Modèles de représentation, sélection d'attributs, classification, catégorisation

Liva Ralaivola

LIF, UMR 6166 CNRS
Université de Provence
liva.ralaivola@lif.univ-mrs.fr

19 décembre 2006

Outline

Outline

Absence/présence de mots

Bits indicateurs de la présence/absence de mots

- Représentation sac de mots
- $\mathcal{D} = \{w_1, \ldots, w_d\}$ index (« dictionnaire ») de d mots
- Un texte **t** est codé selon la forme suivante
$$\mathbf{x} = [x_1 \; x_2 \cdots x_d]$$
où $x_i = 1$ si w_i apparaît dans **t** et 0 sinon

Exemple

- $\mathcal{D}$={avenir, donnee, fouille, image, passe, recherche, structure, text}
- Document : "L'avenir est dans la fouille de données textuelles."
- $\mathbf{x} = [1\ 1\ 1\ 0\ 0\ 0\ 0\ 1]$

Absence/présence de mots

Bits indicateurs de la présence/absence de mots

- Représentation sac de mots
- $\mathcal{D} = \{w_1, \ldots, w_d\}$ index (« dictionnaire ») de d mots
- Un texte **t** est codé selon la forme suivante

$$\mathbf{x} = [x_1 \; x_2 \cdots x_d]$$

où $x_i = 1$ si w_i apparaît dans **t** et 0 sinon

Avantages/incovénients

+ modèle extrêmement simple
+ adapté au modèle le plus simple de Naive Bayes
+ résultats raisonnables
- pas de prise en compte de la fréquence de chaque mot
- longueurs des textes ignorées

Fréquences simples

Comptages

- $\mathcal{D} = \{w_1, \ldots, w_d\}$ index de d mots
- Un texte **t** est codé selon la forme suivante
  $$\mathbf{x} = [x_1 \ x_2 \cdots x_d]$$
  où x_i est la fréquence de w_i dans **t**
- Note : en text mining, « fréquence » désigne généralement un comptage (non une proportion)

Exemple

- $\mathcal{D}$={avenir, donnee, fouille, image, passe, recherche, structure, text}
- Document : "L'avenir est dans la fouille de données textuelles. Cela concerne les textes structurés et semi-structurés"
- $\mathbf{x} = [1\ 1\ 1\ 0\ 0\ 0\ 2\ 2]$

Fréquences simples

Comptages

- $\mathcal{D} = \{w_1, \ldots, w_d\}$ index de d mots
- Un texte **t** est codé selon la forme suivante

$$\mathbf{x} = [x_1 \; x_2 \cdots x_d]$$

où x_i est la fréquence de w_i dans **t**

Avantages/incovénients

\+ prise en compte des fréquences de mots (utile pour la détermination du sujet d'un texte)

\+ adapté au modèle multinomial de Naive Bayes

\+ prend en compte la longueur des textes

\- textes étudiés doivent être à peu près de même longueur

\- sac de mots

Fréquences simples

Comptages

- $\mathcal{D} = \{w_1, \ldots, w_d\}$ index de d mots
- Un texte **t** est codé selon la forme suivante
  $\mathbf{x} = [x_1 \ x_2 \cdots x_d]$
  où x_i est la fréquence de w_i dans **t**

Extensions naturelles

- Normalisation des fréquences :
$$x_i = \frac{n_i}{\sum_i n_i}$$
  avec n_i fréquence d'apparition de w_i dans **t**.
- TF-IDF

TF-IDF

Term-frequency/Inverse document frequency

- $\mathcal{C}$: collection de m documents
- $\mathcal{D}$: index de d mots
- Un texte **t** est codé selon la forme suivante $\mathbf{x} = [x_1 \; x_2 \cdots x_d]$ avec :

$$x_i = \frac{n_i}{\sum_i n_i} \ln \frac{m}{m_i}$$

  - n_i : fréq. w_i dans **t**
  - m_i : nombre de documents dans $\mathcal{C}$ où apparaît w_i

TF-IDF

Effets pondération TF-IDF

- Importance de chaque mot dans le texte normalisée
- Un mot qui apparaît dans tous les documents n'est pas « important » en vue d'une différenciation des textes
- Pertinence des mots peu fréquents globalement mais fréquents dans certains documents

Exemple

Cf. TP.

