

M1 Saclay Calcul Haute Performance, Simulation

Mini Rapport

Intitulé du projet :

Reconnaissance de plantes grâce à un réseau de neurones.

Réalisé par :

Bouchelga Abdeljalil - Aitaidier Ferhat - Aicha Maaoui - Lydia Chouaki - Mohamed Aitmhand.

Sous l'encadrement de :

M.Hugo Bolloré

Année universitaire 2021/2022

I. Présentation sur les réseaux de neurones

1. Définition

Un réseau de neurones est une interprétation machine du cerveau humain, qui contient des millions de neurones qui transmettent des informations sous forme d'impulsions électriques.

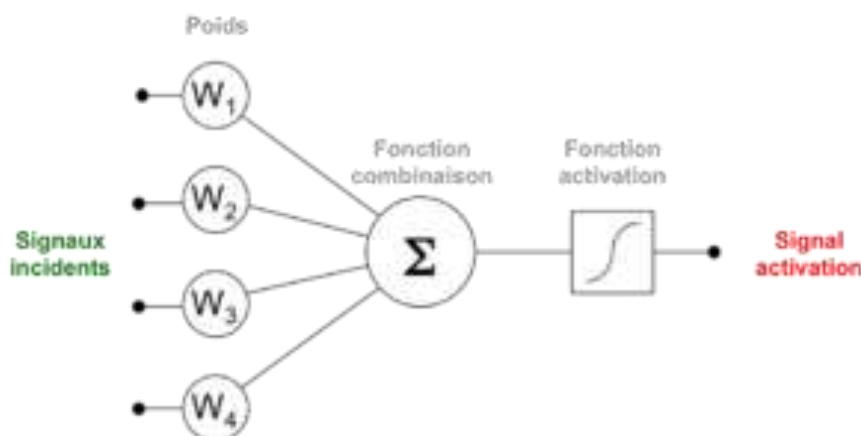
2. A quoi servent les réseaux de neurones ?

Les réseaux de neurones sont utilisés pour résoudre des problèmes complexes qui nécessitent des calculs analytiques similaires à ceux du cerveau humain. Les applications les plus courantes des réseaux de neurones sont :

- **La classification** - la distribution des données par paramètres. Par exemple, un ensemble de personnes est donné à l'entrée et il faut décider laquelle d'entre elles accorder un prêt, et qui ne le fait pas. Ce travail peut être effectué par un réseau de neurones, en analysant des informations telles que l'âge, la solvabilité, Les antécédents de crédit, etc.
- **La prédiction** est la capacité de prédire la prochaine étape. Par exemple, la hausse ou la baisse d'une action en fonction de la situation du marché boursier.
- **Reconnaissance**- actuellement, l'utilisation la plus répandue des réseaux de neurones. Utilisé sur Google lorsque vous recherchez une photo ou dans les appareils photo des téléphones lorsqu'il détecte la position de votre visage et le fait ressortir et bien plus encore.

3. Fonctionnement des réseaux de neurones

Un neurone est une unité de calcul qui reçoit des informations, effectue des calculs simples dessus et les transmet plus loin. Ils sont divisés en trois types principaux : entrée (**x**) avec les deux paramètres **w** (poids) et **b** (bias), caché (**f**) et sortie (**y**).



Dans le cas où un réseau de neurones est constitué d'un grand nombre de neurones, le terme couche est introduit. Par conséquent, il existe une couche d'entrée qui reçoit les informations, n couches cachées (généralement pas plus de 3) qui les traitent et une couche de sortie qui génère

le résultat. Chacun des neurones a 2 paramètres principaux : les données d'entrée et les données de sortie.

Il est important de se rappeler que lors de l'initialisation du réseau de neurones, les poids W sont attribués de manière aléatoire.

