Perspectives en matrices aléatoires et grands réseaux

Gilles Wainrib, Romain Couillet

Perspectives en matrices aléatoires et grands réseaux

Gilles Wainrib [1], **Romain Couillet** [2]

1. *Ecole Normale Supérieure*
   Département d'Informatique
   45 rue d'Ulm, 75005 Paris, France.
   gilles.wainrib@ens.fr

2. *CentraleSupélec*
   91192 Gif sur Yvette, France
   romain.couillet@centralesupelec.fr

RÉSUMÉ. *Dans cet article, de nouvelles perspectives de recherche en matrices aléatoires appliquées à la théorie des graphes sont introduites. Nous nous attachons en particulier à l'analyse spectrale des matrices d'adjacence et laplaciennes de graphes de grandes dimensions pour la détection de communautés dans les réseaux, des matrices aléatoires à noyaux pour la classification non supervisée en big data, ainsi qu'à des applications en réseaux de neurones.*

ABSTRACT. *In this article, several research perspectives in random matrix theory applied to graph theory at large are discussed. Specific focus will be made on the spectrum analysis of the adjacency or Laplacian matrices of large dimensional graphs for community detection in networks, of kernel random matrices for clustering in large datasets, along with applications to neural networks.*

MOTS-CLÉS : *matrices aléatoires, classification spectrale, réseaux de neurones, graphes, graphes aléatoires.*

KEYWORDS: *random matrix theory, spectral clustering, neural networks, graphs, random graphs.*

DOI:10.3166/TS.33.351-376

Extended Abstract

This article introduces basic notions of random matrix theory applied to graph processing, where by "graph processing" we understand here multiple signal processing and machine learning tools ranging from community detection on graphs to kernel methods in machine learning to performance analysis of neural networks with random connections.

The article first provides an introduction to the basic notions of the spectrum of a graph, before applying these notions to the detection of communities within the graph. It is shown there that the leading eigenvectors of the adjacency (or Laplacian) matrix of the graph contains the community structure. The specific and simple case of stochastic block models is covered in full details.

We then move to a different application but with similar mathematical connections in handling the performance evaluation of kernel methods for clustering and classification of large dimensional data $x_1, \ldots, x_n \in \mathbb{R}^p$ in machine learning. There an adjacency matrix can also be defined as the affinity matrix K between every pair of data vector, through the kernel relation $K_{ij} = f(\|x_i - x_j\|^2)$ or $K_{ij} = f(x_i^\mathsf{T} x_j)$ for some non-linear function f. Despite the apparently blocking non linearity of f when analyzing the performance of methods based on K, it is shown that, as $n, p \to \infty$ and x_i are modelled as a mixture of Gaussian vectors with close enough means and covariance (so to ensure difficult class separability), K can be asymptotically approximated by a simpler matrix whose analysis is amenable to random matrix theory. Notably, its structure reveals the possibility to use the dominant eigenvectors of K to perform clustering.

As a last application, we discuss the relations between random matrix theory and neural networks. While the most popular neural networks are essentially data driven (all neuronal connections are learnt from the data themselves), an efficient subclass of neural networks maintains random connection layers and only learns the output layers. Within this class falls for instance the echo-state networks, shown to have powerful performances in time series data analysis. We show that investigating such randomly connected networks is made possible thanks to large dimensional statistical tools, and notably random matrix theory.

As a conclusion, the article promises important aftermaths in the future usage of random matrix theory in the understanding and improvement of multiple methods involving large dimensional datasets and non-linear behaviors, despite the a priori incompatibility between non linearity and classical random matrix techniques. These recent investigations therefore open up a new realm of likely active research in the better understanding of machine learning and signal processing methods in the big data paradigm.

1. Introduction

L'outil des matrices aléatoires a connu depuis le début des années 2000 un fort attrait dans la communauté des communications sans fils (Tulino, Verdú, 2004 ; Couillet, Debbah, 2011). La connexion entre ce domaine et les grandes matrices aléatoires vient du fait que les modèles physiques impliqués prennent des formes matricielles aux entrées aléatoires et surtout que les informations pertinentes aux deux domaines se trouvent bien souvent au cœur des valeurs propres et vecteurs propres de ces modèles matriciels.

En termes de traitement du signal au sens large, les informations pertinentes résident souvent dans des variables non observables du modèle qu'il s'agit donc d'inférer. Ces informations portent par exemple sur la structure statistique du modèle (moyenne, covariance) ou sur une fonctionnelle de cette structure. Comme il s'avère plus compliqué statistiquement d'inférer les paramètres masqués d'un modèle que d'en analyser les conséquences pour les données observées (comme c'est le cas le plus souvent pour les problèmes de communications sans fils), il faudra attendre le milieu des années 2000 pour voir apparaître les premières applications notables (en particulier en traitement d'antennes) connectant inférence statistique et matrices aléatoires pour des jeux de données de grandes dimensions. En particulier, outre des résultats amonts de nature purement statistique proposés dans (Girko, 1987), les résultats en traitement du signal ont tout d'abord concerné la détection de sources (Johnstone, 2001 ; Bianchi *et al*., 2011), puis l'estimation d'angles d'arrivée pour le traitement d'antennes (Mestre, 2008 ; Vallet *et al*., 2012).

La plupart des travaux appliqués mentionnés précédemment reposent néanmoins sur un fort dénominateur commun : les modèles matriciels considérés sont systématiquement des modèles de matrices de covariance empirique (ou diffèrent très peu de ces modèles). Seuls de très rares travaux, souvent demeurés de nature théorique, ont porté jusque là sur des applications de modèles matriciels plus spécifiques. Notons en particulier un récent regain d'intérêt pour l'estimation de séries temporelles multi-variées pour lesquelles des versions "toeplitzifiées" de matrices de covariance empiriques sont au cœur des méthodes d'estimation (Bickel, Levina, 2008 ; Vinogradova *et al*., 2014). D'autres travaux ont également récemment considéré des matrices aléatoires à noyaux (El Karoui, 2010) ou des matrices issues de l'estimation de paramètres de dispersion dans le cadre des statistiques robustes (Couillet *et al*., 2015 ; Couillet, McKay, 2014 ; T. Zhang *et al*., 2014).

La récente impulsion du domaine du big data a cependant modifié de manière importante les modèles matriciels d'intérêt premier. Les méthodes d'apprentissage automatisé sont en effet rarement liées à des modèles de covariance empirique du fait de la non-linéarité intrinsèque des données réelles rencontrées. Néanmoins, l'apprentissage automatisé relève souvent de méthodes d'estimation centrées sur l'analyse spectrale de grandes matrices. Nous nous intéressons particulièrement ici aux matrices d'adjacences et laplaciennes de graphes de données, aux matrices à noyaux, ainsi qu'à des modèles matriciels spécifiques rencontrés dans l'étude de grands réseaux de neurones.

La recherche dans ces domaines d'activité n'en est aujourd'hui qu'à ces balbutiements. L'objectif de cet article est, via l'introduction de quelques exemples concrets mais de contextes très distincts, de susciter l'intérêt du lecteur sur ce que les auteurs estiment être les centres d'intérêt majeurs du domaine des matrices aléatoires appliquées de demain. Spécifiquement, il est question ici de comprendre comment l'outil des matrices aléatoires permet d'analyser les structures de communautés dans des grands réseaux modélisés aléatoirement, de comprendre le fonctionnement des méthodes, dites spectrales, de classification non supervisées exploitant des noyaux, puis de comprendre comment mieux appréhender le fonctionnement de certains modèles de réseaux de neurones.

L'article se divise en trois grandes parties. Dans une première partie, section 2, nous faisons un rappel sur la notion de spectres d'un graphe et rappelons en particulier le lien entre les valeurs propres (et leurs vecteurs propres associés) de la laplacienne d'un graphe et la notion de connectivité d'un graphe. Ces notions de base se révèlent essentielles pour comprendre les applications en détection de communautés et classification non supervisées en section 3. La section 5 sera alors consacrée aux avancées récentes et sujets d'avenir en réseaux de neurones et matrices aléatoires. Des remarques de conclusion seront évoquées en section 6.

2. Le Spectre d'un graphe

2.1. Définitions élémentaires

Commençons tout d'abord par quelques rappels élémentaires sur la définition d'un graphe, sa matrice d'adjacence, ainsi que la matrice laplacienne associée. Le spectre du graphe qui est le sujet de cet article sera alternativement l'ensemble des valeurs propres d'une de ces matrices.

Un graphe $G = (V, E)$ est un ensemble de nœuds, indexés par $V = \{1, \ldots, n\}$ ainsi qu'un ensemble d'arêtes indexées par $E \subset V \times V$, qui est l'ensemble des couples (i, j) tels que le nœud i est connecté au nœud j. On dénotera alternativement $(i, j) \in E$ ou $i \sim j$ et réciproquement $(i, j) \notin E$ ou $i \not\sim j$. Nous considérons exclusivement ici des graphes non dirigés, en ce sens que $i \sim j$ implique $j \sim i$.

La matrice d'adjacence $A \in \{0, 1\}^{n \times n}$ est la matrice d'entrée (i, j) égale à

$$A_{ij} \triangleq \boldsymbol{\delta}_{i \sim j}.$$

Le graphe G n'étant pas dirigé, il est clair que A est une matrice symétrique et a donc toutes ses valeurs propres réelles.

Pour chaque nœud i, on appelle degré d_i le nombre d'arêtes issues de ce nœud, à savoir $|\{j, i \sim j\}|$. On définit alors $D \in \mathbb{N}^{n\times n}$, dite matrice des degrés de G, la matrice diagonale d'entrée

$$D_{ii} = d_i = \sum_{j\sim i} 1 = \sum_{i=1}^{n} A_{ij}.$$

Une première définition de la matrice laplacienne L de G est alors donnée par

$$L = D - A.$$

Cette matrice contient en (i, i) le degré de i, en (i, j), $i \neq j$, la valeur -1 si $i \sim j$ et zéro sinon.

