

Réduction d'un modèle astrophysique par réseaux de neurones

Lucas EINIG^{1,2} Pierre PALUD^{3,4} J. CHANUSSOT¹ J. PETY² E. BRON⁴ P. CHAINAIS³ F. LE PETIT⁴
P.-A. THOUVENIN³ M. GERIN⁵ A. ROUEFF⁵ I. BEŠLIĆ⁵ M. GARCIA SANTA-MARIA⁵ J. H. ORKISZ⁵
A. ZAKARDJIAN⁵ S. BARDEAU⁵ J. R. GOICOECHEA⁵ P. GRATIER⁵ V. V. GUZMAN⁵ A. HUGHES⁵
F. LEVRIER⁵ K. ÖBERG⁵ N. PERETTO⁵ E. ROUEFF⁵ A. SIEVERS⁵

¹ Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-Lab, 38000 Grenoble, France

² Institut de Radioastronomie Millimétrique, 38400 Saint-Martin-d'Hères, France

³ Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France

⁴ LERMA, Observatoire de Paris, PSL Research University, CNRS, Sorbonne Universités, 92190 Meudon, France

⁵ Consortium ORION-B (<https://www.iram.fr/~pety/ORION-B/team.html>)

Résumé – Les modèles numériques nécessitant des ressources importantes en temps, mémoire et puissance de calcul sont présents dans de nombreuses disciplines scientifiques. Nous considérons une simulation astrophysique qui calcule de nombreuses sorties à partir de peu de paramètres d'entrée, et qui peut ponctuellement produire des anomalies. Nous proposons une méthode de réduction de modèle par régression basée sur des réseaux de neurones et une méthode d'apprentissage robuste aux anomalies. Nous informons le choix d'une architecture avec une analyse statistique des prédictions du code. Nous démontrons l'intérêt de l'approche proposée en la comparant avec les méthodes de réduction de modèle couramment utilisées en radioastronomie.

Abstract – Numerical models requiring significant resources in time, memory and computing power are present in many scientific disciplines. We consider an astrophysical simulation that computes many outputs from few input parameters, and that may occasionally produce anomalies. We propose a neural network based regression model reduction method and an anomaly robust learning method. We inform the choice of an architecture with a statistical analysis of the code predictions. We demonstrate the interest of the proposed approach by comparing it with model reduction methods commonly used in radio astronomy.

1 Introduction

Des simulations numériques permettent de modéliser des phénomènes complexes dans de nombreuses disciplines scientifiques, dont l'astrophysique. Les modèles les plus réalistes tiennent compte de nombreux aspects, potentiellement multi-physiques. Les ressources de calcul, de temps et de mémoire nécessaires à l'évaluation de ces modèles peuvent être très importantes, limitant ainsi leurs possibilités d'utilisation. Il est commun de remplacer ces simulations coûteuses par des modèles réduits, approchant le comportement original avec des complexités moindres [1].

Dans ce travail, nous présentons une réduction du code astrophysique Meudon PDR¹ [2]. Cette simulation numérique permet d'étudier le processus de naissance d'étoiles, en particulier l'influence d'une étoile naissante sur son nuage moléculaire parent. Il s'agit d'une simulation déterministe, itérative et multi-physique qui tient compte de processus thermiques, radiatifs et chimiques. À partir de quelques paramètres physiques, ce code calcule l'intensité de nombreuses raies d'émission pour diverses espèces chimiques, chacune correspondant à une transition entre deux niveaux quantiques. En radioastronomie, ce type de modèle est typiquement réduit avec des méthodes d'interpolation à partir d'une grille d'évaluations [3].