TF-IDF

Origines pondération IDF

- Loi de Zipf
- Théorie de l'information (entropie de Shannon)

Avantages/inconvénients

+ en pratique, méthode de représentation la plus utilisée
+ pondère l'importance d'un terme à l'intérieur d'un document et son importance dans un corpus
+ représentation creuse
- sac de mots
- représentation creuse

TF-IDF

Petit retour sur la loi de Zipf

$$f(r; s, C) = \frac{C}{r^s}, C \text{ constante}, s \approx 1$$

- le 50ème terme le plus fréquent en anglais a une proba d'apparition de 0.0018 que vaut C ?
- on suppose C=0.1, sur un corpus de 10000 mots, quel est le rang d'un mot qui apparaît 10 fois ?
- Combien de termes apparaissent 10 fois dans une corpus de 9000 mots ?

TF-IDF

Petit retour sur la loi de Zipf

$$f(r; s, C) = \frac{C}{r^s}, C \text{ constante}, s \text{ proche de } 1$$

- le 50ème terme le plus fréquent en anglais a une proba d'apparition de 0.0018 que vaut C ?
  $C = 0.0018 * 50 = 0.09$
- on suppose C=0.1, sur un corpus de 10000 mots, quel est le rang d'un mot qui apparaît 10 fois ?$k = C * 10000/10 = 100$
- Combien de termes apparaissent 10 fois dans une corpus de 9000 mots ? Poser r_f rang du dernier terme ayant une fréquence f, utiliser $r_9 - r_{10}$

N-grammes

Sous-séquences de mots

- $\mathcal{D}$: index de d mots
- $\mathcal{D}'$: index de séquences ordonnées de N mots construites sur $\mathcal{D}$

$$\mathcal{D}' = \mathcal{D}^N$$

- représentations booléenne, fréquentielle (TF-IDF)

Exemple

- $\mathcal{D}$={avenir, donnee, fouille, image, passe, recherche, structure, text}, $\mathcal{D}' = \mathcal{D}^2$
- Document : "L'avenir est dans la fouille de données textuelles."
- 2-grammes : "avenir fouille", "fouille donnee", "donnee text"

N-grammes

Sous-séquences de mots

- $\mathcal{D}$: index de d mots
- $\mathcal{D}'$: index de séquences ordonnées de N mots construites sur $\mathcal{D}$
$$\mathcal{D}' = \mathcal{D}^N$$
- représentations booléenne, fréquentielle (TF-IDF)

Avantages/inconvénients

+ prise en compte plus poussée de la structure de la phrase
+ très bons résultats (filtrage, catégorisation, etc) pour $N = 2$, $N = 3$
- taille de l'index très grande (gérable uniquement lorsque N est petit)

Outline

Classification et catégorisation

Objectifs

- Déterminer la classe d'un document en fonction de son contenu
    - spam/non spam
    - thématique (exemple : indexation automatique de documents, cf. Yahoo)
    - ...
- Classification : 2 classes
- Catégorisation (text mining) : plus de 2 classes

Classification et catégorisation

Historique

- Au début (jusqu'au début des années 80) : processus semi-automatique reposant sur le codage d'informations a priori
- Ensuite : méthodes inductives type apprentissage automatique
  - apprentissage *supervisé* : détermination d'un modèle à partir d'exemples étiquetés
  - apprentissage *non supervisé* : pas d'information de classe *a priori*
  - performances atteintes du niveau de celles d'experts humains, automatisation du processus

Classification et catégorisation

Techniques utilisées

- Méthodes génératives
  - Naive Bayes
  - mélange de Gaussiennes
  - ...
- Méthodes discriminatives
  - k-plus-proches voisins
  - régression logistique
  - discriminant de Fisher
  - réseaux de neurones
  - Machines à vecteurs de support
  - ...

Approche par modèle génératif

Cadre supervisé

- $\mathcal{X}$: espace de représentation des textes (e.g $\mathbb{R}^d$ ou $\{0,1\}^d$)
- $\mathcal{Y}$: espace des classes (e.g. $\{0,1\}$ ou $\{-1,1\}$ pour le filtrage et $\{1,2,\ldots,k\}$ pour la catégorisation)
- Données de travail : $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\} \in (\mathcal{X} \times \mathcal{Y})^\ell$
- Hypothèse : les éléments $\mathbf{x}_j$ de chaque classe $c_i \in \mathcal{Y}$ générés suivant une distribution de probabilité, de densité p_i (e.g., $\mathcal{X} = \mathbb{R}^d$)
- Objectif : déterminer $\hat{p}_i$ dans une famille $\mathcal{F}$ de distributions qui modélisent chaque classe c_i

