ARIA



# Mini-projet ARIA : Théorie des Graphes

## Utilisations de la transformé de fourrier sur graphe pour évaluer le sur apprentissage d'un réseaux de neurones.

Itai Zehavi

# Table des matières

# 1 Introduction

Ce travail s'inscrit dans le cadre du mini-projet du cours de théorie des graphes d'ARIA, où l'objectif principal est d'explorer la Transformée de Fourier sur Graphe (Graph Fourier Transform, GFT) et d'exploiter ses propriétés pour analyser le phénomène de surapprentissage (overfitting) dans un réseau de neurones artificiel.

La Transformée de Fourier sur Graphe est une généralisation de la Transformée de Fourier classique aux signaux définis sur des structures non euclidiennes, telles que les graphes. Elle repose sur l'analyse spectrale de la matrice de Laplacien associée au graphe, permettant de décomposer un signal en composantes fréquentielles adaptées à la topologie du graphe. Cette approche offre des outils puissants pour l'analyse de signaux complexes sur des réseaux, qu'il s'agisse de données sociales, biologiques ou, comme dans notre cas, des réseaux de neurones.

Dans le contexte des réseaux de neurones artificiels, le surapprentissage se manifeste lorsque le modèle s'ajuste trop étroitement aux données d'entraînement, au détriment de sa capacité à généraliser à des données nouvelles. L'une des hypothèses sous-jacentes de ce travail est que les activations des neurones dans un réseau qui surapprend contiennent une proportion plus importante de composantes de haute fréquence dans le domaine spectral, ce qui reflète une complexité excessive dans l'ajustement des données.

Pour atteindre cet objectif, ce rapport s'articule autour de deux étapes principales. Tout d'abord, nous présenterons les fondements théoriques nécessaires à la compréhension de la Transformée de Fourier sur Graphe, en introduisant notamment les notions de graphe, de matrice de Laplacien et d'analyse spectrale. Ensuite, nous appliquerons ces outils pour étudier et caractériser le surapprentissage dans un réseau de neurones artificiel à l'aide de la GFT, en analysant ses activations dans le domaine spectral.

Ainsi, ce projet met en lumière le lien entre théorie des graphes, transformées spectrales et apprentissage automatique, tout en illustrant l'intérêt pratique de la GFT pour diagnostiquer et comprendre le comportement des modèles d'intelligence artificielle.

# 2 Outils mathématiques sur les Graphes à Signals

Les outils présentés dans cette section sont tirés de l'article [1]. Les démonstrations et interprétations fournies ici sont un mélange de mes propres contributions et de celles proposées dans l'article, qui, dans certains cas, manquent de détails ou sont absentes.

## 2.1 Laplacien combinatoire

**Definition :** Soit un graphe pondéré non orienté avec un Laplacien combinatoire $L$ défini par :

$$L = D - W$$

où :
- $D$ est la matrice des degrés, dont chaque élément diagonal $D_{i,i}$ est égal à la somme des poids des arêtes connectées au sommet $i$, soit $D_{i,i} = \sum_j W_{i,j}$.
- $W$ est la matrice d'adjacence pondérée, où chaque élément $W_{i,j}$ représente le poids de l'arête entre les sommets $i$ et $j$.

**Propriété :** l'action du Laplacien sur le signal $f$ au sommet $i$ est donnée par :

$$(Lf)_i = \sum_{j \in \mathcal{N}_i} W_{i,j}(f(i) - f(j))$$

où $\mathcal{N}_i$ représente l'ensemble des sommets connectés au sommet $i$

**Preuve :**

Soit un graphe pondéré non orienté avec un Laplacien combinatoire $L$ défini par :

$$L = D - W$$

Considérons maintenant un signal $f$ défini sur les sommets du graphe, représenté par un vecteur $f \in \mathbb{R}^N$, où chaque composante $f(i)$ est la valeur du signal au sommet $i$.

L'action du Laplacien $L$ sur le signal $f$ est donnée par le produit matrice-vecteur $Lf$. La $i$-ème composante de ce produit, notée $(Lf)_i$, est :

$$(Lf)_i = \sum_j L_{i,j} f(j)$$

En substituant $L = D - W$, nous obtenons :

$$(Lf)_i = \sum_j (D_{i,j} - W_{i,j}) f(j)$$

Puisque $D$ est une matrice diagonale, seuls les éléments diagonaux $D_{i,i}$ sont non nuls, donc l'expression devient :

$$(Lf)_i = D_{i,i} f(i) - \sum_j W_{i,j} f(j)$$

Rappelons que $D_{i,i} = \sum_j W_{i,j}$, ce qui nous permet de réécrire $(Lf)_i$ comme suit :

$$(Lf)_i = f(i) \sum_j W_{i,j} - \sum_j W_{i,j} f(j)$$

En factorisant $f(i)$ dans le premier terme, nous obtenons :

$$(Lf)_i = \sum_j W_{i,j}(f(i) - f(j))$$

Ainsi, l'action du Laplacien sur le signal $f$ au sommet $i$ est donnée par :

$$(Lf)_i = \sum_{j \in \mathcal{N}_i} W_{i,j}(f(i) - f(j))$$

où $\mathcal{N}_i$ représente l'ensemble des sommets connectés au sommet $i$. □

**Remarque :** Cette expression montre que $(Lf)_i$ mesure la variation locale du signal $f$. Notamment la $i^e$ ligne correspond à la variation entre le sommet $i$ et ses voisins, pondérée par les poids des arêtes. Si $f$ est constant sur tous les sommets voisins de $i$, alors $(Lf)_i = 0$.