Nous proposons d'approcher ce code par un perceptron multicouche, une classe de fonctions polyvalente permettant de ne pas faire d'hypothèses fortes sur le modèle. Le code Meudon PDR peut produire des erreurs pour certaines entrées à cause de sa nature itérative et des systèmes d'équations

considérés. Nous identifions ces points anormaux et les retirons de l'ensemble d'apprentissage. La plupart des travaux sur les réseaux de neurones considèrent de nombreuses entrées et peu de sorties, par exemple en classification d'image ou en détection, tandis que le modèle considéré ici contient peu d'entrées et beaucoup de sorties. Nous proposons plusieurs méthodes pour améliorer la précision du modèle réduit tout en limitant la complexité du réseau.

La Section 2 décrit les propriétés du code Meudon PDR. La Section 3 introduit les méthodes proposées. La Section 4 compare l'approche par régression et les méthodes d'interpolation pour une approximation du code Meudon PDR.

2 Approximation de Meudon PDR

2.1 Code Meudon PDR

Le code Meudon PDR [2] calcule un grand nombre d'intensités de raies d'émission $L > 10^3$ à partir d'un nombre restreint de paramètres physiques $D \lesssim 10$. Il peut être représenté par une fonction $\mathbf{f} : \mathbf{x} \in \mathbb{R}^D \mapsto \mathbf{f}(\mathbf{x}) = \mathbf{y} \in \mathbb{R}^L$. La prédiction $\mathbf{f}(\mathbf{x})$ est un vecteur de prédictions individuelles notées $f_\ell(\mathbf{x})$ avec $\ell = 1, \dots, L$. Dans ce travail, nous cherchons à approcher le code Meudon PDR par rapport à $D = 4$ variables², les autres étant fixées à des valeurs standards. Un total de $L = 5375$ intensités de raies d'émission sont considérées. Ces intensités sont strictement positives et couvrent de très nombreux ordres

²Il s'agit de la pression thermique P_{th} , l'intensité du champ radiatif incident G_0 , l'extinction visuelle totale A_V^{tot} et l'angle d'incidence α

¹<https://ism.obspm.fr>

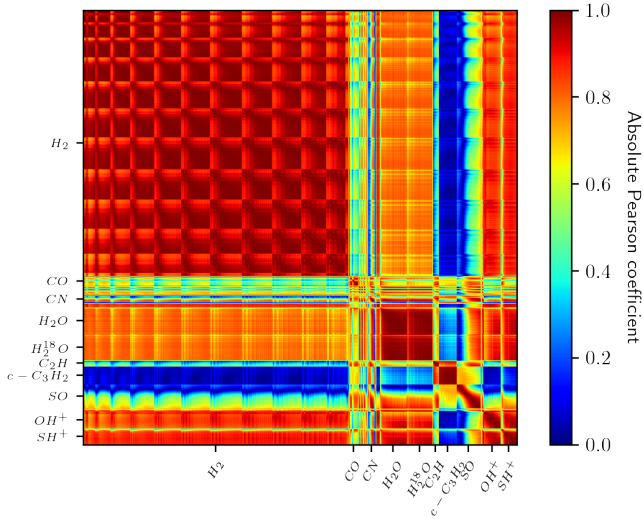


FIGURE 1 : Matrice de corrélation entre les raies de sortie du code Meudon PDR. Les motifs périodiques qui apparaissent dans la matrice proviennent de la manière arbitraire dont ont été organisées les raies d'une même espèce chimique. Pour des raisons de lisibilité, certaines espèces ne sont pas indiquées.

de grandeur, jusqu'à 30 pour certaines raies ℓ . Les valeurs les plus faibles ayant autant d'importance que les valeurs les plus élevées, nous considérons y_ℓ comme le logarithme décimal de l'intensité de la raie ℓ . La fonction f correspondante est hautement non-linéaire. Pour des raisons physiques, nous la supposons continue et différentiable. Chaque évaluation est un calcul intensif et nécessite de quelques heures à quelques jours en raison des nombreux processus pris en compte.

De fortes similarités existent entre certaines raies prédites par le code. La Figure 1 illustre ces similarités avec la matrice de corrélation entre les différentes raies. Certaines sont presque identiques, à un facteur multiplicatif près. Ces similarités sont complexes à décrire ou à prédire physiquement et ne peuvent se résumer à l'appartenance à une même espèce chimique.

