



Septembre 2024

**Étudiant :** Gonzalo Becker  
**Tuteur École :** Dominique Pastor  
**Tuteur Entreprise :** Benoit Sklénard

# Quantification de l'incertitude dans les modèles d'apprentissage profond pour la simulation à l'échelle atomique

Rapport de Stage de fin d'études

TAF MCE



**IMT Atlantique**  
Bretagne-Pays de la Loire  
École Mines-Télécom



© 2024 - *GONZALO BECKER*

ALL RIGHTS RESERVED.

INSTITUT MINES TÉLÉCOM ATLANTIQUE  
BREST, FRANCE

CEA LETI  
GRENOBLE, FRANCE

## Quantification de l'incertitude dans les modèles d'apprentissage profond pour la simulation à l'échelle atomique

### RÉSUMÉ

Ce stage se concentre sur l'utilisation des techniques d'intelligence artificielle pour accélérer la simulation à l'échelle atomique de matériaux utilisés dans les dispositifs microélectroniques. Traditionnellement, ces simulations, basées sur des méthodes comme la théorie de la fonctionnelle de la densité (DFT), sont très coûteuses en temps de calcul. Les réseaux de neurones, notamment les modèles équivariants par rotation et réflexion basés sur des graphes, tels que NequIP et Allegro, offrent une solution prometteuse. Cependant, leur incapacité à estimer la précision de leurs prédictions pose des limites à leur utilisation en chimie et, dans le cas de ce travail, en science des matériaux.

Dans ce contexte, cette étude consiste à développer et à mettre en oeuvre des méthodes d'estimation de l'incertitude pour ces réseaux de neurones. Cette estimation est cruciale pour identifier les prédictions des modèles qui sont moins fiables et qui nécessitent davantage de données pour améliorer leur précision. Ensuite, il est possible d'implémenter un apprentissage actif capable de sélectionner judicieusement les données les plus informatives, ou, autrement dit, avec plus d'incertitude, pour l'entraînement du modèle. Cela permet de concentrer les ressources de calcul sur les aspects les plus critiques, optimisant ainsi le processus global.

L'objectif final est de disposer de modèles d'Intelligence Artificielle très précis pour étudier les phénomènes physiques à l'échelle microscopique ou découvrir de nouveaux matériaux. À terme, cette approche pourrait permettre d'accélérer le développement de nouveaux dispositifs en rendant les simulations atomistiques plus efficaces.

# Table des matières

<b>1</b>	<b>INTRODUCTION</b>	<b>5</b>
1.1	Structure d'accueil . . . . .	5
1.2	Ambiance de travail . . . . .	7
1.3	Mission de stage . . . . .	8
1.4	Planification . . . . .	11
<b>2</b>	<b>RÉSEAUX DE NEURONNES EN GRAPHE POUR LES STRUCTURES MOLÉCULAIRES</b>	<b>12</b>
2.1	Introduction . . . . .	12
2.2	Réseaux de Neurones de Passage de Messages . . . . .	13
2.3	Equivariance . . . . .	14
2.4	Architecture de NequIP . . . . .	15
<b>3</b>	<b>ESTIMATION D'INCERTITUDE</b>	<b>19</b>
3.1	Introduction . . . . .	19
3.2	Méthodes d'estimation . . . . .	20
3.3	Détails du développement . . . . .	27
3.4	Résultats . . . . .	29
<b>4</b>	<b>CONCLUSION</b>	<b>38</b>
4.1	Développement et Défis du Stage . . . . .	38
4.2	Futur du projet . . . . .	39
4.3	Impact sociétal et environnemental . . . . .	40
4.4	Valeur ajoutée à l'entreprise . . . . .	40
4.5	Complémentarité avec la formation et le projet professionnel . . . . .	41
<b>5</b>	<b>ANNEXES</b>	<b>42</b>
5.1	Réseaux de Neurones pour la prédiction moléculaire . . . . .	42
5.2	Estimation de l'Incertitude . . . . .	45
5.3	Détails d'entraînement . . . . .	50
	<b>REFERENCES</b>	<b>53</b>

# 1

## Introduction

### 1.1 STRUCTURE D'ACCUEIL

Le Commissariat à l'Énergie Atomique et aux Énergies Alternatives (CEA) a été créé en 1945 pour promouvoir le développement de l'énergie nucléaire en France. Au fil des décennies, ses activités se sont diversifiées pour inclure de nombreux autres domaines, tels que les énergies alternatives, le numérique, la santé et les sciences du vivant, la matière et l'univers, ainsi que la défense, la sécurité, et le climat et l'environnement. Ce stage s'inscrit dans le cadre de la recherche en numérique et en sciences des matériaux, où l'intelligence artificielle sera appliquée aux matériaux pour des applications en micro-électronique.

Le CEA dispose d'un budget annuel de 6,1 milliards d'euros, réparti sur ses neuf centres de recherche, et bénéficie de la collaboration de plus de 700 partenaires industriels. Cet organisme de recherche est classé premier en Europe en termes de dépôts de brevets, avec 669 brevets enregistrés en 2023. De plus, son engagement en faveur de la création de start-ups et de partenariats industriels constitue un élément fondamental de sa mission. Ce dynamisme permet au CEA de jouer un rôle actif dans la société, en mettant le savoir-faire acquis dans ses laboratoires au service des défis contemporains.

Le stage se déroule au sein du Laboratoire de Simulation et Modélisation (LSM). La structure or-



**Figure 1.1.1 – Site du CEA à Grenoble**

générationnelle du laboratoire dedans le CEA est la suivante :

*Direction de la recherche technologique (DRT)*

*Laboratoire d'électronique des technologies de l'information (LETI)*

*Département Composants Silicium (DCOS)*

*Service Caractérisation, Conception et Simulation (SCCS)*

*Laboratoire de Simulation et de Modélisation (LSM)*

Avec environ 20 chercheurs et 4 thésards, le LSM a l'objectif de développer des outils et des méthodologies pour accélérer la simulation et modélisation dans la microélectronique. Dans ce contexte, le laboratoire, très axé sur la simulation, se concentre sur 5 points, à savoir :

1. **Simulation atomistique** : Utilisation de méthodes de simulation telles que la DFT et l'intelligence artificielle pour la découverte et étude de matériaux, la simulation d'interfaces et l'analyse de propriétés physiques à l'échelle atomique.
2. **Simulation avancée du transport de charges** : prédition du mouvement des charges électriques à travers différents matériaux et structures.
3. **TCAD** : Conception de technologies assistée par ordinateur.
4. **Simulation multi-physique** : Simulation thermique, mécanique et électromagnétique.
5. **Modélisation compacte** : Utilisation des modèles mathématiques qui encapsulent les détails physiques tout en permettant une analyse efficace et rapide des circuits complexes

Ce stage est encadré principalement par l'équipe de simulation atomistique. L'organisation du projet exige des échanges réguliers avec les responsables de l'application de l'intelligence artificielle (IA) à la simulation de matériaux pour la microélectronique. En général, les méthodes développées visent à renforcer la robustesse des modèles en intégrant des métriques d'incertitude, tout en offrant une interface utilisateur accessible. Par conséquent, il est crucial de comprendre comment ces modèles d'IA sont actuellement utilisés et de concevoir une interface simple, afin de ne pas compliquer davantage un processus de simulation déjà complexe. Étant donné que les ingénieurs en simulation

atomistique ne sont pas nécessairement experts en IA, il est essentiel que cette interface soit suffisamment intuitive pour qu'ils puissent utiliser ces outils sans avoir à en maîtriser le fonctionnement interne.

## 1.2 AMBIANCE DE TRAVAIL

Le LSM présente une structure très transversale et multidisciplinaire. Les différents chercheurs collaborent entre eux et guident leurs étudiants (stagiaires, alternants ou thésards) dans leurs développements. Les activités se divisent principalement en trois grands groupes : la simulation avancée, qui couvre les points 1 et 2 de la liste précédente ; la simulation continue, correspondant aux points 3 et 4 ; et enfin, la modélisation compacte. Le laboratoire a un seul supérieur hiérarchique, le chef de laboratoire, qui est également très impliqué dans les activités de recherche. Cela crée une ambiance de travail conviviale et très efficace.

Chaque semaine, des réunions sont organisées au laboratoire, où une personne présente ses avancées les plus récentes, ce qui donne toujours lieu à des discussions très intéressantes. Cela permet également de motiver les autres et de planifier les activités générales. Il faut comprendre que le laboratoire est très orienté vers des développements industriels, avec des entreprises privées qui mènent des projets communs avec le LSM. Cela est certainement dû au haut niveau d'expertise que le LSM peut fournir. Même avec les exigences propres à ces projets, il est évident que tous les collaborateurs sont satisfaits de leur métier et motivés pour pousser au maximum leurs développements dans la microélectronique.

Cette ambiance de travail demande beaucoup d'autonomie et de responsabilité de la part de chaque individu. En effet, il faut planifier, se répartir les tâches et communiquer correctement pour réussir à compléter les projets en cours, même si parfois il y a des défis techniques très difficiles à surmonter. Le laboratoire est également très impliqué dans le dépôt de brevets et la publications d'articles, qui sont très souvent présentés dans des conférences internationales, ce qui est un élément clé pour la motivation collective.

Le cadre de travail est très diversifié sur le plan culturel, de sorte qu'on peut trouver des collaborateurs provenant de tous les continents. Cela crée une ambiance multiculturelle très positive, où tout le monde est à l'écoute et prêt à faire les efforts nécessaires pour que tous se sentent bien accueillis.

En général, il ne fait aucun doute que le LSM est un laboratoire d'excellence, avec des chercheurs passionnés et accueillants. Le cadre de travail propose une planification de tâches stimulantes, équilibrant défis et réalisations, ce qui permet une montée en compétences continue sans pression excessive.

### 1.3 MISSION DE STAGE

Dans le cadre de l'utilisation de l'IA dans la simulation moléculaire, un objectif clé est de prédire les propriétés d'une certaine structure atomique à partir des positions atomiques et des espèces chimiques. Supposons par exemple que nous disposions d'une structure moléculaire comme celle présentée dans la Figure 1.3.1. Une quantité intéressante à analyser d'un point de vue physique est l'énergie potentielle  $E$  stockée dans les liaisons moléculaires. Cette grandeur est cruciale car elle permet de prédire d'autres propriétés de la structure. Par exemple, elle peut être utilisée pour calculer les forces interatomiques comme étant l'opposé du gradient de l'énergie potentielle totale du système par rapport à la position de l'atome concerné. Autrement dit,  $\mathbf{F}_i = -\nabla_{r_i} E$ . De plus, lorsque les positions sont telles que l'énergie est minimisée, on atteint une longueur d'équilibre pour la distance interatomique, permettant ainsi de mesurer la stabilité de la structure atomique.

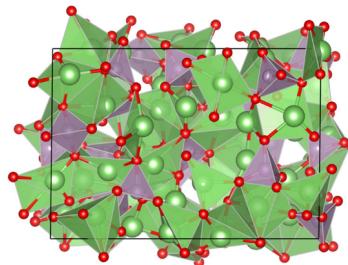
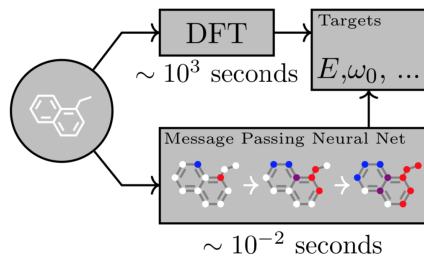


Figure 1.3.1 – Structure de  $Li_3PO_4$  [1]

Globalement, il est clair que l'énergie potentielle est un élément indispensable pour mieux modéliser les systèmes moléculaires. Mais comment peut-on obtenir cette quantité via une méthode de calcul numérique ?

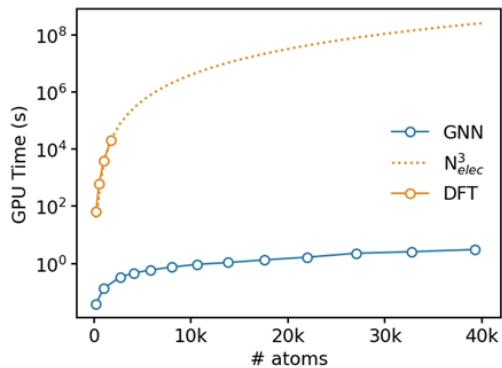
Au fil des années, de nombreuses solutions ont été développées, la plus connue étant la théorie de la fonctionnelle de la densité (DFT). Ces méthodes, basées sur la mécanique quantique, requièrent des temps de calcul excessivement longs, car elles impliquent la résolution d'équations complexes régissant la physique quantique. En effet, l'ordre de complexité de la méthode DFT est de  $\mathcal{O}(n^3)$ , où  $n$  représente le nombre d'électrons, et peut être encore plus élevé pour les méthodes au-delà de la DFT. Cela limite à seulement quelques centaines le nombre d'atomes dans les systèmes à simuler. Aujourd'hui, ce facteur demeure le principal obstacle aux simulations moléculaires.



**Figure 1.3.2 – utilisation des réseaux de neurones pour la prédiction moléculaire [2]**

Toutefois, l'IA a ouvert la voie à une accélération de ces simulations. En particulier, les réseaux de neurones en graphe (GNNs) basés sur les réseaux de neurones de passage de messages (MPNNs), et plus récemment, une implémentation équivariante de ces modèles, ont montré un succès considérable. La Figure 1.3.2 illustre comment les MPNNs peuvent remplacer un calcul DFT pour prédire l'énergie totale d'un système. L'entraînement des MPNNs est fait sur un jeu de données généré à partir de calculs DFT.

Dans la Figure 1.3.3, il est évident que le temps requis par la DFT est proportionnel au cube du nombre d'électrons dans le système. En revanche, les GNNs affichent des temps de calcul significativement réduits, par plusieurs ordres de grandeur.



**Figure 1.3.3 – Temps de calcul en fonction du nombre d'atomes**

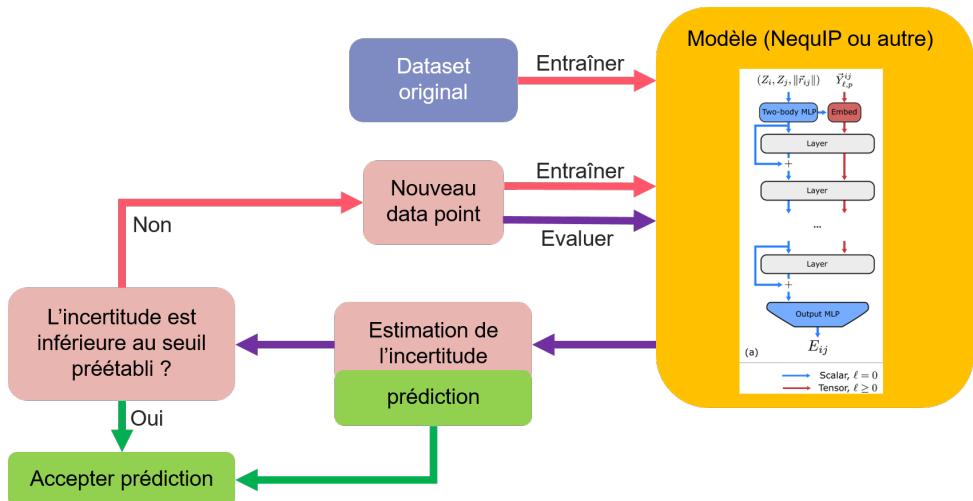
Comme tout modèle d'apprentissage supervisé, ces réseaux requièrent des temps d'entraînement prolongés ainsi qu'une grande quantité de données pour fonctionner correctement. Par conséquent, ces deux facteurs engendrent une consommation énergétique significative.

Dans ce contexte, chaque donnée se correspond à une structure moléculaire distincte. Or, la génération de ces données est elle-même coûteuse, d'où l'intérêt de savoir quelles données sont les plus "intéressantes" en fonction de l'état d'entraînement du modèle à un instant donné. C'est ici que l'apprentissage actif intervient : comment, à partir d'un modèle préentraîné, déterminer quelles données utiliser pour l'entraîner plus efficacement par la suite ?

Une réponse potentielle réside dans l'estimation de l'incertitude des réseaux de neurones. En identifiant les structures moléculaires sur lesquelles notre modèle est le plus incertain, nous pouvons cibler les données sur lesquelles l'entraîner pour l'améliorer rapidement. Il est important de noter que pour les données choisies, il faudra toujours recalculer les sorties attendues avec une méthode telle que la DFT.

Grâce à cette méthodologie, nous pourrions utiliser les réseaux de neurones pour prédire et explorer un nombre illimité des structures moléculaires complexes. Par exemple, pour générer un nouveau matériau avec des propriétés spécifiques, nous pourrions développer des algorithmes explorant diverses structures moléculaires, évaluant leurs propriétés avec un modèle d'IA. Si l'algorithme détecte une incertitude dans la prédiction, il suffirait de ré-entraîner le modèle en temps réel avec des données pertinentes pour travailler toujours avec un modèle précis.

Dans la Figure 1.3.4, un diagramme illustrant le processus d'apprentissage actif est présenté. Les données utilisées pour l'entraînement initial sont représentées en bleu. En rouge, on observe la boucle d'entraînement actif, qui se déclenche uniquement lorsqu'un certain groupe de points dans le jeu de données présente une incertitude trop élevée. En vert, les évaluations sont acceptées car leur incertitude est faible.