## 2.2 Transformée de fourier sur Graphe (GFT)

En faisant l'analogie avec la transformée de fourier d'une fonction, nous pouvons définir celle sur un Graphe de signal.

**Définition : Transformée de Fourier sur Graphe (GFT)**
La Transformée de Fourier sur Graphe (GFT) d'un signal $f$ sur le graphe est définie par :

$$\hat{f}(\lambda_l) = \langle f, u_l \rangle = \sum_{i=1}^{N} f(i) u_l(i),$$

où $u_l$ est le $l$-ème vecteur propre du Laplacien $L$ et $\lambda_l$ la $l$-ème valeur propre associée.

**Propriété : Transformée de Fourier Inverse sur Graphe**
La transformée de Fourier inverse sur graphe permet de reconstruire le signal $f$ en fonction des vecteurs propres du Laplacien :

$$f(i) = \sum_{l=0}^{N-1} \hat{f}(\lambda_l) u_l(i).$$

**Preuve (GFT) :**
La transformée de Fourier sur graphe repose sur la propriété fondamentale de

la décomposition spectrale du Laplacien. Comme $L$ est une matrice symétrique et semi-définie positive, elle possède une décomposition spectrale sous la forme :

$$L = U\Lambda U^\top,$$

où :
- $U = [u_0, u_1, \ldots, u_{N-1}]$ est une matrice dont les colonnes sont les vecteurs propres orthonormés de $L$.
- $\Lambda = \text{diag}(\lambda_0, \lambda_1, \ldots, \lambda_{N-1})$ est une matrice diagonale contenant les valeurs propres $\lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_{N-1}$ de $L$.

Pour un signal $f \in \mathbb{R}^N$, nous pouvons exprimer $f$ dans la base des vecteurs propres de $L$ :

$$f = \sum_{l=0}^{N-1} \langle f, u_l \rangle u_l = \sum_{l=0}^{N-1} \hat{f}(\lambda_l) u_l,$$

où chaque coefficient $\hat{f}(\lambda_l)$ est donné par la projection de $f$ sur le vecteur propre $u_l$, soit :

$$\hat{f}(\lambda_l) = U^\top f.$$

Cette transformation $\hat{f}(\lambda_l) = U^\top f$ fournit les coefficients de $f$ dans le domaine spectral du graphe. La transformation inverse, $f = U\hat{f}$, permet de reconstruire $f$ en combinant ses composantes dans le domaine des vecteurs propres du Laplacien. □

**Interprétation de la Transformée de Fourier sur Graphe :**
La Transformée de Fourier sur Graphe permet de passer du domaine spatial (valeurs du signal sur les sommets du graphe) au domaine spectral (fréquence sur le graphe). Voici quelques interprétations clés :

- **Fréquences sur le graphe :** Les petites valeurs propres $\lambda_l$ correspondent aux fréquences basses et sont associées aux variations lentes du signal sur le graphe. Si le signal est lisse (c'est-à-dire qu'il varie peu entre sommets connectés), la majorité de son énergie se concentre dans les coefficients $\hat{f}(\lambda_l)$ associés aux petites valeurs propres.
- **Modes de variation :** Chaque vecteur propre $u_l$ représente un mode de variation du signal sur le graphe. Pour les fréquences basses (petites valeurs propres), $u_l$ est généralement une fonction lisse sur le graphe. Pour les fréquences hautes (grandes valeurs propres), $u_l$ présente des variations rapides entre nœuds connectés.
- **Applications :** En analysant les coefficients $\hat{f}(\lambda_l)$ dans le domaine spectral, il est possible d'extraire des informations sur la structure du signal, d'appliquer des filtres (passe-bas, passe-haut), et de réaliser des tâches comme le débruitage, la compression, et la détection d'anomalies sur le graphe.

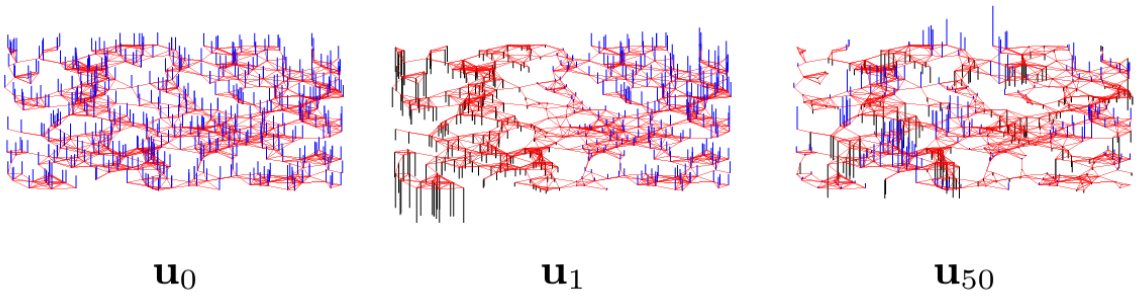



FIGURE 1 – Exemple de vecteur propre sur un graphe. $U_0$ est un vecteur propre associer à une variation quasi nulle. $U_1$ est un vecteur propre associer à une variation lente. $U_50$ est un vecteur propre associer à une variation rapide.[1]