Des valeurs anormales peuvent apparaître dans les prédictions du code Meudon PDR. Ces valeurs peuvent atteindre une erreur d'un facteur 100. Tenir compte de l'existence de ces valeurs est donc crucial pour obtenir un modèle réduit précis. Cependant, ces anomalies n'affectent pas simultanément toutes les raies prédites par le code. Le sous-ensemble des sorties individuelles affectées varie d'un point anormal à un autre. Ces points anormaux sont donc difficiles à identifier.

2.2 Régression par réseaux de neurones

Les perceptrons multicouches [4, chapitre 20] sont une classe de fonctions bio-inspirées. Ils bénéficient de très fortes garanties théoriques pour l'approximation de fonctions, grâce aux théorèmes d'approximation universelle [5]. Encore peu présents en radioastronomie [6], ils sont de plus en plus utilisés dans de nombreuses disciplines scientifiques. Cette popularité vient en partie de leur polyvalence. Des hypothèses sur la fonction à approcher peuvent être exploitées dans le choix d'une architecture. Le choix de la fonction d'activation, en l'occurrence GELU, permet par exemple de contraindre la régularité du modèle. Le choix de l'architecture peut avoir un fort impact sur les performances du réseau.

L'apprentissage d'un perceptron multicouche est une minimisation de fonction de coût réalisée par descente de gradient stochastique. Nous utilisons l'écart quadratique (EQ) comme fonction de coût, ce qui est très courant en régression. Pour un couple (\mathbf{x}, y) et une sortie ℓ , l'écart quadratique s'écrit

$$\mathcal{L}_{\text{EQ}}(\mathbf{f}, (\mathbf{x}, y_\ell)) = (f_\ell(\mathbf{x}) - y_\ell)^2. \quad (1)$$

Cette fonction pénalise fortement les grandes erreurs, ce qui la rend très sensible aux anomalies.

3 Architecture et anomalies

3.1 Choix informé d'architecture

L'architecture d'un réseau de neurones est spécifique à la tâche qu'il réalise. Le code Meudon PDR calcule un grand nombre de raies d'émission à partir de très peu de paramètres d'entrées. Nous proposons trois méthodes qui facilitent le choix d'une architecture adaptée à ce cas particulier.

L'apprentissage de non-linéarités avec un nombre limité de couches cachées est difficile. Fournir au réseau des non-linéarités pré-définies, par exemple des fonctions de noyau, facilite cet apprentissage. Nous appliquons une transformation polynomiale $P_p(\mathbf{x})$ de degré p des entrées. Cette approche est classique en régression mais peu utilisée avec des réseaux de neurones. Elle remplace le vecteur d'entrée de dimension D par un vecteur contenant tous les monômes de degrés au plus p formés à partir des D entrées. Par exemple, pour $D = 3$ et $p = 2$, $\mathbf{x} = (x_1, x_2, x_3)$ est remplacé par $P_2(\mathbf{x}) = (x_1, x_2, x_3, x_1^2, x_2^2, x_3^2, x_1x_2, x_1x_3, x_2x_3) \in \mathbb{R}^9$. Pour $p = 3$ et $D = 4$, $P_3(\mathbf{x}) \in \mathbb{R}^{34}$. Cette transformation est implémentée comme partie intégrante du modèle. Elle est suivie d'une normalisation pour assurer un bon conditionnement. Il est donc possible de calculer le gradient du modèle $\nabla_{\mathbf{x}}\mathbf{f}$ avec une méthode d'auto-différentiation.

