Apprentissage automatique et RI - EARIA 2014

Eric Gaussier

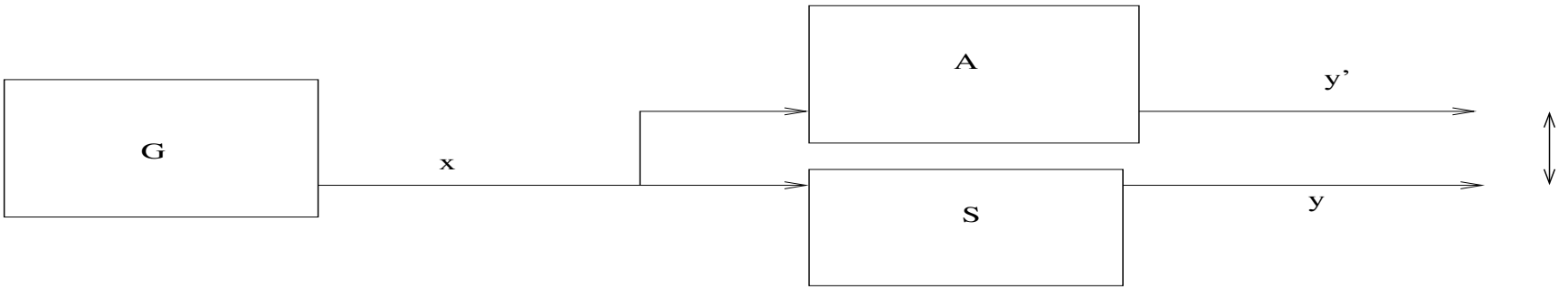
AMA/LIG
Université Grenoble Alpes
UFR-IM^{2}AG
Informatique, Mathématiques et Mathématiques Appliquées de Grenoble
Eric.Gaussier@imag.fr

Octobre 2014

Table des matières

Qu'est-ce que l'apprentissage ?

- **Générateur d'exemples G** produit des exemples x $(P(x))$
- **Système S** produit des valeurs de sortie y pour chaque exemple x $(P(y|x))$
- **Apprenant A** sélectionne, parmi un ensemble donné, la fonction qui lui semble la plus appropriée : $y' = f(x)$



Remarques

1. La seule observation dont on dispose est un ensemble de couples (x, y) (le système S est une boîte noire). Cet ensemble, noté $\mathcal{D} = ((x^{(1)}, y^{(1)}), \cdots, (x^{(n)}, y^{(n)}))$, est *l'ensemble d'apprentissage*
2. $x \in \mathcal{X}$. En général, $\mathcal{X} \subseteq \mathbb{R}^p$. Un document est ainsi représenté par un vecteur dans l'espace vectoriel des termes
3. $y \in \mathcal{Y}$. Dans le cas de la catégorisation binaire, $\mathcal{Y} = \{0, 1\}$

Mesure de la qualité d'une fonction apprise

1. Fonction de coût (*loss function*) :

$$L : \mathcal{Y}x\mathcal{Y} \rightarrow \mathbb{R}^+, \text{telle que } L(y, y') > 0 \text{ pour } y \neq y'$$

Coût 0-1 : $L(y^{(i)}, f(x^{(i)})) = 0$ si $y^{(i)} = f(x^{(i)}), 1$ sinon

2. Risque, risque fonctionnel, erreur de généralisation :

$$R(f) = E_{P(x,y)} L(y, f(x))$$

3. Risque empirique : $\text{Remp}(f; \mathcal{D}) = \frac{1}{n} \sum_{i=1}^{n} L(y^{(i)}, f(x^{(i)}))$

Quelques algorithmes standard

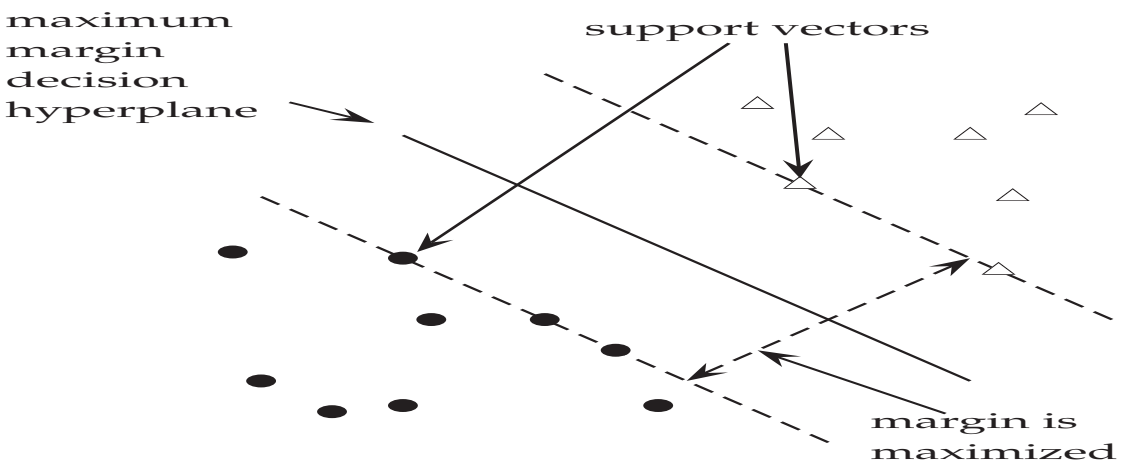
- Les k plus proches voisins (k-PPV / *k-NN*)
- Modèle (log-)linéaire
- Les séparateurs à vaste marge (SVM)