De cette remarque, nous voyons immédiatement que zéro est une valeur propre de L, associé au vecteur propre $\mathbf{1} = (1, \ldots, 1)^T$. Par ailleurs, en évaluant $v^T Lv$ pour un vecteur $v \in \mathbb{R}^n$, on se rend compte que $v^T Lv = \sum_{i\sim j}(v_i - v_j)^2 \geq 0$ et donc L est semi-définie positive. Il vient donc que les valeurs propres $\lambda_1(L) \leq \ldots \lambda_n(L)$ de L s'ordonnent suivant :

$$0 = \lambda_1(L) \leq \ldots \lambda_n(L).$$

Pour certaines raisons, il pourra être plus souhaitable de considérer la laplacienne normalisée L_{norm} définie par

$$L_{\text{norm}} = I_n - D^{-\frac{1}{2}} AD^{-\frac{1}{2}}.$$

Suite aux éléments évoqués précédemment, il vient aisément que zéro est la valeur propre de L_{norm} (de vecteur propre $D^{\frac{1}{2}}\mathbf{1}$) et que les autres valeurs propres sont positives ou nulles.

2.2. Connectivité et spectre

Une information globale importante en analyse de graphes concerne la notion de connectivité. Un premier résultat établit que, si G se représente (de manière minimale) comme l'union de k sous-graphes disjoints (c'est-à-dire sans arête les connectant) $G_1, \ldots, G_k$, alors zéro est une valeur propre de multiplicité exactement k pour L ou L_{norm}. Ceci suit du fait que le vecteur $\mathbf{1}_{G_i} \in \mathbb{R}^n$, composé de uns aux indices de G_i et de zéros ailleurs, est un vecteur propre pour L, et ce pour chaque $i = 1, \ldots, k$. Supposons que $k = 1$, et donc que G est entièrement connecté. Il vient alors, par un raisonnement de perturbation du cas disjoint, que la deuxième plus petite valeur propre, $\lambda_2(L)$, porte une information précieuse sur le degré de connectivité du graphe dans un certain sens. Cette valeur propre est souvent appelée *connectivité algébrique* du graphe. Il peut être montré en particulier que, si on dénote pour $S \subset V$

$$\phi(S) = \frac{\sum_{i\in V} d_i}{\left(\sum_{i\in S} d_i\right)\left(\sum_{i\notin S} d_i\right)} \frac{1}{2} \sum_{i\in S, j\notin S} A_{ij}$$

alors

$$\frac{\left(\min_{S\subset G}\phi(S)\right)^2}{2\max_{i\in V} d_i} \leq \lambda_2(L) \leq \min_{S\subset G}\phi(S).$$

Ce résultat, appelé inégalité de Cheeger, montre combien $\lambda_2(L)$ est intimement lié à des mesures naturelles de connexions du graphe.

La plupart des notions et résultats discutés dans cet article sont fortement liés à cette importante notion de connectivité d'un graphe.

3. Graphes aléatoires

L'une des questions essentielles qui se pose en apprentissage automatisé sur des graphes est celle de la détection de communauté ou la détection d'activité spécifique dans le graphe. En termes de matrices d'adjacence, la question est de découvrir si un ensemble $S \subset V$ de nœuds de G a une propension à se connecter de manière plus forte qu'avec $V \setminus S$. Il s'agit alors à la fois de pouvoir détecter l'existence de telles structures de connexions fortes ainsi que d'identifier l'ensemble des membres de la structure, et donc de connaitre S. Nous allons voir que cette information peut être lue dans les vecteurs propres associés aux valeurs propres extrêmes de la matrice A.

Pour cela, considérons le modèle le plus élémentaire dit "modèle bloc-statistique". Ce modèle consiste en une division de G en deux sous-graphes $G_1 = (V_1, E_1)$ et $G_2 = (V_2, E_2)$ (en particulier $V = V_1 \cup V_2$ mais $V_1 \cap V_2 = \emptyset$) ayant les propriétés suivantes:

– pour chaque paire $(i,j) \in V_1 \times V_1$ ou $(i,j) \in V_2 \times V_2$, $(i,j) \in E$ avec probabilité $p_{\text{in}} = c_{\text{in}}/n$

– pour chaque paire $(i,j) \in V_1 \times V_2$, $(i,j) \in E$ avec probabilité $c_{\text{out}}/n = p_{\text{out}} < p_{\text{in}}$.

Les valeurs c_{in} et c_{out} sont supposées constantes et en particulier indépendantes de n. Comme l'indexation des nœuds est sans importance dans ce qui suit, nous supposerons que les nœuds $1, \ldots, n_1$ ($n_1 = |V_1|$) font partie du graphe G_1 (et donc de la première des deux communautés) tandis que les nœuds $n_1 + 1, \ldots, n$ ($n - n_1 = n_2 = |V_2|$) forment le graphe G_2 de la seconde communauté. Nous faisons également l'hypothèse simplificatrice par la suite que $n_1 = n_2 = n/2$.

Ainsi, la matrice d'adjacence A est désormais une matrice aléatoire dont les entrées non nulles sont tirées avec des probabilités différentes, qui peut s'écrire sous la forme

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

où $A_{11} \in \{0,1\}^{n_1 \times n_1}$ et $A_{22} \in \{0,1\}^{n_2 \times n_2}$ sont des matrices à entrées i.i.d. de moyenne p_{in} et de variance $p_{\text{in}}(1 - p_{\text{in}})$, tandis que $A_{12} \in \{0,1\}^{n_1 \times n_2}$ et $A_{21} \in \{0,1\}^{n_2 \times n_1}$ sont des matrices rectangulaires à entrées i.i.d. de moyenne p_{out} et de variance $p_{\text{out}}(1 - p_{\text{out}})$.

On peut dès lors écrire A sous la forme $A = \mathrm{E}[A] + X$, où

$$\mathrm{E}[A] = \frac{1}{2}\left(p_{\mathrm{in}} + p_{\mathrm{out}}\right)\mathbf{1}\mathbf{1}^T + \frac{1}{2}\left(p_{\mathrm{in}} - p_{\mathrm{out}}\right)(\pm\mathbf{1})(\pm\mathbf{1})^T$$

avec $\pm\mathbf{1} = (\underbrace{1,\ldots,1}_{n_1},\underbrace{-1,\ldots,-1}_{n_2})^T$, et X est une matrice aléatoire d'entrées de moyenne nulle mais de variances distinctes.

Nous allons décrire à présent l'analyse approximative (et formellement inexacte) mais cependant instructive de ce modèle effectuée par (Nadakuditi, Newman, 2012). De manière fortement peu rigoureuse, si on suppose que les quantités c_{in} et c_{out} sont suffisamment larges, il peut alors être admis en utilisant des méthodes classiques de matrices aléatoires (combinatoires ou basées sur la transformée de Stieltjes), que la loi $\mu_n \triangleq \frac{1}{n}\sum_{i=1}^n \boldsymbol{\delta}_{\lambda_i(X)}$, c'est-à-dire la distribution empirique des valeurs propres de X, converge presque surement vers la loi du demi-cercle μ de densité f paramétrée comme suit (Nadakuditi, Newman, 2012)

$$f(x) = \frac{\sqrt{x^2 - 2(c_{\mathrm{in}} + c_{\mathrm{out}})}}{\pi\frac{c_{\mathrm{in}}+c_{\mathrm{out}}}{2}} 1_{\left\{|x| \leq \sqrt{2(c_{\mathrm{in}}+c_{\mathrm{out}})}\right\}}.$$

En particulier, le bord droit du support se trouve en $x = \sqrt{2(c_{\mathrm{in}} + c_{\mathrm{out}})}$, une caractérisation qui va s'avérer bientôt importante. Ce résultat est en fait assez attendu en ce sens que chaque entrée (i, j) de A pris aléatoirement a une probabilité $\frac{p_{\mathrm{in}}+p_{\mathrm{out}}}{2}$ d'être égale à 1. Cependant, la structure de dépendance liée au graphe ne rend pas ce résultat complètement immédiat.

Nous prétendons que ce résultat n'est pas rigoureux en ce sens que la loi limite n'est un demi-cercle que si c_{in} et c_{out} grandissent, même très lentement, avec n. En réalité, dans le modèle qui nous concerne avec c_{in} et c_{out} finis, la loi limite est une version "étalée" du demi-cercle. L'approximation est néanmoins souvent suffisante pour les graphes qui nous intéressent ici, comme en témoignent les simulations effectuées par les auteurs (pour rendre l'approche de (Nadakuditi, Newman, 2012) correcte, il suffirait en fait d'introduire un facteur d'échelle sur c_{in} et c_{out} qui assurent que tous deux tendent vers l'infini avec n à un certain rythme mais que leur différence reste bornée).

Pour caractériser entièrement le spectre de A, il s'agit alors de montrer que toutes les valeurs propres de X ne peuvent asymptotiquement pas quitter le support de la distribution limite, à savoir un résultat équivalent à (Bai, Silverstein, 1998). Ce résultat étant acquis, A consiste donc en une perturbation de la matrice X par la matrice de rang deux $\frac{1}{2}\left(p_{\mathrm{in}} + p_{\mathrm{out}}\right)\mathbf{1}\mathbf{1}^T + \frac{1}{2}\left(p_{\mathrm{in}} - p_{\mathrm{out}}\right)(\pm\mathbf{1})(\pm\mathbf{1})^T$. Ce résultat fait appel ici à la théorie dite des modèles "spiked" de la théorie des matrices aléatoires ; voir par

exemple (Baik, Silverstein, 2006 ; Benaych-Georges, Nadakuditi, 2012). A l'aide de cet outil, il est alors possible de montrer que, dans la limite, la matrice A vérifiera

$$\lambda_n(A) \xrightarrow{\text{p.s.}} \max\left\{\sqrt{2(c_{\text{in}}+c_{\text{out}})}, 1+\frac{1}{2}(c_{\text{in}}+c_{\text{out}})\right\}$$

$$\lambda_{n-1}(A) \xrightarrow{\text{p.s.}} \max\left\{\sqrt{2(c_{\text{in}}+c_{\text{out}})}, \frac{1}{2}(c_{\text{in}}-c_{\text{out}})+\frac{c_{\text{in}}+c_{\text{out}}}{c_{\text{in}}-c_{\text{out}}}\right\}.$$

