



RdF – Reconnaissance des Formes

Semaine 11 : AD suite et fin

Master ASE : <http://master-ase.univ-lille1.fr/>

Master Informatique : <http://www.fil.univ-lille1.fr/>

Spécialité IVI : <http://master-ivi.univ-lille1.fr/>

Plan du cours

1 – Améliorations de l'apprentissage

limites des arbres de décision, boosting, bagging

2 – Forêts aléatoires

principe de construction d'arbres de décision

Limite des arbres

Quelques limites des arbres de décision

test d'**un seul attribut à la fois** : coupes parallèles aux axes

non incrémental: recommencer la construction de l'arbre si on veut **intégrer de nouvelles données**

sensible à de petites **variations** dans les données (instable)

trouver un arbre de décision d'erreur apparente minimale est, en général, un problème **NP-complet**

==> méthodes pour améliorer l'apprentissage

Boosting : un exemple

Courses de chevaux : quelles prédictions?

on interroge plusieurs parieurs professionnels

supposons :

- que les professionnels ne puissent pas fournir **une règle** de pari simple et performante
- mais que face à **des cas** de courses, ils puissent toujours produire des règles **un peu meilleures que le hasard**

comment devenir riche?

Boosting : un exemple

Idée

demander à l'expert des **heuristiques**
recueillir un ensemble de cas pour lesquels ces heuristiques **échouent**
interroger l'expert (ou un autre expert) pour qu'il fournisse des heuristiques pour ces **cas difficiles**
et ainsi de suite...
==> **combiner** l'ensemble de ces heuristiques

comment choisir les courses à chaque étape?

- se concentrer sur les plus « difficiles » (=celles sur lesquelles les heuristiques précédentes sont les moins performantes)

comment combiner les heuristiques en une seule réponse?

- vote (pondéré) majoritaire des réponses

Boosting

Algorithme d'ensembles

principe d'optimisation de l'apprentissage qui s'appuie sur des **ensembles de classifieurs faibles** par itérations successives d'apprentissage pour arriver au classifieur final

un classifieur faible est capable de reconnaître deux classes au moins aussi bien que le hasard ne le ferait

==> c'est-à-dire qu'il ne se trompe pas plus d'une fois sur deux en moyenne...

ils sont pondérés par la qualité de leur classification

==> plus ils classent bien, plus ils sont importants
les exemples mal classés sont *boostés* pour qu'ils aient davantage d'importance au prochain tour

AdaBoost

Ou Adaptive boosting

une des premières méthodes de boosting

- sélection itérative de classifieur faible en fonction de la distribution des exemples d'apprentissage pondérés

Principe

sur un ensemble d'apprentissage donné

initialiser la distribution (uniforme) des exemples

répéter t fois

- trouver le classifieur qui minimise l'erreur de classification**
- mettre à jour la pondération des exemples d'apprentissage**

Bootstrap aggregating

Autrement connu sous le nom de BAGGING

technique d'apprentissage (classification et régression) visant à

- améliorer la **stabilité**
- réduire la **variance**
- éviter le sur-apprentissage (**overfitting**)

utilisable pour n'importe quel type de modèle
utilisé surtout pour les arbres de décision

Principe

étant donné un ensemble d'apprentissage D de taille n ,
on génère m nouveaux ensembles D_i de taille $n' \leq n$ en
échantillonnant **uniformément** les exemples de D **avec remise**

- les m modèles sont entraînés en utilisant les m ensembles
- les réponses des modèles sont combinées (moyenne ou vote)

Forêts aléatoires

Améliorations des arbres de décision

algorithmes récents (années 2000)

utilise les stratégies adaptatives (boosting) ou aléatoires (bagging)

Les « Random Forests » comme bagging de CART...

Breiman propose en 2001 d'utiliser le bagging pour les AD pour chaque ensemble d'apprentissage généré, sélection aléatoire des variables explicatives à chaque noeud

==> choix du meilleur embranchement parmi un petit nombre

==> plus grande variété de modèles

Forêts aléatoires

Avantages

réduction de la variance (influence des données)
simple à mettre en oeuvre

Inconvénients

temps de calcul plus important
interprétabilité diminuée

introduction du caractère aléatoire : objectif de rendre les modèles (arbres) **plus indépendant entre eux**
==> vote des experts (les CART) **plus efficace**

Pour approfondir

Duda, Hart, Stork, « Pattern Classification », 2ème édition, Wiley-Interscience, 2001.

<http://rii.ricoh.com/~stork/DHS.html>

Breiman, « Random Forests », Machine Learning 45(1):5-32, 2001.

<http://download-book.net/quinlan-c4.5-pdf-doc.html>