Les SVMs (1)

On cherche une fonction de décision de la forme :

$$f(x) = \text{sgn}(< w, x > +b) = \text{sgn}(w^T x + b) = \text{sgn}(b + \sum_{j=1}^{p} w_j x_j)$$

L'équation $< w, x > +b = 0$ définit un hyperplan de *marge* $2/||w||$)



Les SVMs (2)

Trouver l'hyperplan *séparateur* de marge maximale revient donc à résoudre le problème d'optimisation quadratique suivant :

$$\begin{cases} \text{Minimize} & \frac{1}{2} w^T w \\ \text{subject to} & y^{(i)}(< w, x^{(i)} > + b) \geq 1, \ i = 1, \cdots, n \end{cases}$$

Cas non séparable

$$\begin{cases} \text{Minimize} & \frac{1}{2} w^T w + C \sum_i \xi_i \\ \text{subject to} & \xi_i \geq 0, \ y^{(i)}(< w, x^{(i)} > + b) \geq 1 - \xi_i, \ i = 1, \cdots, n \end{cases}$$

Les SVMs (3)

La fonction de décision prend la forme primale suivante :

$$f(x) = \text{sgn}(< w, x > + b)$$

et la forme duale :

$$f(x) = \text{sgn}(\sum_{i=1}^{n} \alpha_i y^{(i)} < x^{(i)}, x > + b)$$

Table des matières

Modéliser la RI comme un problème de catégorisation

Quelle est la difficulté ?

1. **Choisir une représentation des exemples**
2. Choisir le nombre de classes
3. Choisir le principe d'apprentissage et l'algorithme associé

Représentation des exemples

Problème crucial : quels types d'exemples considérer ? Docs ? ...
Représentation standard, $x = (q, d) \in \mathbb{R}^m$. Les coordonnées $(f_i(q, d), i = 1, \cdots, p)$ sont très générales. On essaye de se reposer sur un maximum d'information :

- $f_1(q,d) = \sum_{t \in q \bigcap d} \log(t^d)$, $f_2(q,d) = \sum_{t \in q} \log(1 + \frac{t^d}{|\mathcal{C}|})$
- $f_3(q,d) = \sum_{t \in q \bigcap d} \log(\text{idf}(t))$, $f_4(q,d) = \sum_{t \in q \bigcap d} \log(\frac{|\mathcal{C}|}{t^{\mathcal{C}}})$
- $f_5(q,d) = \sum_{t \in q} \log(1 + \frac{t^d}{|\mathcal{C}|}\text{idf}(t))$, $f_6(q,d) = \sum_{t \in q} \log(1 + \frac{t^d}{|\mathcal{C}|}\frac{|\mathcal{C}|}{t^{\mathcal{C}}})$
- $f_7(q,d) = \text{RSV}_{\text{vect}}(q,d)$, ...

Choix du nombre de classes

Le choix du nombre de classes dépend *a priori* :

- Des valeurs de pertinence à disposition (binaires ou multi-valuées)
- Des préférences des concepteurs et développeurs du système

Dans le cas le plus simple, on se contente d'un ensemble de deux classes, correspondant aux documents pertinents et non pertinents
$\rightarrow$ Catégorisation binaire

Choix d'un principe d'apprentissage et de l'algorithme associé

Une fois les données représentées comme ci-dessus, toutes les techniques de catégorisation peuvent *a priori* être utilisées. Deux modèles discriminants standard :

1. Modèle (log-)linéaire (maximisation de la vraisemblance discriminante)
2. SVM (Séparateur à Vaste Marge) (minimisation du risque structurel)

Modèle SVM

L'application de la méthode vue précédemment est directe ici. Chaque $x(=(q,d))$ contenant un document pertinent pour q est associé à la classe $+1$, les exemples avec des documents non pertinents à la classe -1

On obtient alors un hyper-plan séparateur, associé à la fonction de décision :

$$g(R|d,q) = w^T x$$

Remarque : On utilise ici directement la valeur de sortie et non le signe de façon à obtenir un ordre sur les documents

