Classification non supervisée d'un graphe de co-expression avec des méta-données pour la détection de micro-ARNs

Florian Brunet [1], **Jérôme Mariette** [1], **Christine Cierco-Ayrolles** [1], **Christine Gaspin** [1], **Philippe Bardou** [2], **Nathalie Villa-Vialaneix** [1,3]

1. *INRA, Unité MIAT, Castanet Tolosan*
   France
   {florian.brunet,jmariett,christine.cierco,christine.gaspin}@toulouse.inra.fr
2. *INRA, LGC, SIGENAE, Castanet Tolosan*
   France
   philippe.bardou@toulouse.inra.fr
3. *SAMM, Université Paris 1*
   France
   nathalie.villa@univ-paris1.fr

ABSTRACT. *The present article proposes a method to cluster the nodes of a graph that we are using for a specific task in biology. Our purpose is to detect, with an unsupervised approach, possible miRNA yet unknown. To do so, similar candidate miRNAs are clustered into groups. A multi-kernel approach is used to integrate information given by a co-expression network and by additional information describing its nodes. This approach is made more robust by a bagging of clustering. This methodology leads to cluster candidate miRNAs into groups that give a good indication about which candidates are true miRNAs and which ones are false positives.*

RÉSUMÉ. *Nous présentons dans cet article une méthode de classification non supervisée de sommets d'un graphe qui est utilisée dans un contexte biologique particulier. La problématique est de détecter de manière non supervisée des micro-ARNs probables. Pour ce faire, nous utilisons une approche multi-noyaux permettant d'intégrer des informations sur le graphe de co-expression et des informations supplémentaires sur les sommets de ce graphe. Cette approche est rendue robuste par une technique de bagging de classifications. Les résultats obtenus donnent des groupes de miRNAs potentiels dont certains permettent de discriminer avec une bonne confiance les vrais miRNAs des faux positifs.*

KEYWORDS: *clustering, kernel, bagging, bootstrap, miRNA, co-expresion network*

MOTS-CLÉS : *classification, noyau, bagging, bootstrap, miRNA, réseau de co-expression*

DOI:10.3166/HSP..1-12

1. Contexte : la détection de micro-ARNs

L'acide ribonucléique (ARN ou RNA en anglais) est une molécule biologique que l'on trouve dans les cellules des organismes vivants et qui est synthétisée à partir d'une matrice d'ADN dont il est une copie. Les micro-ARNs (miRNAs) sont un ensemble de petits ARNs qui sont impliqués dans la régulation des gènes et dans la manière dont ils s'expriment (i.e., sont utilisés) dans la cellule. Chez les animaux, les miRNAs ont un rôle de régulation majeur durant le développement embryonnaire et sont aussi impliqués dans diverses maladies comme le cancer, les maladies cardio-vasculaires et certaines dégénérescences neurologiques. Il est donc important de pouvoir identifier et caractériser de nouveaux miRNAs afin de pouvoir étudier de manière plus précise leur implication dans le fonctionnement du vivant.

Les méthodes habituelles de détection des miRNAs utilisent un génome de référence ainsi qu'un jeu de données de lectures issues du séquençage haut débit de type sRNA (Friedländer *et al.*, 2008). Cependant, les prédictions ainsi obtenues comprennent très souvent de nombreux faux positifs, qu'il est possible de détecter à l'aide de méthodes d'apprentissage supervisé (Ding *et al.*, 2010; 2011; Mapleson *et al.*, 2013) (SVM) qui nécessitent l'utilisation d'un ensemble d'apprentissage, contenant exemples positifs et négatifs. Nous nous focalisons ici sur un problème intermédiaire dans lequel nous utilisons une approche que nous pourrons qualifier de semi-supervisée : classiquement les chaînes de traitements des données issues des séquenceurs fournissent une liste de miRNAs potentiels et un certain nombre de tests permettent d'éliminer les faux positifs les plus évidents. Par ailleurs, sur les miRNAs potentiels restants, un certain nombre sont identifiés comme étant de vrais micro-ARNs par l'utilisation de banques de données publiques référençant les miRNAs telles que miRBase (Griffiths-Jones *et al.*, 2006) ou des banques plus généralistes sur les ARNs non codants telles que RFAM (Griffiths-Jones *et al.*, 2005), les autres étant de nature inconnue. Le but du travail est de s'appuyer sur cette connaissance préalable pour rechercher, parmi l'ensemble des ARNs dont nous disposons, ceux qui sont les plus probablement des miRNAs. L'originalité de l'approche, par rapport aux approches supervisées décrites plus haut, est que l'on ne dispose pas d'un ensemble d'apprentissage préalable puisque l'on ne dispose que de vrais positifs mais pas d'exemples négatifs.

