Expliquer la classification d'expression de gènes par la méthode des gradients intégrés

Myriam BONTONOU[1,2] Jean-Michel ARBONA[1] Benjamin AUDIT[2] Pierre BORGNAT[2]

[1]Laboratoire de Biologie et Modélisation de la Cellule, ENS de Lyon, CNRS, UMR5239, Inserm U1293, Univ Lyon, Lyon, France

[2]Univ Lyon, ENS de Lyon, CNRS, Laboratoire de Physique, F-69342 Lyon, France

Résumé – Au cours des dernières années, divers réseaux de neurones ont été entraînés à différencier des tissus sains et pathologiques avec précision en utilisant des données d'expression de gènes. Ces modèles prédictifs contiennent très probablement des informations sur les mécanismes moléculaires induisant un état pathologique. Il est donc utile d'expliquer comment leurs décisions sont prises. Plusieurs méthodes permettent d'évaluer l'importance de la contribution de chacun des gènes d'un échantillon à sa classification comme sain ou pathologique. Cependant, un tel classement des gènes n'est pas nécessairement directement interprétable. Dans ce contexte, nous explorons des métriques évaluant la pertinence des classements générés par la méthode des gradients intégrés sur des données d'expression de gènes issues de l'Atlas du Génome du Cancer et sur des données simulées.

Abstract – Over the past years, various neural networks have been trained to accurately differentiate healthy and pathological tissues from gene expression data. These predictive models most likely contain information about the molecular mechanisms inducing a pathological state. It is therefore useful to explain how their decisions are made. Several methods can be used to assess the importance of the contribution of each gene in a sample to its classification as healthy or pathological. However, such an ordering of genes is not necessarily directly interpretable. In this context, we explore metrics to assess the relevance of orderings generated by the integrated gradient method on gene expression data from the Cancer Genome Atlas and on simulated data.

1 Introduction

Pour prévenir, diagnostiquer et traiter des maladies telles que les cancers, il est essentiel de mieux comprendre le fonctionnement du métabolisme cellulaire. Afin de déterminer les mécanismes moléculaires favorisant leur apparition, il est possible de comparer des données génomiques obtenues sur un nombre conséquent de tissus sains ou pathologiques. De nombreuses études ont analysé ces données par des méthodes statistiques afin d'identifier des marqueurs cliniques ; voir par exemple la revue [1] sur le projet The Cancer Genome Atlas (TCGA).

Pour étudier des relations moléculaires plus complexes, il a récemment été proposé de formuler cette question scientifique comme un problème d'apprentissage supervisé sur des données tabulaires (éventuellement vues comme des graphes), par exemple pour la prédiction des phénotypes à partir de données d'expression de gènes [2], ou à partir de modalités multiples [3]. Les décisions issues de réseaux de neurones, entraînés à résoudre ces problèmes supervisés, sont alors expliqués à l'aide de méthodes attribuant un score aux variables d'entrées (e.g. gènes) en fonction de leur contribution [4], [5]. Les gènes ayant une forte contribution sont parfois interprétés comme des marqueurs des phénomènes étudiés, identifiant par exemple différents types de cancer [2].

Notre objectif est d'étudier la pertinence de la hiérarchie obtenue (i.e. classement des gènes par ordre d'importance pour le réseau). A cette fin, nous proposons de calculer systématiquement deux métriques : l'écart de prédiction lié aux variables importantes, noté Prediction Gap on Important features (PGI), et l'écart de prédiction lié aux variables peu importantes, noté Prediction Gap on Unimportant features (PGU) (section 2.6). Pour illustrer l'utilité de ces métriques, nous entraînons des réseaux de neurones à classifier des échantillons cancéreux et des échantillons normaux à partir de données d'expression de gènes. La contribution de chaque gène est estimée à partir de la méthode des gradients intégrés (IG) [5]. Les mêmes expériences sont répétées sur des données simulées où les explications attendues sont contrôlées grâce à un modèle génératif permettant de générer des données appartenant à différentes classes avec des informations discriminantes plus ou moins dispersées dans les variables d'entrées.

Les principales contributions sont[1] : (i) un modèle générant des données avec des informations dispersées et une hétérogénéité contrôlée (section 2.2), (ii) un protocole quantifiant la fiabilité des explications générées (sections 2.5 et 2.6), et (iii) l'application sur des données réelles et simulées (section 3).