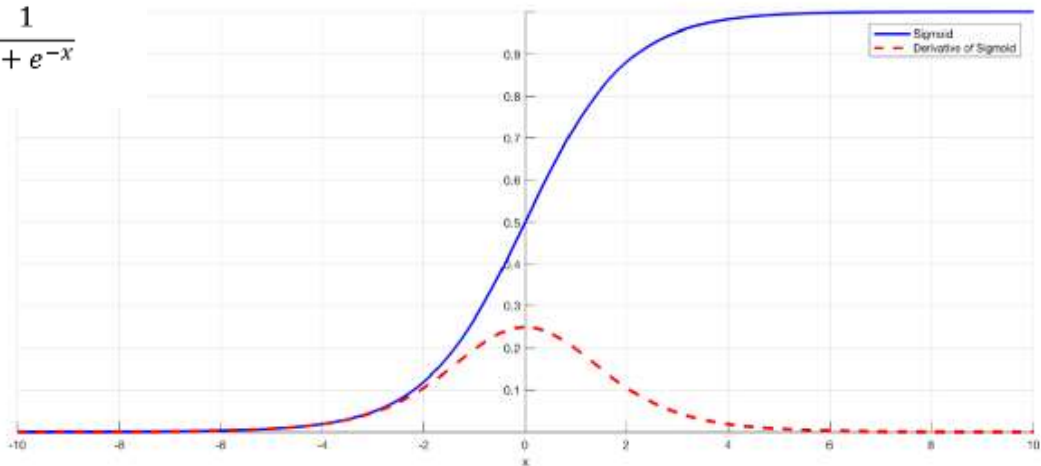
4. Fonction d'activation

Une fonction d'activation est un moyen de normaliser les données d'entrée (nous en avons parlé plus tôt). C'est-à-dire que si vous en avez un grand nombre à l'entrée, après l'avoir passé à travers la fonction d'activation, vous obtiendrez une sortie dans la plage dont vous avez besoin. Il y a beaucoup de fonctions d'activation, nous allons donc considérer les plus basiques : Linéaire, Sigmoid (Logistique) et Tangente hyperbolique. Leur principale différence est la plage de valeurs.

a. Sigmoid

C'est la fonction d'activation la plus courante, sa plage de valeurs est $[0,1]$. Elle montre la plupart des exemples sur le web, elle est aussi parfois appelée la fonction logistique.

$$f(x) = \frac{1}{1 + e^{-x}}$$



La dérivée de la fonction sigmoïde peut aussi être exprimée en fonction de la fonction sigmoïde

$$f(x)' = f(x) \times (1 - f(x))$$

II. Etat de l'art

1. Problématique

La reconnaissance des plantes à partir d'images est une tâche difficile, en raison de l'apparence diversifiée, de la structure complexe des plantes, ou encore le problème de classification, avec un nombre de classes potentiellement élevé. Il s'agit en effet d'un problème bien posé, où les classes sont clairement délimitées et a nous besoin d'information sur les images pour représenter l'objet qu'elle contient et parvenir à l'identifier.

Il y a des méthodes employées pour arriver à ce but, par exemple des méthodes d'analyse de texture sont inspirés par la nature texturale des surfaces de l'écorce et des feuilles, ils font la reconnaissance des plantes comme une classification de texture. Afin de décrire la texture indépendamment de la taille et de l'orientation du motif dans l'image, une description invariante à la rotation et à l'échelle est nécessaire. Il y a aussi des algorithmes qui font l'identification automatique des plantes qui sont développés depuis longtemps : par exemple **LeafSnap**, mais tous les programmes actuellement connus pour l'identification automatique des plantes ont été développés en utilisant de petites bases de données, c'est pourquoi la précision de leur travail est fortement limitée.

2. Travaux parallèles

Des scientifiques de l'**Institut de technologie du Costa Rica** ont présenté une méthode de reconnaissance automatique des plantes à partir d'images d'herbier. Leur algorithme, créé à l'aide de méthodes d'apprentissage en profondeur, entraîné sur 400 000 images de plantes trouvées au Costa Rica et en France, peut identifier correctement les plantes 90 pour cent du temps. (l'article de recherche publié dans BMC Evolutionary Biology)

Les chercheurs ont présenté un algorithme de reconnaissance automatique des plantes à l'aide de la vision par ordinateur. Leur technologie repose sur un réseau de neurones convolutifs (CNN en abrégé), qui se charge du traitement automatique de l'image afin d'isoler certains motifs qui le caractérisent (par exemple, la couleur ou la forme de l'objet représenté).

Cinq bases de données ont été utilisées pour former et tester le réseau de neurones : deux bases de données contenaient des images numérisées de plantes d'herbiers, deux images de plantes vivantes dans la nature et de feuilles numérisées, et une autre base de données (un million d'images extraites d'ImageNet) a été utilisée pour la formation préliminaire. Au total, les chercheurs ont collecté environ 400 000 images de plus de trois mille espèces végétales différentes.

En conséquence, le réseau neuronal, auparavant formé uniquement sur des images d'herbier, a appris à identifier les plantes à partir d'images d'herbier avec une précision de 90 %. Une formation préliminaire basée sur les données d'images de l'herbier a permis au réseau neuronal de déterminer l'image des plantes sur le terrain avec une précision de 66 %. De plus, les chercheurs ont testé la détection automatique de plantes sur des herbiers d'une localité (France) sur la base des données d'herbier d'une autre région (Costa Rica) : le réseau de neurones a appris à résoudre correctement un tel problème avec une précision de 87 %.

