VIL2 Villin 2 (ezrin)"
[14]	"ME491 gene extracted from H.sapiens gene for Me491/CD63 antigen"
[15]	"RETINOBLASTOMA BINDING PROTEIN P48"
[16]	"T-COMPLEX PROTEIN 1, GAMMA SUBUNIT"
[17]	"Zyxin"
[18]	"LEPR Leptin receptor"
[19]	"MCM3 Minichromosome maintenance deficient (S. cerevisiae) 3"
[20]	"C-myb gene extracted from Human (c-myb) gene, complete primary cds"
[21]	"APLP2 Amyloid beta (A4) precursor-like protein 2"
[22]	"MYL1 Myosin light chain (alkali)"
[23]	"X-LINKED HELICASE II"
[24]	"LYZ Lysozyme"
[25]	"TCF3 Transcription factor 3 "

Dans la base de données avec 38 individus (11 AML et 27 ALL), Dudoit et Ge (2004) ont trouvé que 44 gènes ont été identifiés comme différentiellement exprimés en utilisant $\ll MaxT \gg$ et une valeur de p égale à 0.01. Comment est-ce que l'on peut être certain que la plupart de ces gènes sont différentiellement exprimés ? Existe-il une autre méthode pour choisir les gènes qui sont différentiellement exprimés ? Dans la prochaine section nous discuterons la méthode de ≪jackknife≫ pour essayer de répondre à ces questions.

3.3 La méthode de ≪ jackknife ≫

Avant de parler directement de l'utilisation de la méthode de ≪ jackknife ≫ dans notre problème pour distinguer les gènes qui sont différentiellement exprimés dans une expérience de microréseau, nous trouvons qu'il est utile de présenter l'histoire de l'utilisation de cette méthode.

Quenouille (1949) a présenté une méthode, plus tard appelée la méthode de ≪jackknife≫, pour estimer le biais d'un paramètre inconnu en supprimant chaque fois une observation de la base de données originale et en recalculant l'estimateur basé sur le reste des données (voir Efron (1982), Shao et Du (1996) et Efron et Tibshirani (1993)).

Plus explicitement, selon Shao et Du (1996), soit $T_n = T_n(X_1, \ldots, X_n)$ un estimateur d'un paramètre θ inconnu. Le biais de T_n est définie comme suit :

$$biais(T_n) = E(T_n) - \theta.$$

Soit $T_{n-1,i} = T_{n-1}(X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n)$ la statistique basée sur n-1 observations $X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n, i = 1, \ldots, n$. L'estimateur du biais de ≪jackknife≫ est :

$$b_{JACK} = (n-1)(\bar{T}_n - T_n)$$

où $\bar{T}_n = n^{-1} \sum_{i=1}^{n} T_{n-1,i}$. Ceci mène à un estimateur du biais-réduit de ≪jackknife≫

de θ,

$$T_{JACK} = T_n - b_{JACK} = nT_n - (n-1)\bar{T}_n.$$

Les estimateurs de ≪jackknife≫ b_{JACK} et T_{JACK} peuvent être justifiés comme suit. Supposons que :

$$biais(T_n) = \frac{a}{n} + \frac{b}{n^2} + O\left(\frac{1}{n^3}\right)$$

où a et b sont deux inconnus qui ne dépendent pas de n. Puisque les $T_{n-1,i}, i = 1, \ldots, n$, sont identiquement distribuées, alors :

$$biais(T_{n-1,i}) = \frac{a}{n-1} + \frac{b}{(n-1)^2} + O\left(\frac{1}{(n-1)^3}\right) \qquad (1),$$

et le $biais(\bar{T}_n)$ a la même expression que (1). Donc,

$$\begin{aligned} E(b_{JACK}) &= (n-1)[biais(\bar{T}_n) - biais(T_n)] \\ &= (n-1)[(\frac{1}{n-1} - \frac{1}{n})a + (\frac{1}{(n-1)^2} - \frac{1}{n^2})b + O(\frac{1}{n^3}) \\ &= \frac{a}{n} + \frac{(2n-1)b}{n^2(n-1)} + O(\frac{1}{n^2}), \end{aligned}$$

ce qui veut dire que le biais de ≪jackknife≫ est d'ordre n^{-2} et que la méthode de ≪jackknife≫ produit un estimateur du biais-réduit en enlevant le premier terme dans le $biais(T_n)$.

La méthode de ≪jackknife≫ est devenue un outil plus valable depuis que Tukey (1958) a constaté que cette méthode peut également être utilisée pour construire l'estimateur de la variance. Une justification pour l'utilisation de la méthode de ≪jackknife≫ dans l'estimation de la variance est donnée par Tukey (1958).

Notons que T_{JACK} peut s'écrire

$$T_{JACK} = \frac{1}{n}\sum_{i=1}^{n}[nT_n - (n-1)T_{n-1,i}].$$

Tukey (1995) a défini

$$\bar{T}_{n,i} = nT_n - (n-1)T_{n-1,i}, \quad i = 1, \ldots, n.$$

comme étant la pseudo-valeur de ≪jackknife≫ et que :

(I) les pseudo-valeurs $\bar{T}_{n,i}, i = 1, \ldots, n$, peuvent être traitées comme si elles étaient indépendantes et identiquement distribuées.

(II) les $\bar{T}_{n,i}$ ont approximativement les mêmes variances que les $\sqrt{n}T_n$.

Sous les conditions (I) et (II), il est naturel d'estimer la variance $var(\sqrt{n}T_n)$ par la variance échantillonnale basée sur les $\bar{T}_{n,1}, \bar{T}_{n,2}, \ldots, \bar{T}_{n,n}$, c'est-à-dire qu'on peut estimer la variance $var(T_n)$ par :

$$\begin{aligned} v_{JACK} &= \frac{1}{n(n-1)} \sum_{i=1}^{n} (\bar{T}_{n,i} - \frac{1}{n} \sum_{j=1}^{n} \bar{T}_{n,j})^2 \\ &= \frac{n-1}{n} \sum_{i=1}^{n} (T_{n-1,i} - \frac{1}{n} \sum_{j=1}^{n} T_{n-1,j})^2. \end{aligned}$$

Cet estimateur est bien connu sous le nom de l'estimateur de ≪jackknife≫ de la variance pour T_n.

La méthode de ≪jackknife≫ dépend moins du modèle étudié et n'a pas besoin de la formule théorique exigée par l'approche traditionnelle. Cependant, la méthode de ≪jackknife≫ exige le calcul à plusieurs reprises des n statistiques, ce qui était pratiquement impossible à faire avant l'invention d'ordinateurs modernes.

De nos jours, la méthode de ≪jackknife≫, comme son nom l'indique, est devenue un outil populaire et utile dans l'analyse de données statistiques selon Efron (1982) et Efron et Tibshirani (2003). Il est certainement utile en classification pour déterminer la différence entre le vrai taux d'erreur de classification et le taux estimé. Nous voulons adapter le même principe ici. Au lieu de faire l'inférence une seule fois pour déterminer les gènes différentiellement exprimés, nous allons, chaque fois, éliminer un individu et refaire les tests multiples pour l'ensemble des données de Golub et *al.* (1999) avec la méthode de *MaxT* de Westfall et Young (2003). Nous allons faire une liste en ordre décroissant des gènes qui sont choisis au moins plus que 5 fois. Cette liste va déterminer

les gènes les plus probables à être différentiellement exprimés.

On a appliqué la méthode de ≪jackknife≫ à l'ensemble de données de leucémie, on a fait 38 tests multiples sur les données de la matrice x_{ji} 3051×37, c'est-à-dire qu'à chaque test on enlève un échantillon. Le nombre de fois que chaque gène s'est exprimé est présenté dans le tableau ci-dessous :

Tab. 3.4 Nom du gène et le nombre de fois $n \geq 7$ qu'il s'est exprimé pour les données de leucémie avec la méthode de ≪jackknife≫

Nom du gène	nombre de fois que le gène est exprimé
RB1 Retinoblastoma 1 (including osteosarcoma)	38
APLP2 Amyloid beta (A4) precursor-like protein 2	38
C-myb gene extracted from Human (c-myb) gene, complete primary cds, and five complete alternatively spliced cds	38
T-COMPLEX PROTEIN 1, GAMMA SUBUNIT	38
RETINOBLASTOMA BINDING PROTEIN P48	38
TOP2B Topoisomerase (ADN) II beta (180kD)	38
X-LINKED HELICASE II	38

Nom du gène	nombre de fois que le gène est exprimé
Zyxin	38
TCRA T cell receptor alpha-chain	37
Inducible protein mARN	37
LEPR Leptin receptor	37
VIL2 Villin 2 (ezrin)	37
IRF2 Interferon regulatory factor2	35
LYN V-yes-1 Yamaguchi sarcoma viral related oncogene homolog	34
Macmarcks	34
DHPS Deoxyhypusine synthase	34
GB DEF = Homeodomain protein HoxA9 mARN	34
SPTAN1 Spectrin, alpha,non-erythrocytic 1 (alpha-fodrin)	33

