L'intérêt de la diversité des solutions de la co-évolution au boosting

Michèle Sebag, LRI, Orsay

Plan

- Co-évolution
  - antagoniste
  - coopérative
- Rapport avec l'apprentissage
  - Evaluation des hypothèses
  - Dilemme biais variance
  - Boosting
  - Bagging

Co-évolution

Hillis 91

Contexte : Optimisation de programme de tri.

Performance : Nb de cas test résolus.

Difficulté : **Nb de cas tests**

- Evaluation exhaustive ? pas de passage à l'échelle
- Echantillonner ? Fitness chaotique
- Sommer ? mais les cas tests sont de difficultés différentes

→ Evoluer deux populations : les programmes et les cas tests

Co-évolution antagoniste

$X \in \Omega$:

- Programmes
- Prédateurs
- Vaccins
- Solutions

$Y \in \Omega'$:

- Cas tests
- Proies
- Virus
- Contraintes

$$\mathcal{F}_t(X) = Interaction(X, \{Y\}_t)$$
$$\mathcal{F}_t(Y) = Interaction(Y, \{X\}_t)$$

Co-évolution co-opérative

Inter-espèces

$X \in \Omega$: Hôtes / Règles

$Y \in \Omega'$: Parasites / Attributs

Intra-espèces

$\Omega = \Omega'$: Parties de solution / Règles d'une base de règles / Joueurs d'une même équipe

Approches

- Universal Suffrage — Neri-Saitta, ECJ 96
- Avalanches — Boettcher 99
- Approche Parisienne — Lutton et al. 01

Co-évolution

Difficulté centrale

Qui peut le plus peut le moins ? NON :-(

- Battre un virus sophistiqué $\not\Rightarrow$ Battre un virus simple
- Battre un bon joueur $\not\Rightarrow$ Jouer bien
- Beware of the Red Queen Paredis ICGA97

Co-évolution

Recommandations

- Diversité intra population : comme d'habitude

  La préserver à tout prix

- Diversité inter-population : équilibrer les progrès

  Unités de temps différentes, $t(X) << t(Y)$

- Diversité inter-temporelle : accès à l'histoire

  Hall of Fame, Rosin-Belew 97

Co-évolution – Hall of Fame

Rosin-Belew 97

Mémoire (X) :

Liste des meilleurs individus X des générations $1..t$

Performance :

$$\mathcal{F}_t(X) = Interaction(X, \quad \{Y\}_t \bigcup \textit{Mémoire}(Y))$$

Eventuellement : échantillonner *Mémoire*(Y).

Co-évolution - Satisfaction de contraintes

Eiben-Hemert 99

Satisfaction de contraintes

Variables : $\mathcal{X} = \{x_0,\ x_1,\ \ldots, x_n\}$

Domaine d'une variable : $Dom(x_0) = \{a_1, \ldots a_L\}$

Contraintes : $p_0(x_0, x_1) \wedge p_1(x_0, x_3) \wedge p_2(x_1, x_3)$

Relation p_0 : $\{p_0(a_1, a_2),\ p_0(a_2, a_7), ..\}$

Affectation θ : $\{x_0, x_1, .., x_n\} \rightarrow \{a_0, a_1, .., a_L\}$

Solution θ tq

$p_0(\theta(x_0), \theta(x_1)) \in$ Relation p_0
$p_1(\theta(x_0), \theta(x_3)) \in$ Relation p_1
$p_2(\theta(x_1), \theta(x_3)) \in$ Relation p_2

Co-évolution - Satisfaction de contraintes, 2

CSP $\equiv$ minimiser une fonction $\mathcal{F}$

- Nombre de contraintes violées
- Somme des poids des contraintes violées

$$Poids(contrainte) \propto Difficulte(contrainte)$$

Auto-ajustement des poids

- Init : poids uniforme
- $poids(p_i, t) \propto |\{\text{individus (t) satisfaisant } p_i\}|$

Co-évolution

Le rapport avec l'apprentissage

Apprentissage supervisé

Formalisation

X	espace des instances	$P(x)$
$c \subset X$	un concept	$X \mapsto Y = \{1, 0\}$
$\mathcal{H}$	une classe de concepts	
tc	le concept cible	$P(y, x)$

Exemples

X	$\mathbb{R}^2$	$\{0, 1\}^n$
$\mathcal{H}$	ens. des rectangles	CNF à n variables
		conjonction de disjonctions
tc	un rectangle	une CNF

Le cadre de l'apprentissage

Données

$\mathcal{D}$: distribution sur X *quelconque, mais fixe*

Exemples $D = \{(x_1, tc(x_1)), \ldots, (x_n, tc(x_n))\}$, x_i tiré selon $\mathcal{D}$.