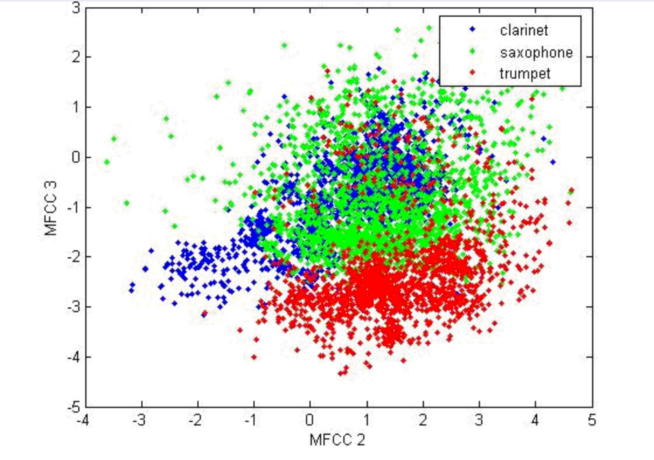
Approche par modèle génératif

Utilisation des distributions « apprises »

- *a priori* (pas d'apprentissage)
  - $\mathbf{x}$ de classe $k \Leftrightarrow k = \text{argmax}_k P(c = k)$
  - ex. : classification d'un mail en spam/non spam en fonction de la proportion de spam uniquement
- Maximum de vraisemblance
  - $\mathbf{x}$ de classe $k \Leftrightarrow k = \text{argmax}_k \hat{p}(\mathbf{x}|c = k) = \text{argmax}_k \hat{p}_k(\mathbf{x})$
- Maximum *a posteriori*, décision Bayésienne
  - $\mathbf{x}$ de classe $k \Leftrightarrow k = \text{argmax}_k P(c = k)\hat{p}(\mathbf{x}|c = k) = \text{argmax}_k P(c = k)\hat{p}_k(\mathbf{x})$
  - préférable aux autres processus de décision

Approche par modèle génératif

Mélange de Gaussiennes



origine : Connexions Ltd

Approche par modèle génératif

Cadre non supervisé

- Pas d'information de classe
- Données de travail : $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_\ell\} \in \mathcal{X}^\ell$
- Hypothèse : éléments $\mathbf{x}_j$ générés suivant une distribution de probabilité, de densité p (e.g., $\mathcal{X} = \mathbb{R}^d$)
- Objectif : déterminer $\hat{p}$ dans une famille $\mathcal{F}$ de distributions qui modélise le processus de génération de $\mathcal{X}$

Approche par méthode discriminante

Contexte

- Classification/catégorisation
- $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\} \in (\mathcal{X} \times \mathcal{Y})^\ell$
- Objectif : déterminer $f \in \mathcal{F}$ qui décrit une « bonne » surface de séparation entre les classes (vs. modélisation de chaque classe)

Naive Bayes : modèle génératif et hypothèse d'indépendance

Contexte

- Classification binaire $\mathcal{Y} = \{0, 1\}$ (se généralise directement à la catégorisation)
- Dictionnaire $\mathcal{D}$, taille d
- $\mathcal{X} = \{0, 1\}^d$ ou $\mathcal{X} = \mathbb{N}^d$

Vraisemblance d'un texte

- Probabilité d'observer un texte **t** représenté par **x** en le supposant de classe y :

$$P(\mathbf{x}|y) = P(x_1, \ldots, x_d|y)$$

Naive Bayes : modèle génératif et hypothèse d'indépendance

Factorisation de la vraisemblance

- Sans aucune hypothèse, factorisation de la vraisemblance :

$$\begin{aligned} P(\mathbf{x}|y) &= P(x_2, \ldots, x_d | x_1, y) P(x_1|y) \\ &= P(x_3, \ldots, x_d | x_1, x_2, y) P(x_2 | x_1, y) P(x_1|y) \\ &= \prod_{i=1}^{d} P(x_i | x_1, \ldots, x_{i-1}, y) \end{aligned}$$

Naive Bayes : modèle génératif et hypothèse d'indépendance

Hypothèse Naive Bayes ! ! !

- Hypothèse très forte (et non vérifiée en générale) : indépendance des attributs étant donné la classe
- Factorisation de la vraisemblance

$$P(\mathbf{x}|y) = \prod_{i=1}^{d} P(x_i|y)$$

Naive Bayes booléen

Représentation booléenne

- $x_i \in \{0, 1\}$ (cf. plus haut)
- On peut poser : $P(x_i = 1|y = k) = \pi_{ik}$ et donc $P(x_i = 0|y = k) = 1 - \pi_{ik}$, modèle de Bernoulli
- Cas binaire : $k = 0$ ou $k = 1$
- Vraisemblance de $\mathbf{x}$: $P(\mathbf{x}|y) = \prod_{i=1}^{d} \pi_{iy}^{x_i}(1 - \pi_{iy})^{1-x_i}$