## 2.3 Notion de dérivée et "smoothness" sur un graphe

**Définition : Dérivée partielle sur un graphe**
La dérivée partielle d'un signal $f$ au sommet $i$ par rapport à une arête $(i, j)$ connectant les sommets $i$ et $j$ est définie par :

$$\frac{\partial f}{\partial e_{i,j}} := W_{i,j} \left( f(j) - f(i) \right),$$

où $W_{i,j}$ est le poids de l'arête entre les sommets $i$ et $j$. Cette dérivée mesure la variation du signal $f$ le long de l'arête reliant $i$ et $j$.

**Définition : Gradient d'un signal sur un graphe**
Le gradient sur graphe du signal $f$ au sommet $i$ est le vecteur défini par :

$$\nabla_i f := \left( \frac{\partial f}{\partial e_{i,j}} \right)_{j \in \mathcal{N}_i},$$

où $\mathcal{N}_i$ représente l'ensemble des sommets voisins de $i$. Ce gradient regroupe les dérivées partielles de $f$ par rapport à chaque arête connectée à $i$, capturant ainsi la variation locale du signal autour du sommet $i$.

**Définition : Norme locale du gradient**
La norme locale du gradient au sommet $i$ est définie par :

$$\|\nabla_i f\|_2 := \left( \sum_{j \in \mathcal{N}_i} \left| \frac{\partial f}{\partial e_{i,j}} \right|^2 \right)^{\frac{1}{2}} = \left( \sum_{j \in \mathcal{N}_i} W_{i,j}^2 \left( f(j) - f(i) \right)^2 \right)^{\frac{1}{2}}.$$

Cette norme fournit une mesure de la variation locale du signal au sommet $i$ : elle est faible lorsque $f$ a des valeurs similaires sur les sommets voisins de $i$, et élevée dans le cas contraire.

**Définition : Forme discrète de Dirichlet pour la variation totale sur un graphe**

La forme discrète $p$-Dirichlet d'un signal $f$ est définie par :

$$S_p(f) := \frac{1}{2} \sum_{i \in V} \sum_{j \in \mathcal{N}_i} (W_{i,j} \left| f(j) - f(i) \right|^2)^{\frac{p}{2}}.$$

Pour $p = 1$, $S_1(f)$ mesure la variation totale du signal sur le graphe :

$$S_1(f) = \frac{1}{2} \sum_{i \in V} \sum_{j \in \mathcal{N}_i} W_{i,j}^{\frac{1}{2}} \left| f(j) - f(i) \right|,$$

tandis que pour $p = 2$, on a :

$$S_2(f) = \frac{1}{2} \sum_{i \in V} \sum_{j \in \mathcal{N}_i} W_{i,j} \left( f(j) - f(i) \right)^2 = f^\top L f,$$

où $S_2(f)$ est connue sous le nom de forme quadratique laplacienne du graphe.

Propriété : La smoothness se note :

$$S_2(f) = \frac{1}{2} \sum_{i \in V} \sum_{j \in \mathcal{N}_i} W_{i,j} \left( f(j) - f(i) \right)^2 = f^\top L f,$$

Pour un signal $f$ défini sur les nœuds d'un graphe, la forme quadratique de Laplacien $S_2(f)$ est définie comme suit :

$$S_2(f) = \frac{1}{2} \sum_{i \in V} \sum_{j \in \mathcal{N}_i} W_{i,j} \left[ f(j) - f(i) \right]^2$$

Étape 1 : Simplification de la Somme Double

Dans la somme double $\sum_{i \in V} \sum_{j \in \mathcal{N}_i}$, chaque arête $(i, j) \in E$ est comptée **deux fois** : une fois dans la somme pour $i$ avec $j$ comme voisin, et une autre fois pour $j$ avec $i$ comme voisin. Pour éviter ce double comptage, nous réécrivons cette somme en comptant chaque arête une seule fois :

$$S_2(f) = \sum_{(i,j) \in E} W_{i,j} \left[ f(j) - f(i) \right]^2$$

Ici, la somme est effectuée uniquement sur les arêtes $(i, j)$ du graphe $E$, ce qui garantit que chaque arête est comptée une seule fois.