Coeur

Algorithme A

Au bout de n exemples, A a appris h_n.

Perte $\ell(tc(x), h_n(x))$ $1_{tc(x) \neq h_n(x)}$

Evaluation : Erreur en généralisation

$$Err(h_n) = E_{\mathcal{D}}[\, \ell(tc, h_n)\,]$$

Un problème d'optimisation, mais...

Objectif

Trouver h = Argmin $Err(h)$

Difficulté

Err(h) est inconnue — On n'a pas tous les exemples

Ce qu'on connait : Erreur empirique

$$Err_{emp}(h) = \frac{1}{n} \sum_i \ell(tc(x_i), h_n(x_i))$$

Première époque

Objectif
minimiser l'erreur empirique

? apprendre par coeur ?
... en restant intelligible : rasoir d'Occam.

Un phénomène gênant : le surapprentissage

Choisir l'espace d'hypothèses $\mathcal{H}$

Le compromis biais variance (intuition)

On voudrait

h^* = Argmin $\{Err(h),\ h\ in\ \mathcal{H}\}$

$$Biais = E_{\mathcal{D}}[\ \ell(tc, h^*)\ ]$$

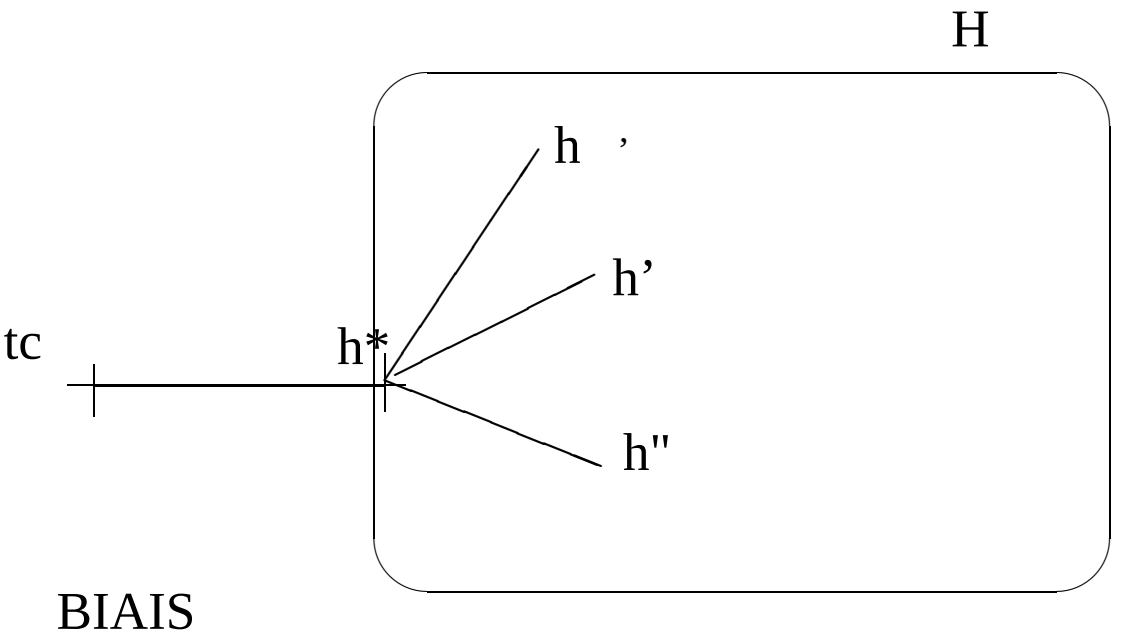
On obtient (au mieux)

h_n = Argmin $\{Err_{emp}(h), h\ in\ \mathcal{H}\}$

$$Variance = E_{\mathcal{D}}[\ell(h^*, h_n)\ ]$$

Erreur = Biais + Variance

Compromis biais variance



Compromis biais variance

Minimiser Biais et Variance simultanément :

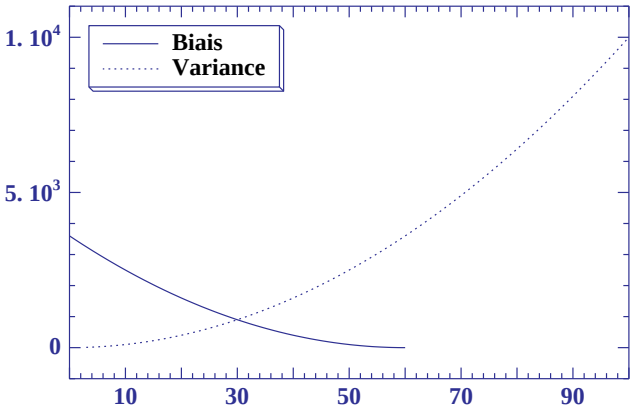
impossible (en général)

$\mathcal{H}$ pauvre : variance faible ; biais grand

$\mathcal{H}$ riche : biais faible ; variance forte

Une optimisation multi-critères...

