

Rapport de projet Deep Learning II

Cadmos Kahalé-Abdou, João Luís Reis Freitas, et Nicolas Oulianov

Institut Polytechnique de Paris

April 4, 2021

1 Instructions d'exécution

Il suffit d'exécuter le code dans le fichier Analyse.py pour obtenir les figures de ce rapport.

2 Analyse des résultats

Nous avons initialisé les poids de deux réseaux de neurones fully connected :

- Aléatoirement pour l'un (Random init).
- A l'aide d'un préentraînement sur un Deep Belief Network pour l'autre (DBN Pretrain).

Nous avons ensuite comparé leur taux d'erreur après 100 epochs d'entraînement par descente de gradient stochastique. Le dataset utilisé est le MNIST.

Pour les Figures 1 et 2, nous avons entraîné les réseaux sur seulement 1000 échantillons de MNIST afin d'accélérer les calculs. Par ailleurs, on utilise la sigmoïde comme fonction d'activation pour les couches intermédiaires.

2.1 Influence du nombre de couches

Les réseaux de neurones profonds sont durs à entraîner, notamment à cause du problème du vanishing gradient. Le réseau de neurone avec initialisation aléatoire illustre ce problème : le taux d'erreur augmente très rapidement dès qu'on dépasse 3 couches.

Le préentraînement DBN a un taux d'erreur bien plus faible et bien plus stable avec l'augmentation du nombre de couches que l'initialisation aléatoire, même avec un réseau très profond.

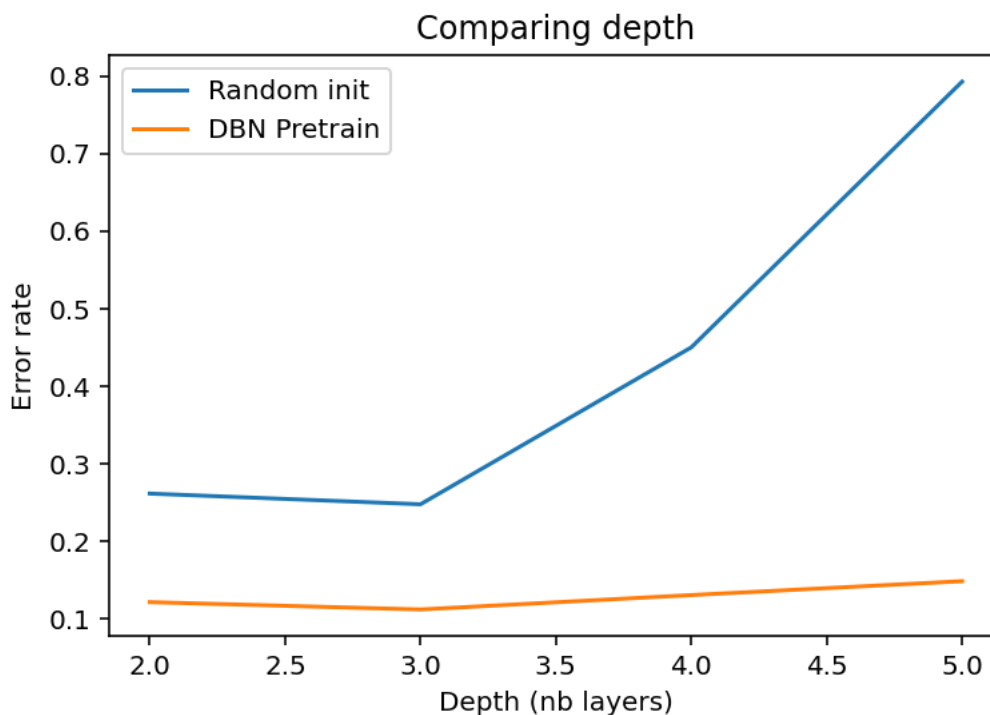


Figure 1: Taux d'erreur en fonction du nombre de couches

2.2 Influence du nombre de neurones par couche

Augmenter le nombre de neurones sur les couches améliore les performances. Vu qu'il n'y a que deux couches cachées, le réseau avec initialisation aléatoire parvient quand même à s'entraîner sur un réseau large : il n'y a pas de problème de vanishing gradient.

Les performances du préentraînement DBN sont, encore une fois, bien meilleures.

