

UNIVERSITÉ DE ROUEN

RAPPORT DE PROJET

---

# Minimal Learning Machine

---



Étudiant  
Tuteur

Laziz Hamdi  
M. Simon Bernard

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>MLM</b>	<b>3</b>
2.1	Formulation . . . . .	3
2.2	La régression . . . . .	3
2.2.1	Construire le modèle . . . . .	3
2.2.2	Estimer les sorties $y$ . . . . .	4
2.2.3	Choix du paramètre K . . . . .	4
2.2.4	Complexité . . . . .	5
2.2.5	Expérimentation . . . . .	5
<b>3</b>	<b>Conclusion</b>	<b>6</b>

# **Chapitre 1**

## **Introduction**

Le Minimal Learning Machine est une technique d'apprentissage supervisé qui est apparue en 2015, qui vise à reconstruire la correspondance qui existe entre la matrice de distance ou de dissimilarité entre les données en entrée et celle entre les données en sortie, ensuite exploiter les positions géométrique des distances prédictes pour estimer les réponses. Le MLM fonctionne sur des problèmes à grande dimension et capable de résoudre des problèmes de régressions et de classifications.

# Chapitre 2

## MLM

### 2.1 Formulation

Pour un ensemble de données en entrée  $X = \{x_i\}_{i=1}^N$  avec  $x_i \in R$  et sa correspondance en sortie  $Y = \{y_i\}_{i=1}^N$  avec  $y_i \in S$ .

On suppose qu'il existe une relation continue entre ces deux espaces qu'on appelle  $f : X \rightarrow Y$ , l'objectif est d'estimer  $f$ .

Le processus du MLM se divise en deux étapes principales :

1. Reconstruire la correspondance entre les matrices de distances entrée et sortie.
2. Estimer les réponses grâce à la configuration géométrique des valeurs prédites.

### 2.2 La régression

Le MLM requiert que l'utilisateur entre un nombre de points références  $k$  pour construire les matrices de distances.

On définit l'ensemble des points références  $R = \{m_k\}_{k=1}^k$  sélectionnés aléatoirement de l'ensemble  $X$  et leurs correspondances  $T = \{t_k\}_{k=1}^k$  de l'ensemble  $Y$ , ensuite on construit la matrice  $D_x \in R^{N \times K}$  des distances euclidiennes entre les observations  $x_i$  en ligne et les références en colonne, de la même manière est définie la matrice des distances en sortie  $\Delta_x \in R^{N \times K}$ .

On définit la relation entre ces matrice ainsi  $\Delta_y = g(D_x) + E$  avec  $E$  qui représente le résidus.

On supposant que cette relation est linéaire on peut la représenté sous forme matricielle  $\Delta_y = D_x B + E$ .

$B$  est la matrice des coefficients du modèle.

#### 2.2.1 Construire le modèle

Pour estimer  $B$  c'est à dire les coefficients du modèle, plusieurs méthodes peuvent être utiliser comme les moindres carrés moyen, moindres carrés récursifs pour calculer la différence entre les vraies distances et les distances prédites. cette différence est exprimé avec cette fonction.

$$RSS(B) = \text{tr}((\Delta_y - D_x B)'(\Delta_y - D_x B))$$

Minimiser cette fonction revient à chercher le point où le gradient est nul, ce qui conduit à résoudre un système d'équations où le nombre d'équations est le nombre d'observations  $N$  et le nombre d'inconnue est le nombre de références  $K$ . La solution est différente selon le nombre  $K$ .

- Pour  $K < N$  :  $\hat{B} = (D_x' D_x)^{-1} D_x' \Delta_y$
- Pour  $K = N$  :  $\hat{B} = D_x^{-1} \Delta_y$
- Pour  $K > N$  : une infinité de solutions.

Ce dernier correspond au cas où après sélection des points références uniquement une partie des données est utilisée pour créer le modèle, ceci donne un problème indéterminé car le nombre d'équations est plus petit que le nombre d'inconnue avec une infinité de solution.

Une fois  $B$  estimé pour un point en entrée  $x$ , on construit un vecteur de distances euclidiennes entre ce point et l'ensemble des points références  $d(x, R) = [d(x, m_1) \dots d(x, m_k)]$ , alors dans le cas où  $K = N$  ou  $k < N$ , le vecteur des distances en sortie est le produit entre le vecteur des distances en entrée et la matrice des coefficients.

$$\hat{\delta}(y, T) = d(x, R)\hat{B} \text{ avec } \hat{\delta}(y, T) = [\hat{\delta}(y, t_1) \dots \hat{\delta}(y, t_k)]$$

### 2.2.2 Estimer les sorties $y$

Pour estimer la position géométrique de  $y$  en connaissant ces distances au point références  $t_k$ , on utilise une multitaréation. C'est une technique qui utilise des mesures de distance pour relever les coordonnées spatiales de positions inconnues. En pratique les distances sont mesurées avec erreur, et les méthodes statistiques peuvent quantifier l'incertitude de l'estimation de la position inconnue. De nombreuses méthodes d'estimation de la position d'un point par multitération peuvent être utilisées comme un estimateur linéaire des moindres carrés, un estimateur des moindres carrés pondéré de manière itérative et une technique non linéaire des moindres carrés. En général la technique des moindres carrés non linéaire est la plus performante.