Le vecteur propre associé à λ_n ne porte que peu d'information, étant donné qu'il provient du vecteur $\mathbf{1}$ évoqué plus haut. Par contre, le vecteur propre associé à λ_{n-1} est lui issu de la perturbation de X par la matrice $(\pm\mathbf{1})(\pm\mathbf{1})^T$ qui porte en sa structure l'information directe de classification entre les nœuds des deux communautés. Par des arguments typiques aux modèles spiked en matrices aléatoires, il vient assez naturellement que, si $\lambda_{n-1}(A) \simeq \sqrt{2(c_{\text{in}}+c_{\text{out}})}$, alors le vecteur propre en question devient asymptotiquement complètement orthogonal à $(\pm\mathbf{1})$ et toute l'information de structure est perdue. A contrario, plus $\lambda_{n-1}(A)$ s'éloigne de la valeur seuil $\sqrt{2(c_{\text{in}}+c_{\text{out}})}$, plus le vecteur propre qu'elle porte s'aligne à $(\pm\mathbf{1})$ et l'information de structure réapparait. Un valeur charnière apparaît donc dans la possibilité asymptotique de répartir les nœuds du graphes dans leurs communautés respectives. Cette valeur est liée à la différence $c_{\text{in}} - c_{\text{out}}$ qui doit être telle que

$$\frac{1}{2}(c_{\text{in}}-c_{\text{out}})+\frac{c_{\text{in}}+c_{\text{out}}}{c_{\text{in}}-c_{\text{out}}} > \sqrt{2(c_{\text{in}}+c_{\text{out}})}$$

ou autrement dit telle que

$$c_{\text{in}} - c_{\text{out}} > \sqrt{2(c_{\text{in}}+c_{\text{out}})}.$$

Dans ces conditions, il est ainsi possible d'inférer la répartition des communautés du graphes à partir du vecteur propre u_{n-1} associé à la valeur propre $\lambda_{n-1}(A)$. Il est intéressant de s'interroger alors sur la quantité de nœuds qui seront bien répartis par l'algorithme de classification qui consiste à établir un seuil en la valeur moyenne des entrées de u_{n-1} et à répartir les nœuds en deux classes selon que la valeur des entrées de u_{n-1} est supérieure ou inférieure au seuil en question. Pour se faire, il s'agit alors de comprendre l'interaction entre u_{n-1} et le vecteur $(\pm\mathbf{1})$. L'étude des modèles spiked se révèle une fois de plus fondamentale ici, en ce sens qu'il est possible de prévoir l'angle $u_{n-1}^T(\pm\mathbf{1})$ créé entre les deux angles en fonction de $c_{\text{in}} - c_{\text{out}}$. En particulier, lorsque $c_{\text{in}} - c_{\text{out}} \simeq \sqrt{2(c_{\text{in}}+c_{\text{out}})}$, $u_{n-1}^T(\pm\mathbf{1}) \simeq 0$, tandis que $u_{n-1}^T(\pm\mathbf{1}) \simeq 1$ lorsque $c_{\text{in}} - c_{\text{out}}$ est suffisamment large. En utilisant la symétrie du problème, il vient alors que la proportion de nœuds mal classés converge vers

$$\Phi\left(\sqrt{\frac{\alpha}{1-\alpha}}\right)$$

où

$$\alpha \triangleq \frac{(c_{\text{in}}-c_{\text{out}})^2 - 2(c_{\text{in}}+c_{\text{out}})}{(c_{\text{in}}-c_{\text{out}})^2}$$

et $\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp(-x^2/2)dx$ est la fonction de distribution d'une variable gaussienne standard. Les courbes de répartitions correctes des nœuds obtenues par cette formule sont représentées en Figure 1.

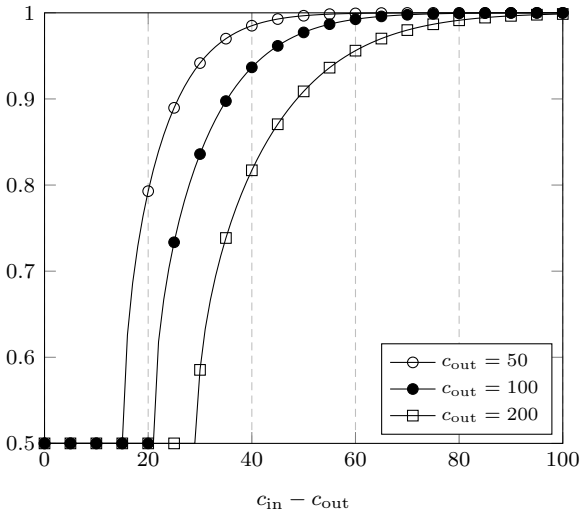

Figure 1. Proportion théorique de nœuds correctement classés en fonctions de $c_{\text{in}} - c_{\text{out}}$ *pour différentes valeurs de* c_{out}

Un certain nombre de travaux ont fait suite à cette analyse simple. En particulier, pour rendre compte de la réalité de réseaux pratiques, l'hypothèse selon laquelle c_{in} et c_{out} sont constants pour chaque nœud du réseau est assez peu réaliste. Il est en effet souvent plus approprié de supposer que les nœuds du réseau ont une probabilité intrinsèque de se lier à tout autre nœud du réseau qui lui est propre. Ainsi, la généralisation de l'étude précédente au cas de graphes ayant une distribution statistique non triviale de degrés a donné lieu à plusieurs résultats plus généraux (Nadakuditi, Newman, 2013 ; X. Zhang *et al.*, 2014). Il apparait en particulier dans ces travaux que la présence de valeurs propres isolées, au-delà de leur capacité à séparer les communautés à l'intérieur du graphe, permet également parfois de détecter la présence de nœuds particulièrement connectés (appelés hubs) dans le réseau. Des propriétés détaillées de ces hubs peuvent se lire dans les vecteurs propres portés par ces valeurs propres (Nadakuditi, Newman, 2013).

D'autres travaux intéressants proposent de généraliser l'étude précédente à des matrices d'adjacence réellement parcimonieuses (à savoir, lorsque c_{in} et c_{out} sont très petits, auquel cas l'erreur commise dans (Nadakuditi, Newman, 2012) est importante et compromet fortement l'analyse), en utilisant des méthodes spectrales sur des graphes plus élaborés. En effet, du point de vue des théoriciens des graphes, il apparait que dans le cas très parcimonieux la matrice d'adjacence (et donc toute méthode liée à cette dernière) voit ces plus grandes valeurs propres dominées par l'effet des rares

nœuds du graphe ayant une forte connectivité, de sorte que les méthodes précédentes ne fonctionnent plus pour n de taille raisonnable. Si l'effet de ces nœuds peut être atténué, la structure globale du graphe pourra réapparaître plus nettement à travers les valeurs propres extrêmes. C'est ce qui a récemment motivé Krzakala et al. (Krzakala *et al.*, 2013) à considérer le spectre de la matrice A_{nbt} dite "non-backtracking" de taille $|E| \times |E|$, indexée sur les arêtes de G et définie, pour tout $e, e' \in E$, avec $e = (a, b)$, $e' = (a', b')$, par

$$(A_{nbt})_{ee'} = \begin{cases} 1 & , \text{ si } b = a' \text{ et } a \neq b' \\ 0 & , \text{ sinon.} \end{cases}$$

La matrice est dite "non-backtracking" du fait de la condition $a \neq b'$ qui interdit de connecter un lien avec son lien réciproque. L'intérêt de cette matrice est d'éviter d'accumuler trop de poids sur les nœuds fortement connectés en établissant des connexions de deuxième ordre uniquement. Grâce à cette matrice, dont l'étude spectrale révèle que certaines valeurs propres isolées portent l'information de communauté dans leurs vecteurs propres, il est possible de recouvrer le seuil $\sqrt{2(c_{\mathrm{int}} + c_{\mathrm{out}})}$ de détection dans le cas très parcimonieux également. Cependant, l'étude spectrale est ici bien plus délicate que dans le cas de A, notamment à cause du fait que A_{nbt} n'est plus hermitienne et a donc un spectre complexe tandis que le spectre de A est réel.

D'autres travaux dans la même veine peuvent également être mentionnés, ceux-ci basés sur des approches relevant de la théorie des graphes plus que de l'analyse spectrale (Guédon, Vershynin, 2014 ; Zhao *et al.*, 2012).

4. Méthodes à noyaux et classification

Une autre direction de recherche connectant matrices aléatoires et structures graphiques consiste en l'analyse de méthodes de classifications non supervisées pour le big data. En effet, alors que de nombreuses méthodes existent pour la classification non supervisées de données nombreuses et de petites dimensions (images en 2D, 3D, vecteurs courts de données, etc.), il est bien plus délicat de gérer la classification de grands vecteurs de données, pour des raisons que nous allons éclaircir plus bas. A l'aide de l'outil des grandes matrices aléatoires, il devient dès lors possible de mieux rendre compte des limitations et espoirs portés par les méthodes existantes et leur spécialisation aux problèmes de grande dimension.

Nous nous attacherons ici précisément à la méthode dite de la classification spectrale (spectral clustering en anglais). Dotons nous de vecteurs $x_1, \ldots, x_n \in \mathbb{R}^p$ (disons) que nous souhaitons classés en groupes de données co-localisées ou étant clairement associées. Une première idée permettant de classifier des données vectorielles consiste à trouver un ou plusieurs hyperplans de l'espace $\mathbb{R}^p$ permettant de séparer les données en groupes disjoints. Cette approche, qui pourrait donner lieu en particulier à des méthodes de type analyse en composantes principales, n'est malheureusement valable que lorsque les données elles-mêmes se prêtent à une telle différentiation linéaire à l'aide d'un hyperplan. Malheureusement, l'idée même de la classification non

supervisée est d'être capable de classer des données moins évidentes que des données linéairement séparables. On peut par exemple souhaiter isoler des données distribuées sur deux sphères concentriques mais de rayons distincts, auquel cas la méthode proposée précédemment ne peut fonctionner.