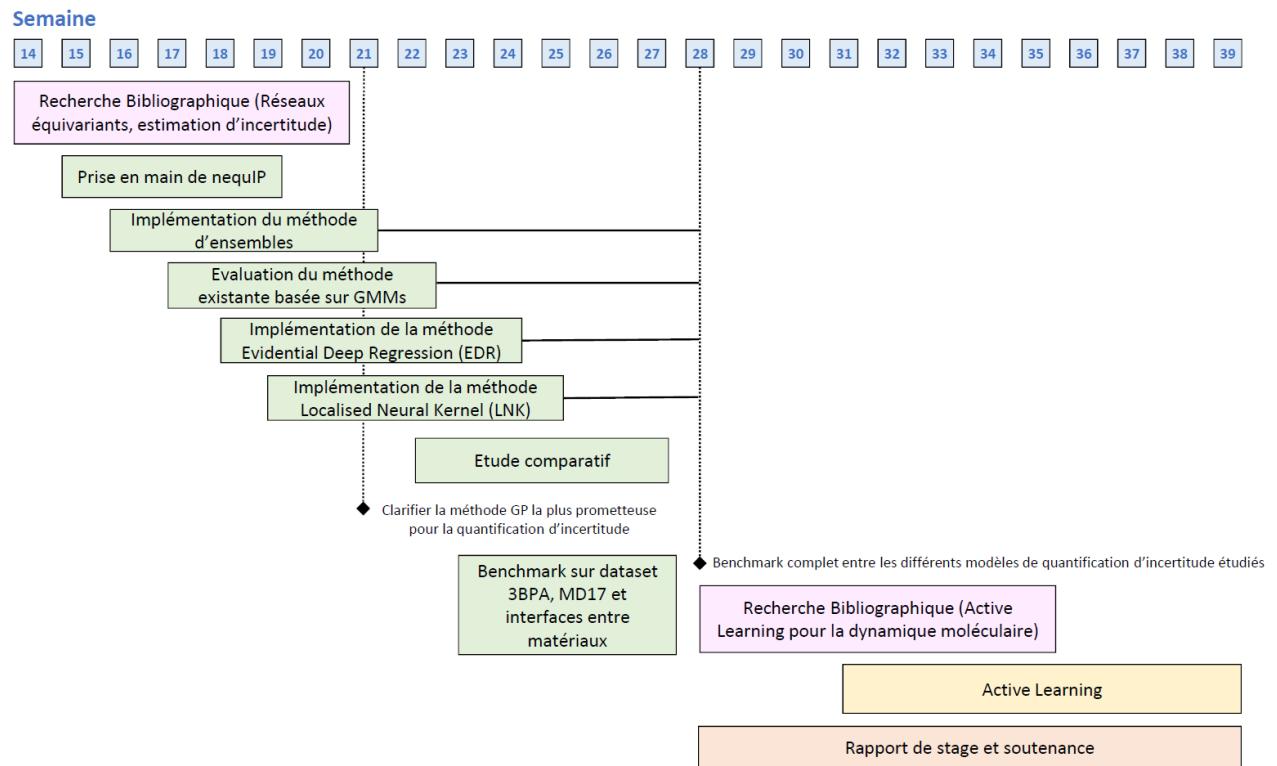
**Figure 1.3.4 – Apprentissage actif appliqué au développement de modèles prédictifs**

Des travaux de recherche ont déjà été réalisés sur cette méthodologie. Dans l'article [3], un développement avancé de l'apprentissage actif, tel que décrit précédemment, est présenté pour la prédiction générale des propriétés atomistiques. Dans une autre étude [4], l'apprentissage actif est utilisé avec des modèles capables de prédire la *stabilité* des matériaux, permettant ainsi de rechercher des matériaux stables. Grâce à cette méthodologie, 381 000 nouveaux matériaux stables ont été découverts, contre 48 000 connus auparavant! La principale limitation de l'apprentissage actif réside dans la difficulté d'obtenir des mesures d'incertitude pour les modèles d'IA qui soient à la fois rapides et

précises. La quantification de l'incertitude, un domaine de recherche en soi, sera la tâche principale du stage, au cours duquel plusieurs méthodes seront évaluées.

## 1.4 PLANIFICATION

La Figure 1.4.1 illustre la planification approximative initiale du stage. Cette planification a été globalement respectée, bien que certaines méthodes de calcul de l'incertitude, telles que la régression évidentielle et le *Localised Neural Kernel* (LNK), aient nécessité plus de temps que prévu.



**Figure 1.4.1 – Planification**

# 2

## Réseaux de neurones en graphe pour les structures moléculaires

### 2.1 INTRODUCTION

Les réseaux de neurones ont la propriété fondamentale d'être des estimateurs universels, ce qui signifie qu'avec un nombre de neurones et une complexité de modèle suffisants, ils peuvent approximer avec une précision infiniment bonne n'importe quelle fonction. En effet, les réseaux de neurones ont la capacité, couche après couche, d'apprendre des dépendances de plus en plus complexes. Cependant, le choix judicieux du modèle reste crucial pour garantir des résultats optimaux. En conséquence, les progrès en intelligence artificielle ne sont pas toujours dus à des avancées technologiques en matière de puissance de calcul, mais souvent à des idées novatrices dans le modèle mathématique sous-jacent.

Récemment, les GNNs ont connu un grand succès dans divers domaines, comme illustré sur la Figure 2.1.1. Ceci est dû à leur capacité à modéliser des relations complexes entre ses noeuds.

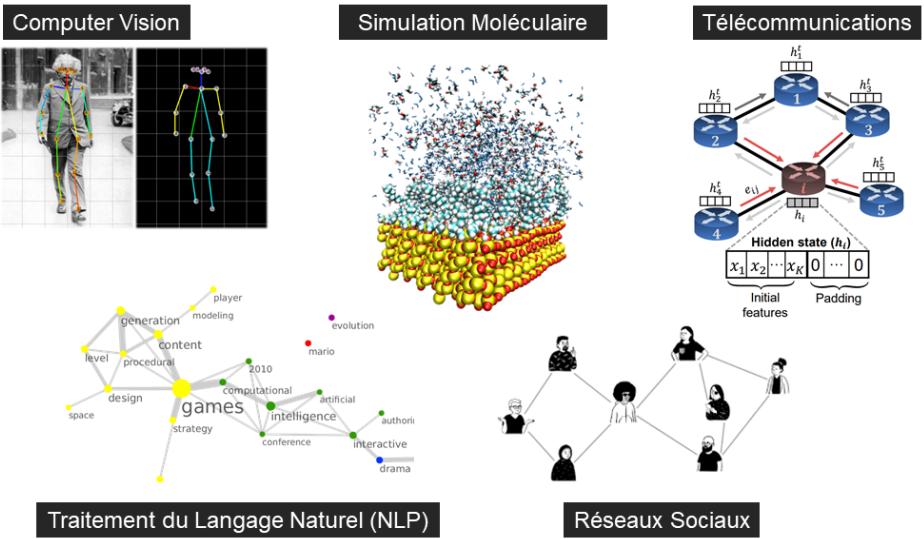


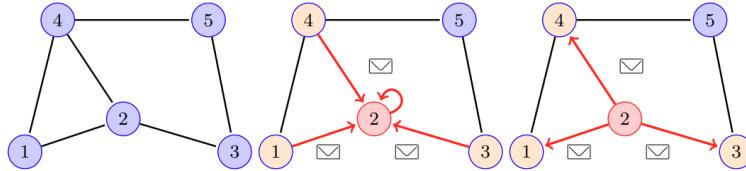
Figure 2.1.1 – Applications des GNNs [5] [6] [7]

Concernant leur utilisation pour les structures atomiques, des modèles basés sur des réseaux de neurones équivariants par rotation et réflexion [1] [8] ont émergé autour de 2022, entraînant une avancée significative dans notre capacité à prédire les propriétés moléculaires. Ces nouveaux modèles sont capables de faire des meilleures prédictions avec moins de données (faire plus avec moins!), ce qui implique une avancée très significative dans le domaine.

Pour les différents développements réalisés au cours de ce stage, le modèle proposé par [8], appelé NequIP, a été utilisé. Le code de NequIP version 0.6.0, disponible sur GitHub à l'adresse suivante <https://github.com/mir-group/nequip>, a servi de base (backbone) pour les différentes implémentations effectuées durant le stage.

## 2.2 RÉSEAUX DE NEURONES DE PASSAGE DE MESSAGES

Les Réseaux de Neurones de Passage de Messages (Message Passing Neural Networks, MPNN) ont été formellement décrits pour la première fois dans [2]. Comme observé dans la Figure 2.2.1, l'idée centrale consiste à ce que, à chaque couche d'interaction, chaque nœud génère des messages à partir de ses caractéristiques internes dans cette couche. Ces messages sont ensuite agrégés à l'aide d'une fonction, qui peut être aussi simple qu'une somme ou plus complexe, selon le cas. Des détails supplémentaires sont fournis en annexe.

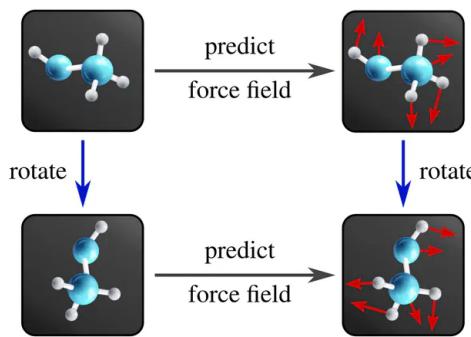


**Figure 2.2.1 – Principe de fonctionnement des réseaux de neurones de type MPNNs [9]**

### 2.3 EQUIVARIANCE

Mais, qu'est-ce qu'un réseau équivariant par rotation et réflexion ? L'équivariance permet au modèle de prédire correctement les propriétés d'une molécule même si celle-ci subit des rotations ou réflexions dans  $\mathbb{R}^3$ . Par exemple, si un modèle est entraîné avec une molécule dans une orientation spécifique, sera-t-il capable de prédire les propriétés de cette même molécule après rotation ? Cela nécessite de comprendre le principe théorique derrière ces modèles et comment chaque atome (nœud du graphe) apprend les caractéristiques de son environnement local.

Pour donner un exemple, considérons les forces interatomiques dans une molécule (Figure 2.3.1). Si la molécule est soumise à une rotation, le champ de forces doit également tourner de manière correspondante. Pour prédire avec précision un objet équivariant telle que la force, il est logique d'utiliser une architecture de réseau également équivariante. Il est important de préciser que ce modèle est équivariant face aux rotations mais invariant face aux translations.



**Figure 2.3.1 – Equivariance pour les forces interatomiques [10]**

Comme on peut l'observer dans l'Figure 2.3.2, l'équivariance implique que les caractéristiques internes du modèle sont modifiées de manière correspondante face aux rotations à l'entrée. Cette correspondance doit être respectée couche après couche afin de garantir une cohérence entre l'entrée et la prédiction finale.

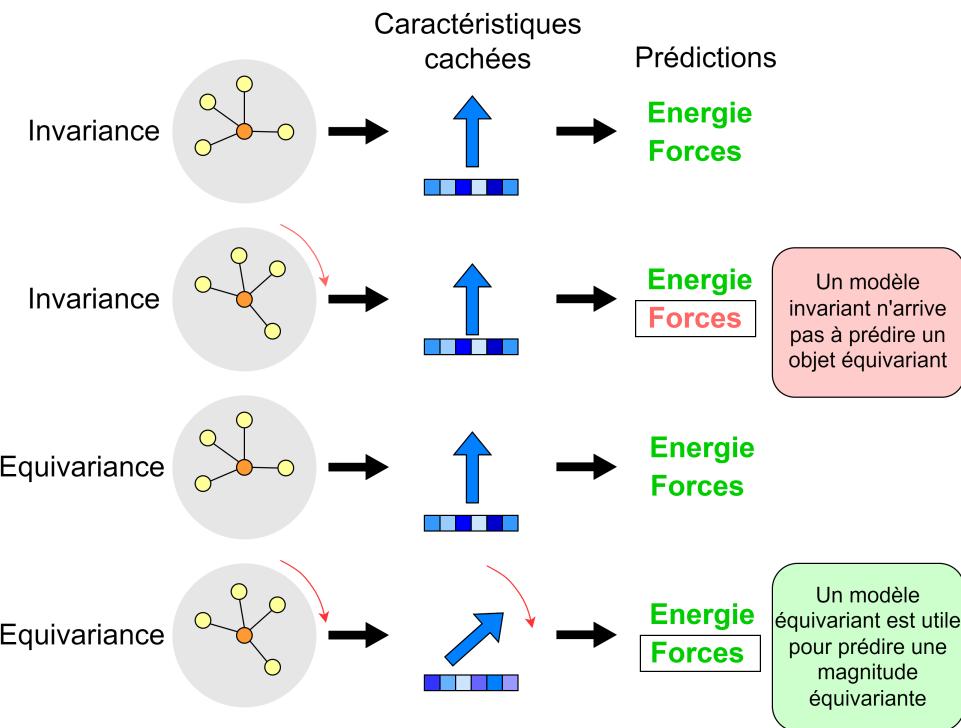


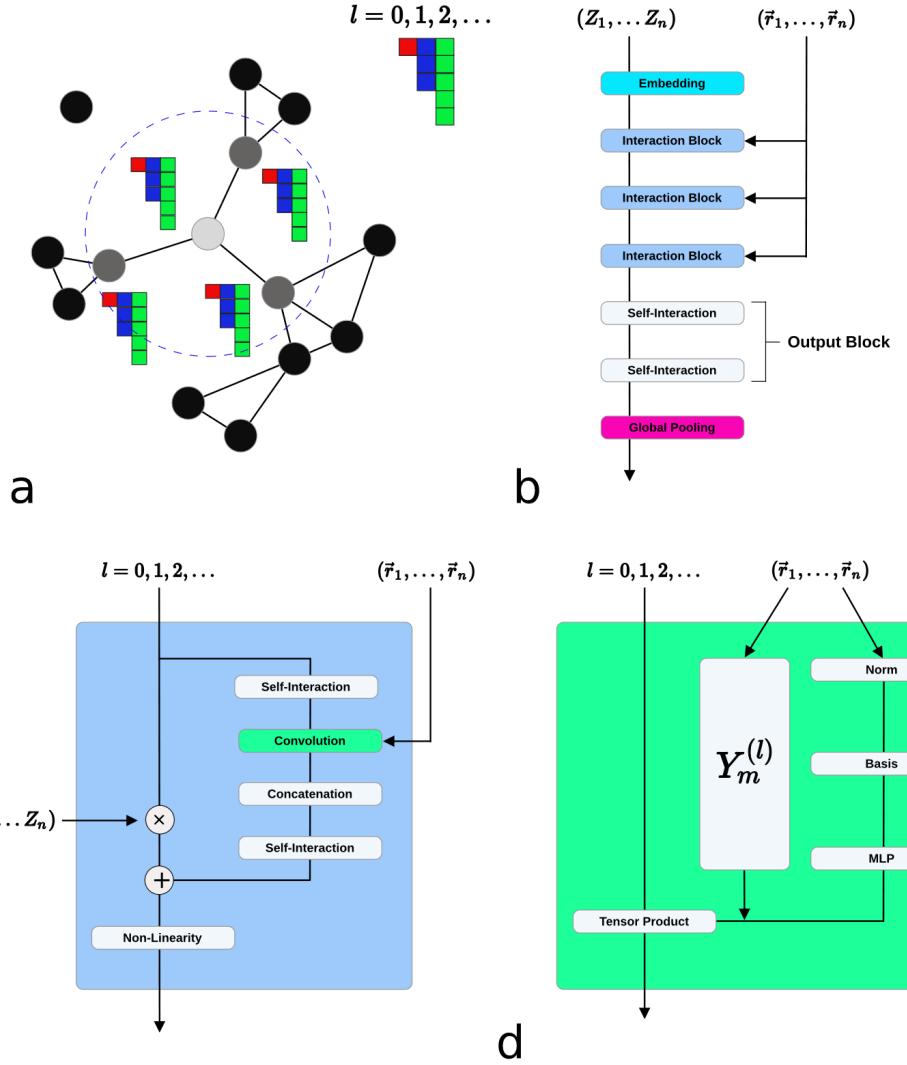
Figure 2.3.2 – Principe théorique de l'équivariance

Les méthodes existantes jusqu'ici étaient de type invariant et fonctionnaient avec des caractéristiques internes qui étaient des scalaires. C'est pourquoi cette nouvelle génération d'architectures est beaucoup plus performante que ses prédecesseurs.

## 2.4 ARCHITECTURE DE NEQUIP

L'architecture de Nequip repose sur des réseaux de neurones en graphe, conçus pour préserver l'équivariance par rapport aux rotations et réflexions, ainsi que l'invariance par rapport aux translations. L'objectif principal de ce modèle est de prédire l'énergie potentielle du système  $E$ , ainsi que les forces  $\mathbf{F}$ , qui sont obtenues par dérivation approximative du réseau de neurones sous la forme  $\mathbf{F}_i = -\nabla_{r_i} E$ . Cette approche garantit la conservation de l'énergie du système lors du calcul des forces.

Pour calculer le gradient de la sortie (énergie totale) par rapport aux entrées (positions atomiques) du modèle, on utilise la *backpropagation*, qui permet d'obtenir les gradients depuis les neurones du modèle. Ainsi, il est possible de définir un réseau de neurones prenant en entrée les positions atomiques et les espèces chimiques, et dont la sortie correspond à l'énergie potentielle totale du système.