Nous proposons, pour aborder cette question, d'utiliser une approche par graphe : les données dont nous disposons sur les ARNs, candidats potentiels à être des miRNAs, sont un graphe de co-expression dans lequel les sommets sont des miRNAs et les arêtes indiquent une forte co-expression entre deux miRNAs (voir la section 3 pour plus de détails sur l'inférence de ce graphe). Par ailleurs, sont également disponibles des méta-données décrivant les ARNs et qui sont généralement pertinentes pour leur identification en tant que miRNA. Ces méta-données sont des variables numériques et catégorielles. Ainsi, les données peuvent être décrites sous la forme d'un graphe dont les sommets sont étiquetés par un ensemble de descripteurs, numériques et non numériques. Nous proposons le développement et la mise en œuvre d'une méthode de classification non supervisée des sommets d'un graphe pour grouper les ARNs en

classes homogènes selon tous leurs descripteurs. Les clusters ainsi obtenus seront ensuite analysés selon qu'ils contiennent ou non un grand nombre de miRNAs déjà connus.

Le reste de l'article est organisé comme suit : la section 2 décrit la méthodologie utilisée pour une classification non supervisée des sommets d'un graphe étiqueté. Cette méthodologie allie approche multi-noyaux, pour intégrer les diverses informations (sur la structure du graphe et sur les attributs des sommets), avec une approche par ré-échantillonnage dont le but est de rendre plus robuste la classification obtenue. Enfin, la section 3 décrit de manière plus précise les données ARN utilisées et les résultats obtenus.

2. Classification non supervisée de sommets dans les graphes étiquetés

2.1. Contexte du problème et état de l'art

Dans cette section, nous introduisons une approche de classification non supervisée pour des sommets d'un graphe étiqueté. De manière plus précise, la méthode aborde le cas d'un graphe $\mathcal{G}$, éventuellement pondéré, contenant n sommets $V = \{x_1, \ldots, x_n\}$. Les sommets sont décrits par un certain nombre de variables additionnelles, appelées attributs, qui sont rangées en deux catégories : d_1 attributs catégoriels où c_i^j désigne le j-ème attribut catégoriel de l'individu i, qui est une valeur prise dans un ensemble fini $\{v_1^j, \ldots, v_{k_j}^j\}$ et d_2 attributs numériques où e_i^j désigne le j-ème attribut numérique de l'individu i, qui est une valeur numérique réelle. On notera également $e_i = (e_i^1, \ldots, e_i^{d_2})$ le vecteur d'attributs numériques du sommet x_i. Enfin, le sommet augmenté de ses attributs catégoriels et numériques sera noté dans la suite $\tilde{x}_i := (x_i, (c_i^j)_j, (e_i^j)_j)$.

Le but ici est de prendre en compte de manière équilibrée les divers types d'informations disponibles : structure du graphe (regrouper des sommets fortement connectés comme dans la problématique standard décrite dans les articles de revue (Fortunato, 2010; Schaeffer, 2007)), attributs catégoriels et attributs numériques (regrouper des sommets dont les attributs sont similaires, ce qui, pour les attributs numériques, est un problème standard de classification non supervisée). La classification de sommets dans des graphes de ce type a été abordée dans plusieurs travaux préalables : (Steinhaeuser, Chawla, 2008) effectue une classification principalement basée sur les attributs des sommets qui est ensuite corrigée par un principe de seuillage basé sur les poids des arêtes entre sommets ; (Ester *et al.*, 2006; Moser *et al.*, 2007; Ge *et al.*, 2008) formalisent cette question sous la forme d'un problème d'optimisation basé sur des distances entre attributs proches de l'algorithme des k-moyennes, en introduisant des contraintes sur la connexité des classes. À l'inverse, d'autres auteurs favorisent la structure du graphe dans leur classification, comme (Cruz *et al.*, 2011; Li *et al.*, 2008). Enfin, d'autres auteurs cherchent, comme nous, à équilibrer les contributions des différents types de données : (Combe *et al.*, 2012; 2013) combinent deux critères (un critère de modularité et un critère d'entropie) pour obtenir un critère

global à optimiser tenant compte des différents objectifs. (Zhou *et al.*, 2009) combinent diverses dissimilarités en une dissimilarité globale qui est utilisée pour la classification. Dans la suite, nous décrivons une approche similaire, qui est basée sur l'utilisation d'une combinaison de *noyaux* comme dans l'extension des cartes auto-organisées présentée dans (Olteanu *et al.*, 2013) : des résultats expérimentaux sur données synthétiques ont montré que la combinaison d'informations par une approche à noyaux multiples permettait d'améliorer la qualité de la classification.

2.2. Classification par k-moyennes à noyaux multiples

Une méthode simple et efficace pour la classification non supervisée est l'algorithme des k-moyennes (Shawe-Taylor, Cristianini, 2004; Dhillon *et al.*, 2004) : cette approche partage les observations en k classes et affecte chaque observation à la classe dont l'individu moyen est le plus proche. De manière plus précise, pour un tableau de données numériques $(x_{ij})_{i=1,\ldots,n,\, j=1,\ldots,d}$, l'algorithme des k-moyennes cherche à trouver la partition $\mathcal{C}_1, \ldots, \mathcal{C}_k$ qui minimise le critère (inertie intra-classes)

$$\sum_{j=1,\ldots,k} \sum_{i \in \mathcal{C}_j} \|x_i - p_j\|^2 \qquad \text{où} \qquad p_j = \frac{1}{|\mathcal{C}_j|} \sum_{i \in \mathcal{C}_j} x_i. \tag{1}$$

L'algorithme procède de manière itérative en alternant l'affectation des observations à la classe dont le centre de classe p_j est le plus proche avec le re-calcul des centres de classe. La convergence de l'algorithme est généralement atteinte en quelques itérations mais l'algorithme présente le désavantage important de fournir des résultats qui dépendent fortement de l'initialisation (généralement aléatoire) de la classification.