Compromis biais variance



Cadre

Rappel :

$$h_B = argmin\ Err(h)$$

$$h^* = argmin_{\mathcal{H}}\ Err(h)$$

Cas restreint :

$$h_B \in \mathcal{H}$$

Cas agnostique : on veut

$$Err(h_n) \rightarrow Err(h^*)$$

Cas universel : on veut

$$Err(h_n) \rightarrow Err(h_B)$$

Apprenabilité

Valiant 1984

ϵ : erreur — approximately correct

δ : confiance — probably

Définition PAC-apprenable :

tc PAC apprenable s'il existe un algorithme A tel que $\forall \mathcal{D},\ \forall\ \epsilon,\ \forall\ \delta,$ $\exists n,$

tel que h_n soit probablement approximativement correcte

$$P(Err(h_n) < \epsilon) > 1 - \delta$$

NB: tc est polynomialement PAC si Coût$(h) = Pol(\frac{1}{\epsilon}, \frac{1}{\delta})$

Apprenabilité forte et faible

Supposons $h_B \in \mathcal{H}$

Apprenabilité forte : trouver h_n, pour ϵ et δ petits

$$P(Err(h_n) < \epsilon) > 1 - \delta$$

Apprenabilité faible : trouver h_n, pour ϵ grand, mais non trivial et δ petit

$$P(Err(h_n) < \epsilon) > 1 - \delta$$

Ex : $\epsilon = \frac{1}{2} - \gamma$

Apprenabilité forte et faible

Théorême — Schapire MLJ 1990

Apprenabilité forte $\Longleftrightarrow$ Apprenabilité faible

Si on sait trouver $h_{1/2-\gamma}$ avec une complexité c,
sous toute distribution
on sait trouver h_ϵ avec une complexité $Pol(c)$.

Apprenabilité forte, Apprenabilité faible

1- On apprend h $Err(h) < \frac{1}{2} - \gamma$

2- On apprend h' sur $\mathcal{D}' = \{x \ / \ Pr(h(x) \neq tc(x)) = \frac{1}{2}\}$

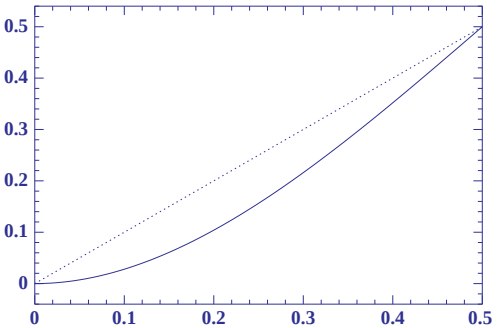
3- On apprend h'' sur $\mathcal{D}'' = \{x / h(x) \neq h'(x)\}$

4- Considérons $Vote(h, h', h'')$: correct si
h et h' corrects, ou ((h ou h' incorrect), et h'' correct)

$$\begin{aligned} Pr(Vote(h, h', h'')\ OK) &= Pr(h\ OK\ et\ h'\ OK) + \\ &\quad Pr(h\ ou\ h'\ \neg OK).Pr(h''\ OK) \\ &\geq 1 - (3\eta^2 - 2\eta^3) \end{aligned}$$

Apprenabilité forte, Apprenabilité faible (2)

$$Err(h) < \eta \Rightarrow Err(Vote(h, h', h'')) < 3\eta^2 - 2\eta^3$$



Algorithme dérivé : le boosting

Prérequis : Algorithme A : apprenti faible.

INIT

Créer $\mathcal{D}_0$ = distribution uniforme

apprendre h_0 sur $\mathcal{D}_0$

BOUCLE

Créer $\mathcal{D}_i$ = altérer $\mathcal{D}_{i-1}$ pour insister sur $\{x / h_{i-1}(x) \neq tc(x)\}$

apprendre h_i sur $\mathcal{D}_i$

OUTPUT

Vote pondéré des h_i,

Poids$(h_i) = f(Erreur_{\mathcal{D}_i}(h_i)$

Le boosting, Algorithme

Input

$\mathcal{E} = \{(x_i, y_i), i = 1..N,\ y_i\ in\ \{0,\ 1\}\}$

$\mathcal{L}$: Algorithme Faible

T : nb d'itérations.

