École nationale supérieure des Mines de Nancy

Rapport de projet 3A

Pierre Gauthier

Algorithme d'apprentissage en chimie quantique et application au screening (sélection) de cellules photovoltaïques

Laboratoire : Institut Élie Cartan

Tuteurs : Dario Rocca et Marianne Clausel

21 Novembre 2018

Table des matières

Introduction :

Ce projet est conduit dans un cadre pédagogique en tant que projet de troisième année à l'Ecole des Mines de Nancy. Il suit la publication scientifique de Mathias Rupp "Machine Learning for Quantum Mechanics in a Nutshell". Dans cette publication Mathias Rupp propose d'allier la mécanique quantique aux méthodes de machines learning pour faire de la prédiction à partir de données et ainsi dépasser les problèmes en terme de puissance de calcul du problème à N-corps. Il vise ainsi à prédire l'énergie d'atomisation de molécules à partir d'un set de données d'entrainement, en utilisant des méthode de linéaire, notamment la régression à vecteur supports (SVM). Le premier objectif du projet est de reproduire les résultats de cette études, et d'explorer des variations dans les paramètres sur l'erreur finale de prédiction. Nous pourrons également dépasser le travail réaliser dans l'étude en travaillant sur des données avec de nouveaux descripteurs qui prennent en compte les propriétés des groupes chimiques des molécules.
Nous allons dans une première partie présenterer les méthodes à vecteur support avec l'astuce des noyaux.

Chapitre 1

Théorie

1.1 Support Vector Machines methods (SVM

Le probléme SVM vise à séparer les données $(x_i,\ y_i)_{1\leqslant i\leqslant N},\ x_i \in \mathbb{R},\ y_i \in \{-1,\ 1\}$ en deux classes $+1$ et -1 à l'aide de la fonction $f(x) = w \cdot x + b$ $(b \in \mathbb{R},\ w \in \mathbb{R}^d)$ telle que $f(x) > 0 \Rightarrow x \in C_{+1}$, et $f(x) < 0 \Rightarrow x \in C_{-1}$

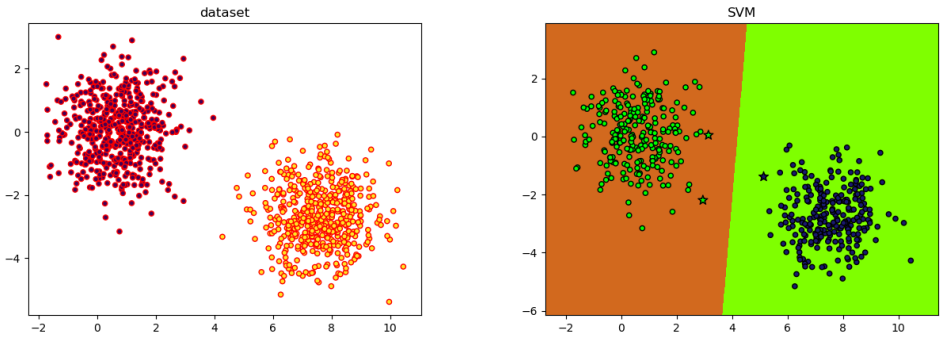

FIGURE 1.1 – sparation de données générées make_blobs du package dataset et séparation de l'espace en deux classe par la méthode des vecteurs support à l'aide de la fonction SVC du package sklearn. Les étoiles sont les vecteurs supports.

Nous voulons trouver l'hyperplan qui sépare le mieux nos données parmi tous ceux compatibles.

Pour juger la qualité d'un hyperplan en tant que séparateur on utilise la distance entre les exemples d'apprentissage et ce séparateur. Plus précisément, la « marge » d'un problème d'apprentissage est définie comme la distance entre le plus proche exemple d'apprentissage et l'hyperplan de séparation.

Pour un hyperplan H, On a :

$$\text{Marge}(H) = \min_{x_i}\ d(x_i,\ H)$$

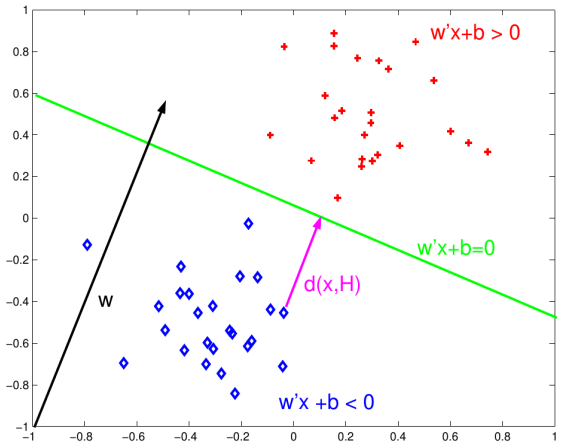

FIGURE 1.2 – Le séparateur idéal correspond intuitivement à l'hyperplan qui passe « au milieu » entre les données sans préférence pour une classe ou une autre. C'est le séparateur de marge maximale.[Cours Cnam RCP209]