Dans le cas où les données ne sont pas numériques, la distance $\|x_i - p_j\|^2$ n'est pas calculable de manière naturelle mais on peut les définir au moyen de *noyaux*. Un noyau K sur l'espace abstrait $\mathcal{X}$ est une application de $\mathcal{X} \times \mathcal{X}$ dans $\mathbb{R}$, symétrique ($K(x, x') = K(x', x)$) et positive ($\forall\, N \in \mathbb{N},\ \forall (\alpha_k)_{k=1,\ldots,N} \subset \mathbb{R},\ \sum_{kk'} \alpha_k \alpha_{k'} K(x_k, x_{k'}) \geq 0$). (Aronszajn, 1950) montre qu'une telle application est un produit scalaire d'une projection ϕ des données de $\mathcal{X}$ dans un espace de Hilbert $(\mathcal{H}, \langle ., . \rangle) : K(x, x') = \langle \phi(x), \phi(x') \rangle$. L'intérêt croissant autour de ce type de similarités vient du fait qu'une fois le noyau choisi, ni ϕ, ni $\mathcal{H}$ n'ont besoin d'être explicites pour pouvoir calculer des distances entre individus x et x' qui sont simplement calculées par :

$$\|\phi(x) - \phi(x')\|^2 = K(x, x) + K(x', x') - 2K(x, x'). \tag{2}$$

Comme dans (Olteanu *et al.*, 2013), nous proposons de décrire les "distances" entre sommets du graphe au moyen de plusieurs noyaux, chacun de ces noyaux cor-

respondant aux proximités selon la structure du graphe ou bien selon les attributs numériques ou bien selon les attributs catégoriels[1].

Dans le cas d'un graphe, plusieurs noyaux décrivant les similarités entre sommets peuvent être utilisés. Les plus populaires sont des versions régularisées du Laplacien L du graphe (Smola, Kondor, 2003), comme le noyau de la chaleur $e^{-\beta L}$, ou *heat kernel*, (Kondor, Lafferty, 2002), ou bien le noyau de temps de parcours ou *commute time kernel*, (Fouss *et al.*, 2007), qui n'est autre que l'inverse généralisée du Laplacien du graphe et s'interprète comme la mesure du temps moyen nécessaire pour relier deux sommets du graphe par une marche aléatoire sur les arêtes.

Pour décrire les similarités entre attributs des sommets, plusieurs choix sont possibles, dépendant de la nature des données : pour les attributs catégoriels c_i^j, une approche courante est de recourir au codage disjonctif de ces variables (c'est-à-dire à leur recodage par modalité en 0/1) et d'utiliser un noyau pour variable numérique. Notons que, dans le cas où le noyau linéaire est utilisé, cette opération conduit à utiliser comme noyau entre deux individus, le nombre d'attributs communs aux deux individus. Pour les attributs numériques e_i^j, le noyau le plus simple est le noyau linéaire : $K_3(e_i, e_{i'}) = (e_{i'})^T e_i$. D'autres noyaux permettent d'appliquer une transformation non linéaire aux données et donc de capter des corrélations plus complexes que la corrélation linéaire comme, par exemple, $K_3(e_i, e_{i'}) = e^{-\beta \|e_i - e_{i'}\|^2}$ (noyau gaussien) ou bien $K_3(e_i, e_{i'}) = (1 + (e_{i'})^T e_i)^P$ (noyau polynomial de degré P).

Le noyau final retenu pour mesurer la similarité globale entre les sommets x_i et $x_{i'}$ est alors $K(\tilde{x}_i, \tilde{x}_{i'}) = \alpha_1 K_1(x_i, x_{i'}) + \alpha_2 K_2(c_i, c_{i'}) + \alpha_3 K_3(e_i, e_{i'})$, où K_1 est le noyau choisi pour mesurer la similarité induite par la structure du graphe, K_2 celui utilisé pour les attributs catégoriels et K_3 celui utilisé pour les attributs numériques. Les $(\alpha_j)_j$ sont des réels positifs tels que $\sum_j \alpha_j = 1$. Il est facile de montrer qu'un tel noyau satisfait aux conditions de l'article (Aronszajn, 1950), toutefois, le choix des valeurs adéquates pour les α_j reste ouvert : (Olteanu *et al.*, 2013) ont proposé l'introduction d'une étape de descente de gradient dans l'algorithme de carte auto-organisatrices. Par nature, cette approche est limitée aux algorithmes de type "stochastique" dans lesquels la classification des observations est mise à jour une par une et non pas de manière globale à chaque itération comme dans l'algorithme des k-moyennes que nous avons décrit ci-dessus. La procédure augmente de plus considérablement le temps de calcul. Dans la mesure où nous n'avons pas d'a priori sur l'importance de chacune des informations introduites dans l'algorithme, nous proposons d'utiliser des poids équilibrés : $\alpha_1 = \alpha_2 = \alpha_3 = 1/3$. Ce choix n'est possible que si les noyaux ont des ordres de grandeur comparables et requiert donc une normalisation préalable de ceux-ci : pour ce faire, nous proposons l'utilisation, pour chacun des trois noyaux, de la normalisation $\widetilde{K}_j(x, x') = \frac{K_j(x,x')}{\sqrt{K_j(x,x) K_j(x',x')}}$