Étape 2 : Développement de $f^\top L f$ en remplaçant $L$ par $D - W$

Nous avons :

$$f^\top Lf = f^\top (D - W)f = f^\top Df - f^\top Wf$$

Développons chacun de ces termes.
Calcul du Terme $f^\top Df$
Le terme $f^\top Df$ est donné par :

$$f^\top Df = \sum_{i \in V} D_{ii} f(i)^2$$

En utilisant la définition de $D_{ii} = \sum_{j \in \mathcal{N}_i} W_{i,j}$, nous avons :

$$f^\top Df = \sum_{i \in V} \left( \sum_{j \in \mathcal{N}_i} W_{i,j} \right) f(i)^2 = \sum_{i \in V} \sum_{j \in \mathcal{N}_i} W_{i,j} f(i)^2$$

Calcul du Terme $f^\top Wf$
Le terme $f^\top Wf$ est donné par :

$$f^\top Wf = \sum_{i \in V} \sum_{j \in \mathcal{N}_i} W_{i,j} f(i) f(j)$$

Étape 3 : Assemblage de $f^\top Lf$

En substituant $f^\top Df$ et $f^\top Wf$ dans l'expression de $f^\top Lf$, nous obtenons :

$$f^\top Lf = \sum_{i \in V} \sum_{j \in \mathcal{N}_i} W_{i,j} f(i)^2 - \sum_{i \in V} \sum_{j \in \mathcal{N}_i} W_{i,j} f(i) f(j)$$

Nous pouvons maintenant regrouper les termes en fonction de $(f(i) - f(j))^2$ en utilisant l'identité :

$$f(i)^2 - f(i)f(j) + f(j)^2 - f(i)f(j) = (f(i) - f(j))^2$$

En appliquant cela à chaque terme, nous obtenons :

$$f^\top Lf = \sum_{(i,j) \in E} W_{i,j} \left[ f(i)^2 - 2f(i)f(j) + f(j)^2 \right] = \sum_{(i,j) \in E} W_{i,j} \left[ f(j) - f(i) \right]^2$$

ce qui est exactement la définition de $S_2(f)$. □

**Définition :** La **semi-norme** associée à la forme quadratique de Laplacien $S_2(f)$ pour un signal $f$ est définie par :

$$\|f\|_L := \|L^{\frac{1}{2}} f\|_2 = \sqrt{f^\top Lf} = \sqrt{S_2(f)}.$$

**Interprétation :** Il est important de noter que la forme quadratique $S_2(f)$ est égale à zéro si et seulement si $f$ est constant sur tous les sommets du graphe (ce qui explique pourquoi $\|f\|_L$ est seulement une semi-norme). Plus généralement, $S_2(f)$ est petite lorsque le signal $f$ a des valeurs similaires aux sommets voisins connectés par une arête de grand poids ; c'est-à-dire lorsque $f$ est lisse.

**Définition :** En revenant aux valeurs propres et vecteurs propres du Laplacien du graphe, le **théorème de Courant-Fischer** nous permet de les définir de manière itérative à l'aide du quotient de Rayleigh comme suit :

$$\lambda_0 = \min_{f \in \mathbb{R}^N, \|f\|_2 = 1} \left\{ f^\top L f \right\},$$

et pour $l = 1, 2, \ldots, N-1$,

$$\lambda_l = \min_{\substack{f \in \mathbb{R}^N \\ \|f\|_2 = 1 \\ f \perp \mathrm{span}(u_0, \ldots, u_{l-1})}} \left\{ f^\top L f \right\},$$

où le vecteur propre $u_l$ est le minimiseur du $l$-ème problème.

Nous allons démontrer ce théorème par réccurence.

**Initialisation :** la Plus Petite Valeur Propre $\lambda_1$

Pour déterminer la plus petite valeur propre $\lambda_1$, nous considérons le problème suivant :

$$\lambda_1 = \min_{\substack{x \in \mathbb{R}^n \\ \|x\| = 1}} x^\top A x.$$

— Puisque $A$ est symétrique, elle possède une base orthonormée de vecteurs propres $v_1, v_2, \ldots, v_n$ associés aux valeurs propres $\lambda_1, \lambda_2, \ldots, \lambda_n$.

— Pour tout vecteur $x \in \mathbb{R}^n$, on peut écrire $x$ comme une combinaison linéaire des vecteurs propres de $A$ :

$$x = \sum_{i=1}^{n} \alpha_i v_i,$$

où $\alpha_i = v_i^\top x$ sont les coordonnées de $x$ dans la base des vecteurs propres de $A$.

— En imposant la condition $\|x\| = 1$, nous avons :

$$\|x\|^2 = x^\top x = \left( \sum_{i=1}^{n} \alpha_i v_i \right)^\top \left( \sum_{j=1}^{n} \alpha_j v_j \right) = \sum_{i=1}^{n} \alpha_i^2 = 1.$$

— La forme quadratique $x^\top A x$ s'écrit alors en fonction des valeurs propres de $A$

et des coefficients $\alpha_i$ :

$$x^\top Ax = \left(\sum_{i=1}^{n} \alpha_i v_i\right)^\top A \left(\sum_{j=1}^{n} \alpha_j v_j\right) = \sum_{i=1}^{n} \alpha_i^2 \lambda_i.$$