Nom du gène	nombre de fois que le gène est exprimé
ACADM Acyl-Coenzyme A dehydrogenase, C-4 to C-12 straight chain	33
ME491 gene extracted from H.sapiens gene for Me491/CD63 antigen	33
Leukotriene C4 synthase (LTC4S) gene	32
PLECKSTRIN	32
INTERLEUKIN-8 PRECURSOR	31
MYL1 Myosin light chain (alkali)	31
Cytoplasmic dynein light chain 1 (hdlc1) mARN	30
Putative enterocyte differentiation promoting factor mARN, partial cds	30
IEF SSP 9502 mARN	29

Nom du gène	nombre de fois que le gène est exprimé
CTSD Cathepsin D (lysosomal aspartyl protease)	29
TCF3 Transcription factor 3 (E2A immunoglobulin enhancer binding factors E12/E47)	29
FAH Fumarylacetoacetate	28
CD33 CD33 antigen (differentiation antigen)	24
CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)	21
INDUCED MYELOID LEUKEMIA CELL DIFFERENTIATION PROTEIN MCL1	20
Lysophospholipase homolog (HU-K5) mARN	20
CCND3 Cyclin D3	17
HKR-T1	17
Interleukin 8 (IL8) gene	17

Nom du gène	nombre de fois que le gène est exprimé
CYSTATIN A	15
GLUTATHIONES-TRANSFERASE, MICROSOMAL	12
CYSTATIN A	11
MEF2A gene (myocyte-specific enhancer factor 2A, C9 form) extracted from Human myocyte-specific enhancer factor 2A (MEF2A) gene, first coding	11
Transcriptional activator hSNF2b	11
FAH Fumarylacetoacetate	10
LYZ Lysozyme	9
TCF3 Transcription factor 3 (E2A immunoglobulin enhancer binding factors E	9
CD33 CD33 antigen (differentiation antigen)	8

Nom du gène	nombre de fois que le gène est exprimé
IGB Immunoglobulin-associated beta (B29)	8
GTBP ADN G/T mismatch-binding protein	7
TCF12 Transcription factor 12 (HTF4, helix-loop-helix transcription factors 4)	7
Uridine diphosphoglucose pyrophosphorylase mARN	7

Pour valider l'utilisation de la méthode de ≪jackknife≫, nous faisons référence aux résultats de Dudoit et Ge (2004) qui contiennent la liste des 44 gènes choisis dont la valeur de p pour $MaxT$ est plus petite ou égale à 0.01%. Nous comparons leur liste à la notre obtenue par la méthode de ≪jackknife≫. Parmi les gènes sur la liste de Dudoit et Ge (2004), toutes sauf 2 sont inclues sur notre liste et 26 sont choisies par le ≪jackknife≫ plus que 29 fois.

C'est clair que la méthode de ≪jackknife≫ est un outil très important pour mettre en évidence les gènes qui sont différentiellement exprimés. Les gènes qui sont choisis chaque fois par la méthode de ≪jackknife≫ ont une très grande probabilité d'être différentiellement exprimés.

3.4 La fonctions ≪ mt.plot≫ : Dudoit et Ge (2004)

On peut se demander si ≪MaxT≫ est une bonne méthode à utiliser et on peut la comparer avec les autres méthodes.

Il est possible d'avoir un certain nombre de sommaires graphiques pour les résultats des méthodes de test multiples et des valeurs de p ajustées correspondantes. La figure 3.3 présente des graphiques des valeurs de p non ajustées assorties de permutation et des valeurs de p ajustées pour les méthodes de Bonferroni, *MaxT* , Benjamini et Hochberg (1995) et de Benjamini et Yekutieli (1999) fait par ≪ mt.plot≫.

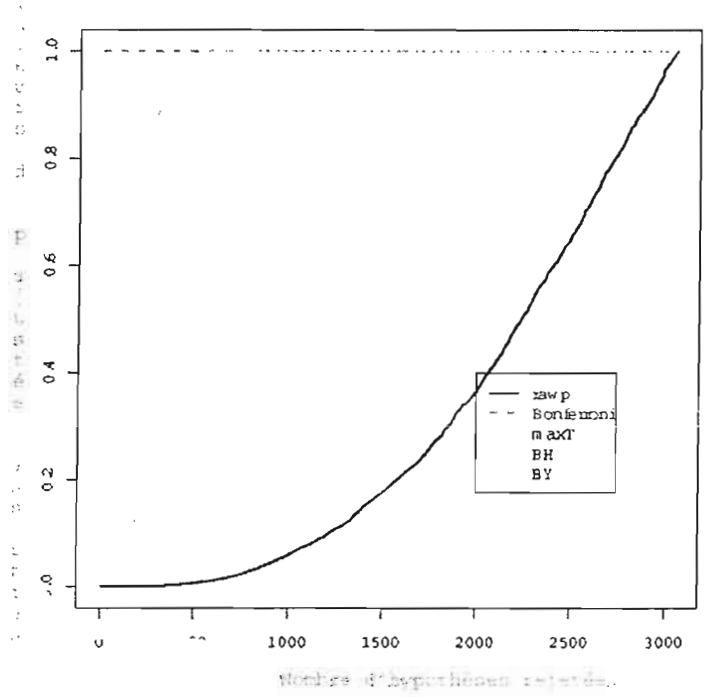

Fig. 3.3 Valeurs de p ajustées assorties versus le nombre d'hypothèses rejetées pour les données de leucémie

Ce graphique montre que la méthode de Bonferroni a tendance de rejeter toutes les hypothèses nulles peu importe la valeur de p ajustée. Par contre ≪rawp≫ ne rejette pas suffisamment d'hypothèses nulles. La méthode de Benjamini et Yekutieli donne des résultats semblables à ≪rawp≫. On peut conclure que les méthode de ≪MaxT≫ et de Benjamini et Hochberg sont très semblables et représentent un bon compromis entre les méthodes extrêmes de Bonferroni et ≪rawp≫.

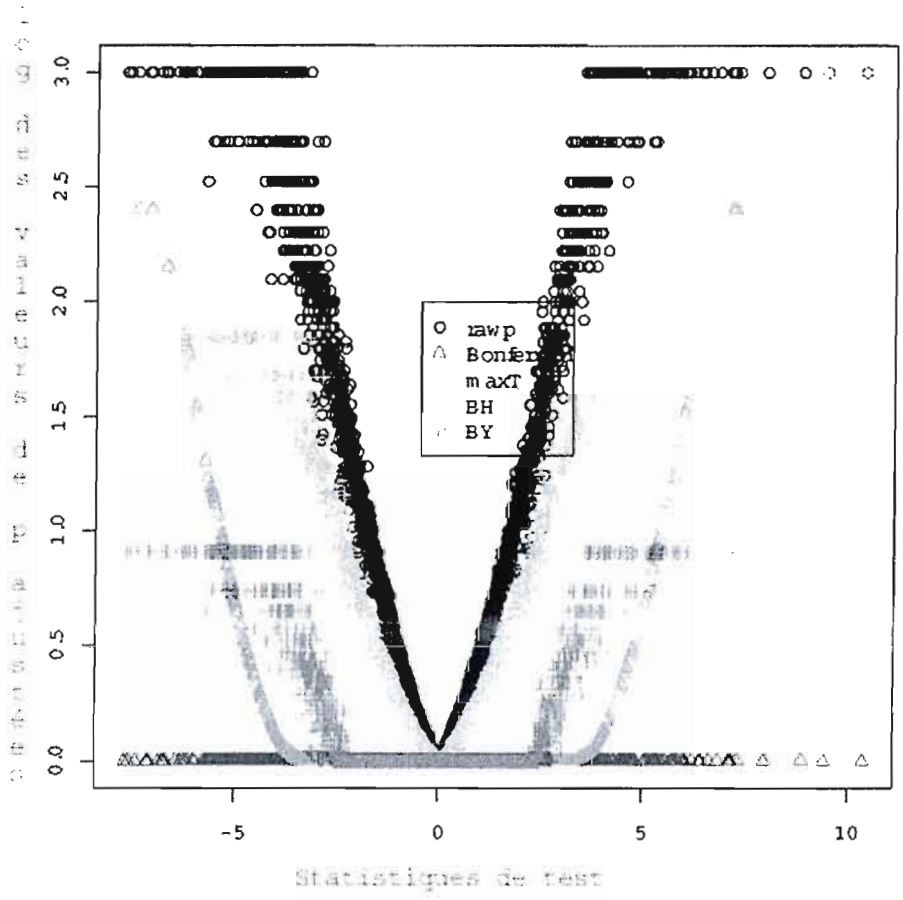

Fig. 3.4 -Logarithme des valeurs de p ajustées versus les statistiques de student t pour les données de leucémie.

La figure 3.4 présente le graphique du (-logarithme des valeurs de p ajustées) versus les statistiques de students t pour les données de leucémie. C'est clair que les valeurs des statistiques de student t ne varient pas beaucoup pour la méthode de ≪rawp≫, les valeurs augmentent très vite. Le graphique de la méthode de ≪MaxT≫ est plus proche de celui de Benjamini et Yekutieli que de celui de Benjamini et Hochberg mais ils sont assez proches.

