Chapitre 18 : Prétraitement et Ingénierie des données

23 février 2026

Plan du chapitre

L'importance vitale du Prétraitement

En Machine Learning, **la qualité et la représentation des données priment souvent sur le choix de l'algorithme**.

D'un point de vue mathématique, les algorithmes font des **hypothèses implicites** sur la distribution et l'échelle des données en entrée. Le non-respect de ces hypothèses a trois conséquences majeures :

1. Distorsion des calculs de distance.
2. Biais isotrope lors de la régularisation.
3. Paralysie des algorithmes d'optimisation.

1. Algorithmes basés sur la distance

Pour les k-NN ou les SVM (noyau RBF), on utilise la distance euclidienne :

$$d(\mathbf{x}, \mathbf{x}') = \sqrt{\sum_{j=1}^{d} (x_j - x'_j)^2}$$

La tyrannie des grandes échelles

Si la variable x_1 (ex : Salaire) a une variance $\sigma_1^2 \approx 10^8$ et x_2 (ex : Âge) a une variance $\sigma_2^2 \approx 10^2$, alors le terme $(x_1 - x'_1)^2$ **absorbe et annule** mathématiquement l'influence de x_2.

$$d(\mathbf{x}, \mathbf{x}') \approx |x_1 - x'_1|$$

L'algorithme devient aveugle à toutes les variables de petite échelle.

2. Impact sur la Régularisation

Les méthodes Ridge, Lasso, ou SVM minimisent un coût pénalisé :

$$J(\mathbf{w}) = L(\mathbf{w}) + \lambda \|\mathbf{w}\|^2$$

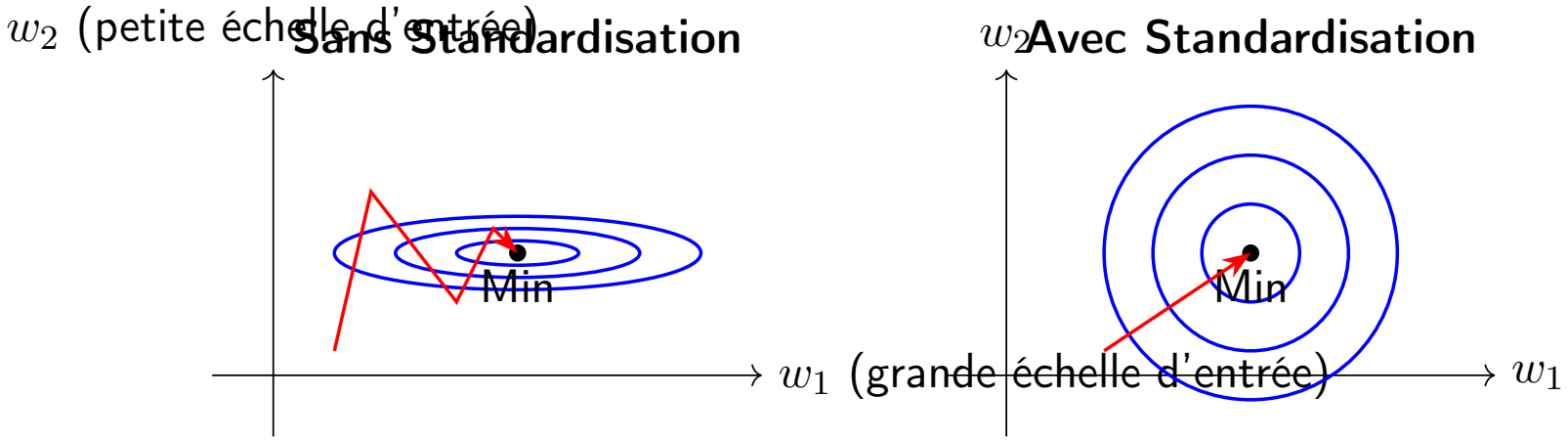
La pénalisation $\|\mathbf{w}\|^2 = \sum w_j^2$ traite tous les poids de manière **isotrope** (symétrique).

Le biais induit

Soit $y = w_1 x_1 + w_2 x_2$. Si x_1 est 100 fois plus grand que x_2, il faut que w_1 soit 100 fois plus petit que w_2 pour avoir le même impact sur y. Or, la régularisation pénalise fortement les *grands* poids (w_2). Sans standardisation, **l'algorithme "tue" artificiellement les variables à petite échelle** !

3. Impact sur l'Optimisation (Descente de Gradient)

La forme de la fonction de coût dépend du conditionnement de la matrice Hessienne H ($\kappa = \lambda_{max}/\lambda_{min}$). Des échelles très différentes rendent κ énorme.



Sans standardisation, les lignes de niveau forment des "canyons". Le gradient rebondit (zig-zag), ce qui ralentit drastiquement la convergence.