Score d'un nouveau document pour une nouvelle requête

Le score d'un nouveau document est directement fondé sur les valeurs des fonctions de catégorisation associées :

- $g(R|d,q)$ pour les séparateurs à vaste marge

La formulation du problème de RI sous forme de problème de catégorisation binaire appelle un certain nombre de remarques

Remarques sur cette approche

1. Cas d'une pertinence multi-valuée : catégorisation multi-classes
2. Méthode qui permet d'attribuer un score, pour une requête donnée, à un document, indépendamment des autres (méthode dite *pointwise*)
3. Résultats comparables à ceux des mdèles probabilistes dans le cas de collections "classiques", meilleurs dans le cas du Web (espace d'attributs plus riches)
4. Méthode qui repose sur une notion "absolue" de pertinence
5. La fonction objectif est "éloignée" de la fonction d'évaluation
6. Disponibilité des annotations ?

Table des matières

Les paires de préférence

• La notion de pertinence n'est pas une notion absolue. Il est souvent plus facile de juger de la pertinence relative de deux documents

• Les jugements par paires constituent en fait les jugements les plus généraux

Représentation des données

On cherche ici une fonction f qui respecte l'ordre des x :
$x^{(i)} \prec x^{(j)} \Longleftrightarrow f(x^{(i)}) < f(x^{(j)})$
Peut-on appliquer les résultats précédents ? Idée : transformer une information d'ordre en une information de catégorie
Elément fondamental couple $x_i = (d_i, q)$ $(1 \leq i \leq n)$; on forme la différence entre deux couples :

$$\{(x_1^{(1)} - x_2^{(1)}, z^{(1)}), \cdots, (x_1^{(p)} - x_2^{(p)}, z^{(p)})\}$$

avec :

$$(x_1^{(i)} - x_2^{(i)}, z = \begin{cases} +1 \text{ si } x_2^{(i)} \prec x_1^{(i)} \\ -1 \text{ si } x_1^{(i)} \prec x_2^{(i)} \end{cases})$$

Ordonnancement et SVMs : *ranking SVM*

$$\begin{cases} \text{Minimize} & \frac{1}{2} w^T w + C \sum_i \xi_i \\ \text{subject to} & \xi_i \geq 0,\ y^{(i)} w^T (x_1^{(i)} - x_2^{(i)})) \geq 1 - \xi_i,\ i = 1, \cdots, p \end{cases}$$

On constitue donc un ensemble d'apprentissage à partir des paires d'exemples. La solution du problème précédent fournit un vecteur w^*.

Comment classer ? $x \prec x'$?

$\Rightarrow x \prec x'$ ssi $\text{sgn}(w^*.(x - x'))$ négatif

Remarques sur Ranking SVM

Comment utiliser w^* en pratique ?

Propriété : $d \succ_{pert-q} d'$ ssi $\text{sgn}(w^*, \overrightarrow{(d,q)} - \overrightarrow{(d',q)})$ positif

Cette utilisation est toutefois coûteuse. On utilise en fait en pratique directement le score « svm » :

$$RSV(q,d) = (w^*.\overrightarrow{(q,d)})$$

Rques

- Pas de différence entre des erreurs faites en tête et en milieu de liste
- Les requêtes avec plus de documents pertinents ont un plus grand impact sur w^*

RSVM-IR (1)

Idée : modifier le problème d'optimisation à la base de *Ranking SVM* (RSVM) pour tenir compte des rangs des documents considérés ($\tau_{k(i)}$) et du type de la requête ($\mu_{q(i)}$)

$$\begin{cases} \text{Minimize} & \frac{1}{2} w^T w + C \sum_i \tau_{k(i)} \mu_{q(i)} \xi_i \\ \text{subject to} & \xi_i \geq 0,\ y^{(i)}(w^T(x_1^{(i)} - x_2^{(i)})) \geq 1 - \xi_i,\ i = 1, \cdots, p \end{cases}$$

où $q(i)$ est la requête associée au $i$ième exemple, et $k(i)$ est le type de rangs associés aux documents du $i$ième exemple

RSVM-IR (2)

En pratique :

- $\mu_{q(i)} = \frac{\max_j \#\{\text{instance pairs associated with } q(j)\}}{\#\{\text{instance pairs associated with } q(i)\}}$

- Pour chaque requête, on établit son ordre optimal (ensemble d'apprentissage) ; on sélectionne ensuite aléatoirement un document pour chaque rang, et on inverse leur position ; ce nouveau ordonnancement induit une baisse de NDCG (ou d'une autre mesure), que l'on moyenne sur toutes les requêtes pour obtenir $\tau_{k(i)}$

RSVM-IR (3)