— Puisque $\sum_{i=1}^{n} \alpha_i^2 = 1$ et que les valeurs propres $\lambda_i$ sont ordonnées de manière croissante ($\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$), l'expression $\sum_{i=1}^{n} \alpha_i^2 \lambda_i$ est minimisée lorsque tout le poids est concentré sur $\lambda_1$. En d'autres termes, le minimum est atteint lorsque $x = v_1$ (c'est-à-dire lorsque $\alpha_1 = 1$ et $\alpha_i = 0$ pour $i \neq 1$).
— Dans ce cas, on a :

$$x^\top Ax = \lambda_1.$$

Nous avons donc montré que :

$$\lambda_1 = \min_{\substack{x \in \mathbb{R}^n \\ \|x\|=1}} x^\top Ax.$$

**Réccurence :**
Pour montrer que la $k$-ième valeur propre $\lambda_k$ est donnée par :

$$\lambda_k = \min_{\substack{x \in \mathbb{R}^n \\ \|x\|=1 \\ x \perp v_1, \ldots, x \perp v_{k-1}}} x^\top Ax,$$

nous allons imposer la contrainte d'orthogonalité. Il suffit de suivre le même raisonnement que pour l'initialisation en considérant que x est orthogonal aux vecteurs précédemment trouvé. ainsi ses composante selon $v_{i \in [0,k-1]}$ sont nulles et donc pour minimiser il faut $x = v_k$.

Ainsi, nous avons montré que :

$$\lambda_k = \min_{\substack{x \in \mathbb{R}^n \\ \|x\|=1 \\ x \perp v_1, \ldots, x \perp v_{k-1}}} x^\top Ax.$$

□

**Interprétation :** Ce théorème permet de mieux comprendre la signification des différentes fréquences d'un graphe. La plus petite valeur propre est associée au signal le plus lisse possible sur le graphe, qui correspond à un signal constant. Ensuite, le second vecteur propre (celui associé à la deuxième plus petite valeur propre) représente le signal minimisant les variations tout en étant orthogonal au vecteur constant. Cela implique que ce vecteur propre prend des valeurs presque constantes sur les deux plus grands clusters du graphe, avec une valeur proche de $-a$ sur un groupe et $a$ sur l'autre, afin de respecter l'orthogonalité avec le premier vecteur propre (le signal constant), tout en minimisant les variations.
Ce processus se poursuit pour les vecteurs propres suivants, chaque vecteur propre ajoutant une fréquence plus élevée au signal tout en minimisant les variations sous la contrainte d'orthogonalité avec les vecteurs précédents. Sur un graphe en ligne, ce phénomène peut être comparé aux modes propres d'une corde vibrante, comme dans l'expérience de Melde, où chaque mode représente une fréquence distincte et une forme particulière d'oscillation.

# 3 Application à un réseau de neurones

Dans cette partie, nous allons utiliser la GFT afin d'étudier le sur-apprentissage d'un réseau de neurones. **L'entièreté des figures seront disponibles dans un jupyter envoyé avec le rapport.**

La GFT, en décomposant un signal en ses composantes fréquentielles propres au graphe, offre un outil puissant pour analyser les activations des neurones dans un réseau. En effet, les valeurs propres de la matrice de Laplacien associée au graphe permettent de définir une notion de fréquence : les petites valeurs propres correspondent aux basses fréquences, c'est-à-dire à des variations douces sur le graphe, tandis que les grandes valeurs propres sont associées aux hautes fréquences, correspondant à des variations rapides et localisées.

Dans le cadre des réseaux de neurones, les activations des neurones peuvent être considérées comme un signal défini sur le graphe représentant la structure du réseau. Lorsque le réseau est bien entraîné et généralise correctement, les activations des neurones présentent généralement une structure cohérente, capturant les principales caractéristiques des données d'entrée. Ce type de signal est principalement composé de basses fréquences dans le domaine spectral, reflétant des variations globales et régulières sur le graphe.

En revanche, lorsqu'un réseau de neurones sur-apprend les données d'entraînement, il ajuste ses poids pour s'adapter étroitement à ces données spécifiques, au détriment de la généralisation. Cela se traduit par des activations de neurones beaucoup plus complexes et localisées, capturant des variations spécifiques à l'échantillon d'entraînement. Ces activations présentent alors une composante spectrale significativement plus importante dans les hautes fréquences, indiquant des variations rapides et moins cohérentes sur le graphe.

Ainsi, en analysant la distribution spectrale des activations des neurones à l'aide de la GFT, il devient possible de détecter le sur-apprentissage. Une concentration anormalement élevée de la composante spectrale dans les hautes fréquences constitue un indicateur fort de sur-apprentissage. Cette approche permet donc de relier directement la structure spectrale des activations neuronales à la capacité du réseau à généraliser ses prédictions à des

données nouvelles.

En résumé, la GFT fournit un cadre mathématique rigoureux pour analyser le comportement des réseaux de neurones dans le domaine spectral, en mettant en évidence les différences entre un réseau bien entraîné et un réseau ayant sur-appris.

## 3.1 Cas pratique : Autoencodeur de signaux

### 3.1.1 Démarche du cas pratique

Afin d'illustrer ce phénomène, nous allons utiliser des autoencodeurs appliqués à des signaux synthétiques. Ces signaux, composés de 128 points, sont générés à partir d'un modèle combinant une tendance linéaire et une composante de saisonnalité.

Un autoencodeur est un type de réseau de neurones conçu pour apprendre une représentation compacte des données en entrée. Il est constitué de deux parties principales : un encodeur et un décodeur. L'encodeur transforme les données d'entrée en une représentation de plus faible dimension (appelée la couche latente), tandis que le décodeur reconstruit les données d'origine à partir de cette représentation. L'objectif de l'entraînement est de minimiser la différence entre l'entrée et la sortie reconstruite, permettant ainsi à l'autoencodeur de capturer les structures essentielles des données.

