

Laboratoire - MLP

Réseau multicouche à rétropropagation des erreurs. Entraînement et utilisation d'un réseau d'extraction de caractéristiques et de classification *Application à la reconnaissance de la parole*

Introduction

Le but du laboratoire est de construire un réseau de neurones qui extrait les paramètres les plus pertinents d'un signal de parole et réalise la reconnaissance de la parole.

Ce laboratoire consiste en la conception d'un réseau de neurones à rétro-propagation du gradient de l'erreur. Le travail sera réalisé à l'aide d'un langage de programmation au choix de l'étudiant(e).

Exposé du problème

La difficulté du processus de traitement du signal de parole réside dans la nature même du signal qui est pseudo-aléatoire. De ce fait, l'extraction des paramètres temporels de l'information contenus dans celui-ci devient problématique. Une façon de solutionner le problème est d'en extraire les paramètres les plus pertinents dans le domaine des fréquences. La FFT (Fast Fourier Transform, Transformée de Fourier Rapide) est un outil qui est largement utilisé dans ce domaine pour représenter le signal temporel en fréquence. En général, on filtre le signal sur une série de bancs de filtres selon une échelle logarithmique. Ceci est justifié par le fait que le système auditif humain est logarithmique quant à sa capacité de percevoir les plages de fréquences.

Une fois les paramètres les plus pertinents du signal extraits, ils sont utilisés pour entraîner le réseau de neurones. La tâche de celui-ci consiste à projeter les données dans un espace de dimension réduite permettant une meilleure discrimination entre les différentes formes du signal de parole.

Constitution de la banque de données pour l'apprentissage

La base de données provient de *Texas Instrument* pour les digits.

Conception du réseau

En vous aidant des algorithmes vus en cours et du support pour le laboratoire (voir Powerpoint), créer un réseau de neurones à rétro-propagation du gradient de l'erreur. Deux étapes importantes sont nécessaires pour la mise en œuvre d'un réseau : **a) étape d'apprentissage** : consiste à présenter au réseau les paramètres extraits du signal de parole selon un procédé à déterminer (aléatoire ou non). **b) étape de généralisation** : consiste à tester la capacité du réseau une fois entraîné, à généraliser pour des entrées non vues lors de l'entraînement.

Deux types de contraintes sont proposés pour élaborer ce laboratoire. Des contraintes obligatoires et des contraintes complémentaires.

A. Contraintes obligatoires

Le réseau doit impérativement respecter ces contraintes :

1. Le réseau doit être très flexible du point de vue nombre d'unités dans la couche d'entrée, couche(s) caché(e)s, ainsi que pour la couche de sortie. Le réseau doit donc être capable d'accepter différent nombre de neurones dans les différentes couches du réseau, y compris la sortie.
2. Au moins deux fonctions d'activations doivent être disponibles. La première est la fonction sigmoïde. La seconde est au choix. Bien entendu, l'utilisateur aura la possibilité de choisir parmi les deux possibilités.

3. Utiliser la validation croisée, en combinaison avec un temps maximum d'apprentissage, pour arrêter l'apprentissage.
4. La possibilité d'examiner les unités de la couche cachée, et pouvoir sauvegarder les informations correspondantes dans un fichier. Cette contrainte est indispensable lorsque la couche cachée est utilisée comme entrée pour une autre méthode (qu'elle soit neuronale, Markovienne ou autre). Cette technique est largement utilisée pour la conception de systèmes hybrides.
5. Générer trois (3) sets de bases de données : 40, 50 et 60. Vous travaillerez dans un premier temps avec des données 40. Pour des fins de comparaison, vous utiliserez les données 50 et 60.
6. Un fichier de configuration qui permet de sauvegarder la dernière configuration du réseau utilisée (nombre d'unités, fonction d'activation utilisée, ..., etc). Ce fichier doit être lu par votre programme à chaque démarrage de celui-ci.

B. Contraintes complémentaires

Sans que la liste ne soit exhaustive, parmi les options souhaitées dans un réseau on peut citer :

1. Une interface graphique pour visualiser la structure du réseau, ainsi que toute autre information jugée utile.
2. Possibilité d'utiliser un MLP avec plus d'une couche cachée (2 couche cachée ou plus). Dans ce cas, les résultats obtenus doivent être présentés.
3. Proposer des améliorations pour la règle d'apprentissage (voir cours). Les éléments à améliorer sont, entre autres, le taux d'apprentissage, l'inertie, la règle d'apprentissage, l'initialisation, la durée d'apprentissage, ..., etc.
4. Toute autre fonction ou option permettant une amélioration des performances du réseau.

Il est demandé d'en produire **3** (contraintes complémentaires) et de clairement les identifier dans le rapport de laboratoire.

Évaluation du laboratoire

La pondération suivante sera utilisée pour évaluer le travail réalisé :

Commentaires pertinents	/5
Code du réseau de neurones	/35
Généralisation du réseau	/10
Contraintes complémentaires	/30
Rapport (incluant analyse)	/15
Démonstration et validation	/5
Notes	/100

Date de remise

Consulter moodle.

Procédure de remise - moodle seulement :

Le travail remis doit contenir :

- le code source qui a servi à concevoir le réseau. Le professeur doit être capable de recompiler votre code. Une attention particulière sera donnée pour l'évaluation du code aux déclarations, variables non significatives, définitions statiques, absence de commentaires, ..., etc.
- un fichier lisez-moi. Celui-ci doit décrire brièvement la structure de vos programmes, l'installation, l'exécution avec un exemple simple pour démarrer le programme, ..., etc.
- le fichier de configuration nécessaire pour le démarrage de votre programme.
- la base de données et les poids obtenus (en format texte), ayant permis d'obtenir la meilleure performance.
- rapport détaillé qui explique comment vous avez procédé au développement de votre laboratoire, dont :
 - ☐ les différentes étapes du développement du laboratoire,
 - ☐ les deux fonctions d'activation utilisées,
 - ☐ les contraintes complémentaires ajoutées,
 - ☐ les améliorations apportées et leur efficacité (ou non),
 - ☐ les différentes simulations effectuées : celles-ci devraient suivre une progression logique (e.g., détermination de la meilleure configuration en tenant compte du nombre de neurones approprié en entrée, couche cachée et sortie, la fonction d'activation, la durée de l'apprentissage, la validation croisée, etc.),
 - ☐ l'évolution de l'erreur au cours de a) l'apprentissage, b) la VC et c) la généralisation,
 - ☐ analyse du comportement du réseau de neurones développé, recommandations, conclusions,
- tout autre document ou information qui peut mettre en valeur votre travail.

Check List pour ne pas perdre des points :

	Document à remettre	Points perdus (si non fourni)
<input type="checkbox"/>	Fichier de configuration	-2
<input type="checkbox"/>	Fichier lisez-moi	-2
<input type="checkbox"/>	Code source commenté	-45
<input type="checkbox"/>	Base de données ayant donné la meilleure performance	-5
<input type="checkbox"/>	Les poids ayant donné la meilleure performance clamée dans le rapport	-5
<input type="checkbox"/>	Contrainte complémentaire #1 : la nommer	-10
<input type="checkbox"/>	Contrainte complémentaire #2 : la nommer	-10
<input type="checkbox"/>	Contrainte complémentaire #3 : la nommer	-10
<input type="checkbox"/>	Évolution de l'erreur au cours de a) l'apprentissage, b) la VC et c) la généralisation	-2

Travail en équipe de deux ou trois étudiants/étudiantes **obligatoire**. Aucun travail individuel n'est accepté.