Apprentissage profond en mathématiques

Institut Camille Jordan

5 ocrobre 2023

Le principe du groupe de travail

Slogan
Comprendre assez de choses de l'apprentissage profond par réseaux de neurones pour en déduire des propriétés sur des questions de recherche en maths.

Fonctionnement
TP et exposés se succèdent une semaine sur deux.

Le travail de Williamson et al

Dans l'article "Advancing mathematics by guiding human intuition with AI", Nature, Vol 600, Dec. 2021, Geordie Williamson et al (entre autres des gens DeepMind de Google) ont fait apprendre la structure d'intervalle du groupe symétrique S_n, en tant qu'ensemble partiellement ordonné et la conjecture d'invariance des polynômes de Kazhdan-Lusztig $P_{x,y} \in \mathbb{N}[q]$.

Conjecture : $P_{x,y}$ ne dépend que de l'intervalle $[x, y]$ dans S_n

Ils se sont rendus compte que selon comment ils faisaient varier certains paramètres dans les réseaux de neurones, la précision de l'algorithme s'améliorait...

Les coefficients de Kronecker

Kyu-Hwan Lee a entraîné des réseaux de neurones standard (Nearest Neighbors, Convolutional Neural Networks,...) pour dire si le coefficient de Kronecker $k^{\nu}_{\lambda,\mu}$ associé à 3 partitions de n était non nul ou pas.

Ce coefficient apparaît dans la décomposition du produit tensoriel de représentations irréductibles de S_n :

$$V(\lambda) \otimes V(\mu) = \bigoplus k^{\nu}_{\lambda,\mu} V(\nu).$$

"Machine-learning Kronecker coefficients" arXiv : 2306.04734

D'autres exemples

- "Machine learning the dimension of a Fano variety" de Coates, Kasprzyk et Veneziale, arXiv : 2309.05473
- "Learning phase field mean curvature flows with neural networks" de Bretin, Denis, Masnou et Terii (nos collègues !), arXiv : 2112.07343
- "Machine learning exploration for affine Deligne-Lusztig varieties" de Dong, He, Jin, Schremmer et Yu, arXiv : 2308.11355
- "Machine learning Clifford invariants of ADE Coxeter elements"
- ...

Le théorème d'approximation universelle

Soit σ une fonction continue de $\mathbb{R}$ dans $\mathbb{R}$. On fixe un entier n. On considère des fonctions de $\mathbb{R}^n$ dans $\mathbb{R}$ du type

$$\tilde{f}(x) = \sum_{k=1}^{d} \sigma(w_k \cdot x + b_k),$$

où $w_k \in \mathbb{R}^n$, $b_k \in \mathbb{R}$ et d un entier qui va varier et représente la taille de la couche cachée du réseau de neurones.

Le théorème dit que $\mathscr{M}(\sigma) = \text{Vect}\,\{\sigma(w_k \cdot x + b_k)\}$ est dense dans l'ensemble des applications continues de $\mathbb{R}^n$ dans $\mathbb{R}$ pour la topologie de la convergence uniforme sur les compacts si, et seulement si, σ n'est pas un polynôme.

La fonction XOR

La fonction XOR est la fonction de $\{0,1\}^2 \to \{0,1\}$ définie par $(a,b) \mapsto a+b \mod 2$. On peut écrire un réseau de neurones avec une couche cachée qui "apprend" cette fonction. Par exemple :

$$\mathbb{R}^2 \longrightarrow \mathbb{R}^2 \longrightarrow \mathbb{R}^2 \longrightarrow \mathbb{R}$$

On commence par

$$\begin{pmatrix} a \\ b \end{pmatrix} \mapsto \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} + \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

Puis on applique $\max(0,\cdot)$ coordonnée par coordonnée. Et au final, on applique

$$v \mapsto \begin{pmatrix} 1 & -2 \end{pmatrix} v.$$

Idées d'exposés

- Preuve du théorème d'approximation universelle
- Réseaux de neurones convolutionnels
- Exposé d'Élie sur le travail de Bretin, Denis, Masnou et Terii
- Exposé de Williamson sur la conjecture d'invariance des polynômes de Kazhdan-Lusztig
- Apprentissage profond géométrique
- Sur les coefficients de Kronecker
- ...

Programme des prochaines séances

- 12/10, William Kengne, Techniques d'apprentissage
- 19/10, TP Python et éventuellement début sur Pytorch
- 26/10, Valentina di Proietto, Réseaux de neurones
- 9/11, TP Pytorch
- 16/11, Théo Lopes Quintas, Descente de gradient
- 23/11, TP Pytorch 2, différentiation automatique
- ...