

# Projet IA : Le Modèle d'Ising

Intelligence artificielle pour la physique

---

A. Cremel-Schlemer (3800159)

G. Carvalho (xxxxxxxxx)

M. Panet (28705836)

7 décembre 2023

## TABLE DES MATIÈRES

Introduction	2
1 Génération de données	2
2 Pré-traitement des données	2
3 Modèles classiques	3
3.1 Analyse par la magnétisation . . . . .	3
3.2 Analyse par l'état individuel des spins . . . . .	4
4 Réseaux de neurones	4
Conclusion	4

## INTRODUCTION

Le modèle d'Ising est un modèle de physique statistique introduit par Rudolph Peierls, Wilhelm Lenz et Ernst Ising dans les années 1920. Ce modèle d'une remarquable simplicité lui permet pourtant de mettre en exergue un comportement de transition de phase.

Sous sa formulation historique, le modèle d'Ising permet d'étudier la transition de phase paramagnétique/-ferromagnétique d'un matériau. Certains matériaux possèdent une aimantation intrinsèque; ils sont dits ferromagnétiques. C'est le cas des aimants. Cette propriété provient d'interactions à courte portée entre les spins d'un cristal. Mais, comme l'a découvert en 1895 le physicien Pierre Curie, ces matériaux perdent leur propriété magnétique au-dessus d'une température caractéristique appelée température de Curie  $T_C$ . On trouve aussi des matériaux dits paramagnétiques. Ces matériaux ne sont pas magnétiques, mais le deviennent au contact d'un champ magnétique extérieur puissant. En l'absence de champ magnétique extérieur, les spins de ces matériaux sont désordonnés, de sorte qu'en moyenne, le moment magnétique du matériau est nul : le matériau n'est pas magnétique. Sous l'action d'un champ magnétique extérieur, ces moments se polarisent, ce qui rend le matériau magnétique.

L'hamiltonien du modèle d'Ising prend donc en compte ces deux propriétés, puisqu'une partie de cet est relative à l'interaction ferromagnétique à courte portée et l'autre partie est relative à l'interaction de chaque spin du cristal avec un champ magnétique extérieur. Dans la plupart des modèles numériques, on travaille sans champ extérieur.

La relative généralité de ce modèle (interactions locales et interaction individuelle avec une force extérieure) transcende la physique et permet de décrire qualitativement et parfois quantitativement une très grande variété de situations (paramagnétisme, gaz réticulaire, agents économiques, modèles écologiques, utilisation en analyse et traitement de l'image). Il donnera même lieu à un modèle plus général : le modèle de Potts (qui généralise l'aspect binaire du modèle d'Ising lié à la binarité des spins).

Le modèle d'Ising 2D est une approche numérique de la résolution de ce problème, où chaque atome d'un cristal idéal plat et de maille carrée est associé à un pixel d'une image, et chaque spin est représenté soit par un 0, soit par un 1. Pour améliorer la validité du modèle, des conditions aux bords périodiques sont utilisées.

À travers notre exposé sur l'intelligence artificielle, nous allons nous intéresser à un jeu de données constitué de 16000 images de  $40 \times 40$  pixels. Chaque image est associée à une température caractéristique allant de 0.25 à 4 (sans unité) et à un label *phase ferromagnétique*, où les spins sont globalement tous alignés ou forment des îlots de spins alignés ( $|M| > 0$ ), et un label *phase paramagnétique*, où les spins sont globalement désordonnés ( $M = 0$ ).

Nous allons explorer différentes méthodes pour voir quelles solutions d'intelligence artificielle nous permettent de trancher, à partir d'une image donnée, si l'on est dans la phase paramagnétique ou ferromagnétique. Nous nous intéresserons aussi à la manière dont sont générées les données du modèle d'Ising. Nous chercherons à voir s'il est possible de prédire la température d'une image de manière fiable et même s'il est possible de générer de nouvelles données sans faire appel à une simulation numérique coûteuse.