Ainsi sont apparues les méthodes dites à noyaux, qui consistent en un sens à déplacer les données de $\mathbb{R}^p$ vers un espace vectoriel de plus grande dimension (parfois infinie) dans lequel on espère qu'un hyperplan pourra séparer les données. Il s'agit donc de transformer x_i en $f(x_i)$ pour une certaine fonction $f : \mathbb{R}^p \to \mathbb{R}^D$, $D > p$, et de construire la matrice de covariance empirique $n^{-1} \sum_{i=1}^n f(x_i) f(x_i)^T$. Cependant, comme l'analyse de données dans des espaces de grandes dimensions est souvent prohibitif en termes calculatoires, on s'attachera plutôt ici à considérer des noyaux f tels que la produit scalaire $f(x_i)^T f(x_j)$ peut s'écrire sous une forme $F(x_i, x_j)$ simple; par exemple $F(x_i, x_j) = G(\|x_i - x_j\|^2)$ pour une fonction G de $\mathbb{R}$ dans $\mathbb{R}$. Il peut être alors montré que l'analyse des valeurs propres et vecteurs propres de la matrice $\mathcal{F} = \{F(x_i, x_j)\}_{1 \leq i,j \leq n}$ donne des informations sur le regroupement des données en classes distinctes, ces classes pouvant être lues directement, comme dans la section précédente, via les entrées des vecteurs propres en question. On parle ici de méthodes de classification spectrale par méthodes à noyaux.

Nous allons nous intéresser à un exemple spécifique de telles méthodes, qui va par ailleurs nous donner une autre façon intuitive de comprendre l'approche par noyaux. Considérons à nouveau les données $x_1, \ldots, x_n$ et supposons qu'elles se séparent en deux classes distinctes. Supposons également que l'expérimentateur soit suffisamment conscient des données qu'il traite pour pouvoir identifier finement une fonction $F(x_i, x_j) = H(\|x_i - x_j\|^2)$, qui est donc fonction de la distance entre les x_i, de manière à ce que $F(x_i, x_j) = 0$ si x_i et x_j sont de classes distinctes, et $F(x_i, x_j) > 0$ sinon. Dans ce cas idéal, remarquons alors que la matrice $\mathcal{L} = \mathcal{D} - \mathcal{F}$, avec $\mathcal{D} = \mathrm{diag}(\{\sum_{j=1}^n F(x_i, x_j)\}_{i=1}^n)$, est la laplacienne d'un certain graphe G (aux arêtes pondérées) des x_i et qu'alors, d'après une remarque faite préalablement, le nombre de valeurs propres nulles de $\mathcal{L}$ coïncide avec le nombre k de sous-graphes $G_1, \ldots, G_k$ connexes de G. Ce nombre k correspond ici précisément au nombre de classes qui nous intéressent. Par ailleurs, nous savons également que l'espace propre engendré par ces valeurs propres est engendré par les vecteurs $\mathbf{1}_{G_i}$, et donc la classification des x_i se lie directement dans l'espace propre en question.

Evidemment, en pratique, $F(x_i, x_j)$ sera rarement nul, mais relativement faible, pour x_i et x_j de classes distinctes. Il vient alors, par un raisonnement de perturbation matricielle, que les k plus petites valeurs propres de $\mathcal{L}$ seront distinctes et suffisamment faibles en comparaison des $n-k$ autres. La classification non supervisée consiste alors à extraire l'information des classes dans les k vecteurs propres associés aux k plus petites valeurs propres. En général, l'extraction de cette information est réalisée à l'aide de l'algorithme dit du k-means. Voir (Ng *et al.*, 2001) pour plus d'informations.

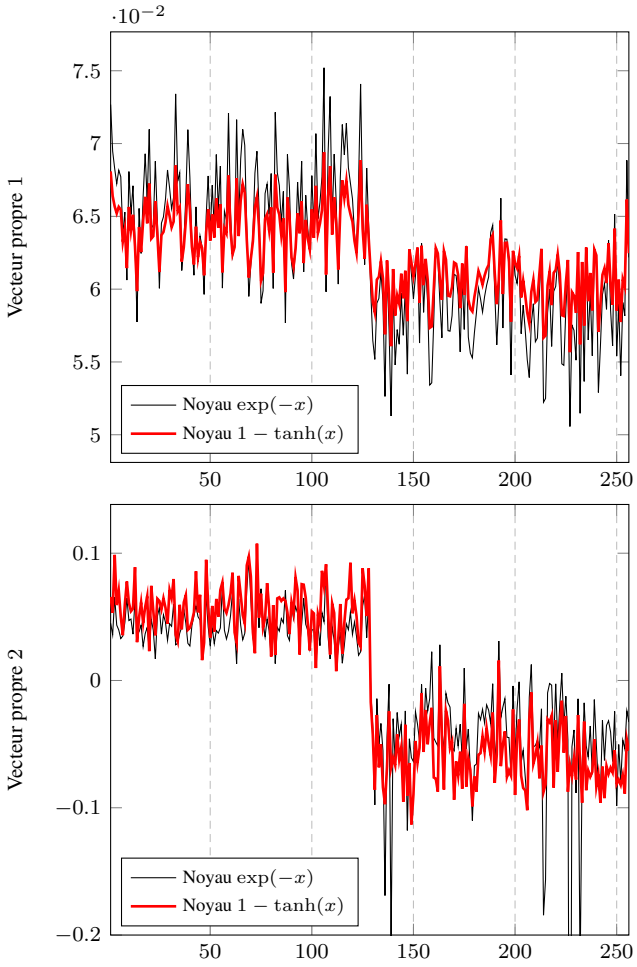

Figure 2. Vecteurs propres dominants (associés aux deux plus grandes valeurs propres) de $\mathcal{D}^{-\frac{1}{2}}\mathcal{F}\mathcal{D}^{-\frac{1}{2}}$, pour les noyaux $f_1(x) = \exp(-x)$ et $f_2(x) = 1 - \tanh(x)$

Lorsque p est de grande taille, le raisonnement précédent n'est cependant plus valable. Supposons en effet que les x_i soient des vecteurs gaussiens de covariance I_p mais de moyennes distinctes, μ_1 ou μ_2. Alors, $\|x_i - x_j\|^2/p \simeq \|\mu_1 - \mu_2\|^2/p + 2$ pour x_i et x_j de classes différentes ou $\|x_i - x_j\|^2/p \simeq 2$ pour x_i et x_j de même classe. Si $\|\mu_i\| = O(1)$, alors ma matrice $\mathcal{F}$ a des entrées convergeant toutes vers 2, en dépit du fait que x_i et x_j soient de même classe ou de classes différentes. La considération "distance" entre deux données due à leur appartenance à une même classe ou non ne tient donc plus ici. Cependant, les simulations numériques montrent que les algorithmes de classification spectrale sont toujours efficaces dans ce cadre. Il est

ainsi fondamental de comprendre quels nouveaux phénomènes régissent cet important régime.

Pour cela, il faut tout d'abord comprendre la structure spectrale de la matrice $\mathcal{F}$ pour des données x_i de grandes dimensions. Pour ce faire, un nouveau type d'analyse de modèle matriciel, celui de $\mathcal{F}$, est requis. Très peu de travaux concernent ce cas d'étude. Cependant, dans un article récent (El Karoui, 2010), il a été établi un premier résultat intéressant concernant le cas d'une seule classe où les x_i sont de moyenne nulle et de covariance C donnée. Dans ce cas, El Karoui montre que, pour des noyaux f suffisamment lisses (au moins trois fois dérivables), la matrice $\mathcal{F}$ devient asymptotiquement équivalente à une matrice aléatoire de type "matrice de covariance empirique perturbée". Précisément, il est établi que

$$\|\mathcal{F} - \mathcal{M}\| \to 0$$

en probabilité, où, avec $X = [x_1, \ldots, x_n]$ et $\psi = \{\psi_i\}_{i=1}^n$, $\psi_i = \frac{1}{p}\|x_i\|^2 - \frac{1}{p}\operatorname{tr} C$,

$$\begin{aligned}\mathcal{M} = {} & f\left(\frac{2}{p}\operatorname{tr} C\right)\mathbf{1}\mathbf{1}^T + f'\left(\frac{2}{p}\operatorname{tr} C\right)\left(\mathbf{1}\psi^T + \psi\mathbf{1}^T - 2\frac{1}{p}X^TX\right) \\ & + f''\left(\frac{2}{p}\operatorname{tr} C\right)\left(\mathbf{1}(\psi\circ\psi)^T + (\psi\circ\psi)\mathbf{1}^T + 2\psi\psi^T + 4\frac{\operatorname{tr} C^2}{p^2}\mathbf{1}\mathbf{1}^T\right) \\ & + \left(f(0) + \frac{2}{p}\operatorname{tr} C\cdot f'\left(\frac{2}{p}\operatorname{tr} C\right) - f\left(\frac{2}{p}\operatorname{tr} C\right)\right) I_n.\end{aligned}$$

Bien que compliquée d'apparence au premier abord, il est important de noter que $\mathcal{M}$, sous un œil "matrices aléatoires", est essentiellement égale à $-2f'(\frac{2}{p}\operatorname{tr} C)\frac{1}{p}X^TX$, à savoir une matrice de covariance empirique très classique à un scalaire près, à laquelle s'additionnent un nombre fini de matrices soient proportionnelles à l'identité, soit de rang un. Cela suggère la possibilité d'une étude poussée de par les outils de la théorie des matrices aléatoires de la matrice $\mathcal{M}$, et ainsi de la matrice $\mathcal{F}$.

Dans le cas de données de classes multiples, il s'agit alors de généraliser le résultat précédent à différentes moyennes et covariances, et ensuite de comprendre le comportement spectral de $\mathcal{F}$ via celui de $\mathcal{M}$. Des travaux sont actuellement en cours dans ce domaine. A titre d'exemple illustratif des attentes de ces études, nous proposons en figure 2 une représentation des deux vecteurs propres dominants de la laplacienne normalisée $\mathcal{D}^{-\frac{1}{2}}\mathcal{F}\mathcal{D}^{-\frac{1}{2}}$ dans le cas de deux classes équilibrées (nombre total de données $n = 256$) de données gaussiennes $\mathcal{N}(\pm 3/\sqrt{p}\mathbf{1}_p, (1 \pm 3/\sqrt{p})I_p)$ de dimension $p = 512$. Deux noyaux sont considérés: $f_1(x) = \exp(-x)$ et $f_2(x) = 1 - \tanh(x)$. Une étude rapide révèle que le vecteur propre 1 (associé à la plus grande valeur propre) porte l'information de séparation des classes sur la base des différences de covariance uniquement, tandis que le vecteur propre 2 contient une information discriminante à la fois sur la moyenne et la covariance. Il apparaît nettement que le choix du noyau importe dans le comportement des vecteurs propres et donc dans la capacité des méthodes à noyaux à gérer des erreurs d'allocation. L'exemple donné en figure 2 révèle

dans ce cas précis des fluctuations plus faibles et une meilleure séparation sur le vecteur propre 2 pour le noyau arc tangente que pour le noyau exponentiel (classiquement utilisé). L'effet sur le vecteur propre 1 est quant à lui moins clair.