Sous l'hypothèse qu'il exixte un hyperplan qui sépare nos données, trouver l'hyperplan qui maximise la marge revient à résoudre le problème suivant :

$$\begin{cases} \arg\min_{w,b}\ \dfrac{1}{2}||w||^2 \\ \forall\ 1 \leqslant i \leqslant N,\ y_i(w \cdot x_i + b) \geqslant 1 \end{cases}$$

On utilise le lagrangien des conditions de Karush, Kuhn et Tucker, qui s'exprime sous la forme suivante :

$$L(w, b, \lambda_i) = \frac{1}{2}||w||^2 - \sum \lambda_i (y_i(w \cdot x_i + b) - 1)$$

On recherche donc le λ qui maximise

$$\max\ L(\lambda) = \sum_i \lambda_i - \frac{1}{2} \sum_i \sum_j \lambda_i \lambda_j y_i y_j x_i \cdot x_j$$

$$\text{sous contraintes } \lambda_i \geqslant 0 \text{ et } \sum_i \lambda_i y_i = 0$$

On utilise alors l'astuce du noyaux qui consiste à remplacer le produit scalaire $x \cdot y$ par un noyaux reproduisant $K(x, y) = \phi(x) \cdot \phi(y)$, $K : \xi \to \mathbb{R}$, $\phi : \xi \to \mathbb{R}$. Le théorème de Mercer assure l'existence d'une telle décomposition du noyaux K

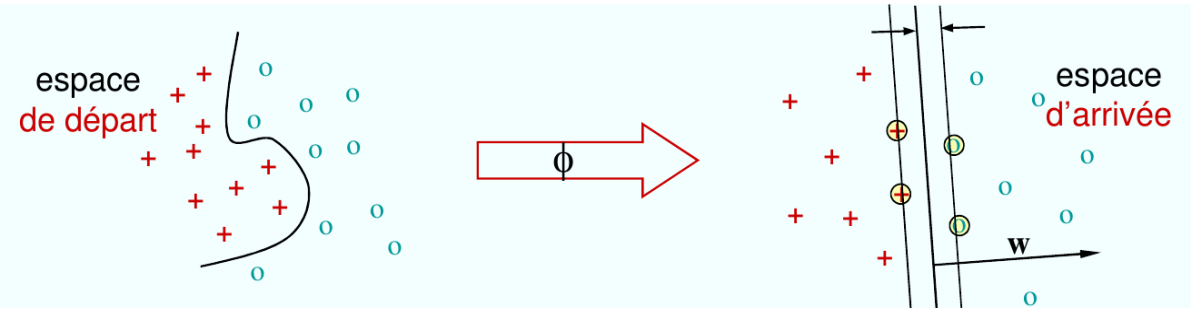

FIGURE 1.3 – Astuce à noyaux : projeter les données dans un espace de dimension beaucoup plus grande, où elles deviennent séparables linéairement.[Cours Cnam RCP209]

L'estimateur devient ainsi en effectuant le produit scalaire sur les données $(x_i)_{1 \leqslant i \leqslant N}$:

$$f(\widetilde{x}) = \sum_{i=1}^{n} \alpha_i \mathrm{k}(x_i, \widetilde{x})$$

1.2 Ridge Regression

Le problème linéaire classique $Y = \sum w_i \cdot x_i$ est optimisé par minimisation de l'erreur :

$$\arg\min_{W \in \mathbb{R}^n} \sum (f(x_i) - y_i)^2$$

La regression ridge consiste à ajouter un poid à cette erreur avec un paramètre $\lambda \geqslant 0$:

$$\arg\min_{W \in \mathbb{R}^n} \sum (f(x_i) - y_i)^2 + \lambda \sum w_i^2$$

En reprenant l'astuce du noyaux précedente ce problème de minimisation peut se réécrire :

$$\arg\min_{\alpha \in \mathbb{R}^n} < K\alpha - y, K\alpha - y > + \lambda \alpha^T K \alpha$$

$$\text{où } K \in \mathbb{R}^{n \times n} \text{ est la matrice des noyaux} \quad K_{i,j} = \mathrm{k}(x_i, x_j)$$

ce qui revient en prenant le gradient à :

$$\alpha = (K + \lambda I)^{-1} y$$

avec I la matrice identité

Chapitre 2

Experimentations

2.1 Base d'entrainement du modèle

Nos données sont initialement contenues dans un fichier .xyz. Chaque élément du fichier est un tableau avec en première ligne le nombre d'atome de la molécule, en deuxième ligne l'énergie d'atomisation et le numéro de la molécule dans le fichier, et pour chaque ligne ensuite correspondantes à chaque atomes les coordonnées cartésienne de l'atome.