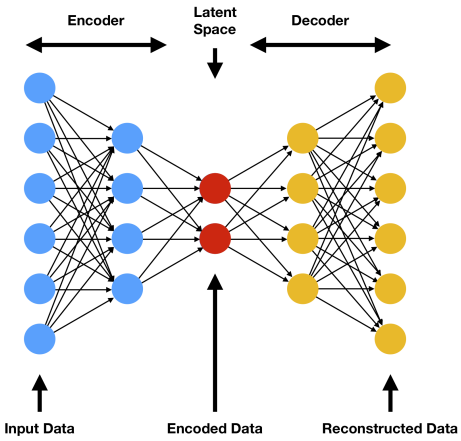



FIGURE 2 – Schéma de la structure d'un autoencodeur. [lien image]

Nous allons entraîner deux autoencodeurs avec la même architecture. Le premier sera entraîné sur seulement deux données, tandis que le second sera entraîné sur un ensemble de 50 000 données. Nous nous attendons à ce que le premier autoencodeur sur-apprenne les deux exemples spécifiques, tandis que le second, grâce à la richesse de son ensemble d'entraînement, parvienne à généraliser.

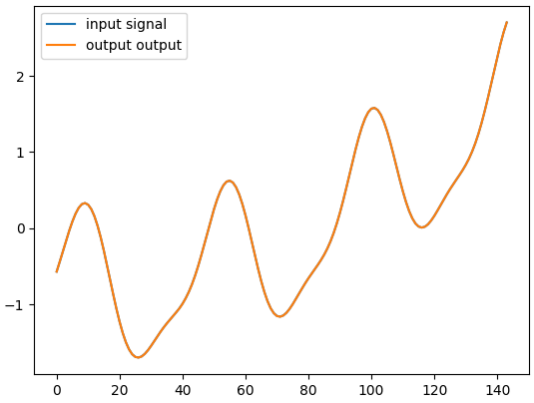
Pour analyser la structure interne de ces réseaux, nous représentons les neurones comme des nœuds d'un graphe, et les poids des couches linéaires comme les liens entre ces nœuds. Ainsi, ce graphe reflète l'organisation interne des connexions dans le réseau. Pour un signal donné en entrée, nous calculons les activations de tous les neurones du réseau, c'est-à-dire les valeurs prises par chaque neurone lorsque ce signal est traité. Ces activations constituent un signal défini sur le graphe du réseau.

Enfin, nous appliquons la GFT au signal d'activation des deux modèles pour étudier leurs composantes fréquentielles respectives. Cette analyse nous permettra de comparer

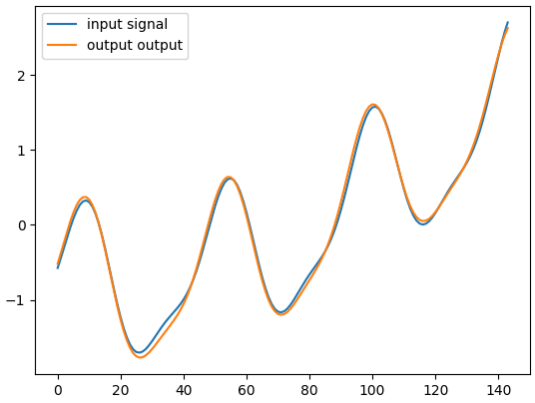
la répartition spectrale des activations entre le modèle ayant sur-appris et celui qui a généralisé correctement.

### 3.1.2 Résultat du cas pratique

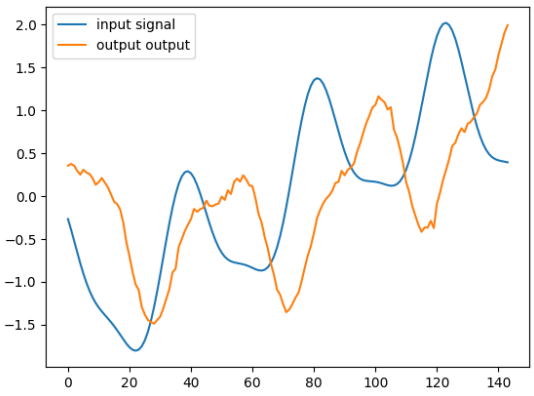
Dans un premier temps, nous avons entraîné nos deux modèles et vérifié que l'un a bien sur-appris tandis que l'autre a généralisé correctement. Pour cela, nous avons comparé la reconstruction d'un signal présent dans leur dataset d'entraînement et celle d'un signal absent de leur dataset d'entraînement.