**Figure 2.4.1 – Architecture de NequIP [8]**

Dans ce contexte, nous pouvons définir une fonction de coût à minimiser, donnée par :

$$\mathcal{L} = \lambda_E \|\hat{E} - E\|^2 + \lambda_F \frac{1}{3N} \sum_{i=1}^N \sum_{\alpha=1}^3 \left\| -\frac{\partial \hat{E}}{\partial r_{i,\alpha}} - F_{i,\alpha} \right\|^2$$

où  $\hat{E}$  représente la prédiction et  $E$  la valeur réelle (cible) de l'énergie totale, et où  $\partial \hat{E} / \partial r_i$  correspond à la prédiction des forces interatomiques pour l'atome  $i$ , avec  $F_i$  comme valeur réelle. La valeur réelle est connue dans le jeu de données d'entraînement. En général, le réseau neuronal utilise des fonctions différentiables dans le modèle qui dépendent des paramètres  $\theta$ . Il est alors possible

de calculer la dérivée de la fonction de perte par rapport aux paramètres  $\theta$ , et donc d'optimiser les paramètres à l'aide d'un algorithme tel que la descente de gradient pour minimiser la fonction de perte  $\mathcal{L}$ .

En entrée du modèle, les vecteurs d'information des atomes sont initialisés dans l'étape d'**embedding**. Ceci se fait sur chaque atome à partir d'une couche d'auto-intéraction qui opère sur les espèces chimiques  $Z_i$ .

Comme le montre la Figure 2.4.1 (a), à chaque couche, chaque atome contient des vecteurs d'information, désormais appelés *caractéristiques internes*, qui représentent ses propres informations combinées à celles des atomes voisins au cours des différentes étapes du modèle. Chaque atome est associé à une sphère de rayon  $D$ , qui délimite la région dans laquelle tous les atomes sont pris en compte pour les opérations d'agrégation entre nœuds. Ainsi, à mesure que le nombre de couches  $N_{couches}$  augmente, le champ de réception (c'est-à-dire le nombre d'atomes sur lesquels un atome unique exerce une influence) croît proportionnellement à  $N_{couches}D$ .

Les blocs d'**interaction** (2.4.1 (b)) sont au cœur de l'architecture. À chaque couche, les atomes et leurs voisins sont combinés via des opérations de **convolution** (2.4.1 (c)). Ces convolutions ne sont pas triviales : elles utilisent une base de fonctions équivariantes, les harmoniques sphériques, pour modéliser les interactions spatiales entre atomes (Figure 2.4.2). En effet, chaque lien peut être modélisé comme étant une fonction égale à 1 dans la direction du lien et à zéro ailleurs, et différents degrés de l'harmonique sphérique ( $\ell = 0, 1, 2, \dots$ ) de ces fonctions peuvent être utilisés pour obtenir cette représentation. Ainsi, les vecteurs reliant deux atomes  $a$  et  $b$ ,  $\mathbf{r}_{ab} = \mathbf{r}_a - \mathbf{r}_b$ , sont décomposés en utilisant ces harmoniques sphériques  $Y_m^{(\ell)}$ , qui sont équivariants par l'action du groupe  $E(3)$ , qui est le groupe de rotations, réflexions et translations en 3D. En particulier, le modèle est aussi invariant face aux translations.

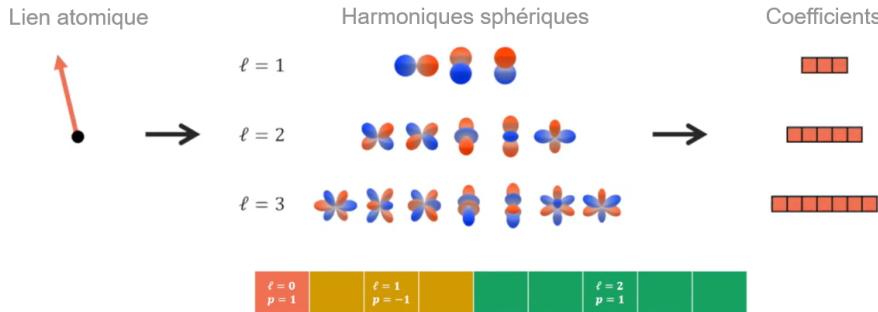


Figure 2.4.2 – Harmoniques sphériques

Les coefficients des filtres de convolution, appliqués à chaque harmonique sphérique, sont appris par le modèle à travers un réseau de neurones auxiliaire de type Multi Layer Perceptron (MLP). Pour compléter l'opération de convolution, il faut effectuer le produit tensoriel entre les caractéristiques

internes et les filtres dérivés des harmoniques sphériques 2.4.1. À ce stade, il est pertinent d'établir un parallélisme avec les MPNNs. En effet, dans le contexte des réseaux de neurones équivariants, la construction des filtres peut être vue comme une construction de messages, similaire à ce qui se fait dans les MPNNs. Dans ce cadre, le produit tensoriel joue le rôle de l'opération d'agrégation, réalisée à travers tous les voisins de chaque atome.

Le bloc d'interaction se termine par une opération d'agrégation entre le produit tensoriel et l'entrée initiale du bloc. Cette technique est souvent appelée "bloc résiduel" (Residual block), car la sortie est obtenue en additionnant l'entrée au résultat obtenu. Ces blocs offrent l'avantage de permettre au modèle d'ajuster en temps réel le nombre effectif de couches optimal pour le problème, offrant ainsi une plus grande flexibilité d'apprentissage.

Ensuite, les couches d'**auto-interaction** consistent à transformer les sorties des blocs d'interaction avec des poids  $W$  à apprendre par le modèle.

Finalement, la phase de **Global Pooling** combine la sortie de tous les atomes pour produire un vecteur de caractéristiques unique qui peut être utilisé dans les couches suivantes pour faire des prédictions.

Il est important de noter qu'il existe une normalisation de l'énergie et des forces à l'entrée du modèle. Lorsque le modèle est utilisé pour l'inférence, cette normalisation est défaite à la sortie.

La question se pose alors : pourquoi ce modèle est-il équivariant ? Pour y répondre, il faut revenir aux harmoniques sphériques. Ces fonctions ont la propriété d'être équivariantes par rapport aux rotations. Il est possible de démontrer que toutes les opérations impliquées dans le modèle, y compris la non-linéarité utilisée à la sortie de chaque couche, forment une structure globalement équivariante. L'équivariance est donc maintenue dans l'ensemble du modèle, et les forces seront prédites correctement en cas de rotations à l'entrée.

En annexe, des justifications plus rigoureuses de ces modèles sont données.

# 3

## Estimation d'incertitude

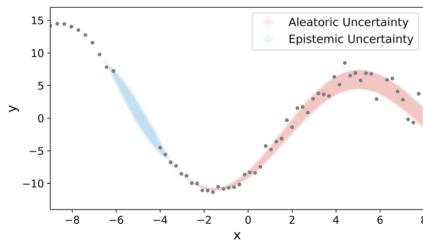
### 3.1 INTRODUCTION

L'estimation de l'incertitude dans un réseau de neurones est essentielle lorsque ces modèles sont utilisés pour des applications complexes, telles que l'apprentissage actif ou la découverte de nouveaux matériaux. Pour d'autres applications critiques, notamment dans le domaine médical, il est crucial de disposer de méthodes capables de différencier en toute sécurité les prédictions non fiables.

Dans ce travail, les efforts seront principalement concentrés sur l'estimation de l'incertitude dans les modèles d'intelligence artificielle pour la prédiction des propriétés moléculaires. Différentes méthodes seront évaluées et comparées afin d'identifier les approches les plus efficaces. Dans tous les cas, les analyses seront effectuées sur le code de NequIP.

L'incertitude peut être classée en deux catégories principales : aléatoire et épistémique. Comme illustré dans la figure 3.1.1, l'incertitude aléatoire correspond à celle induite par les données elles-mêmes, tandis que l'incertitude épistémique est liée aux régions du domaine d'entrée où les données sont absentes, ce qui devrait entraîner une incertitude accrue du modèle. Pour l'apprentissage actif, il est plus pertinent de calculer l'incertitude épistémique, car elle permet de détecter les points qui n'appartiennent pas au jeu de données d'entraînement.

Un autre point pertinent est la propriété de localité de ces méthodes. En effet, il existe un besoin de



**Figure 3.1.1 – Incertitude épistémique et aléatoire [11]**

développer des méthodes locales, c'est-à-dire des méthodes où l'incertitude puisse être prédite au niveau atomique. Cela est crucial car il est nécessaire de pouvoir identifier quels atomes présentent une incertitude plus élevée, afin de les détecter et de leur appliquer un traitement différencié. Par ailleurs, les méthodes globales (qui ne réalisent pas de prédictions locales) disponibles à ce jour ne parviennent pas à fournir une incertitude globale indépendante du nombre d'atomes, ce qui pose problème lorsqu'il s'agit de comparer des structures de tailles atomiques différentes.

## 3.2 MÉTHODES D'ESTIMATION

Pour les réseaux de neurones utilisés dans la simulation atomique, il existe différentes familles de méthodes d'estimation de l'incertitude (Figure 3.2.1). Pour plus de détails sur les fondements mathématiques de ces méthodes, regarder l'annexe.

### 3.2.1 ENSEMBLES PROFONDS

Ces méthodes consistent à entraîner différentes versions du modèle [13], chacune avec une initialisation des paramètres différente, ce qui conduit à des comportements internes variés. Lors de l'évaluation du modèle avec une nouvelle structure atomique, celle-ci est effectuée sur tous les modèles existants. Si les prédictions présentent une variance élevée, cela suggère que le modèle n'est pas familier avec cette molécule ou ce matériau.

### 3.2.2 MÉTHODES BASÉES SUR LA DISTANCE

Ces méthodes tentent d'estimer une densité de probabilité dans l'ensemble de données d'apprentissage, dans l'idée de l'utiliser pour évaluer un ensemble de données de testing. Lorsqu'on teste une nouvelle structure, il devient possible de déterminer si elle est similaire ou différente des molécules utilisées pour l'entraînement. Cependant, il est difficile de comparer directement des structures atomiques, en particulier lorsqu'elles n'ont pas le même nombre d'atomes. Une approche promet-

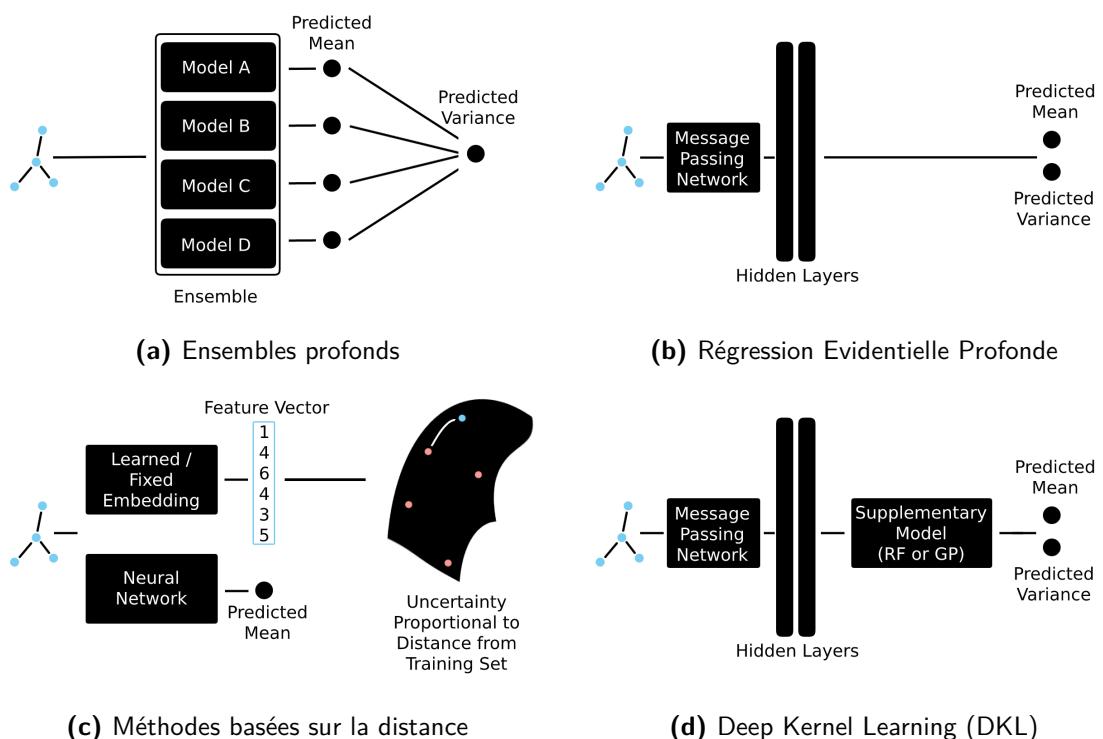


Figure 3.2.1 – Familles des méthodes de quantification d'incertitude [12]

teuse consiste à estimer la distribution de densité dans l'espace latent (espace des caractéristiques internes), généralement à travers la dernière couche du réseau. Nous examinerons une méthode qui ajuste un mélange de gaussiennes (GMM) à ces caractéristiques internes, au niveau de chaque atome.

Autrement dit, si  $\mathbf{h}_i$  représente le vecteur des caractéristiques internes de l'atome  $i$ , l'incertitude peut être estimée par l'expression suivante [14].

$$NLL(h_i|H_i) = -\log \left( \sum_{m=1}^M \mathcal{N}(x|\mu_m, \Sigma_m) \right),$$

où  $\mathbf{H}_i$  désigne les caractéristiques internes d'un certain atome  $i$  déduites du jeu de données d'entraînement. Le nombre de gaussiennes  $M$  est sélectionné à l'aide du critère d'information bayésien (BIC). Les poids  $w$ , ainsi que les paramètres  $\mu$  et  $\Sigma$  (moyenne et matrice de covariance), sont déterminés à partir des données disponibles en utilisant la théorie des mélanges gaussiens, une méthode bien établie et abondamment documentée dans la littérature [15].

La complexité de cette méthode est caractérisée par :

1. Une complexité cubique en fonction du nombre de dimensions (ici, il y a 16 dimensions, corres-

pondant à la dimension de  $h$ ).

2. Une complexité linéaire par rapport au nombre de gaussiennes utilisées ( $M$ ).
3. Une complexité linéaire par rapport au nombre de données utilisées lors de l'entraînement.

Par complexité, nous faisons référence au temps d'entraînement. L'inférence, quant à elle, est supposée être très rapide.

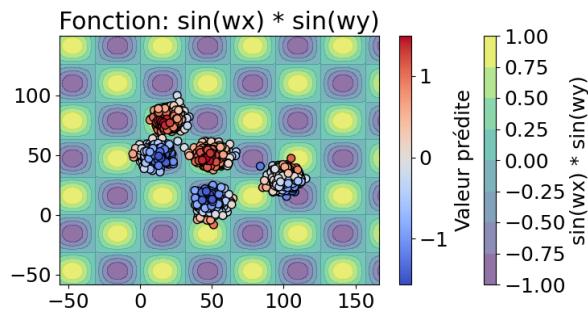
### 3.2.3 RÉGRESSION ÉVIDENTIELLE PROFONDE

Cette technique fait partie d'une famille plus large de méthodes où l'estimation de la moyenne et de la variance est réalisée en parallèle avec les prédictions du modèle. Ainsi, le réseau de neurones est capable de prédire non seulement la sortie attendue, mais aussi la variance associée. Contrairement à d'autres approches, telles que celle décrite dans [16], ce travail propose une méthode locale pour l'estimation de l'incertitude, en la prédisant spécifiquement au niveau de chaque atome à partir des forces interatomiques. Comme détaillé dans l'annexe et dans l'article [17], les étapes pour mettre en œuvre ce développement les suivantes :

1. On suppose une distribution gaussienne pour la quantité à prédire avec le modèle, dans notre cas les forces. Autrement dit,  $\bar{F}_i \sim N(\bar{\mu}_i, \sigma_i^2)$ .
2. En parallèle, d'autres distributions sont supposées sur les paramètres  $\mu$  et  $\sigma^2$ , avec  $\bar{\mu}_i \sim N(-\nabla_{r_i} E, \sigma^2 v^{-1})$  et  $\sigma^2 \sim \Gamma^{-1}(\alpha, \beta)$ .
3. L'objectif est d'utiliser un réseau de neurones pour maximiser  $p(\bar{F}_i | -\nabla_{r_i} E, v, \alpha, \beta)$ , ce qui peut être calculé en utilisant la théorie probabiliste. Les sorties du modèle sont donc  $-\nabla_{r_i} E, v, \alpha$ , et  $\beta$ , tandis que  $\bar{F}_i$  sont des observations disponibles dans la base de données d'entraînement. Un terme lié à l'énergie est également ajouté à la fonction de perte.

Cette méthodologie semble efficace pour les points situés dans la distribution des données d'entraînement (In-Distribution, IN), mais des doutes légitimes subsistent quant à sa capacité à détecter les points hors distribution (Out-of-Distribution, OOD). Les points OOD sont des points très éloignés de ceux utilisés pendant l'entraînement, qui devraient présenter une incertitude épistémique élevée. En pratique, il peut être difficile de détecter ces points avec un modèle qui s'appuie sur un réseau de neurones pour prédire l'incertitude.

