Projet 3A

Boosting et sur-apprentissage

proposé par Sébastien Loustau (loustau@cmi.univ-mrs.fr)

1 Introduction

On peut illustrer le pouvoir d'un algorithme d'apprentissage par un jeu simple : le jeu de pierre-papier-ciseau. Face à face, deux joueurs proposent simultanément la pierre, le papier ou le ciseau. Le papier bat la pierre, la pierre bat le ciseau et le ciseau bat le papier. Ce jeu est ainsi strictement basé sur le hasard. Néanmoins on peut construire un algorithme d'apprentissage capable de bonnes performances. Si un être humain joue contre cet algorithme, son score s'éloignera de celui de la machine au cours du temps. La raison est la suivante : le cerveau humain est incapable de générer de façon indépendante et uniforme sa séquence de pierre, papier, ciseau. La machine va ainsi apprendre au fur et à mesure du jeu le comportement du joueur et prédire relativement bien ses coups à l'avance. La Figure 1 montre l'évolution du score d'un joueur (Sarah) face à une machine [1].

2 Modélisation

On considère le problème statistique suivant. On dispose d'un ensemble d'apprentissage $D_n = \{(X_i,Y_i), i = 1,\dots n\}$ constitué de n réalisations indépendantes et identiquement distribuées d'une variable aléatoire (X,Y) de loi P inconnue. Chaque $Y_i \in \{-1,+1\}$ est la classe correspondante à la variable d'entrée $X_i \in \mathcal{X}$. Le but est de prédire la classe Y d'une nouvelle observation X. En d'autres termes, un algorithme de classification construit une règle de décision ou classifieur $\hat{f}_n$ qui à $x \in \mathcal{X}$ associe $y \in \mathbb{R}$ où le signe de $\hat{f}_n(x)$ determine la classe de x. On dit qu'il généralise.

3 Problématique

Dans ce projet, on s'intéresse à un algorithme très influent dans la communauté de l'apprentissage : Adaboost. Ce programme propose, à partir de règles de décisions très simple, un classifieur pertinent qui permet une bonne généralisation. La principale caractéristique de cet algorithme itératif est qu'il se programme en quelques lignes de codes.

L'étude de ce dernier génère de nombreuses interrogations, aussi bien d'un point de vue théorique que d'un point de vue pratique. Est-ce que Adaboost est performant sur tous les jeux de données ? Est-il victime de sur-apprentissage ? Cette question est au coeur de ce projet. On se penchera plus particulièrement sur :

- des moyens théoriques et numériques d'y répondre,
- des alternatives efficaces.

Si le temps le permet, on pourra regarder l'application d'Adaboost au cas multi-classes et considérer des méthodes plus générales de "Boosting".

1. L'expérience a été réalisée en utilisant l'algorithme d'intelligence artificielle WWW Roshambot de Perry Friedman, de l'Université de Stanford.

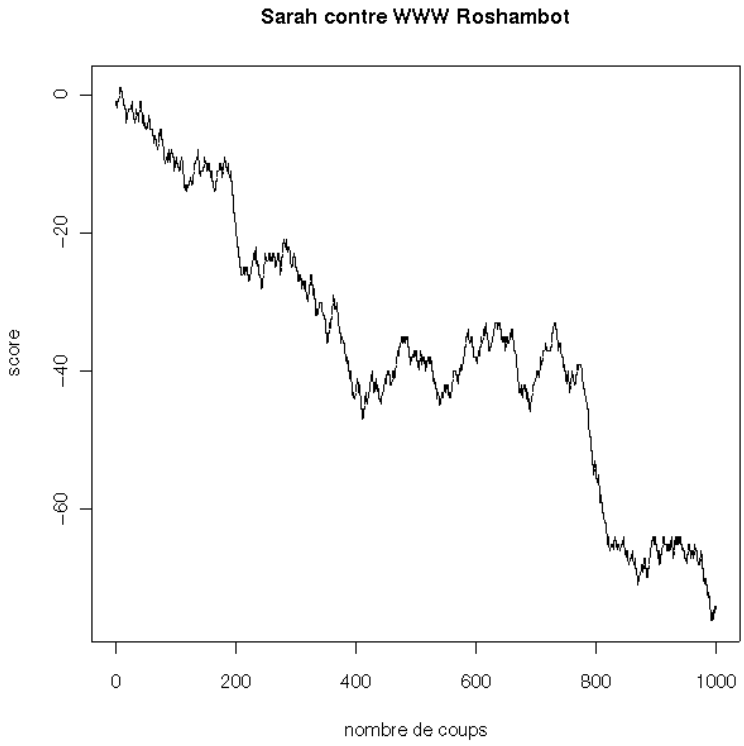

Figure 1 : Evolution du score du joueur face à la machine.

4 Profil du candidat

Ce projet est l'occasion de découvrir la recherche et son mode de fonctionnement. L'autonomie et la motivation seront les principaux atouts des candidat(e)s. Une part importante du travail sera consacrée à la lecture d'articles ou de survols scientifiques du sujet.
De plus, l'utilisation d'un logiciel statistique est indispensable à la programmation de l'algorithme.

Envoyer votre candidature par mail (groupe de 2 personnes maximum).

5 Références

1. S. Boucheron, O. Bousquet, G. Lugosi. Theory of classification: a survey of some recent advances. ESAIM: Probability and Statistics, 9:323-375, 2005.
2. Y. Freund and R. Schapire. Experiments with a new boosting algorithm. In ICML, 148-156, 1996.
3. R. Schapire. The Boosting approach to Machine Learning: An overview. Nonlinear Estimation and Classification. Springer, 2003.
4. R. Meir, G. Ratsch. An introduction to boosting and leveraging. In Advanced Lectures on Machine Learning (LNAI2600), 2003.