3.5 L'analyse des données TIB de Tibshirani

Étant donné que le ≪jackknife≫ a donné d'excellents résultats avec les données de Golub et *al.* (1999), nous traitons dans cette parties les données TIB (http ://www-stat.stanford.edu/tibs/clickwra p/sam/academic/index.html) qui concernent deux types de tumeurs différents de Tibshirani et *al.* (2003) avec le progiciel Biocondutor pour les tests multiples. L'ensemble de données étudiées portent sur des mesures d'expression de gène d'un ensemble d'expériences de microréseau, ainsi que la variable de réponse de chaque expérience. La variable de réponse prend la valeur 1 pour un sujet traité et la valeur 0 pour un sujet non traité. Nous commençons par lire les données. L'ensemble d'étude comporte $n = 200$ échantillons, 150 cas traités (tib.cl=1) et 50 cas non traités (tib.cl=0).

L'ensemble de données de TIB contient les données du niveau d'expression de gène pour les 200 échantillons de mARN de tumeur et 2000 gènes. L'ensemble de données inclut :

Tib : la matrice 2000×200 des niveaux d'expression des gènes

Tib.gnames : la matrice 2000×2 des gènes identifiés, c'est la matrice qui contient les noms des gènes identifiés

Tib.cl : un vecteur des étiquettes de classe de tumeur (0 ou 1).

Nous commençons par présenter un Q-Q graphique des statistiques de test (figure ci-dessous) pour identifier les gènes avec des statistique de test qui sont peu communes.

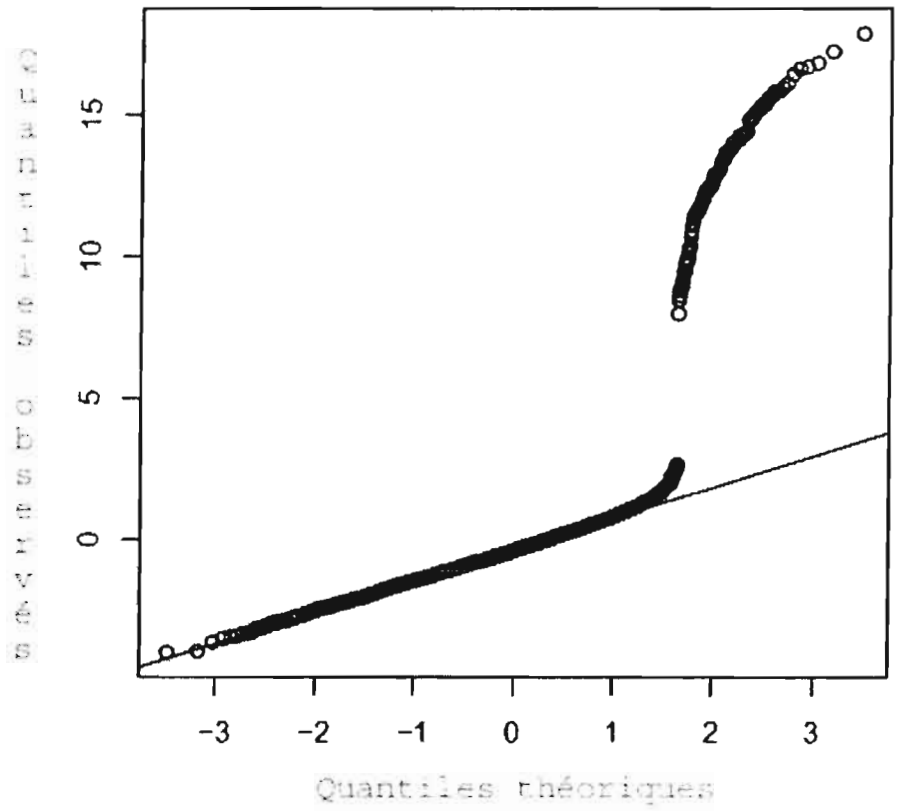

Fig. 3.5 Q-Q graphique pour les données de TIB

Clairement il y a des gènes différentiellement exprimés pour les données TIB à une extrémité seulement.

Nous présentons également le graphique des numérateurs et des dénominateurs des statistiques de test :

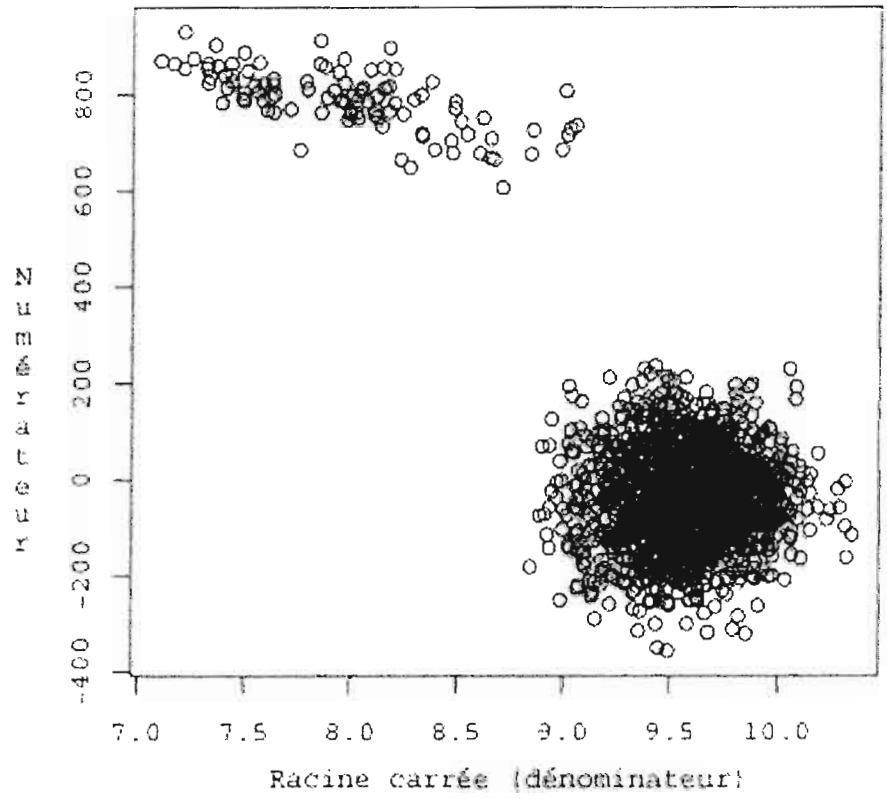

Fig. 3.6 Graphique des numérateurs et dénominateur des statistiques de test pour les données TIB

Le graphique du numérateur versus la racine carrée du dénominateur des statistiques de test pour les données TIB montre qu'il y a plus de gènes sur-exprimées que de sous-exprimés.

On va appliquer la fonction <<mt.$MaxT$ >> aux données TIB de Tibshirani pour calculer les valeurs de p ajustées par permutation pour les méthodes descendantes de test multiple de $MaxT$ et de $MinP$ décrites précédemment, le nombre de permutations B utilisé est égale à 1000.

On utilise la fonction <<mt.reject>> aux données TIB de Tibshirani pour calculer le nombre d'hypothèses rejetées pour plusieurs méthodes de test multiples et différents taux d'erreur nominaux du type 1. Les résultats sont présentés dans le tableau ci-dessous.

Tab. 3.5 Les valeurs des statistiques ≪rawp≫ et ≪$MaxT$ ≫ pour différentes valeurs de α pour les données TIB

p	p non ajustées	maxT
0	0	0
0.1	360	100
0.2	592	100
0.3	801	100
0.4	968	102
0.5	1178	102
0.6	1349	102
0.7	1511	103
0.8	1674	103
0.9	1840	105
1	2000	2000

Le tableau 3.5 montre qu'il y a plus d'hypothèses rejetées avec la méthode de ≪MaxT≫

Les gènes avec des valeurs de p de $\ll MaxT \gg$ inférieures ou égales à 0,01 sont les gènes 0001 à 0100.

Pour avoir les graphiques des résultats des méthodes de test multiples et des valeurs de p ajustées correspondantes, nous allons appliquer la fonction <<mt.plot>> aux données TIB de Tibshirani et *al.* (2003). Les graphiques sont ceux des valeurs de p non ajustées assorties de permutation et des valeurs de p ajustées pour les méthodes de Bonferroni, $MaxT$, Benjamini et Hochberg(1995) et de Benjamini et Yekutieli(1999).