3. Conclusion

Face aux grands volumes de données d'apprentissage qui sont disponibles aujourd'hui on suggère que la reconnaissance des plantes est pratiquement un problème résolu.

III. Classification des Images à l'aide du Conventional Neural Network CNN:

1. DataBase:

La base de données citée dans [2] est riche d'images de plantes provenant de différentes classes. Prenant un cas de classe illustrée dans la figure 1.



Figure 1: *Dilsea carnososa* (Schmidel) Kuntze [2].

L'objectif de cet algorithme est de construire un classificateur d'image capable de déterminer les plantes à partir d'une photo donnée. On peut considérer par titre d'exemple 10 classes de plantes de [2]:

- Saxifraga cotyledon L. : <https://www.gbif.org/occurrence/3013500444>
- Polypodium Vulgare L. : <https://www.gbif.org/occurrence/3013501303>
- Thuidium tamariscinum W.P.Schimper, 1852: <https://www.gbif.org/occurrence/3013504463>
- Dilsea Carnosa Schmidel Kuntze : <https://www.gbif.org/occurrence/3013508598>
- Delesseria Sanguinea Hudson J.V.Lamouroux: <https://www.gbif.org/occurrence/3013511508>
- Anomodon Longifolius C.J. Hartman 1838 : <https://www.gbif.org/occurrence/3013535464>
- Kindbergia praelonga Hedw Ochya : <https://www.gbif.org/occurrence/3013585373>
- Saxifraga Cotyledon L. : <https://www.gbif.org/occurrence/3013600364>
- Fraxinus excelsior L. : <https://www.gbif.org/occurrence/3013679581>
- Hypericum tetrapetalum Lam. : <https://www.gbif.org/occurrence/3017940855>

2. Réseau de Neurones

L'algorithme doit contenir des entrées, des training data utilisées pour entraîner l'algorithme à connaître la correspondance des images à des classes spécifiques et des test data pour tester notre algorithme à la fin.

Les training data contiennent des images multiples de chaque classe. La précision dépend du nombre d'images. En augmentant ces derniers, la précision augmente mais aussi le temps de simulation augmente.

Les test data contiennent des images variées et mixtes des différentes classes. Notre but final est que l'algorithme arrive à préciser la classe de chaque image donnée. Donc la sortie doit être

la sortie désirée. Par conséquent, on va chercher à minimiser l'erreur entre la sortie réelle et la sortie désirée.

Dans [3], la classification d'image est faite selon le nombre de pixels p . Ce dernier doit être égal à la longueur multipliée par la largeur de l'image.

Prenant par exemple le cas de la figure 1. Nous avons 10 classes qui vont être considérées comme des filtres f . La sortie Y obtenu pour une image est [3]:

$$resultat\ Y = \sum_{k=1}^{1788864} p_k \times f_k = P \times F$$

avec:

$$P = (p_1 p_2 \dots p_{1788864})$$

$$F = Transpose(f_1 f_2 \dots f_{1788864})$$

Comme nous avons 10 classes (filtres) donc 10 sorties, on peut intégrer les classes dans une matrice 1788864×10 , tel que:

$$(p_1 p_2 \dots p_{1788864}) \times (f_{i,j}) = (y_1 y_2 \dots y_{10})$$

avec $i = 1 : 1788864$ et $j = 1 : 10$

Ainsi, le réseau de neurones schématisé dans la figure 2 de la plante *Dilsea carnosa* (Schmidel) Kuntze a comme entrée le nombre de pixels de l'image. La sortie va être une des 10 classes.

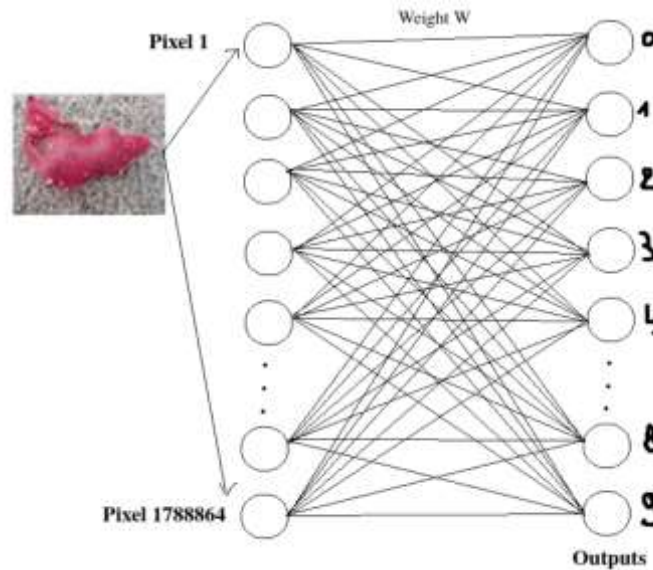


Figure 2: Réseau de Neurones, image d'entrée = *Dilsea carnosa* (Schmidel) Kuntze [2].