2 Matériel et Méthodes

Cette section contient une description des données (sections 2.1 et 2.2), de la méthode d'apprentissage (sections 2.3 et 2.4) et de la méthode d'explicabilité (section 2.5) utilisées lors des expériences. Les métriques permettant d'évaluer les explications obtenues sont définies à la section 2.6.

2.1 Données d'expression de gènes (TCGA)

Le programme TCGA a catalogué des données de génomique liées à plusieurs types de cancers[2]. Ces données ont déjà été utilisées pour entraîner des algorithmes supervisées à distinguer les échantillons cancéreux des échantillons normaux [2].

Ici, deux tâches de classification binaires « cancer versus normal » sont considérées sur des données d'expression de

[1]Code : `https://github.com/mbonto/XAI_for_genomics`

[2]Données accessibles sur `https://portal.gdc.cancer.gov/`

gènes. La première tâche, nommée BRCA, se focalise sur la prédiction de cancers du sein. La deuxième tâche, KIRC, sur celle de carcinomes rénaux. Les données BRCA et KIRC contiennent respectivement 1097/534 échantillons prélevés dans des tumeurs primaires et 113/72 échantillons normaux prélevés dans les tissus solides environnants. Leurs échantillons contiennent les expressions de 58274/58233 gènes. Dans les données disponibles, la somme des expressions de chaque échantillon vaut un million. Afin de ramener tous les niveaux d'expression sur une même échelle, nous les ajustons en prenant le logarithme de leurs valeurs.

2.2 Simuler des données d'expression

Des données sont simulées à l'aide du modèle génératif probabiliste décrit dans notre travail précédent [6]. Ce modèle impose une relation hiérarchique entre les classes et les variables initiales en suivant le schéma du modèle Latent Dirichlet Allocation (LDA) [7].

Dans ce modèle, il y a trois niveaux d'information : les *classes* qui activent des *groupes* contrôlant la distribution d'un petit nombre de *variables*. Les variables exprimées au sein d'un même groupe ont des expressions corrélées. Par analogie avec les données réelles, les classes représentent le phénotype cancéreux ou normal des échantillons. Les variables représentent les gènes exprimés. Les groupes modélisent des voies métaboliques différentes imposant des expressions corrélées au sein des groupes.

Soit un nombre de classes C, un nombre de variables V et un nombre de groupes G. Pour chaque classe c, un a priori sur l'activation relative des groupes est fourni par $\boldsymbol{\alpha}_c \in \mathbb{R}_+^G$. Les variables contrôlées par chaque groupe g sont définies a priori par $\boldsymbol{\eta}_g \in \mathbb{R}_+^V$. La proportion relative de variables apparaissant dans un groupe est déterminée par un tirage aléatoire : $\boldsymbol{\beta}_g \sim \text{Dirichlet}(\boldsymbol{\eta}_g)$. Pour simuler un exemple e appartenant à la classe c, trois étapes sont nécessaires. Tout d'abord, les groupes activés dans cet exemple sont tirés : $\boldsymbol{\theta}_e \sim \text{Dirichlet}(\boldsymbol{\alpha}_c)$. Ensuite, un grand nombre T de tirages de variables est effectué. Pour un tirage t, un groupe g est associé au tirage $g_t \sim \text{Multinomial}(\boldsymbol{\theta}_e)$ et une variable v est tirée par rapport au groupe activé $v_t \sim \text{Multinomial}(\boldsymbol{\beta}_{g_t})$. Enfin, la valeur finale associée à chaque variable de l'exemple e est son nombre de tirage.

Afin d'évaluer précisément l'influence des variables sur les classes, les classes influencent des ensembles distincts de groupes et les groupes contrôlent un nombre restreint de gènes.

Dans cet article, trois simulations sont générées avec chacune 2 classes et 1200 exemples constitués de 50000 variables. 5000 groupes contrôlent l'activation de 10 variables différentes. Dans SimuA, les classes contiennent chacune 600 exemples. Chaque classe est caractérisée par la sur-activation de 500 groupes différents, soit 10% des variables. Dans SimuB, la première classe est constituée de 3 sous-classes de 300 exemples. La seconde classe contient 300 exemples. Dans SimuC, la première classe est constituée de 5 sous-classes de 200 exemples. La seconde classe contient 200 exemples. Les sous-classes sont aussi caractérisées par 500 groupes sur-activés.