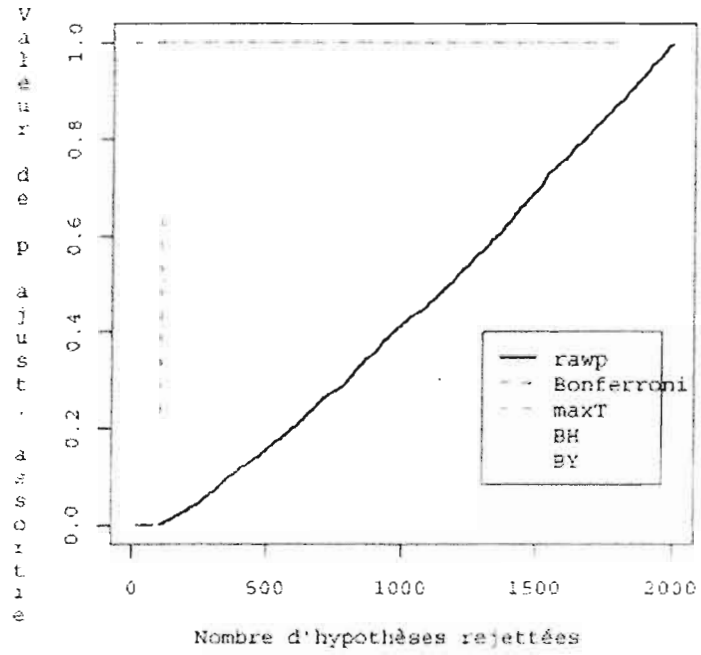

Fig. 3.7 Valeurs de p ajustées assorties pour les données TIB

Le graphique 3.7 montre qu'il y a plus d'hypothèses nulles rejetées par la méthode de Bonferroni et moins d'hypothèses nulles rejetées par la méthode ≪rawp≫. Par contre les méthodes de ≪MaxT≫ et de Benjamini et Hochberg sont semblables et représentent

une alternative aux méthodes extrêmes de Bonferroni et de ≪rawp≫

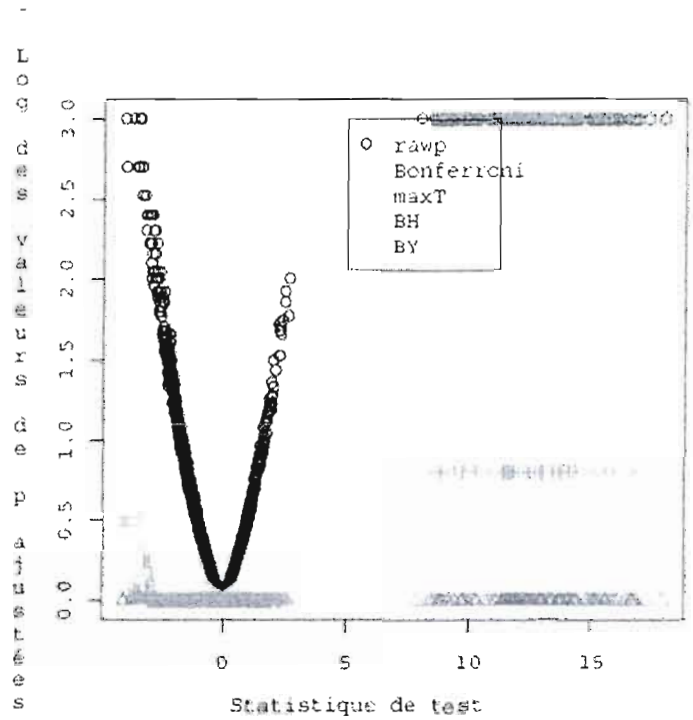

Fig. 3.8 -Logarithme des valeurs de p ajustées versus les statistiques de student t pour les données TIB

La figure 3.8 présente le graphique de (-logarithme des valeurs de p ajustées) versus les statistiques de student t pour les données TIB. On remarque que les statistiques de student t ne varient presque pas pour la méthode de ≪rawp≫. Le graphique de la méthode de ≪MaxT≫ est plus proche de celui de la méthode de Benjamini et Yekutieli que de celui de la méthode de Benjamini et Hochberg.

3.5.1 Méthode de ≪jackknife≫ appliquée aux données TIB de Tibshirani

On a appliqué la méthode de ≪jackknife≫ à l'ensemble de données de TIB de Tibshirani, en faisant 50 tests multiples sur les données de la matrice $\tilde{x}_{ji}$ 2000×50,

échantillonnée d'une manière aléatoire de la matrice des données x_{ji} 2000×200. Sur la matrice des données $\tilde{x}_{ji}$ 2000×50, on a enlevé chaque fois une colonne. Le nombre de fois que chaque gène s'est exprimé est présenté dans le tableau ci-dessous :

Tab. 3.6 Nom du gène et le nombre de fois qu'il s'est exprimé pour les données TIB de Tibshirani avec la méthode de ≪jackknife≫

Nom du gène	nombre de fois que le gène est exprimé
"Gène 0001"	50
"Gène 0003"	50
"Gène 0004"	50
"Gène 0011"	50
"Gène 0013"	50
"Gène 0015"	50
"Gène 0016"	50
"Gène 0017"	50
"Gène 0019"	50
"Gène 0021"	50
"Gène 00023"	50
"Gène 0025"	50
"Gène 0033"	50
"Gène 0035"	50

Nom du gène	nombre de fois que le gène est exprimé
"Gène 0036"	50
"Gène 0038"	50
"Gène 0041"	50
"Gène 0044"	50
"Gène 0048"	50
"Gène 0049"	50
"Gène 0050"	50
"Gène 0052"	50
"Gène 0053"	50
"Gène 0054"	50
"Gène 0055"	50
"Gène 0058"	50
"Gène 0061"	50
"Gène 0065"	50

Nom du gène	nombre de fois que le gène est exprimé
"Gène 0066"	50
"Gène 0069"	50
"Gène 0070"	50
"Gène 0073"	50
"Gène 0076"	50
"Gène 0080"	50
"Gène 0081"	50
"Gène 0082"	50
"Gène 0088"	50
"Gène 0093"	50
"Gène 0095"	50
"Gène 0096"	50
"Gène 0097"	50
"Gène 0100"	50

Nom du gène	nombre de fois que le gène est exprimé
"Gène 0027"	49
"Gène 0047"	49
"Gène 0060"	49
"Gène 0071"	49
"Gène 0091"	49
"Gène 0099"	49
"Gène 0007"	48
"Gène 0045"	48
"Gène 0057"	48
"Gène 0029"	43
"Gène 0098"	41
"Gène 0005"	39
"Gène 0022"	24
"Gène 0014"	22

Nom du gène	nombre de fois que le gène est exprimé
"Gène 0063"	16
"Gène 0006"	15
"Gène 0034"	12

Encore ici la méthode de ≪jackknife≫ semble être une technique prometteuse.

3.6 Étude de simulation

Nous avons illustré les techniques d'analyse de données d'une expérience de microréseau décrite par Dudoit et *al.* (2003) en utilisant deux ensembles de données. Plusieurs auteurs ont analysé ce type de données en utilisant des techniques semblables. On fait maintenant une étude de simulation qui illustre la difficulté de contrôler l'erreur de type 1 en faisant les tests d'hypothèses multiples. On fixe un niveau α nominal du test à plusieurs valeurs différentes et calculer le taux d'erreur par comparaison (PCER), le taux d'erreur par-famille (FWER) et le taux de fausses découvertes (FDR).

3.6.1 Simulation

On commence par générer deux groupes. le groupe I est représenté par une matrice d'ordre 400×25. chaque colonne de cette matrice suit une loi normale $N(0_{400,1}, I_{400,400})$. Pour le groupe II, on génére une matrice d'ordre 400×25, chaque vecteur de cette matrice suit une loi normale $N(\mu, \Sigma)$ où $\mu = (1.5/50, 1.5*2/50, 1.5*3/50, \ldots, 1.5, -1.5/50, -1.5*2/50, \ldots, -1.5, 0, 0, \ldots, 0)^t$ et

$$\Sigma = \begin{pmatrix} \Sigma_1 & \Sigma_2 \\ \Sigma_3 & \Sigma_4 \end{pmatrix}$$

Avec :

$$\Sigma_{1_{50\times 50}} = \begin{pmatrix} 1 & 0.8 & 0.8 & . & . & 0.8 \\ 0.8 & 1 & 0.8 & . & . & 0.8 \\ 0.8 & . & . & . & . & 0.8 \\ . & . & . & . & . & . \\ . & . & . & . & . & 0.8 \\ 0.8 & . & . & . & 0.8 & 1 \end{pmatrix}$$

et :

$\Sigma_{1_{50\times 350}} = 0_{50\times 350}$, $\Sigma_{3_{350\times 50}} = 0_{350\times 50}$ et $\Sigma_{4_{350\times 350}} = I_{350\times 350}$

On fait un ≪multtest≫ avec différents niveaux de α pour les deux groupes pour savoir les gènes qui sont différentiellement exprimés. Soit R_b le nombre de ces gènes et soit V_b le nombre d'erreurs de type 1 et T_b le nombre d'erreurs de type 2 pour les $b = 1, \ldots, 200$ simulations. On définit Q_b comme étant le rapport entre V_b et R_b : $Q_b = \frac{V_b}{R_b}$ si $R_b \neq 0$ et $Q_b = 0$ si R_b =0.

$PCER = \frac{\sum_b V_b/m}{B}$,

$FWER = \frac{\sum_b I(V_b \geq 1)}{B}$,

$FDR = \frac{\sum_b Q_b}{B}$.