1. On pourrait même envisager, si les attributs catégoriels ou numériques sont naturellement répartis en plusieurs groupes de natures très différentes, d'ajouter un noyau pour chacun de ces groupes de variables, la complexité de notre proposition n'augmentant pas réellement avec le nombre de noyaux.

(voir, par exemple (Gärtner, 2008)). Cette normalisation correspond au calcul de l'angle entre x et x' dans l'espace image défini par le noyau K. La méthode de classification finalement utilisée est décrite dans l'algorithme 1 et utilise le noyau $\widetilde{K}(\tilde{x}_i, \tilde{x}_{i'}) = \alpha_1 \widetilde{K}_1(x_i, x_{i'}) + \alpha_2 \widetilde{K}_2(c_i, c_{i'}) + \alpha_3 \widetilde{K}_3(e_i, e_{i'})$.

Algorithm 1 Classification par k-moyennes à noyau

1: **Require**: Nombre de classes k. Noyau multiple $\widetilde{K}$
2: **Initialization**: $\forall\, i = 1, \ldots, n$, classification $\mathcal{C}(i) \in \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$, aléatoire
3: **repeat**
4: **Mise à jour des centres de classes** : $\forall\, j = 1, \ldots, k$, $p_j = \sum_i \beta_{ji} \widetilde{K}(\tilde{x}_i, .)$ avec

$$\beta_{ji} \leftarrow \begin{cases} \frac{1}{|\mathcal{C}_j|} & \text{si } \mathcal{C}(i) = \mathcal{C}_j \\ 0 & \text{sinon} \end{cases}$$

5: **Mise à jour de la classification** : $\forall\, i = 1, \ldots, n$

$$\mathcal{C}(i) \leftarrow \arg\min_j \|\phi(\tilde{x}_i) - p_j\|^2_{\widetilde{K}}$$

6: **until** convergence
7: **return** classification $\mathcal{C}(i)$

L'étape de mise à jour des classes est effectuée sans recours explicite à l'espace image $\mathcal{H}$ ou à l'application ϕ mais par développement du produit scalaire et par utilisation de l'équation (2):

$$\begin{aligned} \|\phi(\tilde{x}_i) - p_j\|^2_{\widetilde{K}} &= \|\phi(\tilde{x}_i) - \sum_l \beta_{jl} \phi(\tilde{x}_j)\|^2_{\widetilde{K}} \\ &= \|\phi(\tilde{x}_i)\|^2_{\widetilde{K}} - 2 \sum_l \beta_{jl} \langle \phi(\tilde{x}_i), \phi(\tilde{x}_l) \rangle_{\widetilde{K}} + \sum_{ll'} \beta_{jl} \beta_{jl'} \langle \phi(\tilde{x}_l), \phi(\tilde{x}_{l'}) \rangle_{\widetilde{K}} \\ &= \widetilde{K}(\tilde{x}_i, \tilde{x}_i) - 2 \sum_l \beta_{jl} \widetilde{K}(\tilde{x}_i, \tilde{x}_l) + \sum_{ll'} \beta_{jl} \beta_{jl'} \widetilde{K}(\tilde{x}_l, \tilde{x}_{l'}) \end{aligned}$$

2.3. Classification robuste par k-moyennes

L'utilisation de l'algorithme de k-moyennes à noyau avec le noyau $\widetilde{K}$ donne des résultats instables qui dépendent fortement de son initialisation. Dans des expériences sur données simulées, nous avons constaté que l'instabilité de la classification était encore plus forte pour l'algorithme à noyau que pour l'algorithme original des k-moyennes. Plusieurs articles ont abordé ce problème en utilisant des approches par ré-échantillonnage (Leisch, 1999; Dubois, Fridlyand, 2003; Jianhua *et al.*, 2011). Le principe est de tirer un grand nombre d'échantillons bootstrap dans les données de départ et de combiner les classifications obtenues sur une instance d'un algorithme de classification appliquée à ces données. La technique du bootstrap (Efron, 1981) consiste à ré-échantillonner les données en créant de nouveaux échantillons, généralement

de même taille que l'échantillon de départ, en tirant aléatoirement avec remise dans l'échantillon de départ. Son but est à l'origine d'estimer la distribution d'une statistique quelconque pour pouvoir obtenir une estimation de sa précision ; elle a toutefois été utilisée avec succès pour combiner des fonctions de prédiction, comme dans les forêts aléatoires (Breiman, 2001). Pour combiner un grand nombre de classifications, les approches décrites dans les articles cités ci-dessus varient : classification basée sur l'ensemble des centres de classes obtenus, classification basée sur une mesure de similarité entre individus qui résume l'intégralité des classifications ou bien minimisation d'un critère d'entropie sur l'intégralité des classifications.