Apprentissage par maximum de vraisemblance

- $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\}$
- recherche des paramètres $\hat{\Pi} = \{\pi_{10} \ldots \pi_{d0}, \pi_{11} \ldots \pi_{d1}\}$ résolvant

$$\hat{\Pi} = \operatorname*{argmax}_{\Pi} P(\mathbf{x}_1, \ldots, \mathbf{x}_\ell | y_1, \ldots, y_\ell)$$

Naive Bayes booléen

Exercice

En supposant que les instances de $\mathcal{S}$ sont iid, montrer que le maximum de vraisemblance pour chaque π_{ik} est donné par :

$$\frac{M_{ik}}{M_k}$$

où

- M_{ik} : nombre de documents de classe k qui contiennent le mot w_i
- M_k : nombre de documents de classe k

Lissage de Laplace

$$\pi_{ik} = \frac{1 + M_{ik}}{2 + M_k}$$

Naive Bayes booléen

Exercice

En supposant que les instances de $\mathcal{S}$ sont iid, montrer que le maximum de vraisemblance pour chaque π_{ik} est donné par :

$$\frac{M_{ik}}{M_k}$$

où

- M_{ik} : nombre de documents de classe k qui contiennent le mot w_i
- M_k : nombre de documents de classe k

Lissage de Laplace

$$\pi_{ik} = \frac{1 + M_{ik}}{2 + M_k}$$

Naive Bayes booléen

Prédiction de la classe d'un exemple **x**

Dans le cas binaire, en fonction du signe de :

$$\log \frac{P(y=1|\mathbf{x})}{1-P(y=1|\mathbf{x})} = \sum_{i=1}^{d} x_i \log \frac{\pi_{i1}}{1-\pi_{i1}} + \log \frac{\pi_1}{\pi_0}$$

où :

- π_k : proportion de textes de classe k

Naive Bayes booléen

Avantages/inconvénients

- \+ modèle extrêmement simple
- \+ apprentissage rapide
- \+ très bonnes performances en pratique
- \+ peu de paramètres à apprendre
- \- performances moins élevées que d'autres méthodes plus évoluées
- \- difficile de faire de la séparation non linéaire

Naive Bayes multinomial

Représentation fréquentielle simple

- x_i : fréquence du mot w_i dans le document étudié
- Hypothèse : un document **t** peut être vu comme le résultat de l'expérience aléatoire consistant à tirer $|\mathbf{t}|$ mots au hasard dans $\mathcal{D}$ avec remplacement en associant la probabilité $P(x = w_i | y = k) = \pi_{ik}$ à chaque terme w_i
- La vraisemblance d'un document **x** est ainsi

$$P(\mathbf{x}|y = k) = \frac{(\sum_i x_i)!}{x_1! \ldots x_d!} \prod_i {\pi_{ik}}^{x_i}$$

Naive Bayes multinomial

Estimation des paramètres avec lissage de Laplace

$$\pi_{ik} = \frac{1 + M_{ik}}{d + M_k}$$

avec

- M_{ik} : nombre de documents de classe k qui contiennent le mot w_i
- $M_k = \sum_i M_{ik}$

Avantages/inconvénients

Les mêmes que ceux liés à la représentation fréquentielle simple

Bilan Naive Bayes

A retenir

- Modèle très simple
- Hypothèse d'indépendance conditionnelle rarement vérifiée
- Apprentissage par maximum de vraisemblance
- Performances bonnes

Plus loin

- NB existe pour vecteurs réels (e.g. une gaussienne par attribut)
- Principe d'inférence Bayésienne possible à mettre en œuvre (i.e. paramètres sont des v.a.)
- Classifieur disponible dans Weka

Régression logistique : séparation linéaire

Hyperplan séparateur

- Régression logistique bien adaptée pour une représentation vectorielle réelle type TFIDF
- Hypothèse d'un modèle probabiliste pour $P(y = 1|\mathbf{x})$ reposant sur un hyperplan séparateur :

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w} \cdot \mathbf{x} - b)}, \mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}$$

Régression logistique : séparation linéaire

Hyperplan séparateur

- Régression logistique bien adaptée pour une représentation vectorielle réelle type TFIDF
- Hypothèse d'un modèle probabiliste pour $P(y = 1|\mathbf{x})$ reposant sur un hyperplan séparateur :

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w} \cdot \mathbf{x} - b)}, \mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}$$