## 1 GÉNÉRATION DE DONNÉES

## 2 PRÉ-TRAITEMENT DES DONNÉES

Maintenant que nous avons généré nos données, nous devons nous faire une idée de la forme de nos données afin de pouvoir les traiter de la meilleure façon possible. Les données fournies au départ sont des vecteurs de taille 1600 contenant des 1 et des 0. Ces vecteurs représentent des configurations de spins se trouvant sur une grille 2D de taille  $40 \times 40$ . Le dataset original est composé de 10000 configurations de spins

pour 16 températures différentes comprises entre 0.25 et 4.00 avec un pas de 0.25. Avec ces configurations, un label est associé à chaque configuration. Ce label est une valeur binaire qui nous indique la phase dans laquelle se trouve le système. De plus, nous avons généré ... Comme on peut le voir sur la figure 2.1a, nos données forment un ensemble bruité mais il apparaît une symétrie par rapport à l'axe horizontal. En effet, à basse température, les spins sont majoritairement alignés de la même façon mais de manière aléatoire en + ou -. Cette symétrie de nos données peut poser un problème à nos modèles qui auront dû apprendre à faire la différence entre deux configurations opposées mais équivalentes. Pour éviter ce problème, nous allons symétriser nos données en inversant les spins de toutes les configurations qui ont une moyenne de spin négative. Ainsi, on se retrouve avec des données symétriques par rapport à l'axe horizontal comme on peut le voir sur la figure 2.1b.

Anatole :  
Complète  
cette ligne  
ou sup-  
primer si  
superflu

Vérifier la  
manière  
de symé-  
triser

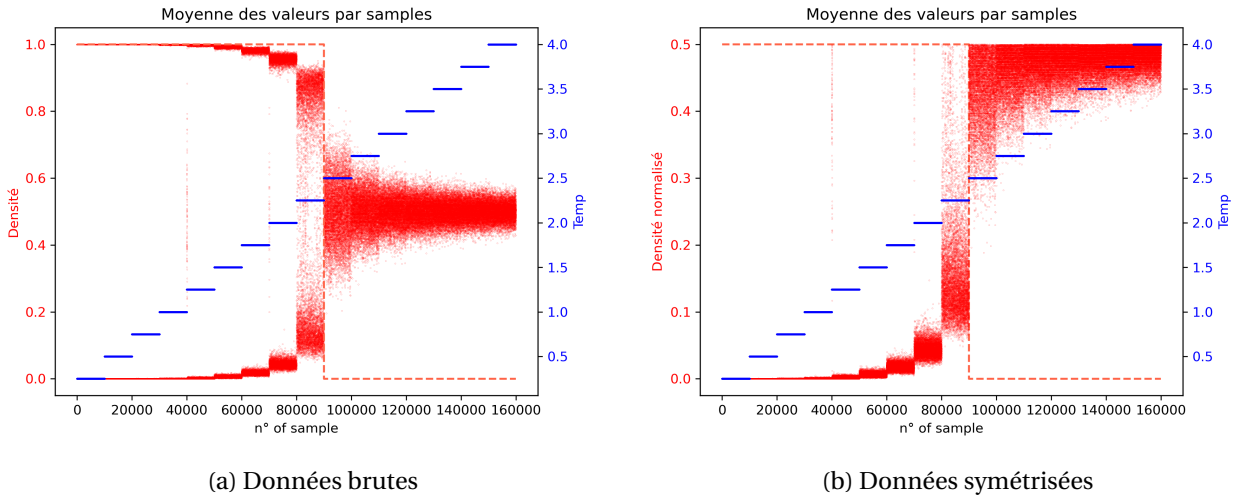


FIGURE 2.1

Dans la partie suivante, nous allons entraîner certains modèles spécifiques sur la densité de spin symétrisée. Dans ce cas, nous allons aussi normaliser afin de rendre les modèles plus performants. Pour cela, nous allons utiliser la méthode *StandardScaler* de la librairie *sklearn* qui permet de centrer et réduire les données. Cette méthode soustrait la moyenne et divise par l'écart-type. Ainsi, on se retrouve avec des données centrées en 0 et de variance 1.