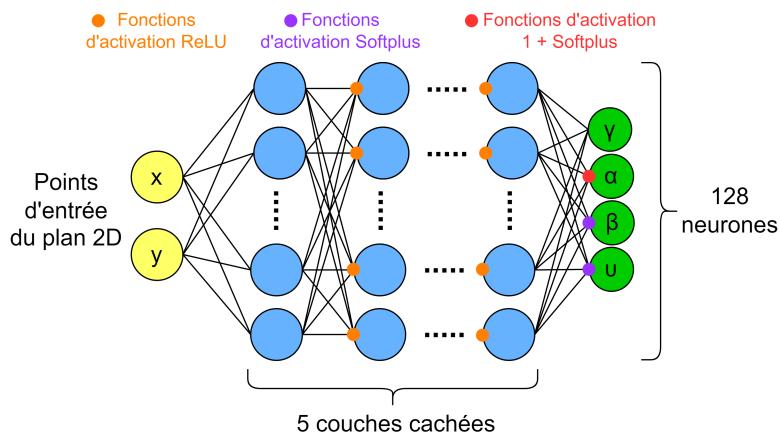
Pour une explication technique, on peut se référer à la référence [18], qui explique que l'efficacité de cette méthode est due à la manière dont le modèle converge. Il semble en effet que le modèle converge plus rapidement autour du centre du nuage de points d'entraînement, ce qui implique un gradient d'erreur plus prononcé vers la frontière. En choisissant un terme approprié comme régulateur dans la fonction de perte, il est possible d'exploiter ce comportement pour obtenir une incertitude qui "explose" en dehors du nuage de points. Cependant, si cette hypothèse est correcte,



**Figure 3.2.3 – Dataset utilisé pour le modèle de régression évidentielle**

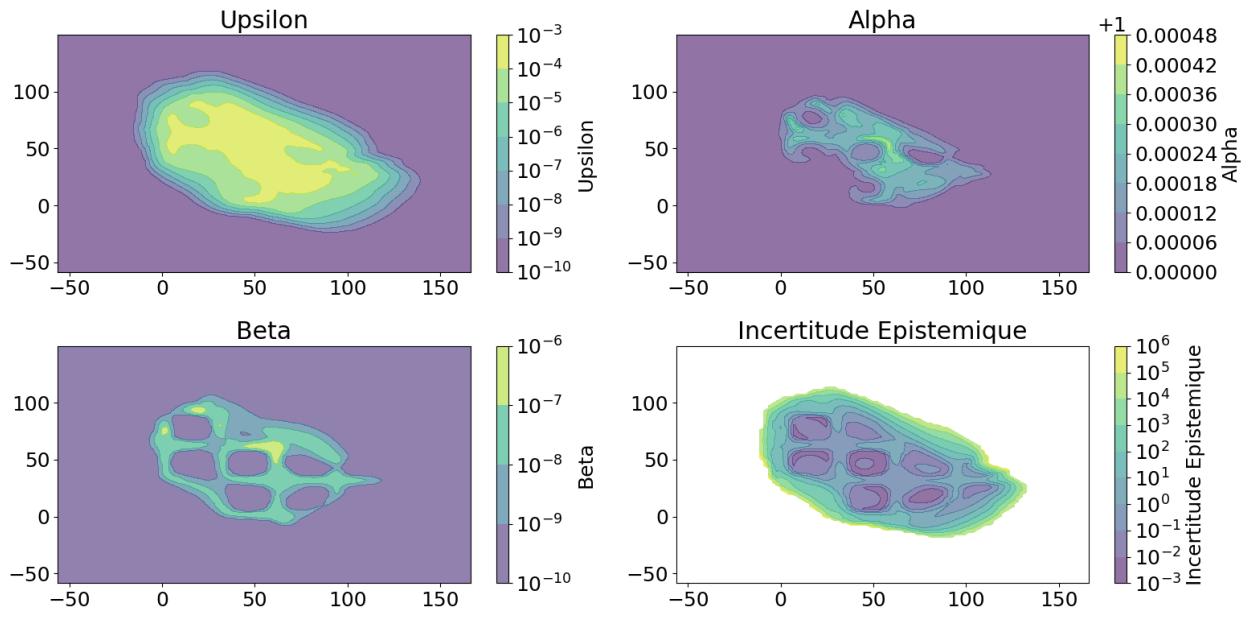
une faiblesse du modèle devient apparente : si le nombre de points est insuffisant pour définir une frontière, comment le modèle peut-il faire des prédictions fiables ?

Étant donné que nous travaillons avec un modèle où le nuage de points est défini dans un espace  $\mathbb{R}^n$ , il n'est pas facile de visualiser ce comportement complexe. Pour surmonter cette difficulté, nous nous tournons vers un cas d'étude plus simple en deux dimensions. Dans ce contexte, nous utiliserons un modèle MLP (Perceptron Multicouche, ou Multi-Layer Perceptron) pour effectuer des prédictions d'incertitude basées sur la régression évidentielle, comme illustré à la Figure 3.2.2.



**Figure 3.2.2 – Architecture d'un modèle simplifié**

Le dataset utilisé pour l'entraînement se compose simplement de 5 clusters, où la valeur de chaque point est donnée par une fonction sinusoïdale dans l'espace 2D (regarder Figure 3.2.3). Les résultats sont affichés dans la Figure 3.2.4. Il est important de noter que la couleur blanche dans les affichages correspond à des valeurs "nan" (not a number), car ces valeurs sont virtuellement infinies. Les para-

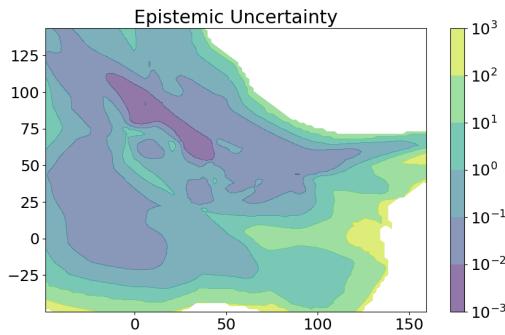


**Figure 3.2.4 – Régression évidentielle dans un cas simple en 2D - 100 epochs et 10000 points**

mètres de sortie ( $v$ ,  $\alpha$ ,  $\beta$ ) sont affichés, ainsi que l'incertitude prédictive issue de ces paramètres.

Très rapidement, une faille importante de ce modèle se révèle : la sur-paramétrisation pousse le modèle à prédire l'incertitude de manière incohérente par rapport à ce qui est attendu pour chaque paramètre. Par exemple, la valeur  $\alpha$  semble chaotique et ne présente aucune correspondance logique avec les clusters définis, et  $\beta$  paraît détecter un cluster en réalité inexistant. Cela suggère que le modèle choisit les prédictions de ces paramètres de manière "chaotique" simplement pour satisfaire l'objectif imposé par la fonction de perte. Cependant, ce n'est pas toujours le cas, car la valeur de  $v$  semble être en corrélation avec l'évidence présente dans l'ensemble de données d'apprentissage, ce qui était attendu de la description théorique de la régression évidentielle. Ici, l'évidence est une quantité qui devrait être plus élevée lorsqu'il y a plus de points d'entraînement dans un endroit spécifique du domaine des données.

Lorsque le nombre de points par cluster est réduit, et le nombre d'epochs est augmenté de manière à ce que  $n_{\text{epochs}} * n_{\text{points}}$  reste constant par rapport à l'entraînement avec 10000 points, il devient évident que les résultats se détériorent, comme on peut le voir dans la figure 3.2.5. Cela démontre clairement qu'une diminution du nombre de points empêche le modèle de détecter correctement les frontières des nuages de points, et par conséquent, d'estimer correctement l'incertitude épistémique.



**Figure 3.2.5 – Régression évidentielle dans un cas simple en 2D - 10000 epochs et 100 points**

Ces inconvénients peuvent avoir un impact négatif dans les applications pour lesquelles la prédiction de l'incertitude est requise. Plus de détails sur le principe mathématique de cette méthode sont donnés en annexe.

### 3.2.4 PROCESSUS GAUSSIEN

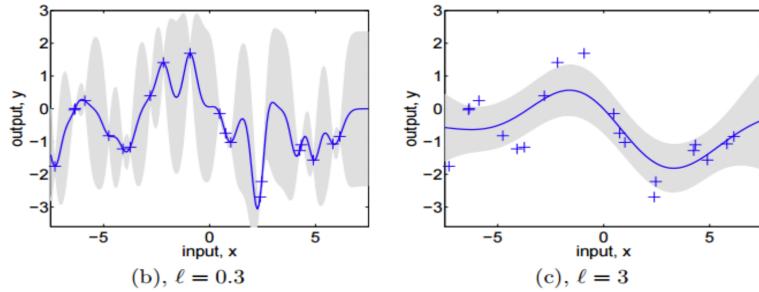
Les réseaux de neurones sont capables d'apprendre des relations complexes dans les données qu'ils traitent. Parallèlement, les processus gaussiens sont reconnus pour leur capacité à prédire la distribution de probabilité d'un jeu de données avec un haut degré de précision [19]. Cette méthode propose donc de combiner ces deux approches, en connectant un processus gaussien à la sortie du réseau de neurones. Cependant, pour rendre le processus gaussien différentiable par rapport à ses paramètres, nous devons utiliser une variante des processus gaussiens connue sous le nom de processus gaussien variationnel. Ainsi, le processus gaussien admet l'optimisation par gradients (backpropagation), et pourra être entraîné en combinaison avec le réseau neuronal.

En utilisant les variables définies précédemment, il est clair que nous cherchons un processus gaussien tel que que l'énergie potentielle totale  $E$  puisse être modélisé par un processus gaussien dont l'entrée est l'espace latent,  $\mathbf{h}$ .

$$E \sim \mathcal{GP}(m(\mathbf{h}), K(\mathbf{h}))$$

Un noyau de type RBF (Radial Basis Function) est choisie pour mesurer la covariance entre les points d'entrée. Ce noyau dépend d'un paramètre à entraîner, le *lengthscale*, qui permet de définir à quelle distance deux points sont corrélés (par exemple, si la distance entre deux points d'entrée est quatre fois supérieure au *lengthscale*  $l$ , alors ces points sont considérés comme décorrélés). Autrement dit, plus le *lengthscale* est petit, plus le processus gaussien aura tendance à faire du surapprentissage sur les données (voir figure 3.2.6). Cependant, c'est le modèle lui-même qui choisit

le *lengthscale* afin de minimiser la fonction de coût.

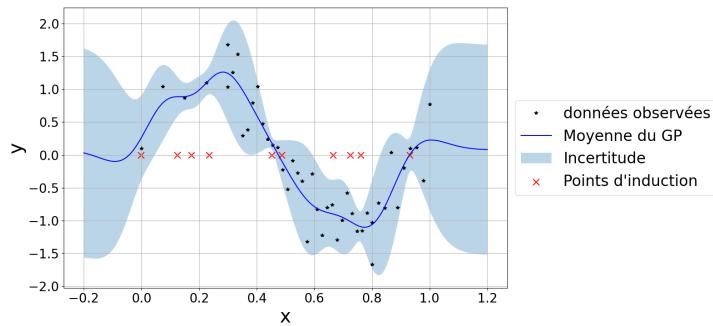


**Figure 3.2.6 –** Effet du *lengthscale* sur la prédition du processus gaussien [20]

Ensuite,  $m$  variables d’induction sont ajoutées aux  $n$  points d’entraînement utilisés dans la formulation de cette méthode. Ces variables sont importantes car, sans elles, la complexité serait  $\mathcal{O}(n^3)$ , tandis que si on utilise  $m$  points d’induction, la complexité chute à  $\mathcal{O}(nm^2)$ . Les points d’induction sont des points appartenant à l’espace d’entrée mais qui ne doivent pas nécessairement faire partie de la base de données d’entraînement, et pour lesquels il n’est pas nécessaire de connaître les prédictions de sortie. La réduction de la complexité provient du fait qu’avec l’ajout des points d’induction, il devient possible de se passer de l’inversion d’une matrice de dimension  $n$  — une opération coûteuse dans la formulation originale des processus gaussiens (voir l’annexe pour plus de détails).

Le modèle est finalement entraîné en utilisant une fonction de coût qui mesure à quel degré l’approximation faite pour les points d’induction est cohérente avec les données disponibles. De plus, cela permet d’obtenir une fonction de coût différentiable et donc d’entraîner le modèle en calculant les gradients par rapport aux paramètres.

Dans la figure 3.2.7, un exemple simple d’application en 2D est présenté. Comme on peut le voir, le modèle parvient à fournir une distribution probabiliste correcte sur la prédition effectuée, en choisissant les bons points d’induction et les bons paramètres du modèle.



**Figure 3.2.7 –** Inférence variationnelle avec un processus gaussien, en utilisant points d’induction

### 3.3 DÉTAILS DU DÉVELOPPEMENT

Dans les figures 3.3.1 et 3.3.2, les architectures des différentes méthodes de calcul d'incertitude basées sur l'architecture NequIP sont présentées.

Pour les ensembles profonds, nous observons que la prédiction est réalisée en entraînant plusieurs modèles similaires avec des initialisations différentes. La fonction de perte classique est utilisée pour l'entraînement, et la variance de toutes les prédictions est prise en compte pour estimer l'incertitude.

Le modèle GMM repose sur l'utilisation d'un seul modèle pré-entraîné. Pendant l'entraînement du GMM, le modèle est figé, c'est-à-dire que ses poids ne sont pas modifiés. La représentation cachée  $h_{rep}$  est utilisée pour ajuster une fonction de densité à l'aide des données d'entraînement, permettant ainsi de mesurer l'incertitude via la Log-vraisemblance Négative (Negative Log-Likelihood, NLL). Ici, la fonction de perte du modèle initial reste celle habituellement utilisée avec NequIP. Les principes théoriques sous-jacents sont décrits en détail dans [14]. Le code de cette méthode est disponible dans la branche de développement du projet NequIP.

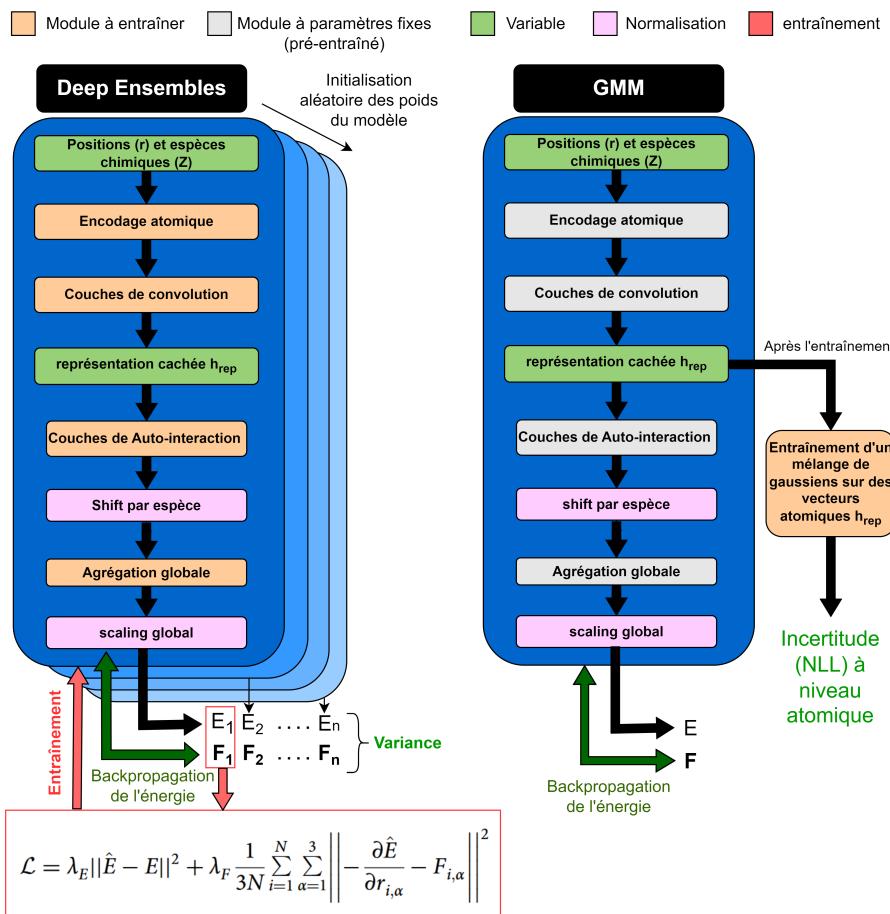


Figure 3.3.1 – Illustration des architectures des ensembles profonds et du modèle GMM.

La régression évidentielle implique une modification du modèle, en particulier de la fonction de perte, afin de prendre en compte les suppositions bayésiennes sur la sortie. Trois sorties additionnelles,  $v$ ,  $\alpha$  et  $\beta$ , sont générées en parallèle à l'énergie atomique dans les couches d'auto-interaction [17]. Des modifications ont été effectuées pour prédire l'incertitude à niveau atomique à partir des forces inter-atomiques, d'une façon très similaire à [21].

Enfin, les processus gaussiens opèrent, comme expliqué dans les annexes, avec des variables d'induction obtenues à partir du vecteur  $h_{rep}$  du modèle original. Les couches d'auto-interaction sont remplacées par le processus Gaussien, qui reçoit en entrée chaque vecteur atomique  $h_{rep}$  et produit la distribution de l'énergie prédictive au niveau local. De cette distribution, la variance est extraite et envoyée en sortie du modèle. Les opérations subséquentes sont réalisées sur cette distribution, et le résultat est finalement envoyé à la fonction de perte. Dans ce modèle, seuls les points d'induction et le processus gaussien sont entraînables, car le reste du modèle est figé.