Les résultats obtenues sont présentées dans le tableau ci-dessous :

Tab. 3.7 Les valeurs des taux d'erreur pour différentes valeurs de α et cov=0.8

α	0.01	0.04	0.05	0.06	0.09	0.1
PCER	0	0.01007	0.01368	0.01676	0.02288	0.02467
FWER	0	0.00861	0.01201	0.01464	0.02034	0.02161
FDR	0	0.001210	0.001764	0.002093	0.002573	0.002793

La représentation graphique des résultats obtenues :

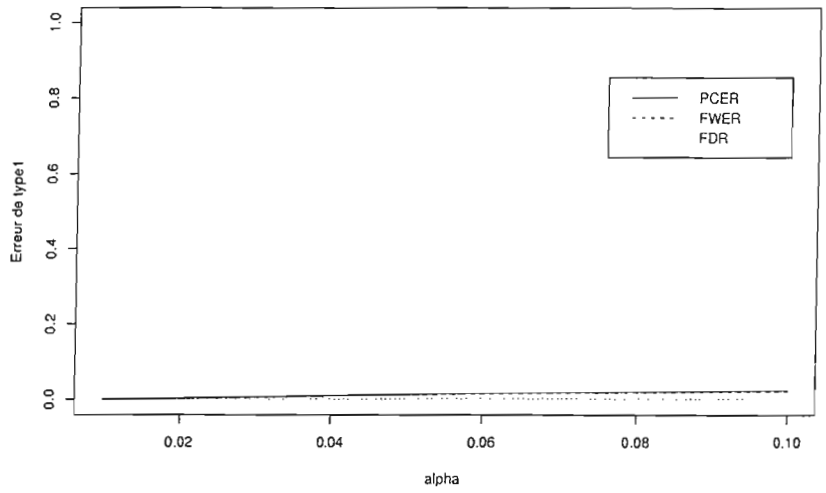

Fig. 3.9 PCER, FWER et FDR versus alpha pour cov=0.8

On refait les mêmes procédures pour différentes valeurs de la matrice Σ :

$$1\text{-}\ \Sigma_{1_{50\times 50}} = \begin{pmatrix} 1 & 0.6 & 0.6 & . & . & 0.6 \\ 0.6 & 1 & 0.6 & . & . & 0.6 \\ 0.6 & . & . & . & . & 0.6 \\ . & . & . & . & . & . \\ . & . & . & . & . & 0.6 \\ 0.6 & . & . & . & 0.6 & 1 \end{pmatrix}$$

et :

$\Sigma_{1_{50\times 350}} = 0_{50\times 350}$, $\Sigma_{3_{350\times 50}} = 0_{350\times 50}$ et $\Sigma_{4_{350\times 350}} = I_{350\times 350}$

Les résultats obtenues sont :

Tab. 3.8 Les valeurs des taux d'erreur pour différentes valeurs de α et cov=0.6

α	0.01	0.04	0.05	0.06	0.09	0.1
PCER	0	0.01020	0.01389	0.0168 0	0.02280	0.02437
FWER	0	0.00845	0.01149	0.01428	0.01983	0.02145
FDR	0	0.001957	0.003178	0.003595	0.004507	0.004334

La représentation graphique des résultats obtenues :

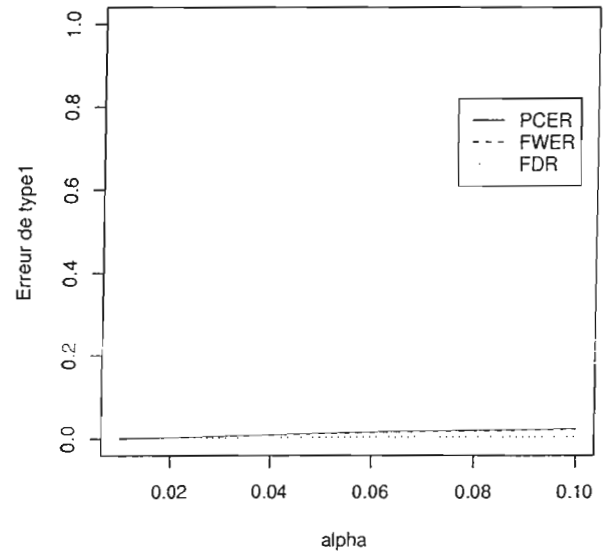

Fig. 3.10 PCER, FWER et FDR versus alpha pour cov=0.6

$$2\text{-}\ \Sigma_{1_{50\times 50}} = \begin{pmatrix} 1 & 0.4 & 0.4 & . & . & 0.4 \\ 0.4 & 1 & 0.4 & . & . & 0.4 \\ 0.4 & . & . & . & . & 0.4 \\ . & . & . & . & . & . \\ . & . & . & . & . & 0.4 \\ 0.4 & . & . & . & 0.4 & 1 \end{pmatrix}$$

et :

$\Sigma_{1_{50\times 350}} = 0_{50\times 350}$, $\Sigma_{3_{350\times 50}} = 0_{350\times 50}$ et $\Sigma_{4_{350\times 350}} = I_{350\times 350}$

Les résultats obtenues sont :

Tab. 3.9 Les valeurs des taux d'erreur pour différentes valeurs de α et cov=0.4

α	0.01	0.04	0.05	0.06	0.09	0.1
PCER	0	0.00957	0.01358	0.01626	0.02252	0.02417
FWER	0	0.00886	0.01182	0.01412	0.02016	0.02165
FDR	0	0.003515	0.004620	0.005033	0.006625	0.006714

La représentation graphique des résultats obtenues :

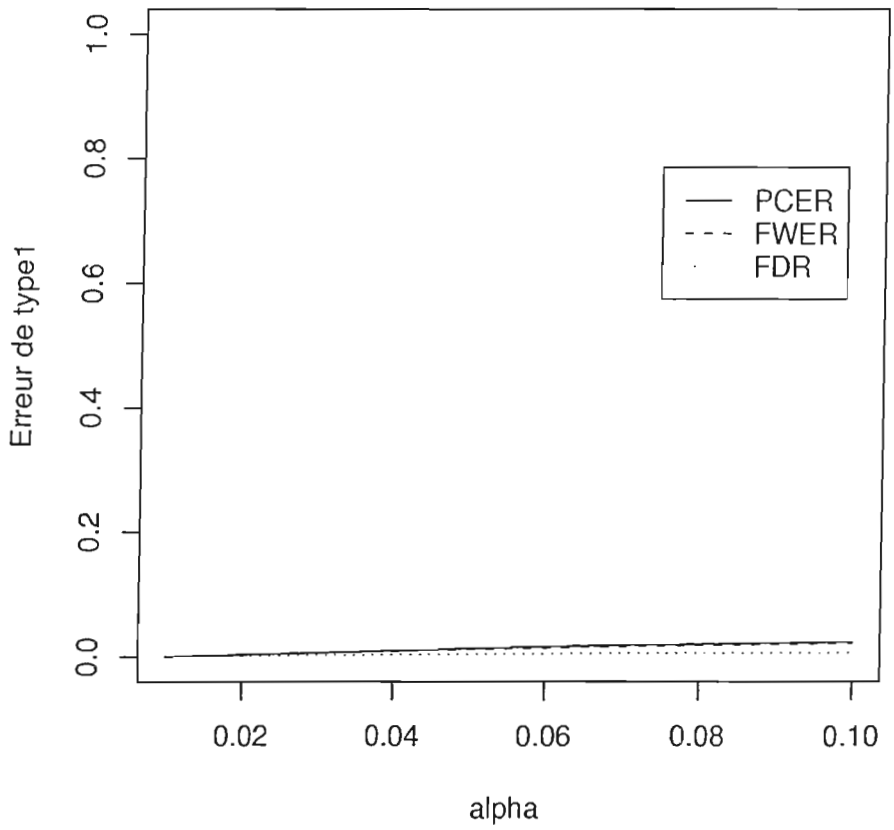

Fig. 3.11 PCER, FWER et FDR versus alpha pour cov=0.6

$$3\text{-}\ \Sigma_{1_{50\times 50}} = \begin{pmatrix} 1 & 0.2 & 0.2 & . & . & 0.2 \\ 0.2 & 1 & 0.2 & . & . & 0.2 \\ 0.2 & . & . & . & . & 0.2 \\ . & . & . & . & . & . \\ . & . & . & . & . & 0.2 \\ 0.2 & . & . & . & 0.2 & 1 \end{pmatrix}$$

et :

$\Sigma_{1_{50\times 350}} = 0_{50\times 350}$, $\Sigma_{3_{350\times 50}} = 0_{350\times 50}$ et $\Sigma_{4_{350\times 350}} = I_{350\times 350}$

Les résultats obtenues sont :

Tab. 3.10 Les valeurs des taux d'erreur pour différentes valeurs de α et cov=0.2

α	0.01	0.04	0.05	0.06	0.09	0.1
PCER	0	0.00948	0.01314	0.01601	0.02206	0.02381
FWER	0	0.00912	0.01239	0.01508	0.02075	0.02225
FDR	0	0.003624	0.004563	0.004845	0.005339	0.005302

La représentation graphique des résultats obtenues :