La plupart des paramètres du réseau appartient à la dernière couche cachée du fait du nombre important de sorties. Le dimensionnement de cette dernière couche est donc critique pour obtenir la meilleure réduction de modèle possible. Une dernière couche trop grande peut mener à un sur-apprentissage et une dernière couche trop petite à un modèle trop régularisé. En régression, la dernière couche d'un réseau de neurone correspond à une transformation affine. Choisir la taille optimale de cette couche revient donc à estimer la dimension $\tilde{L} \leq L$ du sous-espace dans lequel vivent les L intensités de raies. Une analyse en composantes principales (ACP) [4, chapitre 23] sur les intensités y permet d'estimer cette dimension.

Prédire simultanément L raies est une tâche complexe. La Fig. 1 montre que si certaines raies se ressemblent beaucoup, d'autres ne sont pas corrélées. Il peut donc être plus simple d'apprendre à prédire des sous-ensembles de raies homogènes. Un partitionnement spectral [4, chapitre 22] permet de regrouper les L sorties en segments homogènes. La matrice de corrélation des raies est utilisée comme matrice de similarité. La forte corrélation au sein d'un segment réduit le nombre de composantes principales nécessaires et donc le nombre de paramètres à entraîner. Les calculs intermédiaires nécessaires pour un segment sont également mieux exploités. Un ensemble de petits réseaux spécialistes disjoints, chacun dédié à un segment, peut donc être avantageux par rapport à un unique réseau.

3.2 Robustesse aux anomalies

L'apprentissage de modèle en présence de valeurs aberrantes est un sujet crucial en apprentissage statistique [7]. Une approche commune est d'utiliser une fonction de coût robuste, c'est-à-dire pénalisant peu les grandes erreurs, par exemple la log-densité d'une distribution de Cauchy. Pour un couple (\mathbf{x}, \mathbf{y}) et une sortie ℓ , la fonction de Cauchy s'écrit

$$\mathcal{L}_C(\mathbf{f}, (\mathbf{x}, y_\ell)) = \log(1 + (f_\ell(\mathbf{x}) - y_\ell)^2). \quad (2)$$

La limite de cette d'approche est que l'apprentissage peut ignorer des points valides mais difficiles à reproduire, par exemple des changements de régime brusques. Ce comportement est à éviter pour le code Meudon PDR.

Nous appliquons la procédure en trois temps de [8]. Considérons un ensemble d'apprentissage contenant N_{train} couples $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^D \times \mathbb{R}^L$. Un premier réseau de neurones est entraîné avec la fonction de Cauchy (2) pour définir un modèle robuste aux anomalies. Une revue pour manuelle des points de l'ensemble d'apprentissage associés à de grandes erreurs permet d'identifier les anomalies. Un masque binaire $\mathbf{M} \in \mathcal{M}_{N_{\text{train}}, L}(\mathbb{R})$ est construit, avec $m_\ell = 0$ si une valeur est identifiée comme anormale, et 1 sinon. Un masque matriciel $\mathbf{M} \in \mathcal{M}_{N_{\text{train}}, L}(\mathbb{R})$ plutôt qu'un vecteur de dimension N_{train} limite le nombre de valeurs masquées. Il ne rejette que les quelques valeurs affectées par une anomalie au sein d'une simulation plutôt que l'ensemble des L sorties. Cette étape requiert une bonne connaissance de la simulation considérée et de la physique sous-jacente. Elle peut être partiellement automatisée, de sorte à minimiser le nombre d'actions demandées à l'utilisateur. Quelques heures suffisent alors pour obtenir un masque satisfaisant.

Un nouveau réseau est alors entraîné avec une erreur quadratique moyenne masquée, qui pénalise fortement les grandes erreurs et ignore les anomalies. Pour un triplet $(\mathbf{x}, \mathbf{y}, \mathbf{m})$ et une sortie ℓ , elle s'écrit

$$\mathcal{L}_m(\mathbf{f}, (\mathbf{x}, y_\ell, m_\ell)) = m_\ell (f_\ell(\mathbf{x}) - y_\ell)^2. \quad (3)$$

L'ajout de ce masque à l'écart quadratique (1) permet d'exiger du modèle réduit qu'il soit précis aux points correspondant aux changements de régimes, difficiles à reproduire mais très informatifs en astrophysique. La qualité du masque influe cependant grandement sur le modèle réduit obtenu.