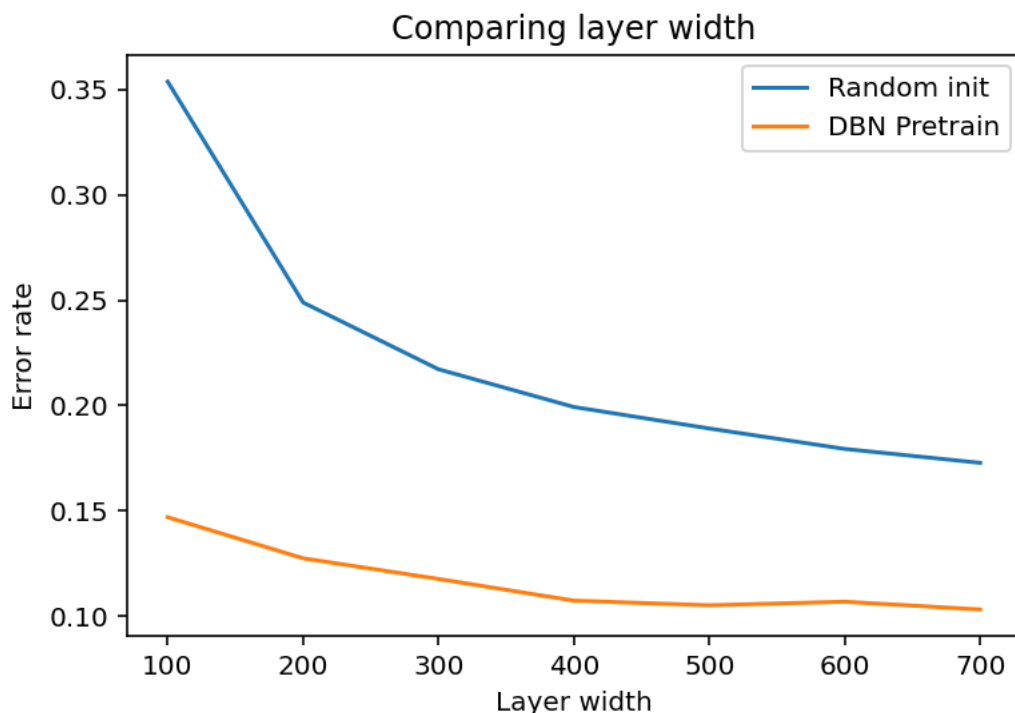


Figure 2: Taux d'erreur en fonction du nombre de neurones par couche

2.3 Influence du nombre de données d'entraînement

Avoir un plus d'échantillons améliore les performances des réseaux de neurones. Comme précédemment, le préentraînement DBN est meilleur que l'initialisation aléatoire.

Sur 1000 données, le réseau avec un préentraînement sur un RBM obtient le même score qu'un réseau avec initialisation aléatoire sur 60 000 données.

Toutefois, le taux d'erreur minimal pour le réseau avec préentraînement DBN est atteint avec 30 000 données. Sans doute il faudrait augmenter le nombre d'époques du préentraînement DBN avec des tailles de datasets plus élevés.

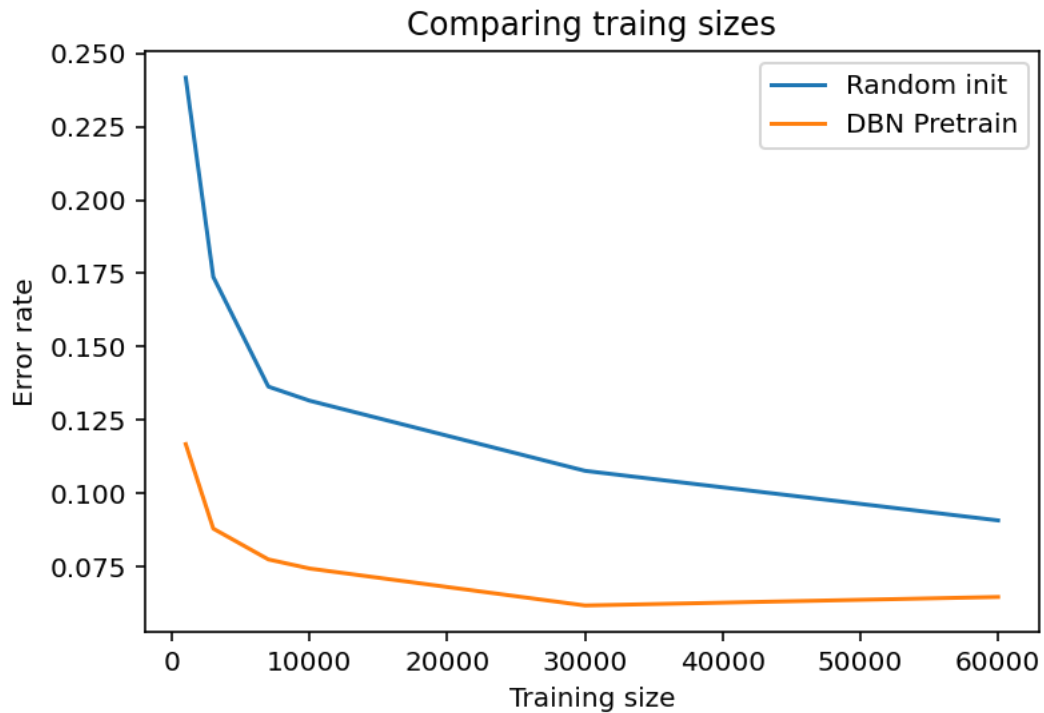


Figure 3: Taux d'erreur en fonction du nombre de données d'entraînement

2.4 Meilleure architecture

On choisit d'entraîner un réseau de neurones à 3 couches cachées de plus en plus petites : 600, 400, 200. C'est une architecture classique. On initialise les poids grâce à un préentraînement DBN sur 5000 samples et 100 epochs. Puis on utilise les 60 000 données d'entraînement, avec 2000 epochs pour le préentraînement DBN et 10 epochs.

Le score final est de 4,9% d'erreur, pour environ 45 minutes d'entraînement.