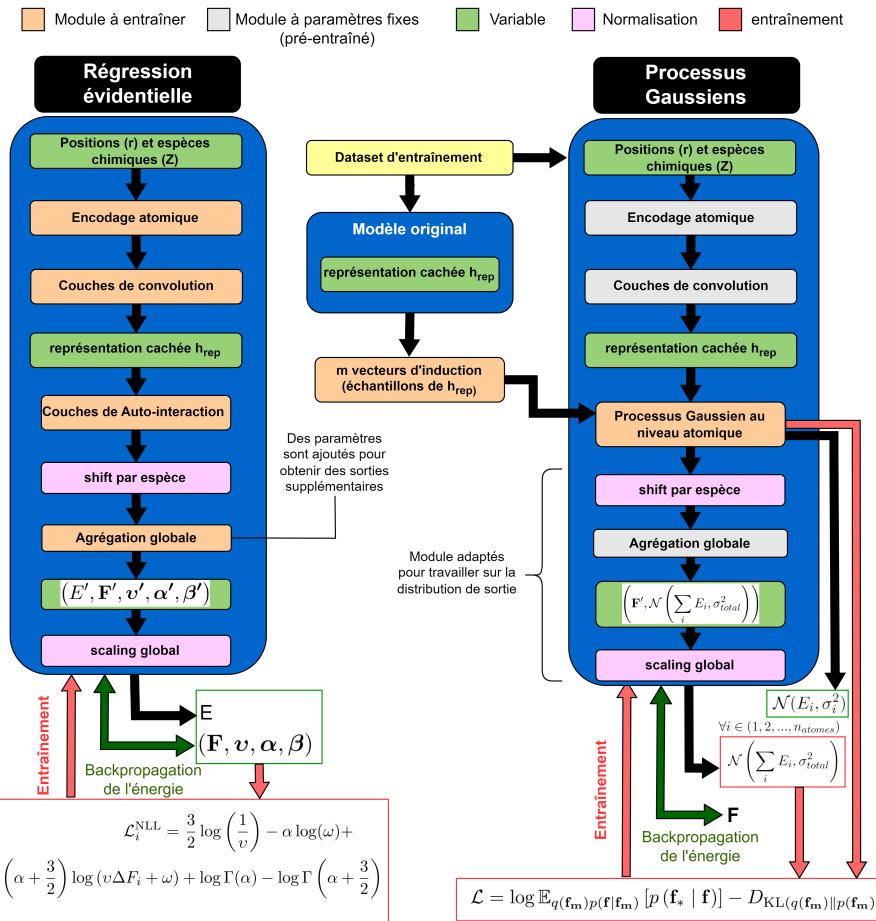


Figure 3.3.2 – Illustration des architectures pour la régression évidentielle et les processus gaussiens.

## 3.4 RÉSULTATS

Pour les résultats, les abréviations suivantes seront utilisés : Ensembles profonds (DE), modèle de mélange de gaussiennes (GMM), Régression Evidentielle Profonde Localisé (LEDL), Processus Gaussiens (GP).

Les analyses sont faites avec des datasets basés sur des molécules isolées, car ils sont plus simples et rapides à traiter, et parce que ce sont les données normalement utilisées dans la littérature. Cependant, à terme l'idée est d'évaluer ces méthodes pour des applications sur des matériaux.

### 3.4.1 ENTRAINEMENT

comme vu précédemment, les différentes méthodes ont dans tous les cas été basés sur le modèle Nequip. Les hyperparamètres utilisés pour chaque modèle sont donnés en annexe.

Pour un entraînement classique, il est possible d'analyser avec une échelle logarithmique comment la fonction de perte décroît pour l'entraînement aussi que pour la validation (Figure 3.4.1).

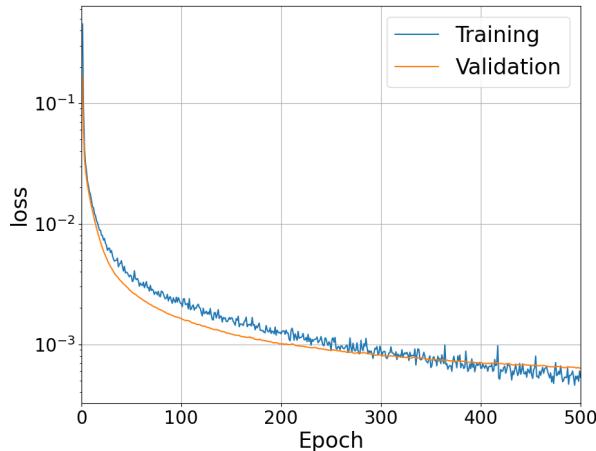


Figure 3.4.1 – Fonction de perte

### 3.4.2 3BPA

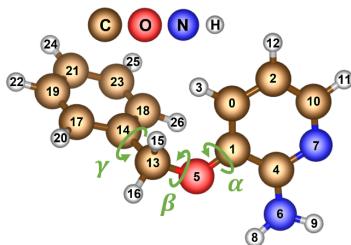
La molécule 3-(benzyloxy)pyridin-2-amine (3BPA), représentée dans l'Figure 3.4.2, est une molécule composée de C, O, N et H. Cette molécule présente de nombreuses configurations géométriques en fonction de trois angles,  $\gamma$ ,  $\beta$  et  $\alpha$ , ce qui la rend idéale pour analyser les méthodes de quantification d'incertitude proposées. Un ensemble de données composé de 500 structures de cette molécule issus de simulations de dynamique moléculaire à 300 K sera utilisé (450 pour l'entraînement

et 50 pour la validation). Pour les tests, trois ensembles de données de 200 molécules chacun seront utilisés, chacun à une température différente : 300 K, 600 K et 1200 K. À mesure que la température augmente, on peut s'attendre à des configurations moléculaires plus instables, avec des déplacements atomiques plus importants. Étant donné que le modèle a été entraîné à 300 K, on s'attend à des prédictions d'incertitude plus élevées à 600 K et 1200 K.

Dans la figure 3.4.3, les résultats pour toutes les méthodes sont présentés. Pour chaque molécule, chaque atome a une prédiction de la force interatomique  $\hat{F}$  qu'il ressent, ainsi qu'une prédiction d'incertitude. De plus, les vraies forces interatomiques  $F$  sont à disposition dans le jeu de données utilisé. Si l'axe  $y$  représente l'incertitude atomique et l'axe  $x$  représente l'erreur commise,  $||\hat{F} - F||$ , il est alors possible de positionner chaque point dans ce plan 2D. En répétant cette opération pour tous les atomes de toutes les molécules, il devient possible de construire un histogramme pour la relation incertitude-erreur en 2D. Cette analyse est répétée pour toutes les méthodes.

Une question importante est de savoir quelles sont les caractéristiques désirables pour une métrique d'incertitude :

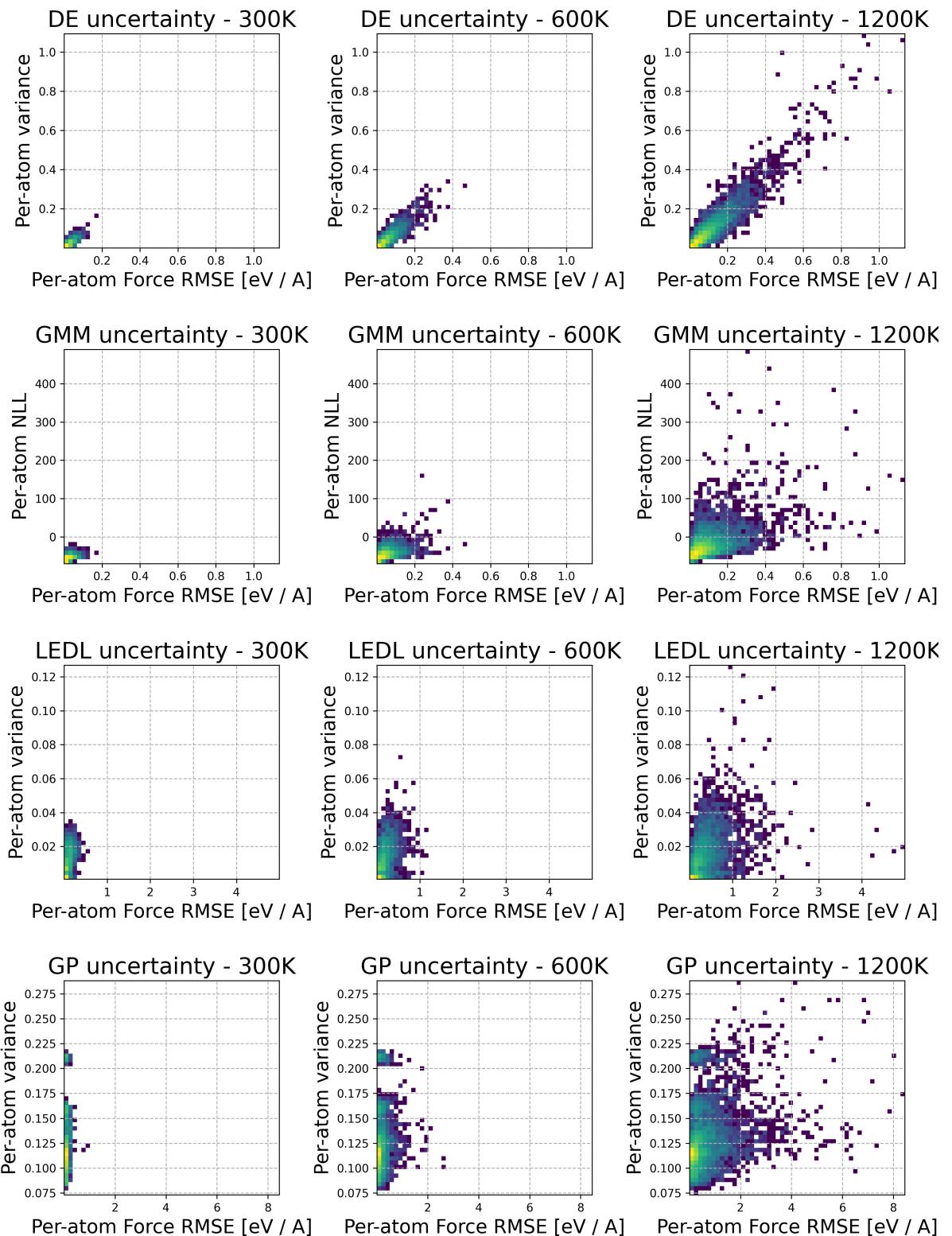
- La relation entre l'incertitude et l'erreur doit être la plus linéaire possible. Si l'histogramme est linéaire, cela signifie que la prédiction d'incertitude permet de bien différencier les points avec une faible et une forte erreur. On suppose qu'une forte erreur correspond à des points hors distribution (OOD).
- À des températures plus élevées, l'incertitude devrait augmenter.



**Figure 3.4.2 –** Représentation de la molécule 3BPA [14]

Les conclusions suivantes peuvent être tirées des résultats obtenus :

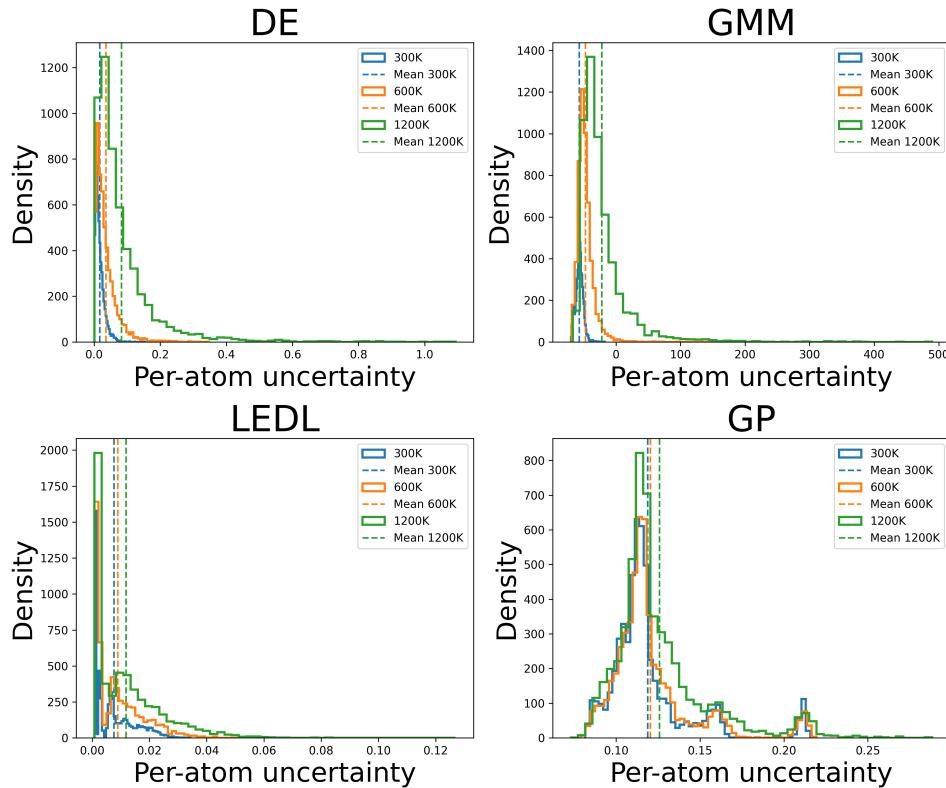
- Les ensembles profonds montrent les meilleurs résultats globaux : les prédictions d'incertitude sont très linéaires par rapport à l'erreur, et l'incertitude augmente avec la température.
- Le GMM montre des résultats très positifs, bien que la linéarité soit moins marquée que pour les ensembles profonds.
- La régression évidentielle ne semble pas respecter la condition de linéarité, mais l'incertitude augmente avec la température.



**Figure 3.4.3 – Résultats des différentes méthodes appliquées à la molécule 3BPA**

- En ce qui concerne les processus gaussiens, les résultats sont mauvais : l'incertitude n'est pas linéaire et il y a une très faible différence lorsque la température augmente ! Une analyse plus approfondie de la méthode et de l'implémentation est nécessaire.

À partir des résultats précédents, il est possible de générer des histogrammes qui représentent les valeurs d'incertitude à chaque température.



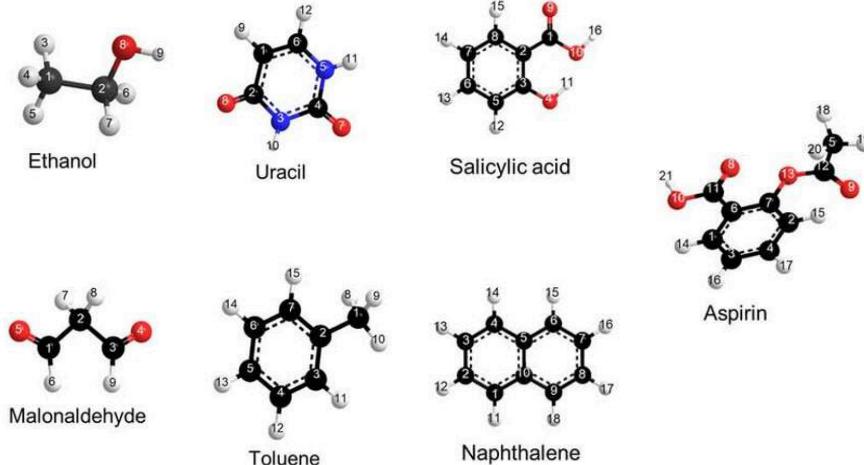
**Figure 3.4.4 – Hisogrammes de l'incertitude prédictive pour chaque méthode et température**

Ces histogrammes permettent de comprendre à quel point l'incertitude augmente avec la température. En effet, il s'avère que les moyennes des incertitudes sont dans l'ordre attendu (300 K - 600 K - 1200 K) pour les quatre méthodes. Cependant, pour la régression évidentielle et les processus gaussiens, cette différence est plus subtile.

### 3.4.3 MD17

Dans ce cas, nous disposons de datasets correspondant à 7 molécules différentes (voir figure 3.4.5). Bien qu'elles partagent certains aspects en commun, ces molécules sont différentes et ne peuvent donc pas être modélisées de la même manière. L'exercice, similaire à celui de [22], que nous réaliserons par la suite consiste à entraîner 7 modèles différents, chacun avec une méthode

d'estimation d'incertitude implémentée, dans le but de détecter les éléments qui n'appartiennent pas à la distribution. Par exemple, si un modèle est entraîné sur l'aspirine et qu'il est testé séparément sur l'aspirine et l'éthanol, on s'attend à ce que l'estimation de l'incertitude soit beaucoup plus élevée pour l'éthanol (car le modèle n'a jamais vu une telle molécule). Autrement dit, l'objectif est de détecter les données hors distribution (OOD) qui sont absentes du dataset d'entraînement.