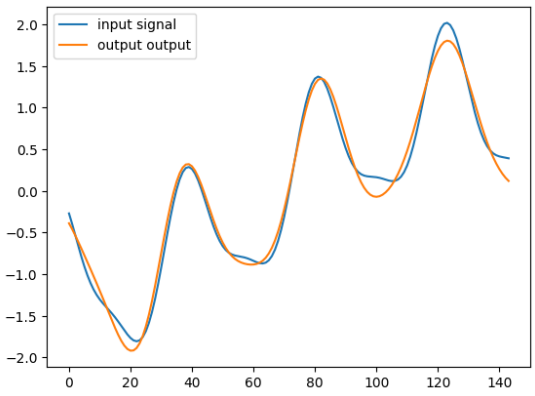

(a) Reconstruction d'un signal d'entraînement par le modèle ayant sur-appris.




(b) Reconstruction d'un signal d'entraînement par le modèle généralisant.




(c) Reconstruction d'un signal de test par le modèle ayant sur-appris.




(d) Reconstruction d'un signal de test par le modèle généralisant.

FIGURE 3 – Comparaison des reconstructions par les deux modèles sur des signaux d'entraînement et de test.

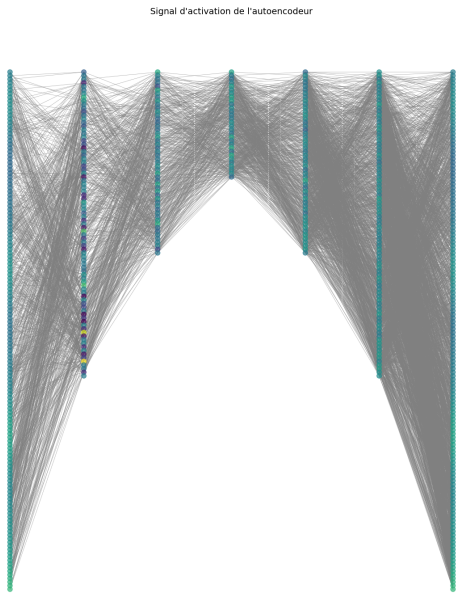
Nous observons, comme attendu, que le modèle ayant sur-appris (figure 3a) restitue parfaitement un signal présent dans son dataset d'entraînement. Cependant, il est incapable de reconstruire fidèlement un signal absent de son dataset, comme le montre figure 3c.

À l'inverse, le second modèle, qui a été entraîné sur un grand nombre de données, parvient à reconstruire de manière satisfaisante les deux signaux (figure 3b et figure 3d).

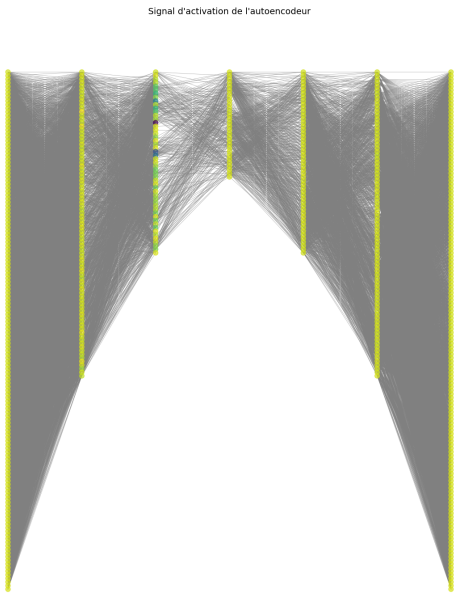
Cela reflète sa capacité à généraliser, capturant les caractéristiques globales des données plutôt que de s'ajuster uniquement aux exemples spécifiques de son dataset.

Ces figures nous démontrent bien que nous avons un modèle ayant sur-appris et un modèle généralisant.

Ensuite, nous calculons les activations des neurones de nos deux modèles pour un même signal d'entrée. Cela nous permet d'obtenir les graphes à signal suivants :




(a) Graphe des activations pour le modèle ayant sur-appris.

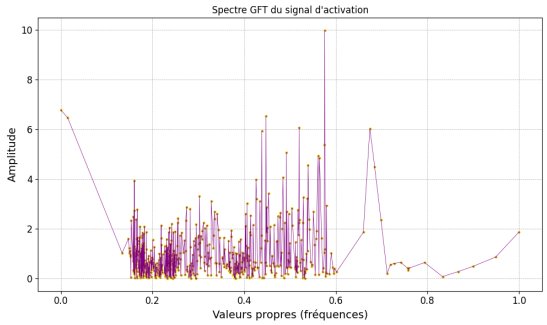



(b) Graphe des activations pour le modèle généralisant.

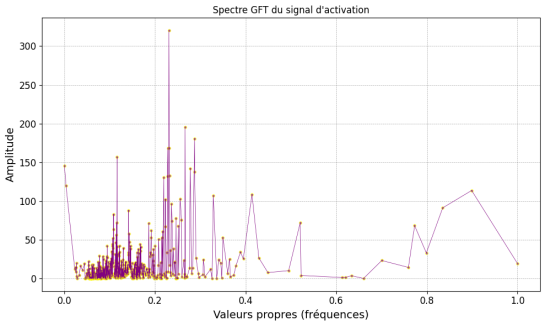
FIGURE 4 – Comparaison des graphes de signal pour un même signal d'entrée, entre le modèle ayant sur-appris et le modèle généralisant.