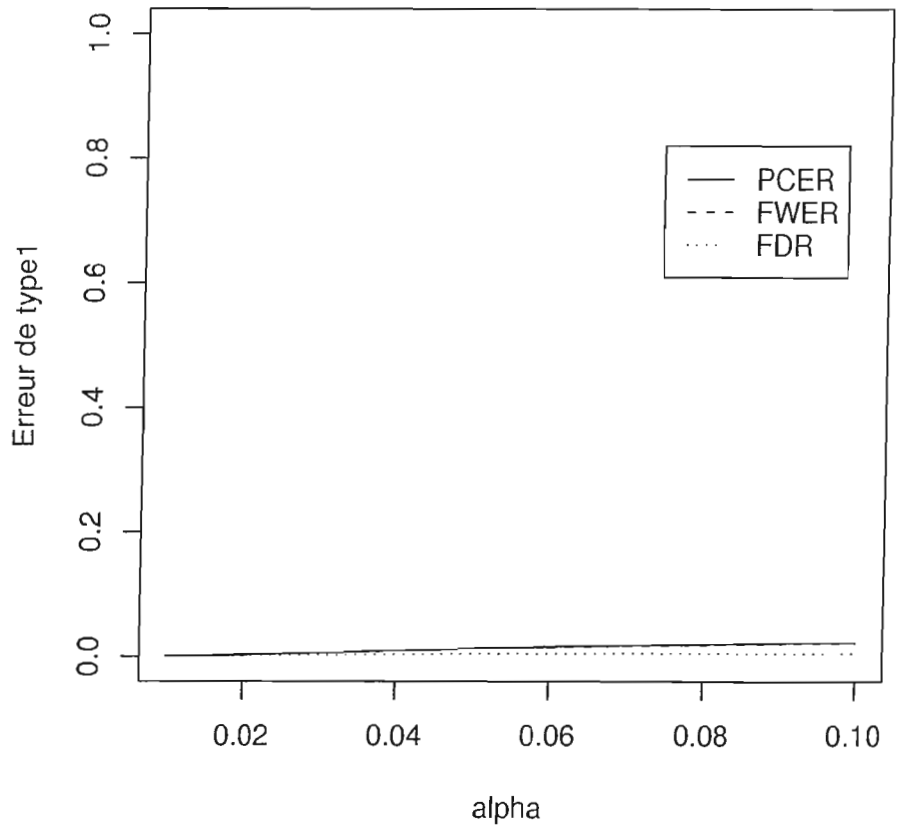

Fig. 3.12 PCER, FWER et FDR versus alpha pour avec cov=0.2

$$4\text{-}\ \Sigma_{1_{50\times 50}} = \begin{pmatrix} 1 & 0 & 0 & . & . & 0 \\ 0 & 1 & 0 & . & . & 0 \\ 0 & . & . & . & . & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & 0 \\ 0 & . & . & . & 0 & 1 \end{pmatrix}$$

et :

$\Sigma_{1_{50\times 350}} = 0_{50\times 350}$, $\Sigma_{3_{350\times 50}} = 0_{350\times 50}$ et $\Sigma_{4_{350\times 350}} = I_{350\times 350}$

Les résultats obtenues sont :

Tab. 3.11 Les valeurs des taux d'erreur pour différentes valeurs de α et cov=0

α	0.01	0.04	0.05	0.06	0.09	0.1
PCER	0	0.00882	0.01228	0.01486	0.02075	0.02225
FWER	0	0.00912	0.01260	0.01536	0.02088	0.02250
FDR	0	0.003041	0.003036	0.002850	0.002859	0.002777

La représentation graphique des résultats obtenues :

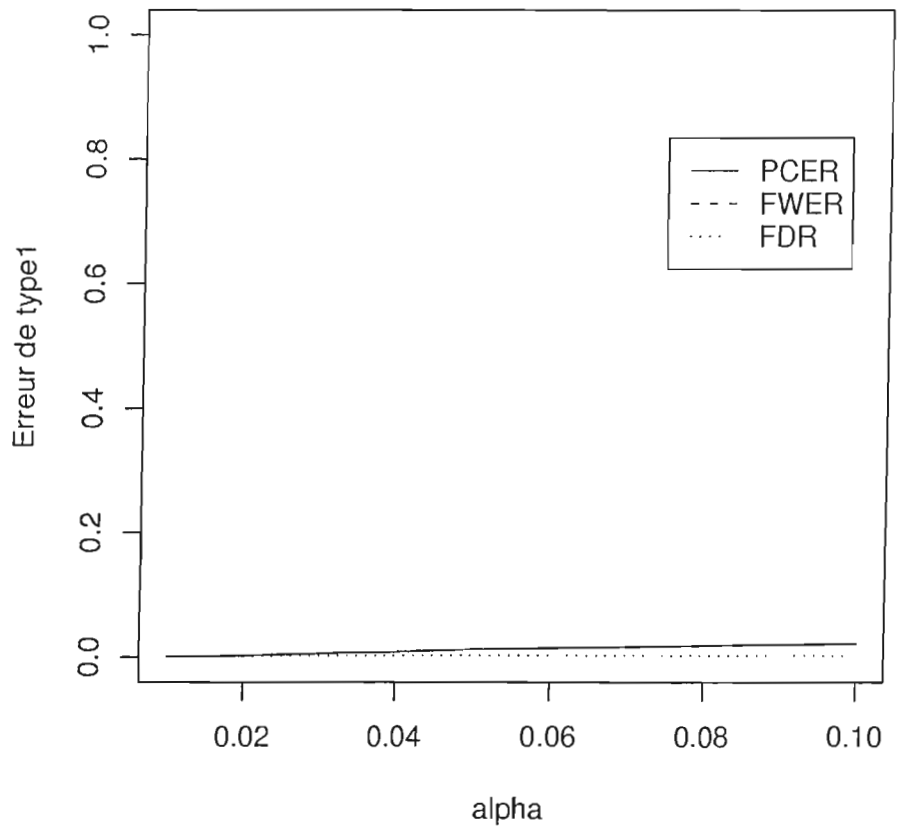

Fig. 3.13 PCER, FWER et FDR versus alpha pour cov=0

Dans chaque cas le PCER et le FWER sont des fonctions croissantes de α. Le même phénomène est vrai pour le FDR. De plus dans cette simulation, si on fixe α, les trois taux d'erreurs sont des fonctions croissantes de la covariance. En général, $FDR < FWER < PCER$ dans cette étude.

CONCLUSION

Dans ce mémoire nous avons étudié des méthodes d'analyse de données d'une expérience de microréseau. Nous avons discuté les problèmes des tests d'hypothèses multiples, le contrôle de l'erreur de type 1 et les taux de faux positifs. Nous avons trouvé que ce domaine de recherche évolue très rapidement en ce moment. Avec le projet de l'étude du génome et l'avance technologique en génétique, il y a une vaste littérature à ce sujet. Essayer de résumer les approches nous mènerait dans beaucoup de directions différentes. Nous avons décidé de se concentrer aux techniques de Dudoit et *al.* (2003), de Hochberg (1988), de Benjamini et Hochberg (1995) et de Westfall et Young (93).

Les problèmes des tests d'hypothèses multiples, les différentes mesures de l'erreur de type 1 et le contrôle de telles erreurs ont été discutés en détail ici. Nous avons décidé de souligner les tests, les mesures d'erreurs de type 1 et leurs contrôles favorisés par Dudoit et *al.* (2003). Nous avons essayé de donner une description assez complète de ces concepts. Nous avons aussi démontré un intérêt à la théorie des inégalités des probabilités dues à Dunn (1967), Jogdeo (1970) et Simes (1986) parce que ces inégalités forment la base de la méthodologie pour contrôler les différentes erreurs de type 1, nous avons inclut les démonstrations de ces inégalités, étant donnée leur importance. En même temps, pour aider le lecteur à mieux comprendre, nous avons mis nos propres démonstrations de certains lemmes et inégalités pour lesquels nous n'avons pas trouvé la démonstration dans la littérature. Une recherche en statistique qui considère les tests n'est pas complète sans une description d'une procédure informatique pour la mise en application. Nous avons décidé d'utiliser le progiciel ≪multtest≫ de Bioconductor (http ://www.bioconductor.org / packages / bioc / stable / src /contrib /html /multtest.html) pour faire une telle analyse de données de microréseau. Nous l'avons décrit en

détail dans le chapitre 3 et l'appendice A. La description trouvée en Dudoit et Ge (2004) dans le progiciel Bioconductor (http ://www.bioconductor.org/packages/bioc/stable /src/contrib/html/multtest.html) est assez détaillée pour aider le lecteur de ce mémoire. Nous avons fait une analyse en utilisant une des méthodes de Dudoit et *al.* (2003) d'un sous ensemble des données de Golub et *al.* (1999) avec les gènes de deux types de leucémie. Nous avons choisi la méthode descendante de maxT de Westfall et Young (1993) à cause de ses bonnes propriétés théoriques. Nous avons aussi introduit la méthode de ≪jackknife≫ comme une alternative pour le contrôle du taux des faux positifs pour cet exemple. La méthode de ≪jackknife≫ a réussi tellement bien dans l'analyse de données que nous avons décidé de l'utiliser pour l'analyse d'un ensemble de données concernant les tumeurs traitées et non traitées de Tibshirani et *al.* (2003). Nous avons conclu que cette méthode de ≪jackknife≫ est une approche très prometteuse.

Pour illustrer les différences entre les mesures de taux d'erreur de type 1 décrites dans le mémoire, nous avons fait une étude de simulation. Nous avons vu que toutes les mesures de l'erreur de type 1, c'est-à-dire, PCER, FWER et FDR croissent avec α et avec la covariance. Nous avons jugé que le FDR est la meilleure mesure.

Il reste beaucoup de directions de recherche future dans ce domaine y compris :

(1) une comparaison très approfondie des méthodes mentionnées ici.

(2) la recherche des méthodes robustes pour faire ces tests multiples.

APPENDICE A

PROGRAMMATION

Ici on donne les programmes utilisés dans le chapitre 3 et 4 pour faire les calculs. Il y a les programmes R utilisés dans le chapitre 3 et les programmes SAS utilisés dans le chapitre 4.
Voici les programmes R utilisés pour analyser les données de Golub, et les données Tib de Tibshirani du chapitre 3. Nous avons utilisé la version R 2.1.1 pour ≪Windows≫.