Pour plus de détail, si le groupe g n'est pas sur-activé dans la classe c $\boldsymbol{\alpha}_c[g] = 2$ et sinon $\boldsymbol{\alpha}_c[g] = 8$. Si la variable v n'est pas contrôlée par le groupe g $\boldsymbol{\eta}_g[v] = 0$ et sinon $\boldsymbol{\eta}_g[v] = 5$. On s'assure que les proportions relatives des variables exprimées au sein d'un groupe $\boldsymbol{\beta}_g$ dépassent 0.001 (choix arbitraire). Si ce n'est pas le cas, un nouveau tirage aléatoire est effectué. On utilise $T = 1500000$, ce qui correspond à l'ordre de grandeur des données expérimentales issues de TCGA.

2.3 Entraîner un réseau à classifier

Considérons une tâche de classification avec C classes sur des exemples $\mathbf{x}$ contenant V variables. Dans le contexte de la génomique, cette tâche peut viser à différencier des échantillons normaux et des échantillons cancéreux à partir de données d'expression de gènes (une variable = l'expression d'un gène).

Pour résoudre la tâche de classification, nous utilisons des réseaux de neurones peu profonds $f(\cdot)$ dotés d'une architecture en perceptrons multicouches. Ils sont entraînés sur 60% des données et testés sur les 40% restants. Pour sélectionner les hyperparamètres du modèle, une validation stratifiée croisée est effectuée en découpant les données d'entraînement en 4 ensembles. Les hyperparamètres testés sont le nombre de couches cachées (1 ou 2) et le nombre d'éléments par couche (10, 20 ou 40). Pour faciliter l'apprentissage, les variables (expression des gènes) sont centrées par rapport aux moyennes calculées sur les données d'entraînement. Les variables des simulations sont également réduites. Le code utilise Pytorch [8].

2.4 Évaluer une performance de classification

Dans nos expériences, le nombre d'exemples est déséquilibré entre les classes. La performance du réseau est donc évaluée par une métrique ajustée, appelée *exactitude équilibrée*, moyennant les scores de rappels obtenus sur chaque classe c :

$$\text{Rappel}_c = \frac{\text{Nombre d'exemples corrects attribués à c}}{\text{Nombre d'exemples appartenant à c}}\,, \quad (1)$$

et

$$\text{Exactitude équilibrée} = \frac{\sum_c \text{Rappel}_c}{C}\,. \quad (2)$$

2.5 Hiérarchiser les variables avec la méthode des gradients intégrés

Les variables contenues dans les données peuvent être classées par ordre d'importance pour un réseau [4]. Cette hiérarchie peut être obtenue pour chaque exemple indépendamment (cadre local) ou pour une classe entière (cadre global).

Au niveau de chaque exemple, un score $\boldsymbol{\phi}_i$ est attribué à chaque variable i par la méthode des gradients intégrés (IG) [5]. Pour simplifier les notations, notons $f(\mathbf{x})$ la sortie du réseau associée à la classe de l'exemple $\mathbf{x}$ et $\mathbf{x}_i$ la valeur de la $i^{\text{ème}}$ variable. Étant donné une référence $\mathbf{x}' \in \mathbb{R}^V$, le score $\boldsymbol{\phi}_i$ attribué à la variable i vaut :

$$\boldsymbol{\phi}_i(\mathbf{x}) = (\mathbf{x}_i - \mathbf{x}'_i) \int_{\alpha=0}^{1} \left.\frac{\partial f(\mathbf{y})}{\partial \mathbf{y}_i}\right|_{\mathbf{y}=\mathbf{x}'+\alpha(\mathbf{x}-\mathbf{x}')} d\alpha\,. \quad (3)$$

La somme des attributions est égale à la différence des sorties du réseau appliqué à l'exemple $\mathbf{x}$ et à la référence $\mathbf{x}'$ [5] :

$$\sum_{i=1}^{V} \boldsymbol{\phi}_i(\mathbf{x}) = f(\mathbf{x}) - f(\mathbf{x}')\,. \quad (4)$$

TABLE 1 : Métriques d'explicabilité calculées sur des données réelles (BRCA, KIRC) et des données simulées (SimuA, SimuB, SimuC). Les modèles entraînés sont des perceptrons avec 1 couche cachée. Tous les résultats sont exprimés en %.