5. Réseaux de neurones

L'histoire des réseaux de neurones en tant que systèmes d'intelligence artificielle (IA) remonte au début des années 1950, avec le célèbre Perceptron de Franck Rosenblatt (Rosenblatt, 1958). On peut même remonter aux années 1940 avec les premiers modèles formels de neurones introduits par Warren MacCulloch et Walter Pitts et les premières idées sur la modélisation de l'apprentissage par Donald Hebb. A cette époque, les ordinateurs tels qu'on les connaît aujourd'hui n'existaient pas encore, et ces recherches, menées pour grande partie dans des laboratoires de psychologie, avaient notamment pour objectif de développer des systèmes de traitement de l'information en s'inspirant directement des dernières connaissances en neurobiologie. En se basant sur l'idée du perceptron de Rosenblatt, Bernard Widrow et Tedd Hoff sont parvenus à mettre au point une technique d'apprentissage supervisé pour effectuer des tâches de type *pattern recognition*. Plus tard, dans les années 1970-1980, après une période dominée par l'approche symbolique de l'IA, des systèmes similaires au Perceptron, mais avec des architectures à plusieurs couches, ont été développées avec succès, en particulier grace à l'invention de l'algorithme de rétropropagation du gradient (Rumelhart *et al.*, 1985). Avec le développement de la puissance de calcul des ordinateurs modernes, l'utilisation de ce type de système s'est considérablement développée, notamment ces dernières années avec des résultats spectaculaires obtenus par les méthodes de *deep learning* sur des tâches classiques d'IA comme la reconnaissance d'image, de la parole et le traitement du langage naturel.

En parallèle de ces avancées récentes, dont la mise en pratique est particulièrement complexe en raison de problèmes d'optimisation non-convexes en grande dimension, des systèmes de réseaux de neurones avec des poids aléatoires ont été développés ces dernières années et offrent un compromis particulièrement intéressant entre leur simplicité d'utilisation et leur niveau de performance sur de nombreuses tâches. Au coeur de ces réseaux de neurones, on retrouve évidemment l'utilisation des matrices aléatoires, représentant les poids de connexions entre deux noeuds du réseau, et qui, contrairement au cas typique provenant du traitement du signal ou de la physique, ne sont plus nécessairement carrées et symétriques. Pour une grande partie, la compréhension mathématique de ces réseaux de neurones aléatoires en tant que systèmes d'intelligence artificielle reste encore incomplète et pose de nouvelles questions à la théorie des matrices aléatoires.

5.1. Réseaux de neurones feedforward

Le réseau de neurones le plus simple, appelé perceptron (Rosenblatt, 1958), est composé uniquement d'une couche de neurones d'entrée et d'une couche de neurones

de sortie, de sorte que la fonction reliant l'entrée $x \in \mathbb{R}^p$ et la sortie $y \in \mathbb{R}^q$ est donnée par :

$$y = \phi(Wx + b) \tag{1}$$

avec:

– $W \in \mathbb{R}^{q \times p}$ la matrice des poids de connexions entre les neurones de la couche d'entrée et ceux de la couche de sortie,

– $b \in \mathbb{R}^q$ un vecteur de biais

– $\phi : \mathbb{R} \to \mathbb{R}$ une fonction d'activation, typiquement une fonction sigmoïde ou de seuillage, qui est appliquée élément par élément au vecteur $Wx + b$.

Du point de vue de la modélisation en neuroscience, le même type de modèles a été proposé pour rendre compte de la relation entre le stimulus d'entrée d'un neurone, ici une somme pondérée de p activités d'autres neurones, et son activité de sortie. Dans le cas où ϕ est une fonction Heaviside, cette formule peut se lire ainsi: si la somme pondérée est plus grande qu'un certain seuil alors le neurone est actif, le perceptron étant alors équivalent à un neurone unique.

Dans le cadre d'un problème d'apprentissage supervisé, on se donne n échantillons des variables d'entrée $x_1, ..., x_n$ et n variables de sortie associées $y_1, ..., y_n$. Par exemple, pour un problème de classification, chaque échantillon x_i peut correspondre à l'enregistrement sonore d'un locuteur A ou B (alors p correspond à la longueur de la série temporelle) et y_i vaut 1 si l'enregistrement provient du locuteur A et 0 sinon.

Dans un tel cadre, l'objectif est d'apprendre les poids (W, b) afin de minimiser une certaine fonction de coût L sur une partie des données E_{app} dont on dispose à la fois des entrées et des sorties associées:

$$(\hat{W}, \hat{b}) = \arg \inf_{(W,b) \in \mathbb{R}^{q \times p} \times \mathbb{R}^q} \sum_{i \in E_{\text{app}}} L\left(g(x_i), y_i\right) \tag{2}$$

avec $g(x_i) = \phi(Wx_i + b)$.

On remarque que si on s'intéresse à un problème de régression, avec une fonction de coût quadratique $L(x, y) = ||x - y||^2$, si ϕ est linéaire alors le perceptron est strictement équivalent à une régression linéaire.

A l'époque de Rosenblatt, la bonne manière de résoudre ce problème numériquement était loin d'être évidente car les ordinateurs étaient encore à leur balbutiement et chaque expérience prenait plusieurs jours de programmation "physique" et parfois plusieurs semaines de calculs. Ce problème a été en grande partie résolu par Hoff et Widrow, qui ont trouvé une manière très simple de configurer les poids de manière itérative, suite à chaque nouvelle présentation d'un couple entrée/sortie.

A ce stade, il est particulièrement intéressant de remarquer que Rosenblatt lui même avait exploré la possibilité de tirer au hasard les poids (W, b), et s'était aperçu que parfois, avec de la chance, il obtenait des résultats intéressants. Il aura fallu en réalité attendre l'introduction des réseaux à une couche cachée, puis les années 2000 avec

les *extreme learning machines*, pour que son intuition d'utiliser des poids aléatoires se concrétise dans la pratique. C'est ce que nous allons présenter maintenant. L'introduction d'un couche cachée permet en réalité de considérer un ensemble de perceptrons connectés. En effet, la fonction reliant l'entrée et la sortie est alors donnée par :

$$y = \psi\left(W_2\phi(W_1 x + b_1) + b_2\right) = f(x) \tag{3}$$

avec $W_1 \in \mathbb{R}^{h\times p}$, $W_2 \in \mathbb{R}^{q\times h}$, $b_1 \in \mathbb{R}^h$ et $b_2 \in \mathbb{R}^q$ et avec des fonctions réelles ϕ et ψ qui sont appliquées élément par élément, et avec h un entier qui correspond au nombre de neurones de la couche cachée. Dans le cadre de l'apprentissage supervisé, on cherche alors à résoudre un problème d'optimisation similaire à (2), mais avec $g = f$, en cherchant les meilleurs poids et biais $\hat{P} := (\hat{W}_1, \hat{W}_2, \hat{b}_1, \hat{b}_2)$. Alors que le perceptron simple était très limité dans sa capacité, notamment à apprendre des problèmes non-linéairement séparables, les réseaux à une couche cachée possèdent de leur côté une propriété très puissante d'approximation universelle (Hornik *et al.*, 1989): en faisant tendre h vers l'infini, l'infimum obtenu dans le problème d'optimisation peut être rendu arbitrairement petit. Bien entendu, cela ne permet pas nécessairement de faire de bonnes prédictions sur des données test et amène son lot de problèmes comme le sur-apprentissage. Il est important de remarquer qu'en général ce problème d'optimisation est non convexe et qu'il existe potentiellement un grand nombre de minima locaux.

Afin de surmonter ces deux problèmes (sur-apprentissage et nombreux minima locaux), tout en préservant la propriété d'approximation universelle, une approche utilisant l'aléatoire a été proposée pendant les années 2000, sous le nom d'*extreme learning machine* (ELM) (Huang *et al.*, 2004 ; Huang, Zhu, Siew, 2006). L'idée est de garder la même structure entrée/sortie (3), mais à la différence que les poids (W_1, b_1) reliant la couche d'entrée à la couche cachée sont désormais tirés au hasard, par exemple suivant des lois i.i.d. gaussienne ou Bernoulli, l'optimisation se faisant uniquement sur les poids (W_2, b_2). Si on s'intéresse par exemple à un problème de régression avec L et ψ convexes (par exemple, respectivement quadratique et linéaire), il est alors possible de démontrer que :

– le problème d'optimisation est convexe ;

– la capacité d'approximation universelle est préservée (Huang, Chen, Siew, 2006).

Sur le plan qualitatif, le phénomène de sur-apprentissage est généralement moins présent car le nombre de paramètres à apprendre est plus faible, de sorte que la projection aléatoire vers la couche cachée opère une forme de régularisation.