Problématique: La taille de l'image est relativement grande, d'où le nombre important de pixels (plus de mémoire, plus de temps de traitement et d'exécution). Il faut donc penser à un redimensionnement des images utilisées.

Dans [4], on a trouvé un exemple de code en Python de la classification des plantes à l'aide de CNN. LA base de données des images utilisées dans le test et le train est donnée dans [5]. 12 espèces de plantes sont utilisées dans ce cas avec un total de 960 images.

Les étapes de cet algorithme sont les suivantes [4]:

- L'importation de bibliothèques (numpy, pandas, matplotlib, etc),
- Obtention des données et redimensionnement des images (cela répond au problématique posée),
- Nettoyage de l'image et suppression de l'arrière-plan,

Remarque: Cette étape de nettoyage et suppression de l'arrière-plan est importante dans notre cas. Dans la figure 3, la partie importante est la reconnaissance de la plante verte de l'image et non pas l'arrière-plan, qui va consommer de ressources de computation.



Figure 3: Saxifraga cotyledon L. [2].

- Conversion des noms en chiffres: Les noms sont des chaînes de caractères, difficiles à traiter, et vont être converti en classification binaire. Dans notre cas de 10 classes par exemple, la classification va être représentée par un tableau de 10 nombres selon la condition: 0 si l'espèce n'est pas détectée, 1 si l'espèce est détectée. Par exemple, si l'espace Dilsea Carnosa Schmidel Kuntze de la figure 1 est détectée (correspond à la classe 4), notre tableau va être [0,0,0,1,0,0,0,0,0,0]
- Définition du modèle et séparation des jeux de données: Dans cette étape, on divise l'ensemble de données d'entraînement pour la validation (test). On peut diviser les données en données de test et données de validation. (Dans [4], 10 % des données totales sont utilisées comme données de test et les 90 % restants comme données d'entraînement)
- Définition du réseau de neurones conventionnels: Dans ce problème, nous allons utiliser un réseau de neurones convolutifs [4]. Ce réseau de neurones prendra des **images en entrée** et fournira la **sortie finale** en tant que **valeur d'espèce**. Dans [4], 4 couches de convolution et 3 couches entièrement connectées sont utilisées aléatoirement.
- Adaptation du CNN aux jeux de données: Le but est que modèle apprenne de l'ensemble de données d'entraînement et les poids soient mis à jour. (Challenges: réduire le taux d'apprentissage, trouver les meilleurs poids pour le modèle et enregistrer ces poids calculés afin que nous puissions les utiliser davantage pour tester et obtenir des prédictions.)
- Vérification des performances du modèle sur les données, Matrice de Confusion: bon moyen pour analyser les erreurs dans le modèle.
- Obtention des prédictions: prédictions sur l'ensemble de données de test à l'aide du modèle entraîné.

IV. Jeu de données

Vous trouverez joint à ce rapport le fichier Excel qui contient notre data fournit par Le Service forestier coréen, contient les caractéristiques suivantes :

- Nom scientifique du taxon
- Pays
- Coordonnées
- Mois & année
- Base de registre
- Classe
- Ordre
- Famille
- Genre
- Espèce

Enregistrement dans le réseau GBIF.

Date d'enregistrement : 17 décembre 2013.

Dernière modification des métadonnées : 18 août 2016.

Licence : CC BY 4.0

Citation : GBIF.org (04 November 2021) GBIF Occurrence Download
<https://doi.org/10.15468/dl.dtvecu>

Références:

[2] Lien, https://www.gbif.org/occurrence/gallery?taxon_key=6

[3] Nicolas Schmid, “Reconnaissance d’images par réseau de neurones”, Lycée cantonal de Porrentruy, novembre 2018.

[4] Lien, https://www.analyticsvidhya.com/blog/2021/05/plant-seedlings-classification-using-cnn-with-python-code/?fbclid=IwAR15rBKrF1GQfy_QSKboG7DMZ1SAXg_tgorIVLBqg26lW3qqAwTqAauf1Eg

[5] Lien, <https://www.kaggle.com/c/plant-seedlings-classification/data>