Dataset	BRCA	KIRC	SimuA	SimuB	SimuC
Exactitude équilibrée (↑)	99.6 ± 0.1	99.6 ± 0.7	100	100	100
Local					
PGI (↑)	98.7 ± 0.3	98.4 ± 0.5	94.9 ± 0.2	97.4 ± 0.2	97.4 ± 0.2
PGU (↓)	0.9 ± 0.2	1.0 ± 0.2	5.5 ± 0.2	6.7 ± 0.3	5.7 ± 0.2
FA (↑)	-	-	79.9 ± 0.3	69.1 ± 0.6	67.8 ± 0.7
Global					
PGI (↑)	98.2 ± 0.3	98.0 ± 0.5	92.1 ± 0.3	96.4 ± 0.2	96.3 ± 0.3
PGU (↓)	1.6 ± 0.3	1.3 ± 0.3	8.3 ± 0.3	9.3 ± 0.6	7.7 ± 0.2
FA (↑)	-	-	100.0 ± 0.0	94.6 ± 0.5	90.5 ± 0.7

Pour une classe, un score commun peut être calculé comme moyenne des scores attribués à tous les exemples. Cependant, comme les amplitudes des scores varient beaucoup d'un exemple à l'autre, nous normalisons d'abord les scores de chaque exemple. Ainsi, pour une classe c avec M exemples, avec $\|\cdot\|$ la norme euclidienne, le score est

$$\boldsymbol{\phi}_i^c = \frac{1}{M} \sum_{k=1}^{M} \frac{\boldsymbol{\phi}_i(\mathbf{x}^k)}{\|[\boldsymbol{\phi}_1(\mathbf{x}^k), \ldots, \boldsymbol{\phi}_V(\mathbf{x}^k)]\|} \tag{5}$$

Ici, la référence $\mathbf{x}'$ est choisie égale à la moyenne des échantillons normaux de l'ensemble d'entraînement pour les données réelles et à la moyenne de la classe homogène pour celles simulées. Les scores sont calculés pour les classes restantes (respectivement classe cancer ou classes hétérogènes pour les données simulées). Le code utilise la bibliothèque Captum [9].

2.6 Évaluer l'explicabilité des variables

Afin d'évaluer la pertinence des hiérarchies (i.e. variables classées selon les scores $\boldsymbol{\phi}$ ou $\boldsymbol{\phi}^c$), plusieurs métriques inspirées de travaux sur des données tabulaires [10], sont étudiées ici.

Les valeurs des variables d'un exemple $\mathbf{x}$ de la classe c peuvent être remplacées par les valeurs de la référence $\mathbf{x}'$ les unes après les autres, en suivant l'ordre de la hiérarchie locale propre à cet exemple ($\boldsymbol{\phi}$) ou de la hiérarchie globale propre à sa classe ($\boldsymbol{\phi}^c$). Notons $\tilde{\mathbf{x}}_p$ l'exemple $\mathbf{x}$ contenant p variables modifiées. Par exemple, $\tilde{\mathbf{x}}_0 = \mathbf{x}$, aucune variable n'est modifiée. L'écart de prédiction pour p variables modifiées est donné par $e_p = \max(f(\mathbf{x}) - f(\tilde{\mathbf{x}}_p), 0)$. Nous appelons *Prediction Gap (PG)* l'aire sous la courbe e_p en fonction de p :

$$\text{PG} = \sum_{p=1}^{V} \frac{\max(f(\mathbf{x}) - f(\tilde{\mathbf{x}}_p), 0)}{V} \ . \tag{6}$$

Le PG est compris entre 0 et 1 car $f(\mathbf{x}) \leq 1$ donc $0 \leq e_p \leq 1$.

Prediction Gap on Important features (PGI) Le PG est calculé en modifiant les variables les plus importantes en premier (score décroissant). Si l'information est concentrée sur un petit nombre de variables, les prédictions du réseau seront très vite erronées. Dans ce cas, le PGI sera proche de 1.

Prediction Gap on Unimportant features (PGU) Le PG est calculé en modifiant les variables les moins importantes en premier. Si les variables les plus importantes sont suffisantes pour prédire correctement, le PGU sera proche de 0.

Le nombre de variables importantes pouvant être modifiées avant de dégrader la performance est estimé par $(1 - \text{PGI}) \times V$. Le nombre de variables peu importantes pouvant être modifiées est quant à lui estimé par $(1 - \text{PGU}) \times V$.