Dans notre cas, nous proposons l'utilisation d'une approche proche de celle de (Dubois, Fridlyand, 2003) : une mesure de dissimilarité, D, basée sur le nombre de classifications communes des données, est calculée. Cette mesure est obtenue à partir des classifications à la classe de centre le plus proche des observations "out-of-bag", c'est-à-dire des observations qui n'ont pas été sélectionnées dans l'échantillon bootstrap courant. Pour que le nombre d'observations "out-of-bag" soit suffisamment grand, les classifications ont été effectuées sur des échantillons de taille 2/3 de la taille initiale. Cette mesure est finalement utilisée comme entrée d'un algorithme de k-moyennes classique : la stabilité des résultats de cet algorithme est assurée de manière standard, en réalisant plusieurs instances de l'algorithme des k-moyennes et en conservant le résultat qui minimise le critère de l'équation (1) : cette approche est rendue possible par le fait que la classification par k-moyennes des données décrites par la dissimilarité D est beaucoup plus simple et stable que la classification réalisée à partir du noyau $\widetilde{K}$. L'intégralité de l'approche est décrite dans l'algorithme 2, où $M_{ii'}$ compte le nombre d'apparitions simultanées des sommets $\tilde{x}_i$ et $\tilde{x}_{i'}$ dans le même échantillon "out-of-bag" et $A_{ii'}$ compte le nombre de classifications simultanées de ces sommets dans la même classe. Ainsi, $D_{ii'}$ est la fréquence des classifications dans la même classe des sommets x_i et x'_i lorsque ceux-ci sont simultanément dans un échantillon "out-of-bag".

3. Application à la détection de miRNAs

3.1. Description des données

Le graphe considéré dans cette section est issu de données d'expression, c'est-à-dire, de données mesurant l'activité d'un miRNA potentiel dans la cellule au moment de l'expérience. Cette expression correspond au comptage du nombre de lectures qui ont été séquencées et qui se sont alignées avec une forte similarité sur le génome de référence. L'expression de 500 miRNAs potentiels (250 annotés comme tels et 250 inconnus) a été obtenue sur 38 échantillons (données non encore publiées). Les données ont été préalablement normalisées par la méthode implémentée dans le package R **DESeq** puis le graphe a été inféré en utilisant une méthode graphique gaussienne (comme implémentée dans le package R **glasso**, voir (Friedman *et al.*, 2008)) bootstrappée. Par ailleurs, chaque miRNA a été caractérisé par un certain nombre de méta-données ; pour notre problématique de classification, nous avons retenu comme

Algorithm 2 Bagging de classification par k-moyennes à noyau

Require: Nombre de classes k. Noyau multiple $\widetilde{K}$
Initialization: $\forall\, i, i' = 1, \ldots, n$, $A_{ii'} \leftarrow 0$ et $M_{ii'} \leftarrow 0$
for $b = 1 \rightarrow B$ **do**
    Tirer aléatoirement un échantillon de taille $n/3$ dans $\{1, \ldots, n\}$ **return** échantillon $\mathcal{B}_b$
    k-moyennes à noyau de noyau $\widetilde{K}|_{\mathcal{B}_b}$ **return** classification $(\mathcal{C}(i))_{i \in \mathcal{B}_b}$
    Prédire la classification des observations "out-of-bag" **return** classification $(\mathcal{C}(i))_{i \notin \mathcal{B}_b}$
    Mettre à jour: $\forall\, j = 1, \ldots, k, \forall\, i, i' \notin \mathcal{B}_b$, $M_{ii} \leftarrow M_{ii'} + 1$
    Mettre à jour: $\forall\, j = 1, \ldots, k, \forall\, i, i' \notin \mathcal{B}_b$ tels que $i, i' \in \mathcal{C}_j$, $A_{ii} \leftarrow A_{ii'} + 1$
end for
$\forall\, i, i' = 1, \ldots, n$, $D_{ii'} \leftarrow 1 - \frac{A_{ii'}}{M_{ii'}}$
for $r = 1 \rightarrow R$ **do**
    k-moyenne pour D **return** classification $\mathcal{C}^r(i)$; inertie intra-classes $\mathcal{E}^r$
end for
return classification $\mathcal{C}^{r^*}(i)$ avec $r = \arg\min_{r^*} \mathcal{E}^r$

attributs catégoriels la présence ou non d'une structure particulière qui est une structure secondaire en tige-boucle, évaluée suite au repliement des régions flanquantes en structure secondaire à l'aide de RNAfold (Lorenz *et al.*, 2011) (appelé "hairpin") et la présence ou non du miRNA star qui est le complèmentaire du miRNA lorsque celui-ci est sous sa forme non mature (appelé "star"). Ces deux informations sont des signaux biologiques qui penchent en faveur de la présence d'un vrai miRNA. Un seul attribut numérique a été utilisé, la taille du pré-miRNA (précurseur du miRNA).