Méthodes Usuelles de Remise à l'Échelle

1. Standardisation (Z-score)

Centre et réduit les données. Exigée pour SVM, PCA, Régressions.

$$x'_j = \frac{x_j - \mu_j}{\sigma_j}$$

Propriétés : Moyenne nulle ($\mu = 0$) et variance unitaire ($\sigma^2 = 1$).
Ne détruit pas les valeurs aberrantes (outliers).

2. Normalisation (Min-Max Scaler)

Comprime les données dans l'intervalle $[0, 1]$.

$$x'_j = \frac{x_j - \min(x_j)}{\max(x_j) - \min(x_j)}$$

Propriétés : Préserve strictement les distances relatives initiales.
Souvent exigée par les Réseaux de Neurones ou le traitement

Variables Catégorielles et structures métriques

Les algorithmes manipulent des vecteurs $\mathbf{x} \in \mathbb{R}^d$. Les mots (Rouge, Vert, Bleu) doivent être numérisés.

Le piège du Label Encoding (Encodage Ordinal)

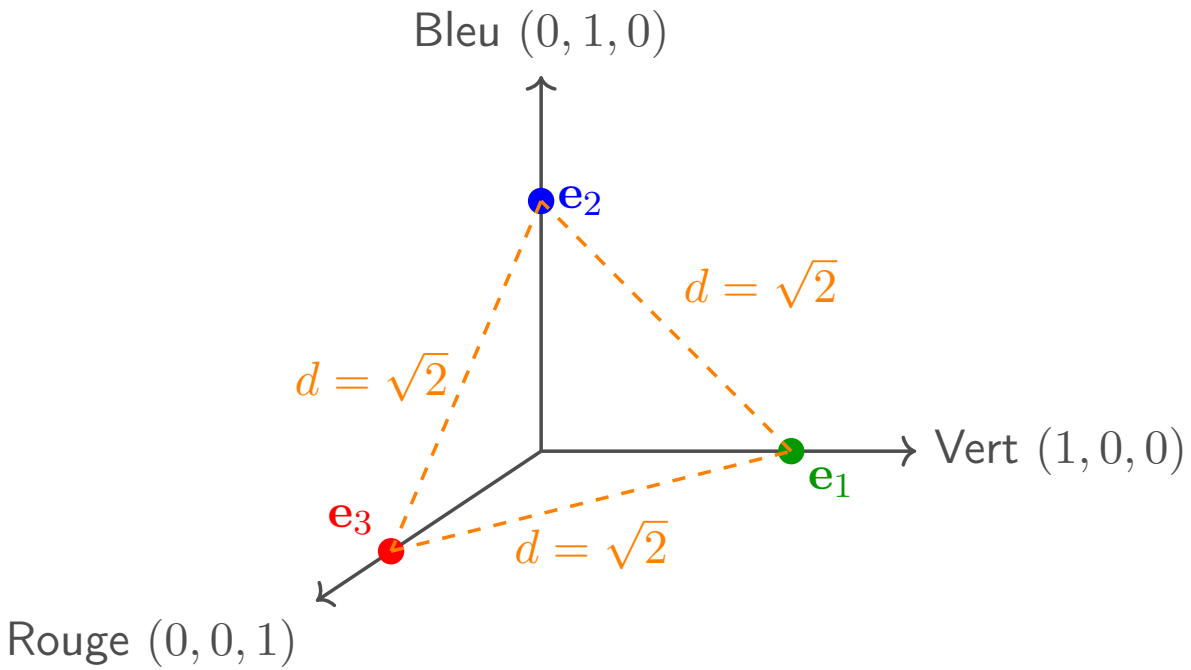
Exemple : Rouge=1, Vert=2, Bleu=3.
L'algorithme va y lire une **structure métrique artificielle** :

- Une relation d'ordre : $1 < 2 < 3$ (Le Bleu serait "supérieur" au Rouge ?).
- Des distances faussées : $|1 - 2| = 1$ mais $|1 - 3| = 2$.

C'est un non-sens mathématique pour des variables nominales.
(Seuls les Arbres de Décision y survivent).

La solution géométrique : One-Hot Encoding

On crée K variables binaires orthogonales. La catégorie k devient le vecteur $\mathbf{e}_k$ de la base canonique dans $\{0,1\}^K$.



- **Orthogonalité** : $\mathbf{e}_i^T \mathbf{e}_j = 0$. Il n'y a plus aucune hiérarchie.
- **Équidistance** : $\|\mathbf{e}_i - \mathbf{e}_j\|^2 = 1^2 + (-1)^2 = 2$. Toutes les classes sont parfaitement à la même distance les unes des autres.