**Figure 3.4.5 – Illustration des 7 molécules du dataset MD17**

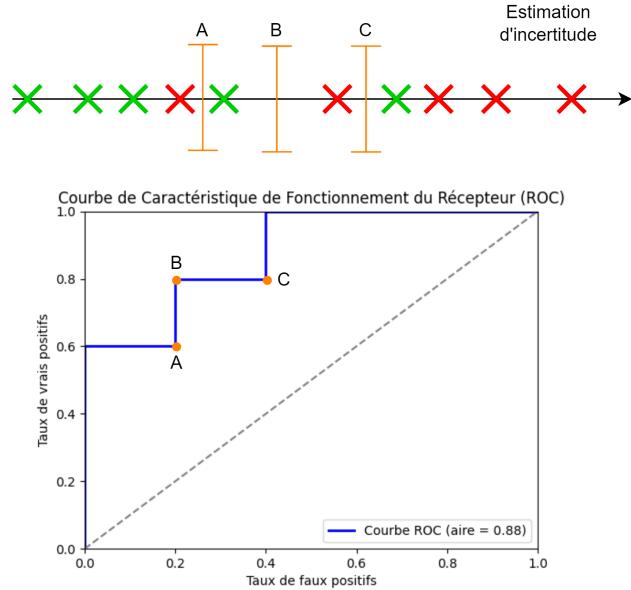
Cependant, la question se pose : comment peut-on mesurer si la classification est bonne ? Pour cela, on peut utiliser les courbes AUC-ROC qui permettent d'évaluer la qualité d'une métrique. Dans ce cas, on utilise la métrique "incertitude" et on considère que toutes les molécules avec une incertitude supérieure à  $\sigma_{\text{cutoff}}$  seront considérées comme OOD ( $y_{\text{pred}} = \mathbf{1}(\sigma_x > \sigma_{\text{OOD}})$ ). Par ailleurs, nous connaissons les véritables molécules OOD,  $y_{\text{vrai}} = \mathbf{1}(x \text{ est OOD})$ , ce qui permet de comparer les classifications  $y_{\text{vrai}}$  et  $y_{\text{pred}}$ . Dans ce contexte, une donnée OOD sera considérée comme un "positif" dans la terminologie de la classification.

Les métriques suivantes sont alors définies. Notez qu'elles dépendent de la valeur de  $\sigma_{\text{cutoff}}$ .

$$\begin{aligned} \text{TPR (True Positive Rate)} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \\ \text{FPR (False positive rate)} &= \frac{\text{FP}}{\text{FP} + \text{TN}} \end{aligned}$$

Évidemment, ces métriques seront optimisées lorsque  $\text{TPR} = 1$  et  $\text{FPR} = 0$ . En faisant varier la valeur de  $\sigma_{\text{cutoff}}$ , cela définit une courbe croissante en fonction du TPR et du FPR, comme illustré dans la figure 3.4.6. Dans le cas d'une classification parfaite, la courbe ROC passera par le point ( $\text{FPR} = 0, \text{TPR} = 1$ ) et l'aire sous la courbe sera exactement égale à 1. Si la classification est aléatoire, on obtiendra la diagonale en gris, et si la classification est parfaitement incorrecte, la courbe

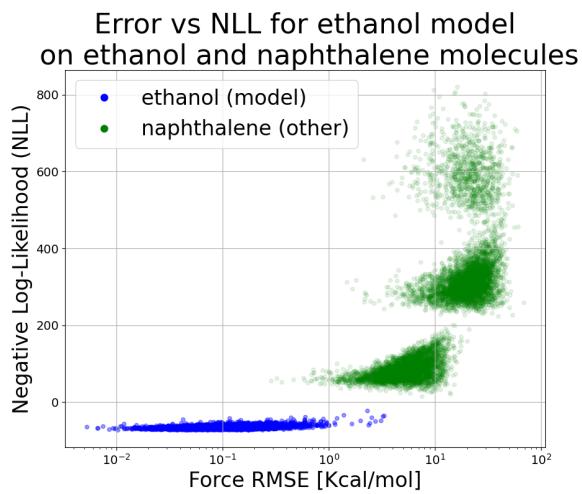
passera par le point ( $FPR = 1, TPR = 0$ ) et l'aire sera égale à 0. Dans la figure 3.4.6, on peut également observer comment les valeurs d'incertitude estimées ( $\sigma_{\text{pred}}$ ) se traduisent en points sur la courbe ROC. Finalement, on prendra l'aire sous la courbe comme métrique pour évaluer si le modèle est capable de détecter les molécules OOD.



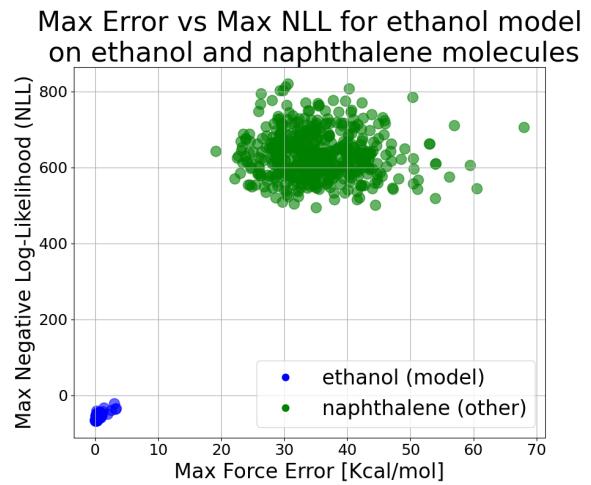
**Figure 3.4.6 – Courbe ROC pour la classification hors distribution**

Tout d'abord, il faut définir comment la métrique d'incertitude sera calculée pour chaque molécule. Sachant que nous disposons d'une estimation d'incertitude pour chaque atome, il suffit de choisir une fonction qui donne une estimation représentative de l'incertitude globale. Pour cela, nous prendrons le maximum parmi tous les atomes de la molécule, mais nous aurions aussi pu prendre la moyenne ou une autre métrique. Par exemple, pour la méthode basée sur les GMMs, les résultats montrent clairement que le maximum est une métrique efficace pour différencier les molécules différentes (voir figure 3.4.7). De plus, dans cette Figure, on peut détecter la présence de "clusters". Cela signifie que les données présentes pour chaque atome sont similaires entre elles, ayant donc des erreurs et des incertitudes similaires. En effet, les informations des caractéristiques cachées,  $h_{\text{rep}}$ , sont regroupées en clusters en fonction des environnements locaux des atomes.

Dans ce qui suit, nous présenterons les résultats de cette détection pour les 4 modèles d'incertitude utilisés, et nous essaierons de comprendre pourquoi chaque modèle a donné des prédictions plus ou moins bonnes. Comme on peut le voir dans la figure 3.4.8, pour chaque méthode l'évaluation est faite pour des modèles entraînés sur une unique molécule et testé sur une molécule différente. L'axe  $y$  se correspond alors à la molécule utilisée pour l'entraînement, alors que l'axe  $x$  à la molécule utilisée pour le test. Cette étude a donné les meilleurs résultats pour les Deep Ensembles et GMM, tandis



(a) Incertitude pour tous les atomes et molécules



(b) Maximum d'incertitude pour chaque molécule

**Figure 3.4.7 – Incertitude prédictive par un modèle d'éthane sur du naphtalène avec la méthode GMM**

que l'evidential deep learning et le deep kernel learning ont produit plus de cas où la classification n'a pas été correcte.

En guise d'observation importante, la diagonale de ces tableaux se correspond aux cas où chaque modèle est testé avec des datasets de test issus de la même molécule utilisée pour l'entraînement. Par exemple, si le modèle a été entraîné avec la molécule d'aspirine, on peut comparer les prédictions d'incertitude entre le dataset utilisé pour l'entraînement et un autre dataset de test de la même molécule, mais que le modèle ne connaît pas a priori. Si le modèle n'a pas fait d'overfitting, on s'attendra à ce que l'AUC-ROC soit égale à 0.5, correspondant à une classification aléatoire. Cela signifierait que le modèle considère que les deux datasets, d'entraînement et de test, appartiennent à la même molécule. C'est pourquoi il est correct que les diagonales soient approximativement égales à 0.5.

Pour l'interprétation des résultats, un aspect très intéressant à considérer est le comportement de la régression évidentielle. Si l'on observe la figure 3.4.8, le tableau LEDL montre que, dans certains cas, la détection des données hors distribution (OOD) est incorrecte. Cependant, en analysant plus en détail les molécules concernées, et en revenant à la figure 3.4.5, on observe que :

- La molécule d'aspirine est la plus complète, dans le sens où sa structure contient plusieurs atomes avec des environnements locaux (voisinages) très similaires à ceux présents dans d'autres molécules. Il ne serait donc pas surprenant qu'un modèle entraîné sur l'aspirine soit capable de faire des prédictions correctes pour le naphtalène ou le toluène. Cela pourrait expliquer pourquoi la régression évidentielle n'assigne pas une incertitude plus élevée à ces structures.
- Le même résultat se confirme pour les modèles entraînés sur le toluène et évalués sur le naphtalène, ainsi que pour les modèles entraînés sur l'acide salicylique et évalués sur le naphtalène.

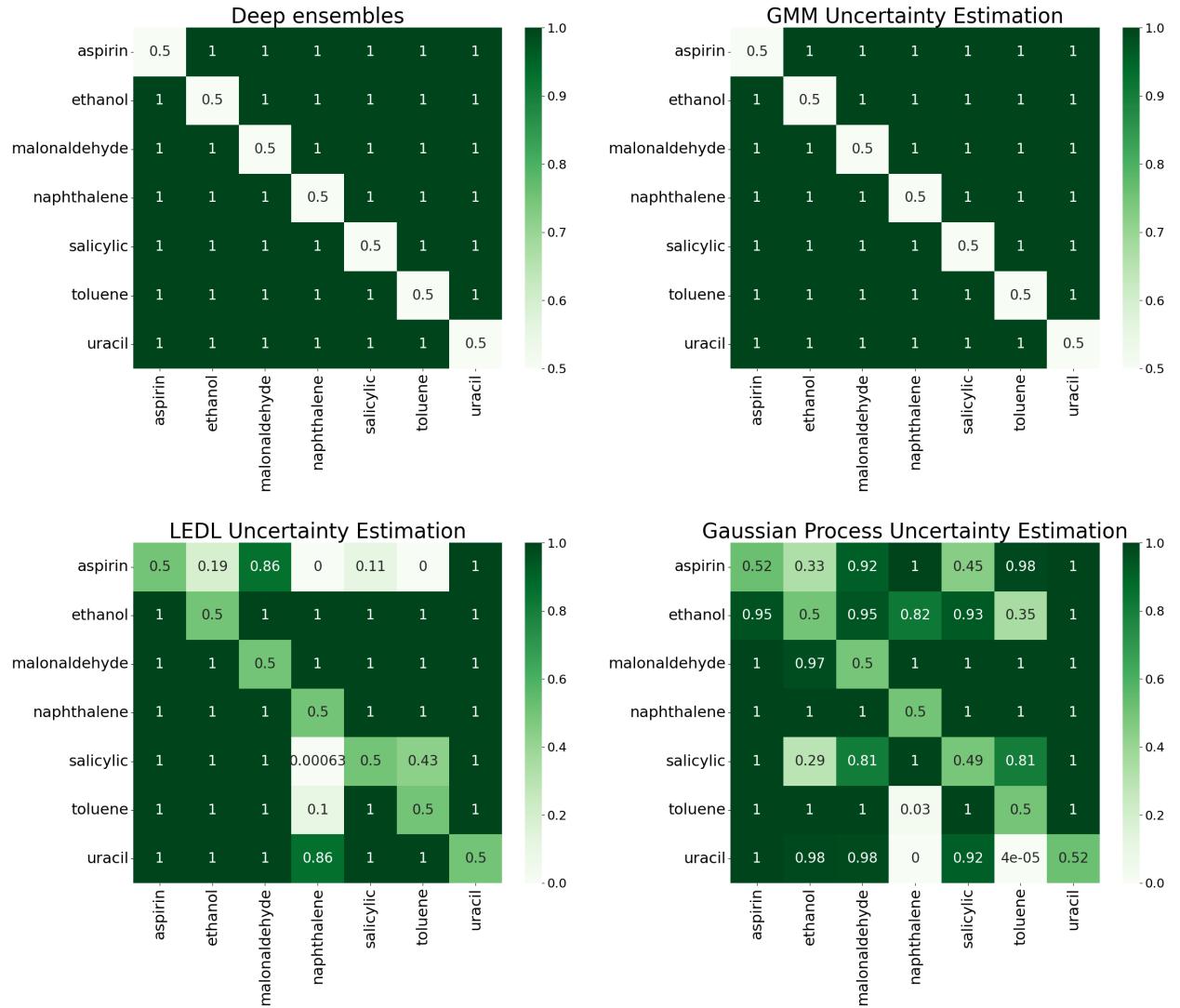
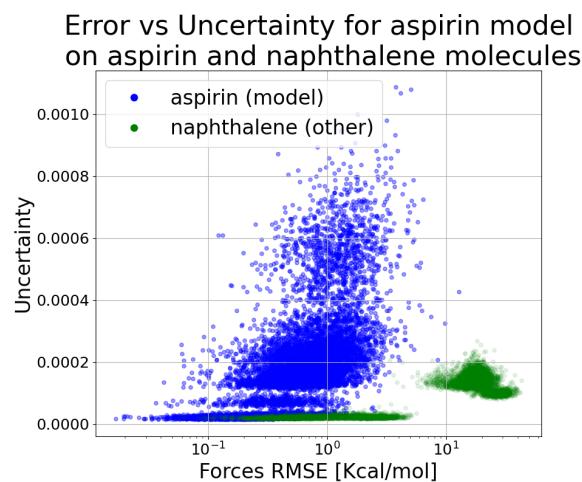


Figure 3.4.8 – Tableaux des métriques AUC-ROC pour la classification des points OOD

Bien que l'on puisse constater une certaine logique dans la régression évidentielle, malheureusement, l'incertitude estimée n'est pas toujours en accord avec l'erreur. Par exemple, si un modèle entraîné sur l'aspirine attribue une faible incertitude au naphtalène, cela devrait impliquer que les erreurs de prédiction sur le naphtalène sont faibles, ce qui n'est pas le cas, comme on peut le voir dans la figure 3.4.9.



**Figure 3.4.9** – Incertitude de la régression évidentielle avec un modèle entraîné sur l'aspirine, évaluée sur l'aspirine et le naphtalène

# 4

## Conclusion

La méthodologie de travail employée au cours de ce stage vise à comparer différentes techniques de quantification de l'incertitude pour des modèles de potentiels inter-atomiques basés sur des réseaux de neurones. Les différentes méthodes proposées seront sans doute d'une grande utilité pour le laboratoire, permettant d'explorer de manière plus efficace et avec plus de confiance l'utilisation de l'IA pour mieux comprendre la physique à l'échelle atomique.

### 4.1 DÉVELOPPEMENT ET DÉFIS DU STAGE

Pour mener à bien ce stage, il a été nécessaire de surmonter plusieurs défis complexes. En effet, les développements n'ont pas toujours fonctionné immédiatement, et les résultats n'ont pas toujours été à la hauteur des attentes. Cela a été particulièrement vrai pour les processus gaussiens, qui ont souvent présenté des inconvénients techniques, notamment en ce qui concerne le choix des hyperparamètres et l'architecture du modèle, y compris les questions liées à la normalisation.

Toutefois, il a été crucial de rester ouvert d'esprit et persévérant, ce qui a permis d'atteindre les résultats attendus. Une communication efficace avec les responsables du projet a également joué un rôle essentiel pour définir la direction à suivre à chaque étape décisive. Comme potentiel point d'amélioration, il apparaît qu'une documentation écrite plus systématique des progrès réalisés aurait pu faciliter les échanges et renforcer la motivation tout au long du projet.

## 4.2 FUTUR DU PROJET

En ce qui concerne les activités de ce laboratoire, j'espère que les outils mis en place permettront d'accélérer l'exploration des composants pour la microélectronique. Tel qu'expliqué précédemment, il sera par la suite possible de :

- Choisir la meilleure méthode d'incertitude pour une application donnée ;
- Réaliser une implémentation intégrée avec les modèles existants (partiellement effectuée durant le stage) ;
- Poursuivre les développements d'apprentissage actif ;
- Étudier la possibilité de construire des modèles de diffusion pour créer des structures à haute incertitude et, ainsi, améliorer l'apprentissage actif.
- Utiliser les méthodes développées pour des application en micro-électronique (nouveaux matériaux, dispositifs, etc.)

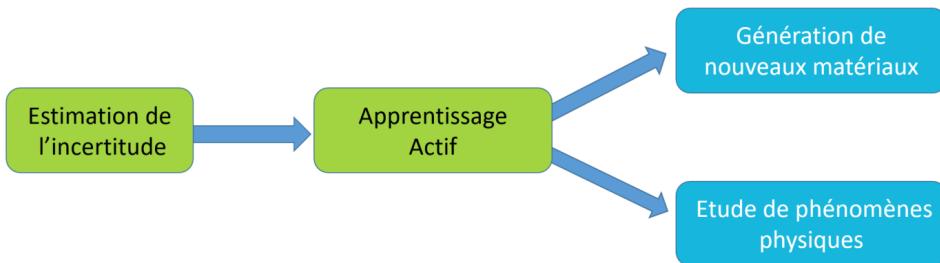


Figure 4.2.1 – Activités potentielles

Dans la figure 4.2.1, les différentes activités abordées au cours du stage ainsi que les chemins à suivre sont présentés de manière simplifiée. La complexité de cette chaîne complète, qui résulte du travail de nombreux experts à travers le monde, nécessite une spécialisation approfondie des chercheurs sur des aspects spécifiques. C'est pour cette raison que ce stage a principalement porté sur l'étude des incertitudes, y compris une initiation à l'apprentissage actif.

L'estimation de l'incertitude dans les réseaux de neurones est un domaine de recherche émergent, caractérisé par une forte demande pour des méthodes qui soient à la fois généralisables, précises, efficaces, rapides et robustes. Bien que des progrès significatifs aient été réalisés, ce domaine reste vaste et en pleine évolution, avec de nombreux défis à relever. Les développements à venir promettent d'ouvrir de nouvelles perspectives technologiques, marquant le début d'une ère où l'intelligence artificielle jouera un rôle encore plus central. Cependant, il est crucial que cette technologie soit utilisée de manière responsable, avec une conscience prononcée des implications sociétales et

environnementales. J'espère que ce travail contribuera à faire avancer cette vision, en guidant le développement de l'IA vers un avenir à la fois innovant et éthique.