D'un point de vue pratique, cette méthode permet donc d'apprendre des relations non-linéaires complexes entre les variables d'entrée et de sortie, tout en restant dans le cadre des problèmes convexes, dont la résolution numérique est beaucoup plus stable et plus aisée. Cependant, sur le plan théorique, de nombreuses questions restent ouvertes. En particulier, on peut se demander comment choisir la loi des poids aléatoires, par exemple leur variance, afin d'optimiser la performance de généralisation du système. Si on prend le point de vue des noyaux (Huang, 2014), on s'attend alors à ce

que le noyau aléatoire $F(x, x') = \phi(W_1 x + b_1)^T \cdot \phi(W_1 x' + b_1)$ joue un rôle important dans la compréhension de l'impact de la loi de (W_1, b_1). Dans certains cas particuliers, il est possible d'obtenir une expression explicite pour F dans la limite où le nombre de neurones de la couche cachée tend vers l'infini (Williams, 1998). Plus précisément, un résultat particulièrement frappant apparaît lorsque l'on choisit $\phi(x) = \mathrm{erf}(x)$ et lorsque W_1 et b_1 sont composés de variables i.i.d. gaussiennes standard. En effet, dans ce cas, lorsque l'on considère un problème de régression avec régularisation l_2, à savoir

$$\inf_{W_2 \in \mathbb{R}^{q \times p} \times \mathbb{R}^q} \|W_2 \phi(W_1 X + b_1) - Y\|^2 + \gamma \|W_2\|^2 \tag{4}$$

on peut alors se passer de calculer le minimiseur $\hat{W}_2$ et montrer que pour prédire une sortie $\tilde{Y}$ à partir d'un nouvel échantillon $\tilde{X}$ il suffit d'évaluer les produits scalaires $\tilde{K} = \tilde{X}^\mathsf{T} X$ et $K = X^\mathsf{T} X$ pour prédire alors $\tilde{Y} = \tilde{K}(K + \gamma I)^{-1} Y$. En utilisant la loi des grands nombres et après un calcul assez technique (Williams, 1998) on obtient une formule explicite pour les *noyaux* K et $\tilde{K}$ dans la limite d'un nombre infini de neurones:

$$K_{k,l} = k(x_k, x_l) \text{ et } \tilde{K}_{k,l} = k(\tilde{x}_k, \tilde{x}_l) \tag{5}$$

avec

$$k(u, v) = \frac{2}{\pi} \arcsin\left(\frac{2u'v}{\sqrt{(1 + 2u'u)(1 + 2v'v)}} \right). \tag{6}$$

Lorsque la dimension des entrées (le nombre de variables) est grande, on s'aperçoit alors que le noyau ainsi obtenu s'approche du noyau "angle" $k(u, v) = 1 - \text{angle}(u, v)$. Ce résultat établit donc une connexion géométrique étonnante entre un réseau de neurones aléatoire gaussien avec une non-linéarité erf et le fait de mesurer des similarités entre les données à l'aide de l'angle. D'un point de vue pratique, utiliser ce noyau dans le cadre d'une régression à noyau sur les données MNIST donne des résultats supérieurs aux SVM gaussiens, avec une erreur de $1,25\ \%$, contre $1,4\ \%$ pour les SVM. Malgré le travail important déjà réalisé dans (Rahimi, Recht, 2007), il reste certainement de nombreuses choses à découvrir sur l'interprétation géométrique des réseaux de neurones aléatoires infinis, et notamment sur l'interaction entre les différentes lois des matrices de connexions et le choix de la non-linéarité.

Ce type de systèmes basés sur des poids aléatoires a été étendu (Cambria *et al.*, 2013) au cas des réseaux "deep networks" à plusieurs couches cachées. A la base de la construction de ces systèmes, on retrouve une idée particulièrement importante qui est celle de l'auto-encodeur. Le principe des auto-encodeurs est d'apprendre à un réseau de neurones à reproduire les signaux d'entrée, c'est-à-dire que $y_i = x_i$. Par exemple, si le nombre de neurones dans la couche cachée h est plus petit que $p = q$ alors il est possible de reconstruire x_i à partir des activations des h neurones cachés, opérant ainsi une opération de réduction de dimension. On peut alors construire un auto-encodeur aléatoire en prenant un ELM auquel on demande d'apprendre $y_i = x_i$. Avec une

fonction de coût quadratique et ψ linéaire, on doit alors résoudre, avec $W_1 \in \mathbb{R}^{h \times p}$ une matrice aléatoire:

$$\hat{W}_2 = \arg \inf_{W_2 \in \mathbb{R}^{q \times h}} \sum_{i \in E_{app}} ||W_2 \phi(W_1 x_i) - x_i||^2$$

où on a volontairement oublié les termes de biais par souci de clarté. Il s'agit d'un problème de moindres carrés, dont la solution est donnée par $\hat{W}_2 = (H^T H)^{-1} H X^T$ avec $H = \phi(W_1 X)$. On peut alors construire un réseau à plusieurs couches en accolant des couches d'auto-encodeurs de tailles différentes, ce qui permet par exemple d'obtenir de bons résultats de classification sur la base MNIST (reconnaissance de chiffres écrits à la main) avec un temps de calcul bien moins important que les méthodes classiques de deep learning. La compréhension mathématique de ce type de systèmes reste encore un sujet ouvert, notamment en ce qui concerne le choix optimal du bon nombre de couches, de neurones par couches, des non-linéarités, etc.

Les ELM et les auto-encodeurs aléatoires ont pour point commun de commencer par une multiplication entre les inputs et une matrice aléatoire, ce qu'on retrouve également dans la technique des projections aléatoires pour la réduction de dimension (Achlioptas, 2003 ; Dasgupta, 2000 ; Bingham, Mannila, 2001), basée sur le lemme de Johnson-Lindenstrauss (Johnson, Lindenstrauss, 1984), ou encore avec la méthode du *compressed sensing* (Donoho, 2006) qui permet la reconstruction de signaux parcimonieux avec un petit nombre de mesures aléatoires.

5.2. Réseaux de neurones récurrents

Les réseaux de neurones récurrents forment *a priori* une classe de réseaux de neurones plus générale que les réseaux feedforward. Dans un réseau feedforward, les connexions ne vont que dans un sens, alors que dans un réseau récurrent on peut considérer des connexions qui vont dans les deux sens (voir figure 3). Ce type de réseaux est particulièrement adapté pour des problèmes impliquant des séries temporelles car la représentation du signal d'entrée à l'instant t dépend du passé de ce signal, contrairement au cas feedforward qui ne prend pas en compte l'ordre dans lequel les entrées sont présentées.

Les réseaux récurrents sont particulièrement difficiles à entraîner par rétropropagation du gradient, en particulier à cause des fameux problèmes du *vanishing gradient* et des bifurcations qui ont lieu pendant l'apprentissage. Ces problèmes numériques rendent l'utilisation pratique de ce type de réseaux très délicat et malgré la promesse d'obtenir des performances supérieures au réseaux feedforward, leur utilisation reste encore relativement marginale, malgré quelques progrès récents (Martens, Sutskever, 2011 ; Sutskever *et al.*, 2011 ; Graves *et al.*, 2013).

Dans la même idée que les ELM, on peut alors considérer un réseau de neurones récurrent connecté aléatoirement et s'intéresser uniquement au problème d'optimisation simple qui cherche à minimiser le coût entre les sorties et une combinaison linéaire de l'activité des différents neurones. Ce type de système, appelé Echo-State

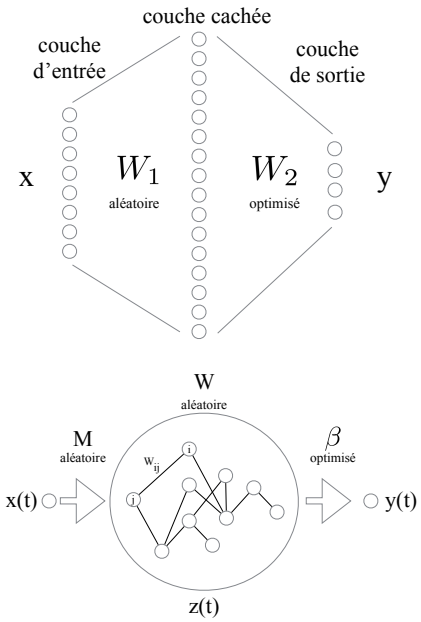

Figure 3. En haut : représentation schématique d'une Extreme Learning Machine. *En bas : représentation schématique d'un* Echo-State Network

Networks (ESN) a été introduit au début des années 2000 et a permis d'obtenir des performances excellentes sur de nombreux problèmes impliquant des séries temporelles (Jaeger, Haas, 2004). Plus précisément, on se place dans un problème d'apprentissage supervisé où l'on cherche à prédire une série temporelle cible $y(t) \in \mathbb{R}^q$ à partir d'une série temporelle d'entrée $x(t) \in \mathbb{R}^p$ pour $1 \leq t \leq T$. Par exemple, pour la prédiction de série temporelle, on prendra $y(t) = x(t+1)$. On considère alors le système dynamique à coefficient aléatoires et stimulé par un signal externe $x(t)$:

$$z(t+1) = S(Wz(t) + Mx(t) + b) \tag{7}$$

avec $W \in \mathbb{R}^{n \times n}$, $M \in \mathbb{R}^{n \times p}$, $b \in \mathbb{R}^n$ des matrices et vecteurs aléatoires, n un entier correspondant au nombre de neurones dans le réseau récurrent et $S : \mathbb{R} \to \mathbb{R}$ une fonction, typiquement sigmoïde, qui est appliquée élément par élément, la variable $z_k(t)$ décrit l'activité du neurone k à l'instant t. On cherche alors à résoudre le problème d'optimisation:

$$\hat{\beta} = \arg \inf_{\beta \in \mathbb{R}^{p \times n}} \sum_{t=1}^{T} ||\beta z(t) - y(t)||^2 \tag{8}$$

Encore une fois, il s'agit d'un simple problème de moindres carrés [1] et la solution $\hat{\beta}$ est explicite. Cette méthode peut donner d'excellents résultats de prédiction si les matrices et vecteurs aléatoires W, M et b sont choisis suivant une bonne loi. En particulier, parmi tous les degrés de liberté que ce choix offre, le rayon spectral de la matrice aléatoire W semble être le paramètre qui a le plus d'impact sur la performance des ESN. La méthode actuelle pour choisir par exemple ce rayon spectral est de procéder à une validation croisée qui peut être très coûteuse en terme de temps de calcul. En effet, on ne connaît pas actuellement de formule théorique qui relierait la performance de généralisation d'un tel système avec des quantités reliées au spectre de la matrice aléatoire W. Par ailleurs, avant même de s'intéresser à la performance, le système dynamique (7) doit d'abord être stable face aux petites perturbations, car on souhaite que deux signaux d'entrée x et x' très proches ne puissent pas générer deux représentations z et z' très différentes. Cette propriété, appelée *Echo-state property*, est en réalité très liée à la notion de chaos et à la présence d'exposants de Lyapunov positifs. Récemment, grâce à des méthodes de champs moyen empruntées à la physique statistique (Massar, Massar, 2013 ; Galtier, Wainrib, 2014) il a été possible d'établir un lien entre le rayon spectral de W et cette transition vers le chaos, dans le cas où W est une matrice de Ginibre réelle, c'est-à-dire à coefficients i.i.d. gaussiens. Certaines études ont également considéré des cas où la matrice de connexion W n'est pas complètement aléatoire et présente une certaine structure (Ozturk *et al.*, 2007 ; Song, Feng, 2010), mais le problème de trouver une manière simple d'ajuster la loi de W en fonction de la tâche à effectuer reste encore largement ouvert.

