HETIC deeplearning diabetic Groupe 3

Livrable :

Option choisie : C – Données Tabulaires

1. Description du dataset et de la tâche

Pour ce projet, j'ai utilisé le **Diabetes Health Indicators Dataset** provenant de Kaggle.

- **Source :** https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset
- **Objectif :** Prédire si un patient est atteint de diabète (0 = non, 1 = oui) en fonction de mesures médicales.
- **Caractéristiques (features) :** 21 variables d'entrée (Nombre de grossesses, Glucose, Pression artérielle, Épaisseur de la peau, Insuline, … Income).
- **Taille :** 253680 échantillons au total.

2. Prétraitement des données

On a utilisé le **CLI Kaggle** (!kaggle datasets download...) directement dans le notebook

Avant l'entraînement, les données ont subi les étapes suivantes :

1. **Nettoyage :** Chargement via Pandas.
2. **Normalisation :** Utilisation de StandardScaler pour mettre toutes les variables sur la même échelle. C'est une étape cruciale pour les réseaux de neurones afin d'éviter que les variables à grandes valeurs (ex : Insuline) ne dominent les autres.
3. **Split :** Division en 80% pour l'entraînement et 20% pour le test.

3. Architecture du modèle de base (Baseline)

Le premier modèle construit est un Multi-Layer Perceptron (MLP) simple :

- **Structure :** 1 couche d'entrée (21), 2 couches cachées (128 et 64 neurones), et 1 couche de sortie (2).
- **Activation :** ReLU pour les couches cachées.
- **Optimiseur :** Adam avec un learning rate de **0.001**.

- **Époques :** 20.
- **Résultat :** Une précision (Accuracy) de **86,56%**.

4- Amélioration Systématique

Pour améliorer les performances, On a modifié l'architecture et les hyperparamètres :

Paramètre	Baseline	Modèle Amélioré
Architecture	128 / 64 neurones	256 / 128 / 64 neurones
Learning Rate	0.001	0.0005
Époques	20	10
Accuracy finale	86,56%	86,65%

Conclusion

Le passage d'un modèle simple à une architecture plus large, combiné à un taux d'apprentissage plus fin, a permis d'augmenter la précision du modèle. L'utilisation de PyTorch a facilité cette itération rapide. Ce projet démontre l'importance du réglage des hyperparamètres dans le succès d'un modèle de classification tabulaire.