#### 4.3 IMPACT SOCIÉTAL ET ENVIRONNEMENTAL

L'IA pose des défis considérables pour notre société. D'une part, sa présence imprègne désormais tous les niveaux de l'activité humaine, soulevant fréquemment des questions morales et environnementales.

Par rapport aux questions éthiques, l'utilisation de l'IA à l'échelle atomique présente des risques importants pour notre société, notamment en matière biologique (potentiel développement d'agents biologiques ou de substances dangereuses pour l'être humain). Concernant les développements en microélectronique, de nombreuses avancées sont à prévoir dans les années à venir, mais pour l'instant il n'existe pas, en revanche, de véritables risques associés à ces améliorations. Toutefois, une utilisation responsable de cette technologie devra jouer un rôle fondamental dans les développements futurs.

Concernant l'impact environnemental, il est important de reconnaître que l'IA est source d'une quantité croissante d'émissions de gaz à effet de serre, car l'entraînement de modèles d'IA requiert énormément de ressources de calcul. En créant des algorithmes conscients de l'incertitude des prédictions, l'apprentissage pourrait être accélérée en utilisant l'apprentissage actif. De plus, les modèles équivariants permettent de diminuer la quantité de données requises pour l'entraînement, ce qui réduit le nombre d'opérations de calcul nécessaires. En combinant ces approches, il est possible de réduire considérablement les émissions associées à l'entraînement de ces modèles d'IA.

Il est également crucial de considérer que les avancées en intelligence artificielle pourraient accélérer les recherches dans divers domaines, ce qui pourrait, en retour, contribuer à résoudre des problèmes environnementaux. Par exemple, il est possible d'envisager les bénéfices potentiels de ces techniques dans la recherche de nouveaux matériaux et/ou procédés de fabrication. À performances équivalentes, par exemple, un matériau pourrait être préféré à un autre parce qu'il est moins polluant ou qu'il émet moins des gaz à effet de serre. Comme pour toute technologie, il est essentiel de réaliser des analyses approfondies pour évaluer si les bénéfices l'emportent sur les impacts négatifs potentiels. À la lumière des progrès récents, il semble que ce sont les avantages qui l'emportent, et je pense qu'il est dans l'intérêt commun d'augmenter les investissements dans cette technologie.

#### 4.4 VALEUR AJOUTÉE À L'ENTREPRISE

Étant donné que l'intelligence artificielle est une technologie en constante évolution et que le CEA se spécialise dans la production technologique, il est évident que le développement d'améliorations

pour ces modèles d'IA permet au laboratoire de rester à la pointe de l'état de l'art, tout en favorisant des recherches plus innovantes et potentiellement plus performantes. Au sein du LSM, les outils développés seront particulièrement utiles pour les équipes de simulation atomistique et de simulation avancée du transport de charges, qui pourront désormais adapter ces méthodologies aux projets en cours. D'ailleurs, j'espère qu'éventuellement ces développements seront utiles non seulement pour le LSM, mais aussi pour d'autres laboratoires et pour la communauté scientifique en général.

#### 4.5 COMPLÉMENTARITÉ AVEC LA FORMATION ET LE PROJET PROFESSIONNEL

Ayant suivi une formation à l'IMT Atlantique en électronique en deuxième année (TAF OPE) et en mathématiques appliquées en troisième année (TAF MCE), cette dernière avec une attention particulière à l'IA, il est clair que ce stage m'a permis de mettre en pratique de nombreux outils acquis pendant ma formation. En effet, au cours de ce stage, j'ai dû à de nombreuses reprises faire preuve d'un esprit critique face aux défis rencontrés, une compétence qui est bien enseignée à l'IMT Atlantique. De plus, la capacité à bien communiquer mes avancées, doutes et propositions pour l'avenir du projet est une autre compétence acquise à l'école que j'ai trouvée extrêmement utile lors de mon stage.

Ma formation à l'IMT Atlantique, en plus de ce stage, sera un atout clé pour mon futur professionnel. En effet, j'ai l'intention de continuer à travailler sur l'utilisation de l'IA pour des applications principalement en simulation et en électronique, mais pas uniquement. Tous les outils dont je dispose aujourd'hui grâce à ma formation et à ce stage seront sans doute fondamentaux dans mon avenir proche.

# 5

## Annexes

### 5.1 RÉSEAUX DE NEURONES POUR LA PRÉDICTION MOLÉCULAIRE

#### 5.1.1 RÉSEAUX DE NEURONES DE PASSAGE DE MESSAGES

Mathématiquement, ces réseaux peuvent être modélisés de la manière suivante :

$$m_v^{t+1} = \sum_{w \in N(v)} M_t(h_v^t, h_w^t, e_{vw})$$

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1})$$

où :

- $m_v^{t+1}$  est le message construit à l'étape  $t$  pour le nœud  $v$ .
- $h_v^t$  représente les caractéristiques internes du nœud  $v$  à l'étape  $t$ .
- $e_{vw}$  désigne les caractéristiques internes de la liaison entre les nœuds  $v$  et  $w$ .
- $M_t$  est la fonction de construction des messages, généralement définie comme un réseau de neurones appris par le modèle.

- $U_t$  est la fonction de mise à jour du nœud, également apprise par le modèle.

Enfin, le modèle se termine par une fonction de sortie donnée par :

$$\hat{y} = R(\{h_v^T \mid v \in G\})$$

où  $R$  est une fonction de lecture (readout) qui agrège les caractéristiques finales  $h_v^T$  de chaque nœud  $v$  appartenant au graphe  $G$  pour produire la prédiction  $\hat{y}$ .

### 5.1.2 RÉSEAUX DE NEURONES ÉQUIVARIANTS SUR DES GRAPHES

Comme expliqué dans le Chapitre 2, certains réseaux de neurones, tels que NequIP, possèdent la propriété d'être équivariants. Dans ce qui suit, nous présentons les fondements mathématiques [23] de ces modèles ainsi que leur équivariance.

En revenant à l'architecture illustrée à la Figure 2.4.1, dans le bloc d'**interaction**, les filtres sont appris en utilisant un réseau auxiliaire  $R_c^{(l_f, l_i)}(r)$  de type Multi-Layer Perceptron, qui attribue des poids aux harmoniques sphériques à partir de la relation suivante :

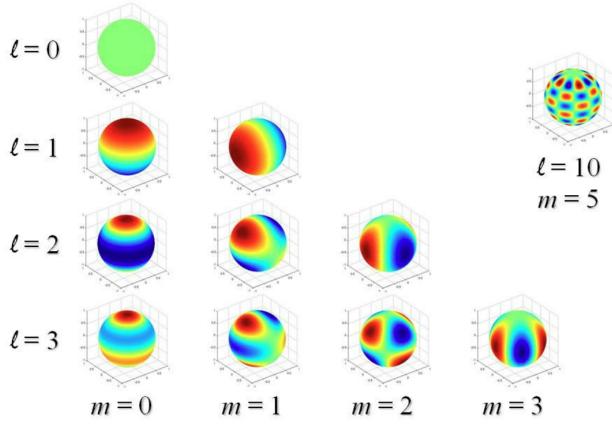
$$F_{cm}^{(l_f, l_i)}(\vec{r}) = R_c^{(l_f, l_i)}(r) Y_m^{(l_f)}(\hat{r})$$

Pour calculer le produit tensoriel dans l'étape de **convolution**, nous utilisons une représentation basée sur les coefficients de Clebsch-Gordan, étroitement liés aux harmoniques sphériques :

$$\mathcal{K}_{acm_o}^{(l_o)}(\vec{r}_a, V_{acm_i}^{(l_i)}) := \sum_{m_f, m_i} C_{(l_o, m_o)}^{(l_f, m_f)(l_i, m_i)} \sum_{b \in S} F_{cm_f}^{(l_f, l_i)}(\vec{r}_{ab}) V_{bcm_i}^{(l_i)}$$

Ici,

- $l_i$  et  $m_i$  sont des constantes associées au moment angulaire et définissent les harmoniques sphériques (voir Figure 5.1.1).
- $\mathcal{K}$  représente la couche convolutionnelle.
- $F_{cm_f}^{(l_f, l_i)}(\vec{r}_{ab})$  est le filtre construit précédemment.
- $V_{bcm_i}^{(l_i)}(\vec{r}_{ab})$  est l'entrée à ce bloc d'interaction.
- $C_{(l_o, m_o)}^{(l_f, m_f)(l_i, m_i)}$  sont les coefficients de Clebsch-Gordan.
- Les paires  $(l_o, m_o)$  sont sélectionnées parmi toutes les combinaisons possibles de  $(l_i, m_i)$  et  $(l_f, m_f)$ . Pour des raisons de complexité, la valeur maximale du moment angulaire est limitée par  $l_o \leq l_{max}$ .



**Figure 5.1.1** – Harmoniques Sphériques

Les couches d'**auto-interaction** sont définies par :

$$\sum_{c'} W_{cc'}^{(l)} V_{ac'm}^{(l)}$$

où  $W_{cc'}^{(l)}$  représente les poids à apprendre par le modèle.

Pour démontrer l'équivariance du modèle, considérons une rotation ou une réflexion. Nous pouvons exprimer la sortie en utilisant une somme de matrices de Wigner et d'harmoniques sphériques :

$$Y_m^{(l)}(R(g)\hat{r}) = \sum_{m'} D_{mm'}^{(l)}(g) Y_{m'}^{(l)}(\hat{r})$$

Grâce aux propriétés des matrices de Wigner et des coefficients de Clebsch-Gordan, toutes les opérations impliquées dans le modèle préservent l'équivariance. La preuve consiste à montrer que :

$$\mathcal{K}_{acm_O}^{(l_o)} \left( \mathcal{R}(g)\vec{r}_a, \sum_{m'_I} D_{m_I m'_I}^{(l_I)}(g) V_{acm'_I}^{(l_I)} \right) = \sum_{m'_O} D_{m_O m'_O}^{(l_O)}(g) \mathcal{K}_{acm'_O}^{(l_o)} \left( \vec{r}_a, V_{acm_I}^{(l_I)} \right).$$

Ici,

- $\mathcal{K}_{acm_O}^{(l_o)}$  représente la sortie associée à l'ordre de rotation  $l_o$  d'une couche d'interaction.
- $\mathcal{R}(g)\vec{r}_a$  correspond à une rotation de l'atome  $a$ .
- $V_{acm_I}^{(l_I)}$  est l'entrée de l'étape concernée, représentant les caractéristiques cachées de l'atome dans la couche précédente.

- $D_{m_O m'_O}^{(l_O)}(g)$  est la matrice de Wigner associée à la rotation  $g$ .

Cela implique qu'une rotation appliquée à l'entrée entraîne une modification des caractéristiques internes via une matrice de Wigner, ce qui est équivalent à une rotation. L'équivariance est alors démontrée.

## 5.2 ESTIMATION DE L'INCERTITUDE

### 5.2.1 ENSEMBLES PROFONDS (DEEP ENSEMBLES)

Les ensembles profonds sont une méthode pour estimer l'incertitude dans les modèles d'apprentissage automatique en entraînant plusieurs modèles indépendamment sur les mêmes données. Chaque modèle dans l'ensemble capture différents aspects des données et apprend différentes représentations en raison des variations dans l'initialisation et les processus d'entraînement. Lors de l'inférence, les prédictions de tous les modèles sont agrégées, généralement par une moyenne. La variance entre ces prédictions sert de mesure d'incertitude : une forte variance indique une grande incertitude, tandis qu'une faible variance suggère une confiance dans la prédiction. Cette approche exploite la diversité des modèles pour fournir une estimation de l'incertitude plus robuste et fiable qu'un seul modèle.

Les ensembles profonds ont déjà été utilisés avec succès dans les dynamiques moléculaires pilotées par l'IA. Les ensembles profonds sont très robustes, étant généralement les méthodes de quantification d'incertitude les plus performantes. Cependant, leur coût computationnel excessif motive la recherche de méthodes alternatives.

### 5.2.2 RÉGRESSION ÉVIDENTIELLE PROFONDE

L'apprentissage profond évidentiel vise à différencier l'incertitude épistémique et aléatoire en supposant une loi a priori sur les paramètres de la distribution de la variable de sortie. Cet algorithme a déjà été utilisé de manière globale pour prédire l'incertitude totale d'une molécule [16]. Cependant, cela ne reconnaît pas que la localité (la capacité d'ajuster l'estimation de l'incertitude en fonction du nombre d'atomes) est une propriété fondamentale pour l'estimation de l'incertitude. Bien que [22] affirme que la régression évidente viole cette propriété, ce n'est plus le cas, car au moins une version localisée de l'apprentissage profond évidentiel a déjà été mise en œuvre pour les dynamiques moléculaires [21]. Dans la présente implémentation, une légère modification est proposée dans les sorties de la régression évidente, de sorte que, suivant la proposition bayésienne générique de [17], la formulation suivante est obtenue :

$$\begin{aligned}\overline{F_i} &\sim N(\bar{\mu}_i, \sigma_i^2) \\ \overline{\mu_i} &\sim N(\nabla_{r_i} E, \sigma^2 v^{-1}) \\ \sigma^2 &\sim \Gamma^{-1}(\alpha, \beta)\end{aligned}\tag{5.1}$$

Maintenant, soit l'erreur de force décrite par

$$\Delta F = \sum_{j=1}^3 \left( (\nabla_{r_i} E)_j - F_{ij} \right)^2\tag{5.2}$$

Ensuite, de manière similaire à [17], la perte peut être calculée comme suit, la principale différence étant que maintenant les forces sont un vecteur et non un scalaire.

$$\begin{aligned}\mathcal{L}_i^{\text{NLL}} &= \frac{3}{2} \log \left( \frac{1}{v} \right) - \alpha \log(\omega) + \\ &\left( \alpha + \frac{3}{2} \right) \log(v \Delta F_i + \omega) + \log \Gamma(\alpha) - \log \Gamma \left( \alpha + \frac{3}{2} \right)\end{aligned}$$

Un régularisateur  $\mathcal{L}_i^R$  est généralement ajouté à la perte à ce stade pour promouvoir une minimisation de l'évidence pour les prédictions incorrectes [17].

$$\mathcal{L}_i^R = \lambda |\Delta F| (2v + \alpha)\tag{5.3}$$

Cependant, la raison mathématique pour laquelle la régression évidente est capable de prédire les points de données hors distribution (OOD) n'est pas encore complètement claire, et une corrélation significative a été trouvée entre ses performances et les schémas de convergence pendant l'entraînement [18]. Récemment, d'autres articles ont aussi fait des analyses plus détaillées pour expliquer les inconvénients de cette méthode [24].

Un régularisateur supplémentaire  $\mathcal{L}^U$  tel que proposé dans [25] est utilisé pour la détection hors distribution, bien qu'aucune amélioration significative n'ait été observée.

$$\mathcal{L}^U = -|y - \gamma| \cdot \log(\exp(\alpha - 1) - 1)$$

### 5.2.3 GMM SUR LES CARACTÉRISTIQUES

La méthode de [8] sera utilisée dans cette étude. Elle consiste à estimer l'incertitude sur une approche distributionnelle sur l'espace latent du modèle. De cette manière, les caractéristiques internes pour chaque atome sont utilisées comme points de données d'entrée d'un modèle de mélange gaussien (GMM), qui définit une fonction de densité de probabilité sur les vecteurs de la représentation cachée des atomes individuels.

### 5.2.4 PROCESSUS GAUSSIENS

Les processus gaussiens (GPs) sont des modèles probabilistes qui permettent de modéliser des distributions sur un jeu de données. Les sorties sont alors modélisées comme des distributions gaussiennes, qui peuvent être utilisées pour faire des prédictions sur des points d'entrée jamais vu pour le modèle. L'une des motivations pour les utiliser est que les GPs ont des propriétés spéciales qui permettent d'obtenir des solutions fermées qui peuvent être proposées dans le contexte de l'estimation de l'incertitude, et qui se sont avérées simples mais puissantes.

Formellement, étant donnée une fonction  $f()$ , des entrées  $\mathbf{x}$  et des sorties  $\mathbf{y} = f(\mathbf{x}) + \epsilon$ , avec  $\epsilon \sim \mathcal{N}(0, \sigma^2)$ , un GP vise à estimer la moyenne et la matrice de covariance de la fonction à une entrée  $\mathbf{x}_*$ , qui ne devrait évidemment pas appartenir à l'ensemble des observations  $\mathbf{x}$ . Ce qui suit présente la formulation exacte du GP, à partir de laquelle la solution épars est dérivée. Enfin, les motivations et la faisabilité de combiner les GPs avec les réseaux de neurones sont explorées.