Init

$w_0(i) = \frac{1}{N},\ i = 1..N$ distr. initiale uniforme

Le boosting, Algorithme, 2

Pour $t = 1..T$

$h_t = Alg(\mathcal{E}, w_t)$

$\varepsilon_t = \sum_{y_i \neq h_t(x_i)} w_t(i)$ Si $\varepsilon_t > 1/2$, recommencer

$$\beta_t = \frac{\varepsilon_t}{1 - \varepsilon_t}$$

Mise à jour des poids:

$$w_{t+1}(i) = w_t(i) \times \begin{cases} \beta_t & \text{si } x_i \text{ est bien classé} \\ 1 & \text{sinon} \end{cases}$$

Normaliser w_t.

Fin de la boucle

Le boosting, Algorithme, 3

Hypothèse finale h^* : Signe de

$$\sum_{i=1}^{T} log(\frac{1}{\beta_t}) \times (h_t - \frac{1}{2})$$

I. Borne de l'erreur d'apprentissage ε

I.1 On borne la somme des $w_t(i)$ en fonction du produit des ε_t:

$\sum_{i=1}^{N} w_{t+1}(i) = \sum_{i=1}^{N} w_t(i) \times \beta_t^{1-|h_t(i)-y_i|}$

$\leq \sum_{i=1}^{N} w_t(i) \times (1 - (1 - \beta_t)(1 - |h_t(x_i) - y_i|))$

car $X^r \leq 1 - (1 - X)r$

(développement Taylor)

$$\leq \sum_{i=1}^{N} w_t(i) \times (1 - (1 - \beta_t)(1 - |h_t(x_i) - y_i|))$$

$$= \sum_{i=1}^{N} w_t(i)(1 - (1 - \beta_t)(1 - \varepsilon_t))$$

d'où

$$\sum_{i=1}^{N} w_{T+1}(i) \leq \prod_{t=1}^{T}(1 - (1 - \beta_t)(1 - \varepsilon_t))$$

I.2 On borne l'erreur de h^* en fonction de la somme des $w_t(i)$.

$$\sum_{i=1}^{N} w_{t+1}(i) \geq \sum_{h^*(x_i) \neq y_i} w_{t+1}(i)$$

avec

$$w_{t+1}(i) = w_0(i) \prod_{t=1}^{T} \beta_t^{1-|h_t(i)-y_i|}$$

Or, l'exemple i est mal classé par h^* ssi

$$\prod_{i=1}^{T} \beta_t^{-|h_t(x_i)-y_i|} \geq \prod_{i=1}^{T} \beta_t^{-\frac{1}{2}}$$

Donc

$$\sum_{i=1}^{N} w_{t+1}(i) \geq \prod_{i=1}^{T} \beta_t^{\frac{1}{2}} \times \varepsilon$$

Finalement,

$$\varepsilon \leq \prod_{t=1}^{T} \frac{(1-(1-\beta_t)(1-\varepsilon_t))}{\beta_t^{\frac{1}{2}}} = \prod_{t=1}^{T} 2\sqrt{\varepsilon_t(1-\varepsilon_t)}$$

Si $\gamma_t = 1/2 - \varepsilon_t$,

$$\varepsilon \leq \prod_{t=1}^{T} \sqrt{1-4\gamma_t^2} \leq exp(-2\sum_{t=1}^{T} \gamma_t^2) \ \rightarrow_{T\rightarrow\infty} 0$$

II. Borne de l'erreur de généralisation

Définition

Erreur de généralisation % distribution $\mathcal{P}$:

$$\varepsilon_g(h) = \sum_{x \in X,\ h(x) \neq y} \mathcal{P}(x)$$

Erreur empirique : (échantillon (x_i, y_i) tiré selon $\mathcal{P}$

$$\varepsilon_e(h) = \frac{|\{x_i \ / \ h_i(x) \neq y_i\}|}{|\{x_i\}|}$$

Théorême (Vapnik):

Si dans un espace H d'hypothèses "simples", une hypothèse est d'erreur empirique faible sur un grand nombre d'exemples, alors l'erreur de généralisation est probablement faible.

$$Pr(|\varepsilon_e(h) - \varepsilon_g(h)| > 2\sqrt{\frac{d(ln\frac{2N}{d} + 1) + ln\frac{9}{\delta}}{N}}) \leq \delta$$

Faiblesses du boosting

Récompense le bruit

points mal classés $\Rightarrow$ poids $\nearrow$

Si $Err(h_i) = 1/2$, on s'arrête (ou on recommence à zéro).