- En pratique, on utilise directement le w appris (tout comme dans RSVM)
- Les résultats obtenus par RSVM-IR sur des collections standard sont très prometteurs (entraînement à partir des données "campagne d'évaluation"), pour l'instant meilleurs que ceux des modèles probabilistes standard (ce qui n'est pas forcément le cas de RSVM ou des approches par catégorisation binaire)
- Approche *pairwise* vs *pointwise* : retour sur la notion de valeur absolue de pertinence

Extensions des approches précédentes

Approche *listwise*

- Traiter directement les listes triées comme des exemples d'apprentissage
- Deux grands types d'approche
  - Fonction objectif liée aux mesures d'évaluation
  - Fonction objectif définie sur des listes de documents
- Mais les mesures d'évaluation sont en général non continues

Fonction objectif et mesures d'évaluation

- Approche standard, pour des problèmes continus et dérivables
  - Fonction objectif borne supérieure de la mesure d'évaluation (SVM-MAP)
  - Lissage des fonctions objectifs (SoftRank)
- Méthodes d'optimisation pour problèmes non continus
  - Algorithmes génétiques (RankGP)

Table des matières

Constituer des données d'apprentissage

- On dispose de données annotées pour plusieurs collections
  - TREC (TREC-vidéo)
  - CLEF
  - NTCIR
- Pour les entreprises (intranets), de telles données n'exitent pas en général $\rightarrow$ modèles standard, parfois faiblement supervisés
- Qu'en est-il du web ?

Données d'apprentissages sur le web

- Une source importante d'information : les clics des utilisateurs
  - Utiliser les clics pour inférer des préférences entre documents (paires de préférence)
  - Compléter éventuellement par le temps passé sur le résumé d'un document (*eye-tracking*)
- Que peut-on déduire des clics ?

Exploiter les clics (1)

Les clicks ne fournissent pas des jugements de pertinence absolus, mais relatifs. Soit un ordre $(d_1, d_2, d_3, \cdots)$ et C l'ensemble des documents cliqués. Les stratégies suivantes peuvent être utilisées pour construre un ordre de pertinence entre documents :

1. Si $d_i \in C$ et $d_j \notin C$, $d_i \succ_{pert-q} d_j$
2. Si d_i est le dernier doc cliqué, $\forall j < i$, $d_j \notin C$, $d_i \succ_{pert-q} d_j$
3. $\forall i \geq 2, d_i \in C, d_{i-1} \notin C, d_i \succ_{pert-q} d_{i-1}$
4. $\forall i, d_i \in C, d_{i+1} \notin C, d_i \succ_{pert-q} d_{i+1}$

Exploiter les clics (2)

- Ces différentes stratégies permettent d'inférer un ordre partiel entre documents
- La collecte de ces données fournit un ensemble d'apprentissage très large, sur lequel on peut déployer les techniques vues précédemment
- La RI sur le web est en partie caractérisée par une course aux données :
  - Indexer le maximum de pages
  - Récupérer le maximum de données de clics

Table des matières

Letor

http ://research.microsoft.com/en-us/um/beijing/projects/letor/

Tao Qin, Tie-Yan Liu, Jun Xu, and Hang Li. *LETOR : A Benchmark Collection for Research on Learning to Rank for Information Retrieval*, Information Retrieval Journal, 2010

Contient des jeux de données maintenant devenus standard pour l'évaluation d'algorithmes d'ordonnancement

Table des matières

Conclusion - Apprentissage et RI

- Des approches qui tentent d'exploiter toutes les informations à disposition (60 attributs pour la collection *gov* par exemple, y compris les scores des modèles *ad hoc*)
- Des approches qui s'intéressent directement à ordonner les documents (*pairwise*, *listwise*)
- Beaucoup de propositions (réseaux neuronaux (Bing), *boosting*, méthodes à ensemble) fonctions proches des mesures d'évaluation - *NDCG_Boost*)
- Comptent parmi les méthodes les plus performantes à l'heure actuelle

Table des matières

Quelques Références (1)

Burges et al. *Learning to Rank with Nonsmooth Cost Functions*, NIPS 2006

Cao et al. *Adapting Ranking SVM to Document Retrieval*, SIGIR 2006

Cao et al. *Learning to Rank : From Pairwise to Listwise Approach*, ICML 2007

Joachims et al. *Accurately Interpreting Clickthrough Data as Implicit Feedback*, SIGIR 2005

Liu *Learning to Rank for Information Retrieval*, tutoriel, 2008.

Manning et al. *Introduction to Information Retrieval*. Cambridge University Press 2008
www-csli.stanford.edu/∼hinrich/information-retrieval-book.html

Quelques Références (2)

Nallapati *Discriminative model for Information Retrieval*, SIGIR 2004

Yue et al. *A Support Vector Method for Optimizing Average Precision*, SIGIR 2007

Workshop LR4IR, 2007 (Learning to Rank for Information Retrieval).