L'estimation de  $y$  peut être estimée en minimisant la fonction objective

$$J(y) = \sum_{k=1}^K ((y - t_k)'(y - t_k) - \hat{\delta}^2(y, t_k))^2$$

Cette fonction de coût possède un minimum en 0 qui est atteint si la valeur estimée est égale à la vraie valeur  $t'y = y$ . Sinon on cherche à avoir un  $t'y$  le plus proche possible avec la technique des moindres carrés non linéaires avec un algorithme de minimisation.

Plusieurs algorithmes de minimisation peuvent être utilisés mais le plus adapté pour ce problème est l'algorithme de Levenberg-Marquardt qui permet de trouver une solution numérique à un problème de minimisation d'une fonction non linéaire dépendant de plusieurs variables. Cet algorithme est plus stable et trouve une solution même s'il démarre très loin du minimum.

### 2.2.3 Choix du paramètre K

Le principale avantage du MLM est qu'il ne possède qu'un seul hyper paramètre  $K$  à optimiser le nombre de point références que l'utilisateur doit rentrer pour. Pour un nombre de références optimal la validation croisée est utilisée, en divisant l'ensemble des données en  $F$  sous-ensembles ensuite en testant avec différentes valeurs de  $K$  les taux de réussites sont calculés avec les moindres carrés moyens pour les vecteurs de distances en sortie  $\hat{\delta}$  et l'estimation des  $\hat{y}$

$$AMSE(\delta) = \frac{1}{K} \sum_{k=1}^K \frac{1}{N_v} \sum_{i=1}^{N_v} (\delta(y_i, t_k) - \hat{\delta}(y_i, t_k))^2$$

$$AMSE(y) = \frac{1}{4} \sum_{s=1}^S \frac{1}{N_v} \sum_{i=1}^{N_v} (y_i^{(s)} - \hat{y}_i^{(s)})^2$$

Les points références sont sélectionnés aléatoirement des données.

## 2.2.4 Complexité

La complexité pour l'étape de formation du modèle dépend fortement de la méthode utiliser pour le calcules des inversions de matrice surtout qu'on construit des matrices de distances toujours plus grandes selon la taille des données et le nombre de points références sélectionnés, puisque on construit des matrices de tailles  $N, K$ .

L'une des méthodes les plus connue est l'inverse de Moore-Penrose qui estime une pseudo inverse de la matrice car dans certains cas les matrices ne sont pas inversibles. L'une des constructions les plus connues de cette méthode est la décomposition en valeurs singulières  $SVD$  qui est très précise mais très gourmande en temps de calcule qui est plusieurs fois plus éléver que le produit matrice-matrice.

Pour accélérer le calcule, plusieurs méthodes ont été proposé comme le produit entre un type spécial de tenseur et une décomposition  $QR$  ou encore un algorithme basé sur une factorisation Cholesky.

La complexité de la phase d'entraînement du MLM est  $\Theta(K^2N)$  c'est similaire à celle d'un algorithme de machine learning lorsque le nombre de neurones cachés est égal au nombre de référence  $K$ .

## 2.2.5 Expérimentation

Le MLM est testé sur 12 ensembles de données les plus fréquemment utiliser dans le monde ensuite ces performance sont comparées à celle de cinq autre méthode de références le machine learning extrême ELM, le réseau de fonction à base radial RBF, les machines à vecteur de support SVM, les processus gaussiens GP et le perceptron multicouches MLP. Tous les ensembles de données sont pré traités de la même manière pour reproduire les expériences à l'identique supprimer les données manquantes, supprimer les données catégorielles, normaliser de la même façon et utiliser la même proportion de données pour l'entraînement et le test.

Pour ces tests le seul hyper paramètre  $K$  du MLM est optimisé avec une validation croisé sur 10-Fold, avec une sélection aléatoire des références depuis l'ensemble de données pour  $k$  allant de 5% à 100% avec un pas de 5%. Tous les modèles sont évalués en utilisant l'erreur quadratique moyenne MSE sur 10 tests indépendants. Le MLM obtient le plus petit taux d'erreur pour 5/8 des problèmes de régressions et pour les autre problèmes il obtient des résultats proches des résultats obtenue par les autres méthodes.

Ces expériences montre qu'en utilisant 20% des points d'apprentissages comme points références semble être un bon choix pour la plus part des ensembles de données.

# **Chapitre 3**

## **Conclusion**

Les résultats des expériences montre que le Minimal Learning Machine peut être un réel atout pour résoudre des problèmes d'apprentissages supervisés. La complexité du MLM lors de la formation du modèle est faible concurrençant les techniques de machine learning les plus rapides. Pour la phase de test une sélections aléatoire des points références et une optimisation avec l'algorithme de Levenberg-Marquardt permet d'atteindre des performances de pointes.