Prise en compte connaissances du domaine

Choisir l'espace $\mathcal{H}$

pas de miracle !

Du boosting aux comités d'experts

Le bagging

Breiman 1996

D_1 = exemples tirés uniformément avec remise dans D

h_1 : appris à partir de D_1

. . .

D_T = exemples tirés uniformément avec remise dans D

h_T : appris à partir de D_T

$$h = Vote(h_1, ..h_T)$$

Combinaison de classifieurs

Input : $h_1, \ldots h_T$

Objectif : $h_A = sgn(\sum \alpha_i h_i)$

Prérequis : Diversité des classifieurs h_i

Heuristiques de diversité :

- Divers ensembles de données
- Divers ensembles de descripteurs
- Décorréler les h_i pendant l'apprentissage

Autre regard sur la diversité

Apprentis stables
- Plus proche voisin
- Analyse discriminante linéaire

Apprentis instables
- réseaux neuronaux
- arbres de décision

Pourquoi ca marche ?

Stabilité $\Rightarrow$

Variance faible

$$h_A \approx h_i$$

Biais : ca dépend (de la richesse de $\mathcal{H}$)

analyse discriminante linéaire : parfois trop pauvre

Instabilité $\Rightarrow$

Variance large

Biais petit (e.g. arbre ou nn approximateurs universels)

Intérêt

la combinaison réduit la variance (supposant les h_i décorrélées).

$$Variance(h_A) \approx \frac{1}{T} Variance(h_i)$$

Pourquoi ca marche ? (2)

Boosting : classifieur réel

$$h : X \mapsto \mathbb{R} \mapsto \{0, 1\}$$

Information supplémentaire : $|h(x)|$: confiance

Marge : confiance minimum.

Maximiser la marge $\approx$ réduire la richesse de $\mathcal{H}$
$\Longleftrightarrow$ diminuer la variance

(dimension de Vapnik Cervonenkis)

Différence bagging - boosting

1/ Bagging : h_i indépendants

parallélisation possible

2/ Boosting : h_i liés

(h_i "rattrappe les erreurs" de $h_1 \ldots h_{i-1}$).

Illustration de leur diversité :

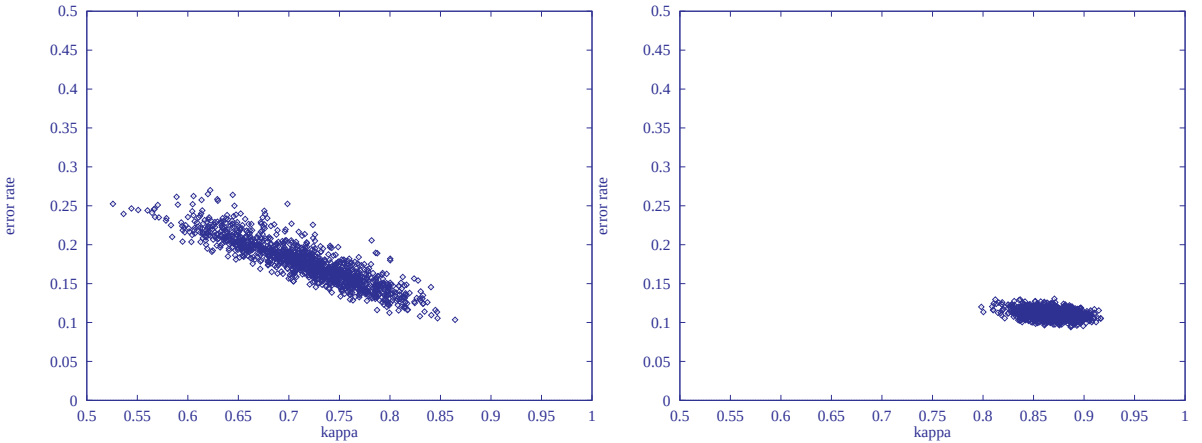


Figure 1: Kappa-Error diagrams for AdaBoost (left) and bagging (right) on the Expf domain.

Apprentissage, Evolution et Diversité

- Co-évolution des hypothèses et des exemples
- La solution est la population : ensemble d'hypothèses
- Apprentissage et optimisation multi-objectif
  Compromis biais-variance

Sources

- Ron Meir, Boosting Tutorial, Machine Learning Summer School, 2002
- Breiman, Arcing classifiers, Annals of Statistics, 1998
- Schapire, MLJ 1990
- Margineantu Dietterich ICML 1997
- `www.boosting.org`