Programmes pour analyser les données du chapitre3

Lecture des données

> data(jeu_de_données)

#calcul du statistique de test pour chaque gène #

> teststat <− mt.teststat(jeu_de_données, jeu_de_données.cl)

Q-Q graphe des statistiques de test en format eps

```
> postscript("mtQQ.eps")

> qqnorm(teststat)

> qqline(teststat)

# Q-Q graphe des statistiques de test en format pdf #

> pdf("mtQQ.pdf")

> qqnorm(teststat)

> qqline(teststat)

# Graphique des numérateurs et des dénominateurs des statistiques de test#

> tmp <- mt.teststat.num.denum(jeu_de_données, jeu_de_données.cl, test = "t")

> num <- tmp$teststat.num

> denum <- tmp$teststat.denum

> postscript("mtNumDen.eps")
```

```
> pdf("mtNumDen.pdf")

> plot(sqrt(denum), num)

> plot(sqrt(denum), num)

# Calcul des valeurs de p des statistiques de test #

> rawp0 <- 2 * (1 - pnorm(abs(teststat)))

# Calcul des valeurs de p ajustées des statistiques de test pour les différentes
méthodes #

> procs <- c("Bonferroni", "Holm", "Hochberg", "SidakSS", "SidakSD","BH",
"BY")

> res <- mt.rawp2adjp(rawp0, procs)

> adjp <- res$adjp[order(res$index), ]

> round(adjp[1 :10, ], 2)

# Calcul des valeurs de p de permutation #
```

> resT < − mt.maxT(jeu_de_données, jeu_de_données.cl, B = 1000)

> ord < − order(resT$index)

> rawp < − resT$rawp[ord]

> maxT < − resT$adjp[ord]

> teststat < − resT$teststat[ord]

Calcul du nombre d'hypothèses nulles rejetées par les méthodes de rawp et maxT

> mt.reject(cbind(rawp, maxT), seq(0, 1, 0.1))$r

Identification des gènes avec des valeur de p ≤ 0.01

> which < − mt.reject(cbind(rawp, maxT), 0.01)$which[, 2]

> jeu_de_données.gnames[*which*, 2]

Graphique des valeurs de p ajustées pour les méthodes de Bonferroni, maxT, Benjamini et Hochberg et Benjamini et Yekutieli

```
> res <- mt.rawp2adjp(rawp, c("Bonferroni", "BH", "BY"))

> adjp <- res$adjp[order(res$index), ]

> allp <- cbind(adjp, maxT)

> dimnames(allp)[[2]] <- c(dimnames(adjp)[[2]], "maxT")

> procs <- dimnames(allp)[[2]]

> procs <- procs[c(1,2,5,3,4)]

> cols <- c(1, 2, 3, 5, 6)

> ltypes <- c(1, 2, 2, 3, 3)

> postscript("mtpvsr.eps")

# Graphique des valeurs de p ajustées contre les statistiques de test #

> mt.plot(allp[, procs], teststat, plottype = "pvsr", proc = + procs, leg = c(2000,
0.4), lty = ltypes, col = cols, lwd = 2)

> pdf("mtpvsr.pdf")
```

> mt.plot(allp[, procs], teststat, plottype = "pvsr", proc = +procs, leg = c(2000, 0.4), lty = ltypes, col = cols, lwd = 2)

> postscript("mtpvst.eps")

> mt.plot(allp[, procs], teststat, plottype = "pvst", logscale = +TRUE, proc = procs, leg = c(-0.5, 2), pch = ltypes, col = +cols)

> pdf("mtpvst.pdf")

> mt.plot(allp[, procs], teststat, plottype = "pvst", logscale = + TRUE, proc = procs, leg = c(-0.5, 2), pch = ltypes, col = + cols)

On décrit aussi les programmes SAS utilisés pour obtenir la simulation du chapitre 4. Nous avons utilisé SAS version 8.2 pour ≪Windows≫. Nous avons utilisé également quelques macros pour SAS qui prennent en entrée les différents valeurs et retournent les valeurs des statistiques de test : FWER, PCER et FDR .

Programme de la Simulation du chapitre 4

%macro mvn3 (varcov=, /*les données pour la matrice de variances-covariances*/

means=, /* les données pour le vecteur moyenne */

n=, /* la taille de l'échantillon */

seed=, /* valeur initiale pour générer un nombre aléatoire */

```
sample=) ;          /* le nom d'ensemble de données */

/* Donner une valeur génératrice initiale «seed». si seed ¡ 0, alors on génére cette
valeur du système d'horloge. */

data _null_ ;

if &seed le 0 then do ;

seed = int(time()) ;          /* le temps de l'horloge est un entier */

put seed= ;

call symput('seed',seed) ;       /* on met les ≪seed≫ dans une variable macro */

end ;

run ;

/* Générer une base de données normalement multivariées dans SAS/IML */

proc iml worksize=100 ;

use &varcov ;          /* lire la matrice de variance et covariance */

read all into cov ;

use &means ;          /* lire le vecteur des moyennes */

read all into mu ;

v=nrow(cov) ;          /* calculer le nombre des variables */

n=&n ;

seed = &seed ;

l=t(root(cov)) ;          /* la décomposition de Choleski de la matrice de variance-
covariance*/
```

```
z=normal(j(v,&n,&seed)) ;          /* taille d'échantillon * nombre de variable
normales*/
x=l*z ;          /* multiplier par la racine carré de Choleski */
x=repeat(mu,1,&n)+x ;          /* ajouter la moyenne */
tx=t(x) ;
create &sample from tx ;          /* écrire les données d'échantillon dans un
l'ensemble de données de SAS */
append from tx ;
quit ;
%mend mvn3 ;
/* Construire le premier bloc de la matrice de variance-covariance*/
data varcovsig1 ;
array sigma150 ;
do j=1 to 50 ;
do i=1 to 50 ;
if i eq j then
sigma1i=1 ;else sigma1(i)=0.3 ;end ;output ;end ;
drop i j ;run ;
data varcovsig1 ;set varcovsig1 ;id=_N_ ;run ;
/* Construire le deuxième bloc de la matrice de variance-covariance*/
data varcovsig2 ;
```

```
array sigma250 ;
do j=1 to 50 ;
do i=1 to 50 ;
if i eq j then
sigma2i=1 ;else sigma2(i)=0.3 ;end ;output ;end ;
drop i j ;run ;
data varcovsig2 ;set varcovsig2 ;id=_N_+50 ;run ;
/* Construire le troisième bloc de la matrice de variance-covariance*/
data zeroh1 ;
array sigma250 ;
do j=1 to 50 ;
do i=1 to 50 ;
sigma2i=0 ;end ;output ;end ;
drop i j ;run ;
data zeroh1 ;set zeroh1 ;id=_N_ ;run ;
/* Construire le quatrième bloc de la matrice de variance-covariance*/
data zeroh2 ;
array sigma150 ;
do j=1 to 50 ;
do i=1 to 50 ;
```

```
sigma1(i)=0;end;output;end;

drop i j;run;

data zeroh2;set zeroh2;id=_N_+50;run;

/* Fusionner les blocs pour avoir une première partie de la matrice de variance-
covariance */

data part1;merge varcovsig1 zeroh1;by id;run;

/* Fusionner les blocs pour avoir une deuxième partie de la matrice de variance-
covariance */

data part2;merge zeroh2 varcovsig2;by id;run;

/*Fusionner les deux parties pour avoir la partie supérieure gauche*/

data leftup;set part1 part2;by id;run;

/*Construire la partie supérieure droite */

data rightup;

array sigma3300;

do j=1 to 100;

do i=1 to 300;

sigma3i=0;end;output;end;

drop i j;run;

data rightup;set rightup;id=_N_;run;

/*Construire la partie inférieure droite */

data rightdown;
```

```
array sigma3300;
do j=1 to 300;
do i=1 to 300;
if i eq j then
sigma3i=1;else sigma3(i)=0;end;output;end;
drop i j;run;
data rightdown;set rightdown;id=_N_+100;run;
/*Construire la partie inférieure gauche */
data lefdown;
array sigma150;
array sigma2(50);
do j=1 to 300;
do i=1 to 50;
sigma1i=0;
sigma2(i)=0;end;output;end;
drop i j;run;
data lefdown;set lefdown;id=_N_+100;run;
/*Construire la partie supérieure de la matrice de variance-covariance*/
data up;merge leftup rightup;by id;run;
/*Construire la partie inférieure de la matrice de variance-covariance*/
```

```
data down ;merge lefdown rightdown ;by id ;run ;
/* Fusionner les 2 parties pour obtenir la matrice de variance-covariance */
data varcovid ;set up down ;by id ;
drop id ;run ;
/* Construire le vecteur Moyenne */
data up1 ;
do i=1 to 50 ;
mu=1.5*i/50 ;output ;end ;run ;
data up1 ;set up1 ;id=_N_ ;run ;
data up2 ;
do i=1 to 50 ;
mu=-1.5*i/50 ;output ;end ;run ;
data up2 ;set up2 ;id=_N_+50 ;run ;
data up3 ;
do i=1 to 300 ;
mu=0 ;output ;end ;run ;
data up3 ;set up3 ;id=_N_+100 ;run ;
data meansid ;set up1 up2 up3 ;by id ;
drop i id ;run ;
%macro simu(iter) ; /* nombre de simulations */
```

```
%do i=1 %to &iter;

/*Générer les 25 premiers échantillons aléatoires de taille 400 */

%mvn3 (varcov = varcovid, means = meansid, seed = 1736117 + 7 * (&i) + 13 *
(&i) * (&i), n = 25, sample = test3) ;

/*Générer les 25 deuxième échantillons aléatoires de taille 400 */

%mvn3 (varcov = varcovid, means = meansid, seed = 5637931 + 17 * (&i) *
(&i), n = 25, sample = test4);

data thrvec(rename=(col1-col400=xone1-xone400)) ;set test3 ;id=_N_ ;run ;

data forvec(rename=(col1-col400=xtwo1-xtwo400)) ;set test4 ;id=_N_ ;run ;

data vec1 ;merge thrvec forvec ;by id ;

drop id ;run ;

data vec1 ;set vec1 ;id=1 ;run ;

data vecyew1 ;set vec1 ;

/* Calcul des moyennes des vecteurs */

array moyu(400) ;

array moyd(400) ;

array x1(400) xone1-xone400 ;

array x2(400) xtwo1-xtwo400 ;

by id ;

if first.id then ;do i=1 to 400 ;moyu(i)=0 ;moyd(i)=0 ;end ;

do i=1 to 400 ;
```

```
moyu(i)=moyu(i)+x1(i);
moyd(i)=moyd(i)+x2(i);end;
do i=1 to 400;
moyu(i)=moyu(i)/25;
moyd(i)=moyd(i)/25;end;
if last.id;
keep id moyu1-moyu400 moyd1-moyd400;
run;
/* Calcul des variances */
data vecmu1;merge vec1 vecyew1;by id;run;
data vechew11;set vecmu1;
array mu(400) moyu1-moyu400;
array md(400) moyd1-moyd400;
array x1(400) xone1-xone400:
array x2(400) xtwo1-xtwo400;
array sig(400);
by id;
if first.id then;do i=1 to 400;sig(i)=0;end;
do i=1 to 400;
sig(i)=sig(i)+(x1(i)-mu(i))*(x1(i)-mu(i))+(x2(i)-md(i))*(x2(i)-md(i));end;
```

```
if last.id ;
keep id sig1-sig400 ;
run ;
/* Calcul des écart-types */
data vechew1 ;set vechew11 ;
array su(400) sig1-sig400 ;
array s(400) ;
do i=1 to 400 ;
s(i)=sqrt(su(i)/48) ;end ;
keep id s1-s400 ;run ;
/* Calcul des t-tests, des valeurs de p, du nombre de gènes qui sont différentiellement exprimés, des taux d'erreur de type 1 et de type 2 et des statistiques de test */
data vecmusig1 ;merge vecyew1 vechew1 ;by id ;run ;
data ttest1 ;set vecmusig1 ;
array mu(400) moyu1-moyu400 ;
array md(400) moyd1-moyd400 ;
array sd(400) s1-s400 ;
array ttestu(50) ;
array ttestd(50) ;
array pvalu(50) ;
array pvald(50) ;
```

```
 array ru(50) ;

 array rd(50) ;

 den=sqrt(2/45) ;

 do i =1 to 50 ;

 ttestu(i)=(mu(i)-md(i))/(sd(i)*den) ;

 pvalu(i)=1-cdf('T',ttestu(i),48,0) ;

 ru(i)=(pvalu(i)¡0.05) ;end ;

 do i =1 to 50 ;

 ttestd(i)=(mu(i+50)-md(i+50))/(sd(i+50)*den) ;

 pvald(i)=1-cdf('T',ttestd(i),48,0) ;

 rd(i)=(pvald(i)¡0.05) ;end ;

 r1val=(sum(of ru1-ru50)) ;

 r2val=(sum(of rd1-rd50)) ;

 keep r1val r2val ;

 proc append base=result data=ttest1 ;

run ;

%end ;

%mend simu ;

%simu(200) ;

proc univariate noprint data=result ;
```

```
var r1val r2val;output out=res1 sum=r1s r2s;run;

proc print;var r1s r2s;run;
```

