Présentation de l'article *An efficient boosting algorithm for combining preferences* (Y. Freund et al., 1998)

François Rousseau   Jérémie Decock

UPMC

13 octobre 2010

Plan

Le boosting

Rankboost

Résultats

Le boosting

Le boosting

Face à un problème de classification complexe, il est difficile de construire un classifieur

- efficace
- qui généralise bien

En revanche, il est plus facile de construire un classifieur « faible »

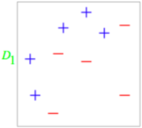
- classant un peu mieux que le hasard
- sur un sous ensemble des données

Boosting

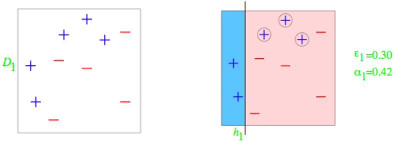
On peut résoudre un problème complexe en combinant intelligement plusieurs classifieurs faibles

Exemple

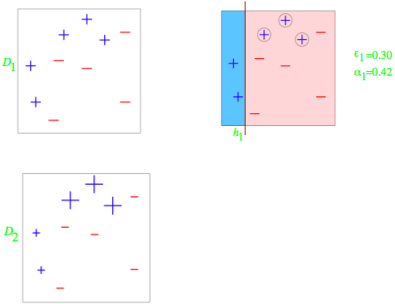
Exemple



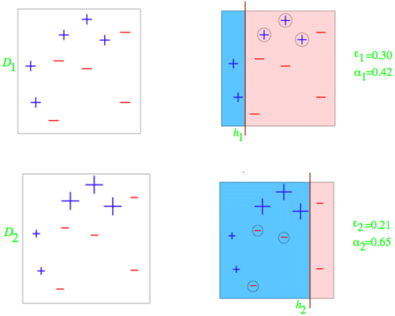
Exemple



Exemple

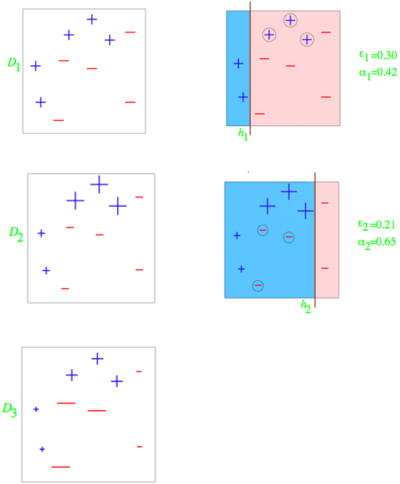


Exemple



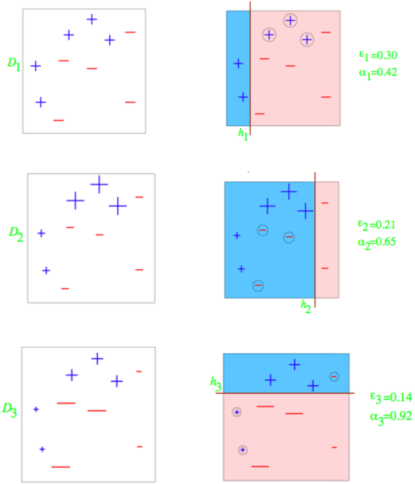
D_1
h_1
$\varepsilon_1=0.30$
$\alpha_1=0.42$
D_2
h_2
$\varepsilon_2=0.21$
$\alpha_2=0.65$

Exemple



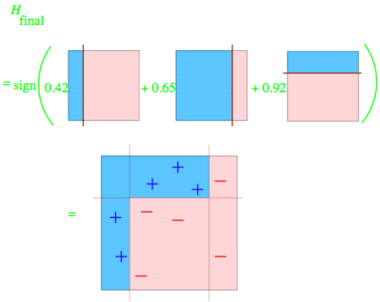
D_1
h_1
ε_1=0.30
α_1=0.42
D_2
h_2
ε_2=0.21
α_2=0.65
D_3