Afin d'appliquer exactement la même transformation sur les données de test, nous allons créer un pipeline qui va appliquer la méthode de symétrisation puis la méthode de normalisation. Ainsi, on pourra appliquer le pipeline sur les données de test sans avoir à les modifier. Finalement, certains modèles seront plus réceptifs à des données présentées sous forme de vecteur de taille 1600, d'autre sous forme de matrice de taille  $40 \times 40$ . Pour éviter de devoir modifier les données à chaque fois, nous allons créer un pipeline qui va transformer les données en matrice si nécessaire et appliquer les autres méthodes de pré-traitement expliquées ci-dessus.

### 3 MODÈLES CLASSIQUES

#### 3.1 Analyse par la magnétisation

Notre première approche de ce problème est de considérer la magnétisation comme une fonction de la température. Cette approche nous permet de réduire grandement la complexité du problème. En effet, nous n'avons plus qu'une seule variable à considérer : l'état moyen des spins. Dans cette partie, nous allons donc essayer de prédire la température à partir de la magnétisation. Il suffit ensuite de comparer la valeur prédite à la valeur de la température critique pour déterminer la phase du système.

Avant tous, nous allons établir un modèle naïf qui va nous servir de référence. Ce modèle va simplement renvoyer la valeur moyenne de la température du jeu de données. Ainsi, on pourra comparer les performances de nos modèles avec ce modèle naïf. On obtient une erreur quadratique moyenne de 1.32.

Du fait du bruit de nos données et du faible nombre de températures distinctes, nous devons porter une attention particulière au sur-apprentissage de nos modèles. Prenant en compte le grand nombre de données, nous sommes partis sur un modèle de forêt aléatoire. Ce modèle est très robuste et permet de limiter le sur-apprentissage. De plus, il est très rapide à entraîner et à tester. Nous avons aussi limité la profondeur de l'arbre à 5. De plus, la méthode de *Bootsrap* est activée. Cette méthode permet de créer des sous-ensembles de données de la taille de l'ensemble de données originale. Ainsi, on peut entraîner plusieurs arbres sur des données différentes et les combiner pour obtenir un modèle plus robuste. On peut donc limiter le sur-apprentissage tout en gardant un modèle performant. Les résultats obtenus sont présentés sur la figure 3.1.

Avec une MSE de 0.108, on obtient un modèle qui est 12 fois plus performant que le modèle naïf. Cependant, on peut voir que le modèle a du mal à distinguer les hautes températures. En effet, les valeurs prédites sont toujours inférieures à 3.5.

Compléter avec explication de la partie d'Anatole

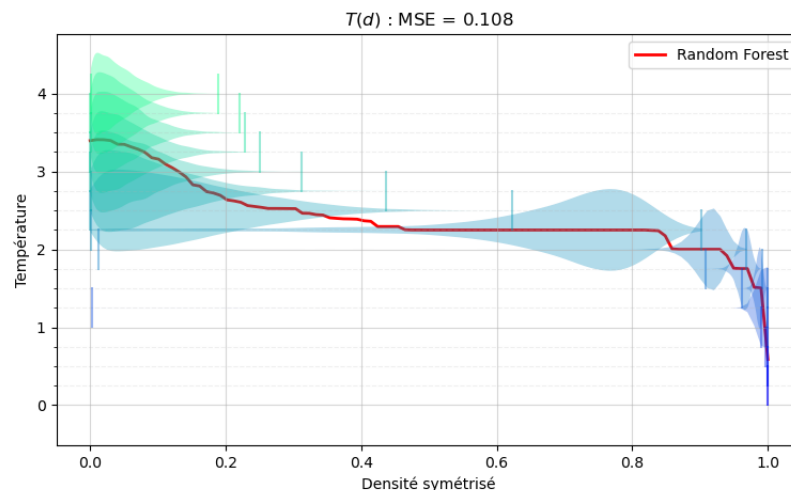


FIGURE 3.1 – Résultats du modèle de forêt aléatoire

### 3.2 Analyse par l'état individuel des spins

## 4 RÉSEAUX DE NEURONES

## CONCLUSION