Plus largement, au-delà de la forme précise de l'équation (7), on peut considérer un cadre plus général, appelé *reservoir computing* (LukošEvičIus, Jaeger, 2009) dans lequel le signal d'entrée stimule un certain système dynamique, créant ainsi une représentation en grande dimension de ce signal, à partir de laquelle on peut chercher à résoudre un problème d'apprentissage supervisé. On peut considérer par exemple les *liquid-state machines* composés de réseaux de neurones à potentiel d'action (Maass *et al.*, 2002), des approches de *physical computing* avec des systèmes basés sur la dynamique des fluides (Fernando, Sojakka, 2003) ou les propriétés optiques des matériaux désordonnés (Paquot *et al.*, 2012).

5.3. Modélisation en neuroscience

Au-delà des applications en intelligence artificielle, le type de modèles mathématiques présentés dans les deux sections précédentes est aussi intéressant du point de vue de la modélisation du cerveau. En effet, un des enjeux centraux des neurosciences est de comprendre comment les capacités de traitement de l'information et les fonctions cognitives peuvent émerger d'un système composé d'un très grand nombre d'éléments dynamiques interconnectés de manière particulièrement complexe et apparemment désordonnée.

1. On considère souvent en pratique la version régularisée de ce problème.

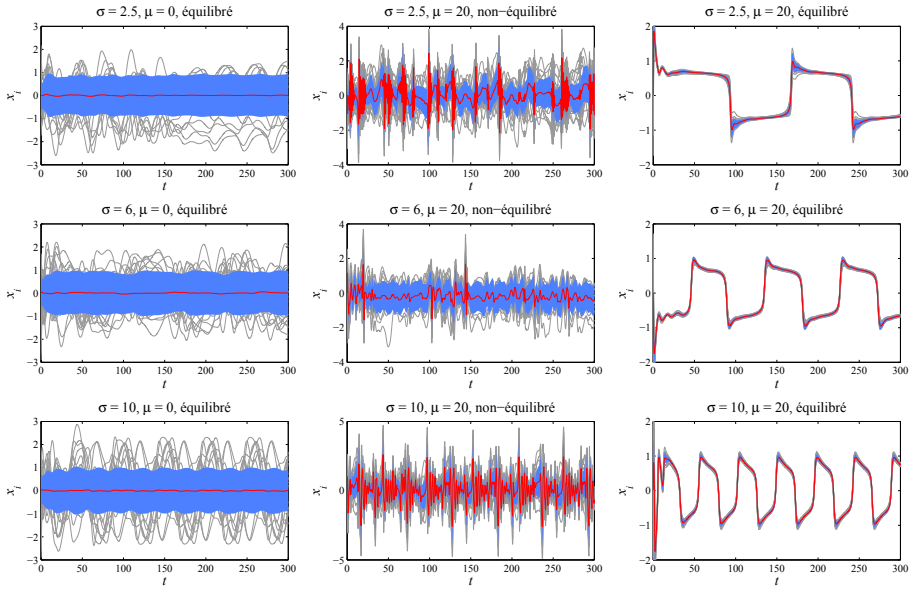

Figure 4. Exemple de dynamiques de réseaux de neurones connectés aléatoirement avec différentes lois pour J et m. La première ligne correspond à $J_{ij} \sim \mathcal{N}(0, 1/n)$ i.i.d. et $m_i = \pm 1/\sqrt{n}$. La deuxième ligne correspond à J est de plus supposée parcimonieuse avec un coefficient $p = 0.5$ et, pour la troisième ligne, on suppose une structure de graphe small world *pour J. Pour ces deux dernières lignes, les J_{ij} non nuls sont uniformes dans $[-1, 1]$ et m est un vecteur gaussien normalisé à somme nulle. D'après (Del Molino* et al.*, 2013)*

En particulier, l'équation (7) sans entrée, ou sa version à temps continu, qui s'inspire des modèles de verres de spin dynamiques en physique statistique, est considérée comme un modèle particulièrement important pour étudier les liens entre la manière dont les neurones sont connectés et les propriétés dynamiques et de calcul du réseau (Sompolinsky *et al.*, 1988 ; Rajan *et al.*, 2010 ; Toyoizumi, Abbott, 2011). Dans ce modèle, si l'on suppose que les poids de connexion W_{ij} sont des variables aléatoires indépendantes centrées et de variance σ^2/n, et si S est une fonction sigmoïde, nulle et de pente unité en zéro, alors il existe une transition de phase, dans la limite des grands réseaux, avec apparition de trajectoires chaotiques dans la dynamique du réseau de neurone lorsque σ excède 1. On retrouve cette transition en observant la matrice jacobienne en zéro $-I + W$, dont les valeurs propres se répartissent suivant une loi circulaire centrée en -1 et de rayon σ: la transition de phase correspond alors à la déstabilisation du point d'équilibre trivial zéro. La théorie des matrices aléatoires permet d'éclaircir un peu plus cette transition, notamment en ce qui concerne le nombre de points d'équilibre (Wainrib, Touboul, 2013) ou les effets de taille finie (Wainrib, Del Molino, 2013). En particulier, la probabilité p_0 que le point d'équilibre trivial 0 soit linéairement stable pour le système dynamique $\dot{x} = -x + WS(x)$ où $x \in \mathbf{R}^n$ doit

tendre vers 1 lorsque $n \to \infty$ si $\sigma < 1$. En effet, il s'agit d'une conséquence de la loi circulaire: le maximum des parties réelles des valeurs propres de $\mathbf{W}$ tend vers $\sigma < 1$ lorsque $n \to \infty$, et dès lors la Jacobienne en 0 n'a pas de valeurs propres à partie réelle positive. Certes, pour n petit, cette probabilité n'est pas égale à 1 et on pourrait alors raisonnablement penser que plus n est grand plus p_0 s'approche de 1. Or pour σ suffisamment proche de 1, on observe qu'il existe une valeur intermédiaire de n qui minimise cette probabilité, de sorte que cette taille de réseau est plus favorable à l'apparition de dynamiques complexes. Ce résultat est démontré dans (Wainrib, Del Molino, 2013) à l'aide de la théorie des valeurs extrêmes pour les valeurs propres des matrices aléatoires récemment développée dans (Bender, 2010 ; Rider *et al.*, 2014).

Au-delà de ces modèles matriciels élémentaires, la théorie des matrices aléatoires apparaît centrale pour comprendre comment les différentes contraintes de structure des réseaux de neurones biologiques influencent le comportement du système. Parmi ces contraintes, on peut mentionner la parcimonie du graphe de connexions, la distribution des degrés, le rôle de l'asymétrie des connexions, l'équilibre entre excitation ou inhibition, ou encore le fait que les neurones sont généralement soit excitateurs, soit inhibiteurs. Arrêtons nous quelques instants sur cette dernière propriété, appelée principe de Dale. Celle-ci se traduit par l'introduction de matrices aléatoires $W_{ij} = \mu m_j + \sigma J_{ij}$ avec $m = (m_j)$ un vecteur normalisé de somme nulle et J_{ij} des variables aléatoires indépendantes centrées de variances éventuellement différentes, et σ, μ deux paramètres positifs. Ce modèle de matrice aléatoire W a été introduite dans (Rajan, Abbott, 2006) pour modéliser des réseaux constitués de neurones qui ne peuvent avoir qu'une action ou bien excitatrice (colonne avec une moyenne $m_j > 0$) ou bien inhibitrice (colonne avec une moyenne $m_j < 0$). Il s'agit là d'un cas de perturbation de rang fini, qui a suscité l'intérêt des mathématiciens (Tao, 2013). Dans le cas de variances identiques, ce dernier a montré que la distribution spectrale de ce modèle spiked converge vers la combinaison de la loi circulaire et d'un nombre fini de valeurs propres spiked présentes en dehors du disque unité. Lorsqu'on impose la condition de balance stricte $\sum_i W_{ij} = 0$, les valeurs propres isolées disparaissent, révélant un lien très important entre des propriétés de structure du réseau et des propriétés du spectre de la matrice de connectivité qui influence la stabilité des états stationnaires et les bifurcations du système. Par ailleurs, sur le plan de la dynamique, ce modèle a été étudié dans (Del Molino *et al.*, 2013) où il a été montré l'existence d'une transition vers des dynamiques synchronisées oscillatoires (voir la figure 4), qui ne peut pas se déduire directement de l'étude du spectre de W, montrant ainsi une limitation dans l'outil matrice aléatoire pour l'étude de ce type de modèles.

6. Conclusion

En somme, il apparaît que les domaines très actifs aujourd'hui de l'apprentissage supervisé (ou non supervisé) et du big data de manière générale regorgent de questions relevant de l'analyse de modèles matriciels aléatoires. La raison principale de l'omniprésence de ces modèles réside dans l'impossibilité de gérer des données déterministes de tailles colossales sans un minimum de régularité et d'invariance struc-

turelle, que l'aléa offre lorsque les dimensions du système considéré augmentent (par des principes standards de loi des grands nombres) ; cette remarque est d'autant plus marquée dans le cadre des réseaux de neurones dits ESN abordés ici pour lesquels les connexions neuronales sont volontairement prises aléatoirement. Dans de nombreux cas, ces modèles se distinguent cependant du corpus des travaux d'application des matrices aléatoires au traitement du signal, encore très limité à ce jour autour de modèles de covariances empiriques. De nombreuses voies de recherches s'ouvrent ainsi, sous l'impulsion notamment de quelques exemples clés, qui, nous en sommes convaincus, promettent d'ouvrir de nouveaux champs de perspectives dans l'intégration naturelle et obligatoire de l'aléa pour l'analyse de grands systèmes de données.

Cette révolution est en réalité déjà en marche lorsque l'on sait que l'outil moteur, aujourd'hui en big data, qu'est celui des méthodes parcimonieuses requiert un principe fondamental d'invariance, dit RIP (*restricted isometry property*), rencontré uniquement pour des modèles matriciels suffisamment aléatoires.

Remerciements

Ce travail a été conjointement soutenu par le projet ANR RMT4GRAPH (ANR-14-CE28-0006) ainsi que par le projet "Jeunes Chercheurs" du GdR ISIS–GRETSI.