4 Expériences

4.1 Critères d'évaluation

Les réseaux de neurones proposés sont comparés avec plusieurs méthodes d'interpolation : plus proche voisin, linéaire, par splines [9] et par fonction de base radiale (RBF) [10]. Ces méthodes d'interpolation sont fréquemment utilisées dans la communauté astrophysique pour remplacer un modèle numérique à partir d'une grille d'évaluations [3].

La comparaison est basée sur trois critères principaux : la durée d'évaluation, le nombre de paramètres nécessaires pour décrire le modèle, et la précision d'approximation. La durée d'évaluation est mesurée sur l'ensemble des L raies pour 1 000 points. Par souci d'équité, les réseaux de neurones sont évalués sur CPU. Leurs durées peuvent être encore réduites en les utilisant sur GPU. Le nombre nécessaire de paramètres pour une

méthode d'interpolation est égal au nombre de données dans la grille d'apprentissage. La précision des modèles réduits est évaluée sur l'ensemble de test. Pour quantifier les précisions, nous définissons une nouvelle métrique appelée facteur d'erreur (EF) qui est une version symétrisée de l'erreur relative. Pour un couple (\mathbf{x}, \mathbf{y}) et une sortie ℓ , le facteur d'erreur s'écrit

$$\text{EF}(\mathbf{f}, (\mathbf{x}, y_\ell)) = 10^{|f_\ell(\mathbf{x}) - y_\ell|}, \quad (4)$$

avec $f_\ell(\mathbf{x})$ et y_ℓ des intensités en échelle logarithmique. Le facteur d'erreur peut également être exprimé en pourcentage grâce à la transformation $100 \times (\text{EF} - 1)$. Dans la suite, le facteur d'erreur est indiqué en pourcentage s'il est inférieur à 100%, sinon il est converti en facteur multiplicatif et précédé par le signe \times .

La distribution d'erreurs obtenue est résumée par sa moyenne, son percentile à 99% et son *maximum*. Le percentile à 99% et le *maximum* permettent d'estimer une borne supérieure sur l'erreur du modèle réduit. Le percentile à 99% est plus robuste que la valeur maximale aux points anormaux potentiellement présents dans les données de test. Il fournit donc une borne supérieure plus informative que le *maximum*.

Dans des observations récentes [11], les erreurs d'échantonnage conduisent à des erreurs multiplicatives moyennes d'environ 10%. Pour qu'une réduction de modèle soit pertinente pour l'analyse des observations, son facteur d'erreur moyen doit donc être inférieur à 10% ($\text{EF} = 1, 1$).

4.2 Résultats

Deux ensembles de données sont construits à partir d'évaluations du code Meudon PDR. L'ensemble d'apprentissage est utilisé pour construire les modèles réduits. Il contient $N_{\text{train}} = 14 \times 14 \times 14 \times 7 = 19\,208$ points formant une grille régulière dans l'espace des paramètres. L'ensemble de test contient $N_{\text{test}} = 3\,192$ simulations. Ces points sont tirés aléatoirement et indépendamment d'une loi uniforme à l'intérieur de l'enveloppe convexe de la grille d'apprentissage.

La Table 1 présente les résultats sur l'ensemble de test. On observe que l'approche par régression avec un réseau de neurones surpasse les méthodes d'interpolation pour toutes les métriques. Les réseaux de neurones sont 100 à 1 000 fois plus rapides que les méthodes d'interpolation tout en nécessitant 20 fois moins de paramètres en mutualisant les calculs intermédiaires. Ils sont également beaucoup plus précis, avec des erreurs moyennes jusqu'à deux fois plus faibles et des erreurs maximales jusqu'à trois fois plus faibles. Les choix d'architecture informée et la gestion des anomalies provoquent une diminution de l'erreur moyenne sur les données de test et une diminution significative des erreurs les plus importantes.