Feature Agreement (FA) Dans le cas des données simulées, la métrique *Feature Agreement* mesure la concordance de la hiérarchie avec les variables simulées réellement importantes pour classer un exemple. Soit un ensemble $\mathcal{E}_r$ de variables réellement importantes et un ensemble $\mathcal{E}_i$ constitué des $|\mathcal{E}_r|$ variables les plus importantes identifiées par IG,

$$\text{FA} = \frac{|\mathcal{E}_r \cap \mathcal{E}_i|}{|\mathcal{E}_r|}. \tag{7}$$

3 Résultats

Les résultats présentés dans la Table 1 sont les moyennes et les écarts types des métriques obtenues sur 10 entraînements du même modèle. A chaque fois, les métriques sont calculées sur les exemples du jeu test correctement classifiés.

Métriques locales Les données simulées permettent d'estimer la pertinence des variables pour un réseau donné. Chaque exemple simulé contient 50000 variables dont 10000 informatives ; ainsi 20% des variables sont informatives. Supprimer ces variables revient à supprimer toute information des données. Si ces variables étaient identifiées comme les plus importantes dans la hiérarchie, les PGU seraient égaux à 20% ou inférieurs si un plus petit nombre de variables suffit au réseau pour faire une prédiction correcte. Les PGI seraient égaux ou supérieurs à 80%, si la modification des premières variables les plus informatives induisait une prédiction erronée. Les résultats sont cohérents avec cela (Table 1). Par exemple dans le cas de SimuA, en moyenne, d'après le PGU, toutes les variables sauf les 5.5% les plus importantes peuvent être modifiées sans dégrader la prédiction ; 2750 variables non modifiées suffisent pour une prédiction correcte. Lorsque 5.1% des variables les plus importantes sont modifiées alors que les autres valeurs ne le sont pas, la prédiction est erronée ; le nombre de variables importantes pouvant être modifiées est estimé à 2550. D'après la métrique FA, les 10000 variables informatives n'apparaissent pas en tête des hiérarchies. Par exemple, pour SimuA, 20.1% des variables discriminantes (soit 2010 variables) sont absentes. Au vu des écarts de prédiction, il est probable que certaines variables soient trop bruitées dans les exemples étudiés pour être exploitées par le modèle.

Pour les données réelles, les PGU sont proche de 1%. Tous les gènes à l'exception des 600 plus importants peuvent être modifiés avant de confondre les classes. Les PGI sont entre 98 et 99%. Entre 1 et 2% des gènes les plus importants peuvent être modifiés avant de confondre les classes. Autrement dit, 1000 gènes environ sont utilisés par les modèles : ils sont suffisants et si on les modifie, les autres gènes ne suffisent pas.

Métriques globales Les métriques globales estiment la pertinence d'une hiérarchie commune pour tous les exemples d'une même classe. Lorsque les données sont hétérogènes, les valeurs des métriques globales auront tendance à s'éloigner des valeurs locales. Par exemple, dans les simulations, on pourrait s'attendre à ce que les métriques locales et globales soient similaires dans le cas de SimuA (classe homogène) et que les valeurs des PGU augmentent et que celles des PGI diminuent dans le cas des SimuB et SimuC (classe composée de sous-classes sur-exprimant des variables distinctes). Cependant, les différences observées sont limitées. Comme le PGU reste faible, certaines des variables associées à des scores importants contiennent probablement une information discriminante suffisante. De même, le PGI reste élevé : toutes les variables informatives ne suffisent probablement pas à distinguer les classes à elles-seules. La métrique FA, calculée ici pour chacune des sous-classes indépendamment (i.e. en calculant une hiérarchie propre à chaque sous-classe) montre tout de même que les variables informatives ont été mieux prises en compte par le réseau pour SimuA que pour SimuB et SimuC.

Pour les données réelles, les PGU et les PGI varient légèrement par rapport aux métriques locales. On pourrait penser que les classes sont relativement homogènes. Cependant, la Figure 1 montre une visualisation des valeurs d'expression des échantillons issus de BRCA et des scores attribués aux gènes des échantillons cancéreux bien classés. Ces visualisations sont effectuées à l'aide de l'algorithme t-SNE [11] après une analyse en composantes principales. Deux groupes, déjà présents dans les données, apparaissent dans les explications.