BIBLIOGRAPHIE

Anderson, T. W. (1955). The integral of a symmetric unimodal function over a symmetric convex set and some probability inequalities. *Proceding of the American Mathematical Society 6* , pp. 170-176.

Benjamini, Y. et Hochberg, Y. (1995). Controlling the false discovery rate : A practical and powerful approach to multiple testing. *J. Roy. Statist. Soc., Ser. B 57* , pp. 289-300.

Benjamini, Y. et Yekutieli (2001). The control of the false discovery rate in multiple testing under dependency. *Ann. Statist. 29*, pp. 1165-1188.

Brown, P.O. et Botstein, D. (1999). Exploring the new world of the genome with DNA microarrays. *Nature Genetics*, Vol. 21, pp. 33-37.

Callow, M. J., Dudoit, S., Gong, E. L., Speed, T. P. et Rubin, E. M. (2000). Microarray expression profiling identifies genes with altered expression in HDL-deficient mice. *Genome Research 10*, pp. 2022-2029.

Dabrowski, A. (2004). Statistics for comparison of two independant cDNA filter microarrays, dans Statistical Modeling and Analysis for Complex Data Problems (Eds. P. Duchesne et B. Remillard). Kluwer, 161-178.

Dudoit, S., Yang, Y. H., Callow, M. J. et Speed, T. P. (2000). Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statist. Sinica 12*, pp. 111-139.

Dudoit, S., Shaffer, J. P. et Boldrick, J. C.(2002). Multiple hypothesis testing in microarray experiments. Technical Report 110, Division of Biostatistics, Univ. California, Berkeley. Available at http ://www.bepress.com/ucbbiostat/ paper110/.

Dudoit, S., Shaaffer, J. P. et C. Boldrick (2003). Multiple hypothesis testing in microarrays experiments. *Statistical Science*, Vol. 18, No. 1, pp. 71-103.

Dudoit, S. et Ge, Y. (2004). Bioconductor's multtest package. *www.bioconductor.org / packages / bioc / stable / src / contrib / html / multtest.html.*

Dunn, O. J. (1958). Estimation of the means of dependent variables. *Ann. Math Statist, 29*, pp. 1095-1111.

Efron B. (1982). *The jackknife, the bootstrap, and other resampling plans.* Philadelphia : Society for Industrial and Applied Mathematics.

Efron B. et Tibshirani R. (1993). *An introduction to the bootstrap.* New York : Chapman and Hall.

Hochberg, Y. (1988). A sharper Bonferroni procedure for multiple tests of significance. *Biometrika 75*, pp. 800-802.

Hochberg, Y. et Tamhane, A. C. (1987). *Multiple Comparison Procedures.* New York : Wiley.

Hogg R. V. et Tanis E. A. (2001). *Probability and Statistical Inference.* New Jersey : Prentice Hall.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scand. J. Statist. 6*, pp. 65-70.

Jogdeo, K. (1977). Association and probability inequalities. *Ann. Statist. 5*, pp. 495-504.

Jogdeo, K. (1970). A simple proof of an inequality for multivariate normal probabilities of rectangles. *Ann. Statist. 41*, pp. 1357-1359.

Lange K. (2002). *Mathematical and Statistical Methods for Genetic Analysis.* New York : Springer.

Lee, M-L.T., Kuo, F.C., Whitmore, G.A., et Sklar, J. (2000). Importance of microarray gene expression studies : Statistical methods and evidence from repetitive cDNA hybridizations. *Proccedings of the National Academy of Sciences, 97*, pp. 9834-9839.

Lehmann, E. L. (1986). *Testing Statistical Hypotheses.* New York : 2nd ed. Wiley.

Shao J. et Tu D. (1996). *The Jackknife and Bootstrap.* Springer Series in Statistics, pp. 71-103.

Sidak, Z. (1967). Rectangular confidence regions for the means of multivariate normal distributions. *J. Amer. Statist. Assoc. 62*, pp. 626-633.

Simes, R. J. (1986). An improved Bonferroni procedure for multiple tests of significance. *Biometrika 73*, pp. 751-754.

Slonim R., D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D. et Lander, E. S. (1999). Molecular classification of cancer : Class discovery and class prediction by gene expression monitoring. *Science 286*, pp. 531-537.

Storey, J. D. et Tibshirani, R. (2001). Estimating the positive false discovery rate under dependence, with applications to DNA microarrays. Technical Report 2001-28, Dept. Statistics, Stanford University.

Tibshirani R., Hastie T., Narasimhan B. et Chu G. (2003). Class prediction by nearest

shrunken centroids, with applications to DNA microarrays. *Statistical Science, Vol. 18*, pp. 104-117.

Westfall, P. H. et Young, S. S. (1993). *Resampling-Based Multiple Testing : Examples and Methods for p-Value Adjustment.* New York : Wiley.

Wright, S. P. (1992). Adjusted p-values for simultaneous inference. *Biometrics 48*, pp. 1005-1013.

Yekutieli, D. et Benjamini, Y. (1999). Resampling-based false discovery rate controlling multiple test procedures for correlated test statistics. *J. Statist. Plann. Inference 82*, pp. 171-196.