Pour la totalité des $L = 5\,375$ sorties, $\tilde{L} \simeq 1\,000$ composantes principales suffisent pour une reconstruction avec un facteur d'erreur moyen inférieur à 0,1%. Si la fonction de coût utilisée pour l'apprentissage du réseau est l'écart quadratique moyen, alors \tilde{L} est aussi le nombre minimal de neurones dans la dernière couche cachée assurant une erreur d'approximation moyenne inférieure à 0,1% lors de la transformation linéaire.

Nous choisissons de fixer l'architecture à trois couches cachées de même dimension, obtenue par la méthode précédente. L'objectif de ce travail n'est pas de produire un réseau optimal, mais de proposer des méthodes génériques et performantes de

TABLE 1 : Comparaison des approches par régression par réseaux de neurones et par interpolation. Le temps de calcul des méthodes par splines n’est pas indiqué car l’implémentation de Scipy ne permet une comparaison équitable pour des fonctions à valeurs vectorielles. Pour les réseaux de neurones, R correspond au dimensionnement de la dernière couche cachée par ACP, P au développement polynomial avec $p = 3$, M au masque et S à la segmentation des raies.

Méthode	Facteur d’erreur (%)			Param. (million)	Temps (ms cpu)	
	moy.	99% per.	max			
Voisin	×13,1	×11,7	×3e5	103	62	
Linéaire	15,7	×2,3	×143	103	1,5e3	
Spline	Linéaire	15,7	×2,3	×144	103	-
	Cubique	11,2	×2,2	×122	103	-
	Quintique	19,1	×2,9	×304	103	-
RBF	Linéaire	10,2	96,8	×99	103	1,1e4
	Cubique	10,4	×2,1	×112	103	1,1e4
	Quintique	10,9	×2,1	×118	103	1,1e4
Réseaux	R	7,3	64,8	×81	7,4	12
	R+P	6,2	49,7	×84	7,4	13
	R+P+M	5,5	42,3	× 41	7,4	13
	R+P+M+S	4,9	44,5	×44	3,2	14

réduction de tels modèles. La Table 1 montre que cet objectif est atteint sans optimisation d’hyperparamètres, comme le nombre de couches ou leurs tailles respectives, qui nécessiterait une validation croisée et davantage de données.

L’identification des anomalies sur l’ensemble d’apprentissage produit un masque retirant 57 498 valeurs, soit 0.39% des données d’apprentissage. Le masque assure l’apprentissage du modèle sur des données valides. Il fournit également aux utilisateurs du code Meudon PDR un indicateur de la fiabilité des résultats obtenus au voisinage de certains paramètres.

Les raies sont partitionnées en quatre segments de sorte à maximiser le rapport entre les corrélations moyennes intra et inter-segments. Les corrélations moyennes intra et inter-segments sont de 0,895 et 0,462 respectivement, tandis que la corrélation moyenne entre toutes les raies est de 0,73. Les segments contiennent respectivement 3712, 1272, 241 et 150 raies. Ce déséquilibre entre segments provient du déséquilibre entre molécules. Par exemple, H_2 génère plus de la moitié des raies d’émission, et ces raies sont toutes hautement corrélées, comme le montre la Fig. 1. Pour chaque segment, le nombre de composantes principales assurant une erreur de reconstruction inférieure à 0.1% est respectivement d’environ 500 ($\simeq 13\%$ des 3712 raies du segment), 350 ($\simeq 28\%$), 100 ($\simeq 41\%$), 75 (50%). Les plus gros segments sont les plus homogènes, ce qui permet de grandement réduire le nombre de paramètres à entraîner. La Table 1 montre les performances obtenues en utilisant ces valeurs pour définir la taille de la dernière couche cachée de chacun des sous-réseaux. L’utilisation de réseaux spécialistes permet de diviser par 2 le nombre de paramètres en maintenant le niveau de précision.