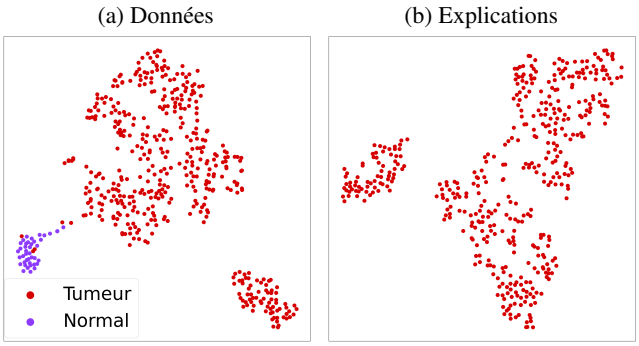

FIGURE 1 : Visualisation t-SNE des données BRCA à partir des valeurs exprimées par les gènes des échantillons (a) et des scores attribués aux gènes des échantillons cancéreux (équation 3) (b).

4 Conclusion

En résumé, la méthode des gradients intégrés (IG) est utilisée pour expliquer les décisions de réseaux de neurones peu profonds sur des données d'expression de gènes. Les scores attribués aux gènes par cette méthode permettent de les hiérarchiser en fonction de leur contribution aux prédictions. Sur les données simulées, les mesures de FA proches de 100% indiquent que IG ordonne bien les variables en fonction de leur caractère informatif pour le réseau. Les métriques PGU et PGI permettent de mettre en avant l'utilité d'environ 8% des variables, à comparer avec les 20% informatifs. Ici, une explicabilité visant à maintenir la performance de classification ne coïncide pas avec une explication mécanistique cherchant à révéler l'ensemble des variables informatives. Sur les données de TCGA, ces deux métriques suggèrent qu'environ 1000 gènes sont utilisés pour différencier les échantillons pathologiques et sains. Il est très probable que l'ensemble des gènes informatifs soit plus large. Nous avons précédemment illustré ce phénomène sur un jeu de données d'expression de gènes pour 33 classes de cancer [6]. Ici, nous avons uniquement exploré des groupes formés selon l'ordre hiérarchique généré par la méthode des gradients intégrés. D'une manière plus générale, trouver les groupes les plus informatifs est un problème combinatoire ouvert.

Au-delà des métriques, la méthode des gradients intégrés dépend de l'exemple utilisé en référence. Ici, l'état cancéreux est expliqué en prenant comme référence les échantillons sains. Les mêmes gènes explicatifs émergeraient-ils si l'état sain était expliqué à partir des échantillons cancéreux ? On pourrait aussi explorer dans quelle mesure la comparaison des variables utilisées pour classer un exemple par un réseau et des variables importantes déterminées globalement pour la classe prédite permet de détecter des erreurs de classification. Enfin, lorsque la classification dépend de relations complexes (e.g. si, parmi deux gènes, un seul des deux doit être exprimé à la fois), il pourrait être plus pertinent d'utiliser des méthodes d'explicabilité mesurant des relations d'ordre supérieur pour estimer des groupes de gènes utiles.

Références

[1] K. TOMCZAK et al., "Review The Cancer Genome Atlas (TCGA) : an immeasurable source of knowledge", *Contemporary Oncology/Współczesna Onkologia*, 2015.

[2] R. RAMIREZ et al., "Classification of cancer types using graph convolutional neural networks", *Frontiers in physics*, 2020.

[3] M. ZITNIK et al., "Machine learning for integrating data in biology and medicine : Principles, practice, and opportunities", *Information Fusion*, 2019.

[4] S. M. LUNDBERG et al., "A unified approach to interpreting model predictions", *NeurIPS*, 2017.

[5] M. SUNDARARAJAN et al., "Axiomatic attribution for deep networks", in *International conference on machine learning*, PMLR, 2017.

[6] M. BONTONOU et al., "Studying Limits of Explainability by Integrated Gradients for Gene Expression Models", *arXiv preprint*, 2023.

[7] D. M. BLEI et al., "Latent dirichlet allocation", *JMLR*, 2003.

[8] A. PASZKE et al., "Pytorch : An imperative style, high-performance deep learning library", *NeurIPS*, 2019.

[9] N. KOKHLIKYAN et al., *Captum : A unified and generic model interpretability library for PyTorch*, unpublished, 2020.

[10] C. AGARWAL et al., "OpenXAI : Towards a Transparent Evaluation of Model Explanations", in *NeurIPS Datasets and Benchmarks Track*, 2022.

[11] L. VAN DER MAATEN et al., "Visualizing data using t-SNE.", *JMLR*, 2008.

Ce travail a été financé par CHIST-ERA-19-XAI-006, pour le projet GRAPHNEX ANR-21-CHR4-0009.