#### PROCESSUS GAUSSIENS EXACT

L'objectif est d'obtenir un GP tel que, pour toute entrée  $\mathbf{z}$ ,

$$\mathbf{f}(\mathbf{z}) \sim \mathcal{GP}(m(\mathbf{z}), K(\mathbf{z}))$$

où  $K(\mathbf{z})$  est une matrice de covariance définie par un kernel prédéfini tel que

$$K_{\mathbf{a}, \mathbf{b}}(i, j) = k(a_i, b_j)$$

Ce kernel permet de mesurer à quel point 2 points d'entrée sont correlés. Dans ce code, le kernel RBF (*Radial basis function*) est utilisé, qui est défini par

$$k_{\text{RBF}}(x_1, x_2) = \exp\left(-\frac{1}{2}(x_1 - x_2)^T \Theta^{-2} (x_1 - x_2)\right)$$

Soient maintenant  $\mathbf{f} = f(\mathbf{x})$  et  $\mathbf{f}_* = f(\mathbf{x}_*)$ . Une propriété clé des GPs est que la distribution conjointe de deux variables peut être calculée de manière directe :

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N} \left( 0, \begin{bmatrix} K_{\mathbf{x}, \mathbf{x}} + \sigma^2 I & K_{\mathbf{x}, \mathbf{x}_*} \\ K_{\mathbf{x}_*, \mathbf{x}} & K_{\mathbf{x}_*, \mathbf{x}_*} \end{bmatrix} \right)$$

En suivant le raisonnement de [19], une formule fermée est obtenue pour calculer les paramètres de la distribution de  $f_*$ , lorsque conditionnée sur les observations connues.

$$\begin{aligned} \mathbf{f}_* | \mathbf{x}, \mathbf{y}, \mathbf{x}_* &\sim \mathcal{N} (\bar{\mathbf{f}}_*, \text{cov}(\mathbf{f}_*)) , \\ \bar{\mathbf{f}}_* &\triangleq \mathbb{E} [\mathbf{f}_* | \mathbf{x}, \mathbf{y}, \mathbf{x}_*] = K_{\mathbf{x}_*, \mathbf{x}} [K_{\mathbf{x}, \mathbf{x}} + \sigma_n^2 I]^{-1} \mathbf{y}, \\ \text{cov}(\mathbf{f}_*) &= K_{\mathbf{x}_*, \mathbf{x}_*} - K_{\mathbf{x}_*, \mathbf{x}} [K_{\mathbf{x}, \mathbf{x}} + \sigma_n^2 I]^{-1} K_{\mathbf{x}, \mathbf{x}_*}. \end{aligned}$$

Bien que ce développement soit élégant, il pose un coût computationnel élevé. En effet, le calcul de l'inverse d'une matrice de rang  $n$  a une complexité  $\mathcal{O}(n^3)$  et des exigences de stockage  $\mathcal{O}(n^2)$ . Lorsque  $n$  est de l'ordre de milliers, millions ou même plus dans les applications modernes de big data (y compris les dynamiques moléculaires !), cette méthode devient ingérable.

## PROCESSUS GAUSSIENS ÉPARS

Afin de réduire la complexité et les besoins de stockage de l'algorithme, des approximations peuvent être proposées en se basant sur  $m \ll n$  pseudo-entrées  $f_m$ , qui seront appelés à partir de maintenant points d'induction. Ces points n'ont pas besoin de faire partie des données d'entraînement  $(\mathbf{x}, \mathbf{y})$ . Comme démontré dans [26], l'égalité suivante est valable :

$$p(\mathbf{f}_* | \mathbf{y}) = \int p(\mathbf{f}_* | \mathbf{f}_m, \mathbf{f}) p(\mathbf{f} | \mathbf{f}_m, \mathbf{y}) p(\mathbf{f}_m | \mathbf{y}) d\mathbf{f} d\mathbf{f}_m$$

À ce stade, l'idée clé de l'approximation prend place. En effet, une hypothèse est faite que les points inducteurs  $f_m$  sont une statistique suffisante, ce qui implique que  $p(\mathbf{f}_* | \mathbf{f}_m, \mathbf{f}) = p(\mathbf{f}_* | \mathbf{f}_m)$ . En même temps, une deuxième approximation est faite sur  $\phi(\mathbf{f}_m) = p(\mathbf{f}_m | \mathbf{y})$ , qui est supposée suivre une distribution gaussienne de moyenne  $\mu$  et de matrice de covariance  $A$ . En suivant le raisonnement de [26], une solution fermée dépendant de ces paramètres est obtenue :

$$\begin{aligned} \bar{\mathbf{f}}_* &= K_{\mathbf{f}_*, \mathbf{f}_m} K_{\mathbf{f}_m, \mathbf{f}_m}^{-1} \mu, \\ \text{cov}(\mathbf{f}_*) &= K_{\mathbf{f}_*, \mathbf{f}_*} - K_{\mathbf{f}_*, \mathbf{f}_m} K_{\mathbf{f}_m, \mathbf{f}_m}^{-1} K_{\mathbf{f}_m, \mathbf{f}_*} + K_{\mathbf{f}_*, \mathbf{f}_m} B K_{\mathbf{f}_m, \mathbf{f}_*} \end{aligned}$$

avec  $B = K_{\mathbf{f}_m, \mathbf{f}_m}^{-1} A K_{\mathbf{f}_m, \mathbf{f}_m}^{-1}$ .

La question se pose alors, comment choisir  $\mu$  et  $A$ ? L'idée clé est qu'une forte approximation ( $p(\mathbf{f}_* | \mathbf{f}_m, \mathbf{f}) = p(\mathbf{f}_* | \mathbf{f}_m)$ ) a été faite sur  $p(\mathbf{f}_* | \mathbf{y})$ . Renommerons l'approximation en tant que  $q(\mathbf{f}_*) \approx p(\mathbf{f}_* | \mathbf{y})$ . Il devrait maintenant être clair que l'objectif principal est d'obtenir  $q(\mathbf{f}_*)$ , qui dépend des paramètres variationnels ( $\mu, A$ ), aussi proche que possible de  $p(\mathbf{f}_* | \mathbf{y})$ . De manière équivalente, nous pouvons essayer de rendre  $q(\mathbf{f}, \mathbf{f}_m)$  aussi similaire que possible à  $p(\mathbf{f}, \mathbf{f}_m | \mathbf{y})$ .

Il est alors possible d'utiliser la divergence de Kullback-Leibler entre ces deux distributions comme fonction objectif à minimiser. Cela devrait alors permettre d'optimiser les paramètres variationnels en utilisant un algorithme d'optimisation, tel que la descente de gradient, par rapport à la fonction de perte définie. Cependant, cela peut être un défi de calcul, donc une borne inférieure de la divergence KL est utilisée [26]. Bien que cette méthodologie conduise à une fonction de perte efficace, ce n'est pas l'algorithme utilisé dans la plupart des applications modernes. En réalité, des paramètres variationnels supplémentaires et des approximations sont généralement ajoutés dans les développements les plus récents, comme défini dans [27].

Dans la présente implémentation, la borne inférieure de l'évidence variationnelle de la Predictive-LogLikelihood (ELBO) est utilisée ([Log-Vraisemblance Prédictive](#)), qui est définie par

$$\mathcal{L} = \log \mathbb{E}_{q(\mathbf{f}_m)p(\mathbf{f}|\mathbf{f}_m)} [p(\mathbf{f}_* | \mathbf{f})] - D_{\text{KL}}(q(\mathbf{f}_m) \| p(\mathbf{f}_m))$$

Avec quelques manipulations algébriques intelligentes et des approximations, cette fonction de perte peut être minimisée efficacement et les paramètres variationnels peuvent être trouvés, à l'aide d'un algorithme d'optimisation. La complexité de calcul est  $\mathcal{O}(m^3)$ , ce qui est efficace puisque  $m \ll n$ .

## PROCESSUS GAUSSIENS

Nous avons déjà vu que les processus gaussiens peuvent être optimisés via l'inférence variationnelle (VI) pour minimiser une fonction de perte via un algorithme basé sur la descente de gradient. En même temps, les réseaux de neurones sont entraînés en suivant une méthodologie similaire. Cela implique que les réseaux de neurones et les processus gaussiens pourraient être combinés, obtenant les représentations internes puissantes des réseaux de neurones tout en étant capable de calculer des distributions de probabilité représentatives sur la prédiction du modèle dans son ensemble. Cette stratégie s'appelle Deep Kernel Learning (DKL).

Dans ce cas, les processus gaussiens seront combinés avec l'architecture neuronale NequIP afin d'obtenir des estimations d'incertitude. À cette fin, chaque représentation de caractéristique interne  $\mathbf{h}_{\text{rep}}(i)$  correspondant à l'atome  $i$  est passée à travers un processus gaussien. Comme indiqué dans [[22]], la sortie du processus gaussien peut ensuite être calculée en agrégant toutes les distri-

butions atomiques individuelles comme suit.

$$p(E_\star | \mathbf{X}, \mathbf{H}) = \sum_i^n \mathcal{GP}_\phi \circ h_{\text{rep}}(\mathbf{X}, \mathbf{H})_i \\ \sim \mathcal{N} \left( E_\star | \sum_i^n \mathbb{E}[E_\star]_i, \sum_{ij}^n \text{Cov}(E_\star)_{ij} \right)$$

où  $\mathbf{X}$  sont les données d'entraînement,  $E$  l'énergie et  $\mathbf{H}$  les représentations cachées.

### 5.3 DÉTAILS D'ENTRAÎNEMENT

Dans ce qui suit, les hyperparamètres utilisés pour les différents modèles pour la classification OOD (dataset MD17) sont donnés. Pour le dataset 3BPA, les hyperparamètres sont les mêmes, sauf  $n_{\text{train}}$  et  $n_{\text{val}}$ .

Paramètre	DE & GMM	LEDL	GP
$n_{\text{train}}$	1000	1000	1000
$n_{\text{val}}$	500	500	500
batch size	5	5	32
Forces Loss	Per atom RMSE	Modified evidential loss	-
Total Energy Loss	Per atom MSE	Per atom MSE	-
$\rho_{\text{force}}$	0.99	0.9	0.5
Learning Rate (lr)	0.005	0.005	0.01
lr Scheduler	ReduceOnPlateau	ReduceOnPlateau	ReduceOnPlateau
lr patience	100	100	10
lr factor	0.5	0.5	0.1
Optimisateur	Adam	Adam	Adam
Per Species Rescale Shift	Moyenne de $E_{\text{totale}}$	Moyenne de $E_{\text{totale}}$	Moyenne de $E_{\text{totale}}$
Per Species Rescale Scale	No	No	No
Epochs	100	100	50
$\lambda$	-	0.01	-
Likelihood	-	-	Gaussian
Fonction d'optimisation	-	-	Predictive log likelihood
Kernel	-	-	RBF
$n_{\text{inducing points}}$	-	-	1000
Points d'induction	-	-	Aléatoire sur les données d'entraînement

**Table 5.3.1** – hyperparamètres pour les méthodes DE, GMM, LEDL, et GP pour le dataset MD17

# Bibliographie

- [1] Musaelian, a., batzner, s., johansson, a. et al. learning local equivariant representations for large-scale atomistic dynamics. *nat commun* 14, 579 (2023).
- [2] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry, 2017. URL <https://arxiv.org/abs/1704.01212>.
- [3] Han Yang, Chenxi Hu, Yichi Zhou, Xixian Liu, Yu Shi, Jielan Li, Guanzhi Li, Zekun Chen, Shui-zhou Chen, Claudio Zeni, Matthew Horton, Robert Pinsler, Andrew Fowler, Daniel Zügner, Tian Xie, Jake Smith, Lixin Sun, Qian Wang, Lingyu Kong, Chang Liu, Hongxia Hao, and Ziheng Lu. Mattersim : A deep learning atomistic model across elements, temperatures and pressures, 2024. URL <https://arxiv.org/abs/2405.04967>.
- [4] Amil Merchant, Simon Batzner, Samuel Schoenholz, Muratahan Aykol, Gowoon Cheon, and Ekin Cubuk. Scaling deep learning for materials discovery. *Nature*, 624 :1–6, 11 2023. doi : 10.1038/s41586-023-06735-9.
- [5] Fan Liang, Cheng Qian, Wei Yu, David Griffith, and Nada Golmie. Survey of graph neural networks and applications. *Wireless Communications and Mobile Computing*, 2022, 07 2022. doi : 10.1155/2022/9261537.
- [6] Lingfei Wu, Yu Chen, Kai Shen, Xiaojie Guo, Hanning Gao, Shucheng Li, Jian Pei, and Bo Long. Graph neural networks for natural language processing : A survey. *CoRR*, abs/2106.06090, 2021. URL <https://arxiv.org/abs/2106.06090>.
- [7] Maciej Krzywda, Szymon Lukasik, and Amir H. Gandomi. Graph neural networks in computer vision - architectures, datasets and common approaches. In *2022 International Joint Conference on Neural Networks (IJCNN)*. IEEE, July 2022. doi : 10.1109/ijcnn55064.2022.9892658. URL <http://dx.doi.org/10.1109/IJCNN55064.2022.9892658>.
- [8] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E. Smidt, and Boris Kozinsky. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*, 13(1), May 2022. ISSN 2041-1723. doi : 10.1038/s41467-022-29939-5. URL <http://dx.doi.org/10.1038/s41467-022-29939-5>.
- [9] Stefan Bloemheuvel, Jurgen van den Hoogen, and Martin Atzmueller. A computational framework for modeling complex sensor network data using graph signal processing and graph neural networks in structural health monitoring. *Applied Network Science*, 6, 12 2021. doi : 10.1007/s41109-021-00438-8.

- [10] Maurice Weiler. URL [https://maurice-weiler.gitlab.io/blog\\_post/cnn-book\\_1\\_equivariant\\_networks/](https://maurice-weiler.gitlab.io/blog_post/cnn-book_1_equivariant_networks/).
- [11] Chu-I Yang and Yi-Pei Li. Explainable uncertainty quantifications for deep learning-based molecular property prediction. *Journal of Cheminformatics*, 15, 02 2023. doi : 10.1186/s13321-023-00682-3.
- [12] Lior Hirschfeld, Kyle Swanson, Kevin Yang, Regina Barzilay, and Connor W. Coley. Uncertainty quantification using neural networks for molecular property prediction. *Journal of Chemical Information and Modeling*, 60(8) :3770–3780, 2020. doi : 10.1021/acs.jcim.0c00502. URL <https://doi.org/10.1021/acs.jcim.0c00502>. PMID : 32702986.
- [13] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles, 2017. URL <https://arxiv.org/abs/1612.01474>.
- [14] Albert Zhu, Simon Batzner, Albert Musaelian, and Boris Kozinsky. Fast uncertainty estimates in deep learning interatomic potentials. *The Journal of Chemical Physics*, 158(16), April 2023. ISSN 1089-7690. doi : 10.1063/5.0136574. URL <http://dx.doi.org/10.1063/5.0136574>.
- [15] Douglas Reynolds. *Gaussian Mixture Models*, pages 659–663. Springer US, Boston, MA, 2009. ISBN 978-0-387-73003-5. doi : 10.1007/978-0-387-73003-5\_196. URL [https://doi.org/10.1007/978-0-387-73003-5\\_196](https://doi.org/10.1007/978-0-387-73003-5_196).
- [16] Ava P. Soleimany, Alexander Amini, Samuel Goldman, Daniela Rus, Sangeeta N. Bhatia, and Connor W. Coley. Evidential deep learning for guided molecular property prediction and discovery. *ACS Central Science*, 7(8) :1356–1367, 2021. doi : 10.1021/acscentsci.1c00546. URL <https://doi.org/10.1021/acscentsci.1c00546>.
- [17] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. Deep evidential regression, 2020. URL <https://arxiv.org/abs/1910.02600>.
- [18] Nis Meinert, Jakob Gawlikowski, and Alexander Lavin. The unreasonable effectiveness of deep evidential regression. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(8) :9134–9142, June 2023. ISSN 2159-5399. doi : 10.1609/aaai.v37i8.26096. URL <http://dx.doi.org/10.1609/aaai.v37i8.26096>.
- [19] C. E. Rasmussen and C. K. I. Williams. Gaussian processes for machine learning, 2006. URL [www.GaussianProcess.org/gpml](http://www.GaussianProcess.org/gpml).
- [20] Cláudia Neves. *Structural Health Monitoring of Bridges : Model-free damage detection method using Machine Learning*. PhD thesis, 04 2017.
- [21] Aik Rui Tan, Shingo Urata, Samuel Goldman, Johannes C. B. Dietschreit, and Rafael Gómez-Bombarelli. Single-model uncertainty quantification in neural network potentials does not consistently outperform model ensembles. *npj Computational Materials*, 9(1), December 2023. ISSN 2057-3960. doi : 10.1038/s41524-023-01180-8. URL <http://dx.doi.org/10.1038/s41524-023-01180-8>.
- [22] Tom Wollschläger, Nicholas Gao, Bertrand Charpentier, Mohamed Amine Ketata, and Stephan Günemann. Uncertainty estimation for molecules : Desiderata and methods, 2023. URL <https://arxiv.org/abs/2306.14916>.

- [23] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks : Rotation- and translation-equivariant neural networks for 3d point clouds, 2018. URL <https://arxiv.org/abs/1802.08219>.
- [24] Mira Jürgens, Nis Meinert, Viktor Bengs, Eyke Hüllermeier, and Willem Waegeman. Is epistemic uncertainty faithfully represented by evidential deep learning methods ?, 2024. URL <https://arxiv.org/abs/2402.09056>.
- [25] Kai Ye, Tiejin Chen, Hua Wei, and Liang Zhan. Uncertainty regularized evidential regression, 2024. URL <https://arxiv.org/abs/2401.01484>.
- [26] Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In David van Dyk and Max Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL <https://proceedings.mlr.press/v5/titsias09a.html>.
- [27] James Hensman, Alex Matthews, and Zoubin Ghahramani. Scalable variational gaussian process classification, 2014. URL <https://arxiv.org/abs/1411.2005>.