5 Conclusion

Dans ce travail, nous avons considéré une réduction de modèle contenant peu d’entrées et beaucoup de sorties. Une régression avec réseaux de neurones est plus performante qu’une interpolation pour toutes les métriques considérées. Fournir des non-linéarités pré-définies au réseau facilite l’apprentis-

sage. Une ACP permet d’informer le choix d’une architecture. Une segmentation des sorties permet de définir des réseaux spécialistes disjoints plutôt qu’un réseau général, ce qui réduit le nombre de paramètres à entraîner. La définition semi-automatique d’un masque tient compte des données anormales. Les dérivées du modèle réduit obtenu sont calculables avec de l’auto-différentiation. L’approche présentée est suffisamment générale pour être appliquée à des modèles similaires dans d’autres disciplines.

Remerciements

Ce travail a été soutenu par l’Agence Nationale de la Recherche sous la subvention DAOISM (ANR-21-CE31-0010), le programme PCMI du CNRS/INSU avec l’INC/INP et un projet 80Prime du CNRS. Nous remercions l’Institut MIAI@Grenoble Alpes (ANR-19-P3IA-0003), la Chaire IA Sherlock (ANR-20-CHIA-0031-01), ainsi que le programme national d’investissement d’avenir (ANR-16-IDEX-0004 ULNE) et la Région Hauts-de-France.

Références

- [1] P. BENNER, S. GUGERCIN et K. WILLCOX, “A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems”, *SIAM Review*, t. 57, p. 483-531, juin 2015.
- [2] F. LE PETIT, C. NEHME, J. LE BOURLOT et al., “A Model for Atomic and Molecular Interstellar Gas : The Meudon PDR Code”, *The Astrophys. J. Supp. Series*, t. 164, n° 2, p. 506-529, juin 2006.
- [3] R. WU, E. BRON, T. ONAKA et al., “Constraining physical conditions for the PDR of Trumpler 14 in the Carina Nebula”, *A&A*, t. 618, A53, oct. 2018.
- [4] S. SHALEV-SHWARTZ et S. BEN-DAVID, *Understanding Machine Learning : From Theory to Algorithms*, 1^{re} éd. Cambridge University Press, mai 2014.
- [5] M. LESHNO, V. Y. LIN, A. PINKUS et al., “Multilayer feed-forward networks with a nonpolynomial activation function can approximate any function”, *Neural Networks*, t. 6, n° 6, p. 861-867, jan. 1993.
- [6] C. J. FLUKE et C. JACOBS, “Surveying the reach and maturity of machine learning and artificial intelligence in astronomy”, *WIREs Data Mining Knowl Discov*, t. 10, n° 2, mar. 2020, arXiv :1912.02934 [astro-ph].
- [7] H. WANG, M. J. BAH et M. HAMMAD, “Progress in Outlier Detection Techniques : A Survey”, *IEEE Access*, t. 7, p. 107 964-108 000, 2019, Conference Name : IEEE Access.
- [8] H. J. MOTULSKY et R. E. BROWN, “Detecting outliers when fitting data with nonlinear regression – a new method based on robust nonlinear regression and the false discovery rate”, *BMC Bioinformatics*, t. 7, n° 1, p. 123, mar. 2006.
- [9] B. D. BOJANOV, H. A. HAKOPIAN et A. A. SAHAKIAN, *Spline Functions and Multivariate Interpolations*. Dordrecht : Springer Netherlands, 1993.
- [10] G. E. FASSHAUER, *Meshfree Approximation Methods with Matlab*. 2007.
- [11] J. PETY, V. GUZMAN, J. ORKISZ et al., “The anatomy of the Orion B Giant Molecular Cloud : A local template for studies of nearby galaxies”, *Astronomy & Astrophysics*, t. 599, nov. 2016.