3.2. Méthodologie

La méthodologie mise en place a été la suivante :

– les noyaux choisis pour chacun des types de données ont été, le noyau de temps moyen de parcours ("commute time kernel" comme dans (Fouss *et al.*, 2007)) pour le réseau de co-expression, le noyau gaussien pour la variable numérique et le noyau gaussien sur le codage disjonctif des variables qualitatives. La normalisation présentée en section 2.2 a été appliquée aux trois noyaux ;

– 1 000 échantillons bootstrap ont été générés pour produire une classification sur la base de la méthode de bagging décrite dans la section 2.3. La classification finale, basée sur la similarité issue du bagging, était la meilleure de 100 itérations sur la base de l'inertie intra-groupes.

Par ailleurs, de nombreux travaux ont proposé des heuristiques pour le choix du nombre de classes dans l'algorithme des k-moyennes : beaucoup sont basés sur une approche dans laquelle on fait varier le nombre de classes en minimisant un critère donné (Minimum Message Length (Figueiredo, Jain, 2002), Modèle de Mélange Gaussien

(Wallace, Boulton, 1968; Wallace, Freeman, 1987), Minimum Description Length (Hansen, Yu, 2001), BIC/AIC, Gap Statistics (Tibshirani *et al.*, 2001)...). Mais comme souligné dans l'article de référence sur l'algorithme des k-moyennes (Jain, 2010), aucun de ces critères ne permet réellement de manière parfaite de retrouver le nombre de classes le plus adéquat et sur certains exemples simples présentés dans cet article, plusieurs valeurs sont parfois pertinentes et à pondérer suivant les objectifs de l'utilisateur. Aussi, privilégiant l'interprétation à un critère *ad hoc*, nous avons fait varier le nombre de classes de la classification de 2 à 10 et avons regardé l'évolution de deux critères pour étudier la qualité de la classification en fonction du nombre de classes : la modularité (Newman, Girvan, 2004), qui est un critère de la qualité d'une classification de sommets dans un graphe, et l'inertie intra-classes, calculée dans l'espace image induit par le noyau (ce dernier critère est le critère cible minimisé dans la classification par k-moyennes à noyau). La modularité optimale n'est pas un critère monotone en le nombre de classes et donne donc une indication du nombre de classes optimal, du moins pour la classification des sommets basée sur la structure seule du réseau, tandis que l'inertie intra-classes optimale est un critère qui décroit avec le nombre de classes. L'évolution de la modularité et de l'inertie intra-classes induite par le noyau sont données dans la figure 1. À la vue de ces graphiques,

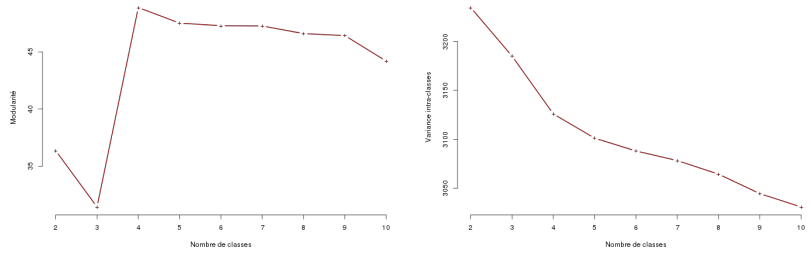

Figure 1. Évolution de la modularité (à gauche) et de l'inertie intra-groupes (à droite)

d'une étude systématique de l'homogénéité des classes selon les variables initiales et pour simplifier la présentation des résultats, une classification a été retenue : celle à 9 classes (qui se trouve à la fin d'un "plateau" pour la modularité et avant une diminution de la pente dans la décroissance de l'inertie intra-classes).

3.3. Résultats et discussion

Les caractéristiques des classes de la classification à 9 classes sont analysés dans la table 1. En particulier, on s'intéresse à la troisième colonne qui correspond au pourcentage de miRNAs annotés (ce qui signifie qu'ils ont été validés comme étant des vrais miRNAs), information qui n'a pas été utilisée comme entrée de l'algorithme de classification. La classification donne des résultats intéressants dans le sens où elle permet bien de regrouper des miRNAs de caractéristiques similaires : fortement co-exprimés (avec des sous-graphes de forte densité, sauf pour les classes 1 et 4), avec