Ces deux graphes montrent clairement que le signal d'activation de l'autoencodeur ayant sur-appris (figure 4a) varie de manière beaucoup plus importante que celui de l'autoencodeur généralisant (figure 4b). Cette observation reflète le fait que le premier modèle capture des détails spécifiques aux exemples d'entraînement, tandis que le second, grâce à sa capacité à généraliser, produit un signal plus régulier.

Enfin, nous avons calculé la GFT du signal d'activation pour chacun des modèles. Les résultats sont présentés ci-dessous :




(a) Spectre GFT du signal d'activation pour le modèle ayant sur-appris.




(b) Spectre GFT du signal d'activation pour le modèle généralisant.

FIGURE 5 – Comparaison des spectres GFT des signaux d'activations entre un modèle ayant sur-appris et un modèle généralisant.

Nous observons que le spectre du signal d'activation de l'autoencodeur ayant sur-appris (figure 5a) présente une proportion nettement plus importante de composantes en haute fréquence. Cela reflète la nature complexe et localisée des activations, due à un ajustement excessif aux données d'entraînement spécifiques.

En revanche, le spectre du modèle généralisant (figure 5b) est dominé par des composantes en basse fréquence. Cela indique que ses activations sont plus régulières et cohérentes sur le graphe, reflétant une meilleure capacité de généralisation et une moindre sensibilité aux détails spécifiques des données d'entraînement.

Nous avons faiT le test pour d'autres signaux et nous obtenons des résultats similaires.

À travers cette analyse, nous avons montré que la Transformée de Fourier sur Graphe (GFT) constitue un outil puissant pour caractériser le comportement interne des réseaux de neurones. En particulier, elle nous a permis d'identifier des différences significatives entre les modèles ayant sur-appris et ceux généralisant correctement, en analysant leurs spectres d'activations neuronales.

Le modèle ayant sur-appris présente un spectre dominé par des composantes de haute fréquence, traduisant des activations complexes et spécifiques aux données d'entraînement. À l'inverse, le modèle généralisant affiche un spectre principalement composé de basses fréquences, indiquant des activations plus régulières et globales. Ces observations confirment que la GFT est non seulement une approche théorique intéressante, mais également un outil pratique pour diagnostiquer le sur-apprentissage dans les réseaux de neurones.

# 4 Conclusion

Dans ce rapport, nous avons étudié l'application de la Transformée de Fourier sur Graphe (GFT) pour analyser le comportement des réseaux de neurones et, en particulier, pour identifier le phénomène de sur-apprentissage. En utilisant un autoencodeur comme modèle d'étude, nous avons démontré que la GFT permet de mettre en évidence des différences spectrales significatives entre un modèle ayant sur-appris et un modèle généralisant correctement. Ces résultats confirment que la GFT est un outil puissant pour caractériser les activations des neurones dans un réseau sous l'angle spectral.

En analysant les spectres GFT des activations neuronales, nous avons observé que les

modèles sur-appris présentent une forte proportion de composantes en haute fréquence, traduisant des activations localisées et spécifiques aux données d'entraînement. À l'inverse, les modèles généralisants affichent des spectres dominés par des basses fréquences, reflétant des activations plus globales et cohérentes, témoins d'une meilleure capacité de généralisation.

Cependant, bien que cette approche soit théoriquement solide et efficace dans notre cadre expérimental, elle présente également certaines limites. En particulier, pour des modèles de grande taille tels que les architectures profondes modernes (par exemple GPT ou BERT), la construction du graphe devient un défi computationnel majeur. En effet, le nombre de nœuds du graphe correspond au nombre total de neurones du modèle, ce qui rend l'approche difficilement applicable pour des réseaux contenant des millions, voire des milliards de paramètres. Dans notre cas, avec un modèle de 590 neurones, la méthode était encore réalisable, mais son extension à des modèles de grande échelle nécessiterait des adaptations ou des simplifications.

### Perspectives et travaux futurs

Cette étude ouvre la voie à plusieurs perspectives de recherche. Une première direction pourrait consister à appliquer cette méthode à d'autres architectures de réseaux de neurones, comme les réseaux convolutifs ou les réseaux récurrents, pour évaluer si les conclusions obtenues restent valables dans des contextes différents.

Une seconde piste serait de développer un outil capable de déceler le sur-apprentissage en temps réel pendant l'entraînement. En analysant dynamiquement les spectres GFT des activations, il pourrait être possible de concevoir un nouveau critère d'arrêt basé sur la proportion des hautes fréquences, permettant ainsi de prévenir le sur-apprentissage avant qu'il ne se manifeste pleinement.

Enfin, des recherches pourraient être menées pour rendre cette méthode plus applicable à grande échelle. Cela pourrait inclure des techniques pour approximer la GFT ou des approches permettant de réduire la taille du graphe, tout en conservant les informations essentielles sur la structure et les activations du réseau.

En conclusion, bien que cette méthode présente des défis, elle offre un cadre innovant et rigoureux pour explorer et comprendre le comportement des réseaux de neurones sous un nouvel angle. Avec des améliorations et des extensions, elle pourrait devenir un outil précieux pour la recherche en apprentissage automatique et en optimisation des réseaux de neurones.

## Références

[1] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs : Extending high-dimensional data analysis to networks and other irregular domains. *arXiv preprint arXiv :1211.0053*, 2013.