Exemple



D_1
h_1
ε_1=0.30
α_1=0.42
D_2
h_2
ε_2=0.21
α_2=0.65
D_3
h_3
ε_3=0.14
α_3=0.92

Exemple



Historique

Historique

- [Kearns and Valiant 88] : *does weak learnability imply strong learnability ?*
- [Schapire 90] : premier algorithme prouvé *The strength of weak learnability* (boosting par sous ensembles)
- [Freund and Schapire 95] : Adaboost (l'algorithme de référence)
- [Freund and Schapire 98] : Rankboost

Adaboost

Ensemble d'apprentissage étiqueté

$\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$, $\mathbf{x}_i \in \mathcal{X}$, $y_i \in \{+1, -1\}$, $i = 1, m$

Initialisation la distribution des exemples

$D_1(i) = \frac{1}{m}, i = 1, \ldots, m$

Déroulement

Pour $t = 1, \ldots, T$

- tirer un échantillon d'apprentissage $\mathcal{S}_t$ dans $\mathcal{S}$ selon D_t
- trouver une *hypothèse faible* $h_t : \mathcal{X} \rightarrow \{+1, -1\} / h_t = \underset{\epsilon_t}{argmin}$
- calculer le poids α_t de h_t : typiquement $\alpha_t = \frac{1}{2} \ln \frac{1-\epsilon_t}{\epsilon_t}$
- $D_{t+1}(i) = \frac{D_t(i)\, e^{-\alpha_t y_i h_t(\mathbf{x}_i)}}{Z_t}$, Z_t un facteur de normalisation

Hypothèse finale

$H(\mathbf{x}) = sign(\sum_{t=1}^{T} \alpha_t h_t(\mathbf{x}))$

Rankboost

Rankboost

On ne fait plus de la classification mais du *ranking*

- ensemble d'instances $\mathcal{X} = \{x_0, \ldots, x_m\}$
- ensemble de classements $\mathcal{F} = \{f_0, \ldots, f_n\}$
  $f_i(x_0) > f_i(x_1) \Leftrightarrow$ « i préfère x_0 à x_1 »
- distribution des exemples $D(x_i, x_j) = c.max(0, \phi(x_i, x_j))$
  avec $\phi : \mathcal{X}^2 \rightarrow \mathbf{R}$ une relation de préférence
- $h(x) = \begin{cases} 1 & \text{si} \quad f_i(x) > \theta \\ 0 & \text{si} \quad f_i(x) \leq \theta \\ q_{def} & \text{si} \quad f_i(x) = \phi \end{cases}$
  avec $\theta \in \mathbf{R}$ et $q_{def} \in \{0, 1\}$

Rankboost

Initialisation la distribution des exemples

$D_1(x_i, x_j) = c.max\{0, \phi(xi, xj)\}$

Déroulement

Pour $t = 1, \ldots, T$

- *weak hypothesis* $h_t : \mathcal{X} \to \mathbf{R}, h_t = \underset{r_t}{argmax}, h_t \in \mathcal{H},$
  $$r_t = \sum_{x_i, x_j} D_t(x_i, x_j)(h_t(x_j) - h_t(x_i))$$
- *poids du classifieur* : $\alpha_t \in \mathbf{R}, \alpha_t = \frac{1}{2}\ln\frac{1+r_t}{1-r_t}$
- $D_{t+1}(x_i, x_j) = \frac{D_t(x_i, x_j)\, e^{\alpha_t(h_t(x_i) - h_t(x_j))}}{Z_t}$

Hypothèse finale

$H(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$

Résultats

Résultats

Données

- recommandation de films (sources : *Digital Equipment Corporation*)
- 61 625 utilisateurs
- 1628 films
- 2 811 983 classements

Résultats

Méthodes comparées

- régression
- K plus proche voisin

Critères de performance

- *disagreement*
- *predicted-rank-of top*
- *coverage*
- *rank-of-predicted-top*

Résultats

Rankboost $>$ KNN $>$ Regression

Conclusion

Conclusion

- résultats probants sur les exemples de ranking
- le boosting permet d'améliorer sensiblement les performances des lors que l'hypothèse faible est bien choisie

Critiques

- comparaison avec des versions très simplifiées des algorithmes (surtout pour KNN)
- valeurs non mentionnées et à priori arbitraires pour *truly top-rated instances*
- sensibilité au bruit/valeur aberrante ($<$ exponentielle)