Table 1. Analyse des résultats : classification à 9 classes

Classe	Nb de sommets	% annotés	densité (%)	% hairpin	% star	taille pre-miRNA
1	140	68,57	0,03	52,86	5,71	73,04
2	95	2,11	0,62	28,42	2,11	74,17
3	88	4,55	0,65	32,95	5,68	73,42
4	75	100,00	0,06	94,67	100,00	61,56
5	37	78,38	0,44	40,54	2,70	73,65
6	30	36,67	0,26	50,00	0,00	74,27
7	18	100,00	0,58	100,00	100,00	63,17
8	14	100,00	0,48	57,14	7,14	73,43
9	2	50,00	1,00	50,00	50,00	66,50

des pourcentages de hairpin ou bien de star très forts ou très faibles (en particulier pour les classes 4 et 7) et des tailles de pré-miRNAs qui peuvent être assez différentes (en particulier ici aussi pour les classes 4 et 7). Une étude systématique de l'homogénéité des différentes classes (non détaillée pour des questions de place) nous a convaincus de la pertinence de cette classification : elle est cohérente et homogène (hormis pour une classe ne contenant que deux miRNAs) tout en étant suffisamment riche pour ne pas fournir que des classes annotées à 100% ou 0% comme dans les classifications avec moins de classes. Toutefois, à l'exception de la classe 6 (et également de la classe 9 mais celle-ci est une classe atypique, réduite à deux éléments seulement), le pourcentage d'annotation parmi les miRNAs identifiés est soit fort, soit faible. Un test de Fisher indique que la classe 1 et la classe 5 sont significativement enrichies en miRNAs annotés par rapport à l'ensemble des ARNs disponibles au départ (au risque de 1%). Cela indique que les ARNs non annotés de ces classes ont un bon potentiel d'être de vrais miRNAs et sont donc des candidats à privilégier pour être analysés plus précisément par les biologistes afin de les valider comme nouveaux miRNAs.

4. Conclusion

Nous avons mis en place une méthode de classification non supervisée sur les sommets d'un graphe de co-expression pour lequel des informations additionnelles sur les sommets étaient disponibles. La méthodologie employée a produit des classes dont la composition en miRNAs annotés était souvent tranchée (faible ou forte), ce qui donne des indications, pour les miRNAs potentiels non encore annotés. Les perspectives méthodologiques de ce travail sont multiples avec d'une part l'incorporation de méta-données qui sont plus complexes à extraire mais qui permettraient d'obtenir des classes plus fines et de mieux repérer les miRNAs les plus probables, d'autre part, une calibration adaptative des poids des divers noyaux et enfin, l'étude de critères permettant de sélectionner de manière automatique le nombre de classes le plus pertinent.

References

Aronszajn N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, Vol. 68, No. 3, pp. 337-404.

Breiman L. (2001). Random forests. *Machine Learning*, Vol. 45, No. 1, pp. 5-32. http://www.springerlink.com/content/u0p06167n6173512/fulltext.pdf

Combe D., Largeron C., Egyed-Zsigmond E., Géry M. (2012). Getting clusters from structure data and attribute data. In *Proceedings of international conference on advances in social networks analysis and mining (ieee/acm)*, p. 731-733.

Combe D., Largeron C., Egyed-Zsigmond E., Géry M. (2013). ToTeM: une méthode de détection de communautés adaptées aux réseaux d'information. In *Proceedings of 13e conférence francophone sur l'extraction et la gestion des connaissances (egc)*, p. 305-310.

Cruz J., Bothorel C., Poulet F. (2011). Entropy based community detection in augmented social networks," , 2011 international conference, pp.163-168 doi: 10.1109/cason.2011.6085937. In *Proceedings of computational aspects of social networks (cason)*, p. 163-168.

Dhillon I., Guan Y., Kulis B. (2004). Kernel k-means, spectral clustering and normalized cuts. In *Proceedings of international conference on knowledge discovery and data mining*, p. 551-556. http://www.cs.utexas.edu/~kulis/pubs/spectral_kdd.pdf

Ding J., Zhou S., Guan J. (2010). MiRenSVM: towards better prediction of microRNA precursors using an ensemble SVM classifier with multi-loop features. *BM*, Vol. 11, pp. S11. http://www.biomedcentral.com/1471-2105-11-S11

Ding J., Zhou S., Guan J. (2011). miRFams: an effective automatic miRNA classification method based on n-grams and a multiclass SVM. *BMC Bioinformatics*, Vol. 12, pp. 216.

Dubois S., Fridlyand J. (2003). Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, Vol. 19, No. 9, pp. 1090-1099.

Efron B. (1981). Nonparametric estimates of standard error: the jackknife, the bootstrap and other methods. *Biometrika*, Vol. 68, pp. 589-599.

Ester M., Ge R., Gao B., Hu Z., Ben-Moshe B. (2006). Joint cluster analysis of attribute data and relationship data: the connected k-center problem. In *Siam international conference on data mining*, p. 25-46. ACM Press.

Figueiredo M., Jain A. (2002). Unsupervised learning of finite mixture models. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 3, pp. 381-396.

Fortunato S. (2010). Community detection in graphs. *Physics Reports*, Vol. 486, pp. 75-174. http://arxiv.org/pdf/0906.0612v2

Fouss F., Pirotte A., Renders J., Saerens M. (2007). Random-walk computation of similarities between nodes of a graph, with application to collaborative recommendation. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 19, No. 3, pp. 355-369.