Nombre d'atome			
numéro de la molécule	énergie d'atomisation		
atome 1	x(1)	y(1)	z(1)
atome 2	x(2)	y(2)	z(2)
.	.	.	.
.	.	.	.
.	.	.	.
atome n	x(n)	y(n)	z(n)

Par exemple pour la première molécule du fichier que l'on traite :

5						
0001	-417.031					
C	1.04168000	-0.05620000	-0.07148000	1.04168200	-0.05620000	-0.07148100
H	2.15109000	-0.05620000	-0.07150000	2.13089400	-0.05620200	-0.07149600
H	0.67187000	0.17923000	-1.09059000	0.67859800	0.17494100	-1.07204400
H	0.67188000	0.70866000	0.64196000	0.67861300	0.69474600	0.62898000
H	0.67188000	-1.05649000	0.23421000	0.67861400	-1.03828500	0.22864100

FIGURE 2.1 – Premier élément du fichier .xyz correspondant à la première molécule. Le bloc de coordonnées à gauche correspond aux coordonnées du champ de force et le bloc de droite correspond aux coordonnées DFT.

Nous allons ensuite mettre en forme ces données pour entrainer le modèle. On utilise ainsi les matrices de Coulomb pour représenter les molécules. Les matrices de Coulomb sont représentés comme suit :

$$M_{ij} = \begin{cases} 0.5Z_i^{2.4} & i = j \\ \dfrac{Z_i Z_j}{||R_i - R_j||_2} & i \neq j \end{cases}$$

avec Z_i le numéro atomique correspondant et R_i la position des atomes.

Cependant pour chaque molécules les coordonnées des atomes sont déterminé par rapport à un atome de référence ce qui fait que ces matrices ne sont pas insensible à l'inversion de ligne pour l'entrainement du modèle. Pour cela après le calcul des matrices nous trions les ligne par norme descendante.
De plus les matrices de Coulomb étant symétrique nous ne pouvons garder que la partie inférieure de la matrice dans un vecteur colonne de taille égale au plus grand vecteur possible sur nos données.

2.2 entraînement du modèle

Nous allons dans un premier temps travailler avec un noyaux gaussien définis par :

$$\mathrm{k}(x, z) = \exp -\frac{||x - z||_2^2}{2\sigma^2} \quad \sigma \geqslant 0$$

Nous avons donc deux hyperparamètres dans notre modèle d'après le chapitre1 à la page 3

1. σ du noyaux gaussien
2. λ de la ridge régression

Le vecteur α dans l'estimateur $f(\widetilde{x}) = \sum_{i=1}^{n} \alpha_i \mathrm{k}(x_i, \widetilde{x})$ est directement déterminé par $\alpha = (K + \lambda I)^{-1} y$

nous allons ainsi rechercher directement le meilleur couple (λ, σ) sur un ensemble Ω en mimisant l'erreur en utilisant la cross-validation.

Nous allons dégager une base de test de notre dataset (de taille 7000 molécules) une base de taille 1000 sur laquelle nous allons rechercher nos hypermaramètre. Nous décomposons encore cette base de taille 1000 en une base de taille 900 pour le choix des hyperparamètres et une base de taille 100 sur laquelle nous allons verifier la pertinence du choix de nos hyperparamètres, pour se prémunir contre des problème d'overfitting.

De plus si nous nous intéressons à la répartition du nombre d'atome qui ne sont pas des atomes d'hydrogène dans notre dataset, nous pouvons voir sur la Figure 2.2 que cette répartition est trés inégale. En pratique nous prendrons toutes les molécules qui comprenent moins de 5 non-H atomes car leur nombre est trés réduit (59).

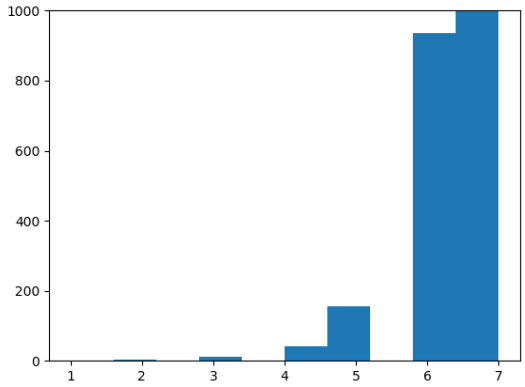

FIGURE 2.2 – Histogramme du nombre d'atome non-H par molécule