Bibliographie

Achlioptas D. (2003). Database-friendly random projections: Johnson-lindenstrauss with binary coins. *Journal of computer and System Sciences*, vol. 66, n° 4, p. 671–687.

Bai Z. D., Silverstein J. W. (1998). No eigenvalues outside the support of the limiting spectral distribution of large dimensional sample covariance matrices. *The Annals of Probability*, vol. 26, n° 1, p. 316-345.

Baik J., Silverstein J. W. (2006). Eigenvalues of large sample covariance matrices of spiked population models. *Journal of Multivariate Analysis*, vol. 97, n° 6, p. 1382-1408.

Benaych-Georges F., Nadakuditi R. R. (2012). The singular values and vectors of low rank perturbations of large rectangular random matrices. *Journal of Multivariate Analysis*, vol. 111, p. 120–135.

Bender M. (2010). Edge scaling limits for a family of non-hermitian random matrix ensembles. *Probability theory and related fields*, vol. 147, n° 1-2, p. 241–271.

Bianchi P., Najim J., Maida M., Debbah M. (2011). Performance of some eigen-based hypothesis tests for collaborative sensing. *IEEE Transactions on Information Theory*, vol. 57, n° 4, p. 2400-2419.

Bickel P. J., Levina E. (2008). Regularized estimation of large covariance matrices. *The Annals of Statistics*, vol. 36, n° 1, p. 199–227.

Bingham E., Mannila H. (2001). Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the seventh acm sigkdd international conference on knowledge discovery and data mining*, p. 245–250.

Cambria E., Huang G.-B., Kasun L. L. C., Zhou H., Vong C.-M., Lin J. *et al.* (2013). Extreme learning machines. *IEEE Intelligent Systems*, vol. 28, n° 6, p. 30–59.

Couillet R., Debbah M. (2011). *Random Matrix Methods for Wireless Communications* (first éd.). New York, NY, USA, Cambridge University Press.

Couillet R., McKay M. (2014). Large dimensional analysis and optimization of robust shrinkage covariance matrix estimators. *Journal of Multivariate Analysis*, vol. 131, p. 99-120.

Couillet R., Pascal F., Silverstein J. W. (2015). The random matrix regime of Maronna's M-estimator with elliptically distributed samples. *Journal of Multivariate Analysis*, vol. 139, p. 56–78.

Dasgupta S. (2000). Experiments with random projection. In *Proceedings of the sixteenth conference on uncertainty in artificial intelligence*, p. 143–151.

Del Molino L. C. G., Pakdaman K., Touboul J., Wainrib G. (2013). Synchronization in random balanced networks. *Physical Review E*, vol. 88, n° 4, p. 042824.

Donoho D. L. (2006). Compressed sensing. *Information Theory, IEEE Transactions on*, vol. 52, n° 4, p. 1289–1306.

El Karoui N. (2010). The spectrum of kernel random matrices. *The Annals of Statistics*, vol. 38, n° 1, p. 1–50.

Fernando C., Sojakka S. (2003). Pattern recognition in a bucket. In *Advances in artificial life*, p. 588–597. Springer.

Galtier M., Wainrib G. (2014). A local echo state property through the largest lyapunov exponent. *arXiv preprint arXiv:1402.1619*.

Girko V. L. (1987). Introduction to general statistical analysis. *Theory of Probability & Its Applications*, vol. 32, n° 2, p. 229–242.

Graves A., Mohamed A.-R., Hinton G. (2013). Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, p. 6645–6649.

Guédon O., Vershynin R. (2014). Community detection in sparse networks via grothendieck's inequality. *arXiv preprint arXiv:1411.4686*.

Hornik K., Stinchcombe M., White H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, vol. 2, n° 5, p. 359–366.

Huang G.-B. (2014). An insight into extreme learning machines: random neurons, random features and kernels. *Cognitive Computation*, vol. 6, n° 3, p. 376–390.

Huang G.-B., Chen L., Siew C.-K. (2006). Universal approximation using incremental constructive feedforward networks with random hidden nodes. *Neural Networks, IEEE Transactions on*, vol. 17, n° 4, p. 879–892.

Huang G.-B., Zhu Q.-Y., Siew C.-K. (2004). Extreme learning machine: a new learning scheme of feedforward neural networks. In *Neural networks, 2004. proceedings. 2004 ieee international joint conference on*, vol. 2, p. 985–990.

Huang G.-B., Zhu Q.-Y., Siew C.-K. (2006). Extreme learning machine: theory and applications. *Neurocomputing*, vol. 70, n° 1, p. 489–501.

Jaeger H., Haas H. (2004). Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, vol. 304, n° 5667, p. 78–80.

Johnson W. B., Lindenstrauss J. (1984). Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics*, vol. 26, n° 189-206, p. 1.

Johnstone I. M. (2001). On the distribution of the largest eigenvalue in principal components analysis. *Annals of Statistics*, vol. 99, n° 2, p. 295-327.

Krzakala F., Moore C., Mossel E., Neeman J., Sly A., Zdeborová L. *et al.* (2013). Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, vol. 110, n° 52, p. 20935–20940.

LukošEvičIus M., Jaeger H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, vol. 3, n° 3, p. 127–149.

Maass W., Natschläger T., Markram H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural computation*, vol. 14, n° 11, p. 2531–2560.

Martens J., Sutskever I. (2011). Learning recurrent neural networks with hessian-free optimization. In *Proceedings of the 28th international conference on machine learning (icml-11)*, p. 1033–1040.

Massar M., Massar S. (2013). Mean-field theory of echo state networks. *Physical Review E*, vol. 87, n° 4, p. 042809.

Mestre X. (2008, novembre). Improved estimation of eigenvalues of covariance matrices and their associated subspaces using their sample estimates. *IEEE Transactions on Information Theory*, vol. 54, n° 11, p. 5113-5129.

Nadakuditi R. R., Newman M. E. J. (2012). Graph spectra and the detectability of community structure in networks. *Physical review letters*, vol. 108, n° 18, p. 188701.

Nadakuditi R. R., Newman M. E. J. (2013). Spectra of random graphs with arbitrary expected degrees. *Physical Review E*, vol. 87, n° 1, p. 012803.

Ng A. Y., Jordan M., Weiss Y. (2001). On spectral clustering: Analysis and an algorithm. *Proceedings of Advances in Neural Information Processing Systems. Cambridge, MA: MIT Press*, vol. 14, p. 849–856.

Ozturk M. C., Xu D., Príncipe J. C. (2007). Analysis and design of echo state networks. *Neural Computation*, vol. 19, n° 1, p. 111–138.

Paquot Y., Duport F., Smerieri A., Dambre J., Schrauwen B., Haelterman M. *et al.* (2012). Optoelectronic reservoir computing. *Scientific reports*, vol. 2.

Rahimi A., Recht B. (2007). Random features for large-scale kernel machines. In *Advances in neural information processing systems*, p. 1177–1184.

Rajan K., Abbott L. (2006). Eigenvalue spectra of random matrices for neural networks. *Physical review letters*, vol. 97, n° 18, p. 188104.

Rajan K., Abbott L., Sompolinsky H. (2010). Stimulus-dependent suppression of chaos in recurrent neural networks. *Physical Review E*, vol. 82, n° 1, p. 011903.

Rider B., Sinclair C. D. *et al.* (2014). Extremal laws for the real ginibre ensemble. *The Annals of Applied Probability*, vol. 24, n° 4, p. 1621–1651.

Rosenblatt F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, vol. 65, n° 6, p. 386.

Rumelhart D. E., Hinton G. E., Williams R. J. (1985). *Learning internal representations by error propagation*. Rapport technique. DTIC Document.

Sompolinsky H., Crisanti A., Sommers H. (1988). Chaos in random neural networks. *Physical Review Letters*, vol. 61, n° 3, p. 259.

Song Q., Feng Z. (2010). Effects of connectivity structure of complex echo state network on its prediction performance for nonlinear time series. *Neurocomputing*, vol. 73, n° 10, p. 2177–2185.

Sutskever I., Martens J., Hinton G. E. (2011). Generating text with recurrent neural networks. In *Proceedings of the 28th international conference on machine learning (icml-11)*, p. 1017–1024.

Tao T. (2013). Outliers in the spectrum of iid matrices with bounded rank perturbations. *Probability Theory and Related Fields*, vol. 155, n° 1-2, p. 231–263.

Toyoizumi T., Abbott L. (2011). Beyond the edge of chaos: Amplification and temporal integration by recurrent networks in the chaotic regime. *Physical Review E*, vol. 84, n° 5, p. 051908.

Tulino A. M., Verdú S. (2004). Random matrix theory and wireless communications. *Foundations and Trends in Communications and Information Theory*, vol. 1, n° 1.

Vallet P., Loubaton P., Mestre X. (2012). Improved subspace estimation for multivariate observations of high dimension: the deterministic signals case. *IEEE Transactions on Information Theory*, vol. 58, n° 2, p. 1043–1068.

Vinogradova J., Couillet R., Hachem W. (2014). Estimation of Toeplitz covariance matrices in large dimensional regime with application to source detection. *IEEE Transactions on Signal Processing*. Consulté sur http://arXiv.org/pdf/1403.1243

Wainrib G., Del Molino L. C. G. (2013). Optimal system size for complex dynamics in random neural networks near criticality. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 23, n° 4, p. 043134.

Wainrib G., Touboul J. (2013). Topological and dynamical complexity of random neural networks. *Physical review letters*, vol. 110, n° 11, p. 118101.

Williams C. K. (1998). Computation with infinite neural networks. *Neural Computation*, vol. 10, n° 5, p. 1203–1216.

Zhang T., Cheng X., Singer A. (2014). Marchenko-Pastur Law for Tyler's and Maronna's M-estimators. *http://arxiv.org/abs/1401.3424*.

Zhang X., Nadakuditi R. R., Newman M. E. J. (2014). Spectra of random graphs with community structure and arbitrary degrees. *Physical Review E*, vol. 89, n° 4, p. 042816.

Zhao Y., Levina E., Zhu J. *et al*. (2012). Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics*, vol. 40, n° 4, p. 2266–2292.

Article soumis le 26/03/2015

Accepté le 2/03/2016