Friedländer M., Chen W., Adamidi C., Maaskola J., Einspanier R., Knespel S. *et al. (2008). Discovering microRNAs from deep sequencing data using miRDeep.* Nature Biotechnology*, Vol. 26, pp. 407-415.*

Friedman J., Hastie T., Tibshirani R. (2008). Sparse inverse covariance estimation with the graphical lasso. Biostatistics*, Vol. 9, No. 3, pp. 432-441.*

Gärtner T. (2008). Kernel for structured data *(Vol. 72). World Scientific.*

Ge R., Ester M., Byron J., Hu Z., Bhattacharya B., Ben-Moshe B. (2008). Joint cluster analysis of attribute data and relationship data: the connected k-center problem, algorithms and applications. ACM Transactions on Knowledge Discovery from Data*, Vol. 2, No. 2, pp. 7.*

Griffiths-Jones S., Grocock R., Dongen S. van, Bateman A., Enright A. (2006). mirbase: microrna sequences, targets and gene nomenclature. Nucleic Acids Research*, Vol. 34, No. suppl 1, pp. D140-D144.*

Griffiths-Jones S., Moxon S., Marshall M., Khanna A., Eddy S., Bateman A. (2005). Rfam: annotating non-coding RNAs in complete genomes. Nucleic Acids Research*, Vol. 33, No. suppl 1, pp. D121-D124.*

Hansen M., Yu B. (2001). Model selection and the principle of minimum description length. Journal of the American Statistical Association*, Vol. 96, No. 454, pp. 746-774.*

Jain A. (2010). Data clustering: 50 years beyond k-means. Pattern Recognition Letters*, Vol. 31, pp. 651-666.*

Jianhua J., Xiao X., Liu B., Jiao L. (2011). Bagging-based spectral clustering ensemble selection. Pattern Recognition Letters*, Vol. 32, pp. 1456-1467.*

Kondor R., Lafferty J. (2002). Diffusion kernels on graphs and other discrete structures. In Proceedings of the 19*th* international conference on machine learning*, p. 315-322.*

Leisch F. (1999, August). Bagged clustering. *(Working Paper 51, SFB "Adaptive Information Systems and Modeling in Economics and Management Science")*

Li H., Nie Z., Lee W., Giles C., Wen J. (2008). Scalable community discovery on textual data with relations. In Proceedings of the 17th acm conference on information and knowledge management*, p. 1203-1212.*

Lorenz R., Bernhart S., Siederdissen C. Höner zu, Tafer H., Flamm C., Stadler P. et al. (2011). ViennaRNA package 2.0. *Algorithms for Molecular Biology*, Vol. 6, pp. 26.

Mapleson D., Moxon S., Dalmay T., Moulton V. (2013). MirPlex: a tool for identifying miRNAs in high-throughput sRNA datasets without a genome. *Journal of Experimental Zoology, Part B: Molecular and Developmental Evolution*, Vol. 320B, pp. 47-56.

Moser F., Ge R., Ester M. (2007). Joint cluster analysis of attribute and relationship data without a-priori specification of the number of clusters. In *Proceedings of the 20th acm sigkdd conference on knowledge discovery and data mining*, p. 510-519. San Jose, CA, USA.

Newman M., Girvan M. (2004). Finding and evaluating community structure in networks. *Physical Review, E*, Vol. 69, pp. 026113. http://www.citebase.org/abstract?id=oai%3AarXiv.org%3Acond-mat%2F0308217

Olteanu M., Villa-Vialaneix N., Cierco-Ayrolles C. (2013). Multiple kernel self-organizing maps. In M. Verleysen (Ed.), *Xxist european symposium on artificial neural networks, computational intelligence and machine learning (esann)*, p. 83-88. Bruges, Belgium, d-side publications.

Schaeffer S. (2007, August). Graph clustering. *Computer Science Review*, Vol. 1, No. 1, pp. 27-64.

Shawe-Taylor J., Cristianini N. (2004). *Kernel methods for pattern analysis*. Cambridge, UK, Cambridge University Press.

Smola A., Kondor R. (2003). Kernels and regularization on graphs. In M. Warmuth, B. Schölkopf (Eds.), *Proceedings of the Conference on Learning Theory (COLT) and Kernel Workshop*, p. 144-158.

Steinhaeuser K., Chawla N. (2008). Community detection in a large real-world social network. In H. Liu, J. Salerno, M. Young (Eds.), *Social computing, behavioral modeling, and prediction*, p. 168-175. Springer US.

Tibshirani R., Walther G., Hastie T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society Series B*, Vol. 63, No. 2, pp. 411-423.

Wallace C., Boulton D. (1968). An information measure for classification. *Computional Journal*, Vol. 11, pp. 185-195.

Wallace C., Freeman P. (1987). Estimation and inference by compact coding (with discussions). *Journal of the Royal Statistical Society Series B*, Vol. 49, pp. 240-251.

Zhou Y., Cheng H., Yu J. (2009). Graph clustering based on structural/attribute similarities. In *Proceedings of the vldb endowment*, Vol. 2, p. 718-729.