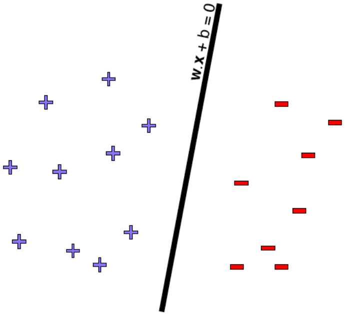
Comportement

$\mathbf{w} \cdot \mathbf{x} + b = 0 \quad P(y = 1|\mathbf{x}) = 0.5$

$\mathbf{w} \cdot \mathbf{x} + b \rightarrow +\infty \quad P(y = 1|\mathbf{x}) \rightarrow 1$

$\mathbf{w} \cdot \mathbf{x} + b \rightarrow -\infty \quad P(y = 1|\mathbf{x}) \rightarrow 0$

Régression logistique : séparation linéaire



Comportement

$\mathbf{w} \cdot \mathbf{x} + b = 0 \quad P(y = 1|\mathbf{x}) = 0.5$

$\mathbf{w} \cdot \mathbf{x} + b \rightarrow +\infty \quad P(y = 1|\mathbf{x}) \rightarrow 1$

$\mathbf{w} \cdot \mathbf{x} + b \rightarrow -\infty \quad P(y = 1|\mathbf{x}) \rightarrow 0$

Régression logistique : séparation linéaire

Autre perspective

Les logits sont décrits par une application linéaire :

$$\log \frac{P(y = 1|\mathbf{x})}{P(y = 0|\mathbf{x})} = \mathbf{w} \cdot \mathbf{x} + b$$

Apprentissage régression logistique

Maximisation de vraisemblance

- Ensemble d'apprentissage : $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\}$ iid
- Vraisemblance $\mathcal{L}$ des données :

$$\begin{aligned}\mathcal{L}(\mathcal{S}; \mathbf{w}; b) &= P(y_1, \ldots, y_\ell | \mathbf{x}_1, \ldots, \mathbf{x}_\ell; \mathbf{w}, b) p(\mathbf{x}_1, \ldots, \mathbf{x}_\ell) \\ &= \prod_{i=1}^{d} P(y_i | \mathbf{x}_i; \mathbf{w}, b) \times cte\end{aligned}$$

- Trouver $\hat{\mathbf{w}}, \hat{b} = \mathrm{argmax}_{\mathbf{w},b}\, \mathcal{L}$

Apprentissage régression logistique

Exercice
Montrer que $\hat{\mathbf{w}}, \hat{b}$ sont solutions de

$$\hat{\mathbf{w}}, \hat{b} = \underset{\mathbf{w},b}{\operatorname{argmax}} \sum_{i=1}^{\ell} \left[y_i \log \left(\frac{1}{1 + \exp(-\mathbf{w} \cdot \mathbf{x}_i - b)} \right) \right.$$
$$\left. + (1 - y_i) \log \left(\frac{1}{1 + \exp(\mathbf{w} \cdot \mathbf{x}_i + b)} \right) \right]$$

Apprentissage régression logistique

Optimisation

- Descente de gradient
- Méthode de Newton-Raphson (information d'ordre 2)
- Minima locaux

Apprentissage régression logistique

Avantages/inconvénients

+ adapté au modèle TFIDF
+ classification linéaire simple
+ interprétation probabiliste
+ apprentissage en ligne possible
+ programmé dans Weka
- minima locaux
- séparation linéaire (peut être arrangée grâce aux noyaux)

K-ppv : Méthode simplisssssssssiiiiiime

Fonctionnement

- $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\}$ iid
- *k* choisi *a priori*
- Classification d'un nouvel exemple $\mathbf{x}$:
  1. trouver les k vecteurs de $\mathcal{S}$ les plus proches de $\mathbf{x}$
  2. affecter à $\mathbf{x}$ la classe majoritaire parmi les k voisins

K-ppv : Méthode simplisssssssssiiiiiime

Choix de la fonction de similarité/distance

- Il en existe des milliers (quelle que soit la représentation)
- Les plus connues
  - produit scalaire standard : $sim(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}' = \sum_{i=1}^{d} x_i x_i'$
  - cosinus : $sim(\mathbf{x}, \mathbf{x}') = \frac{\mathbf{x} \cdot \mathbf{x}'}{||\mathbf{x}|| \, ||\mathbf{x}'||}$
  - coefficient de Dice : $sim(\mathbf{x}, \mathbf{x}') = \frac{2\mathbf{x} \cdot \mathbf{x}'}{||\mathbf{x}||^2 ||\mathbf{x}'||^2}$
  - Tanimoto (repr. booléenne) : $sim(\mathbf{x}, \mathbf{x}') = \frac{|\mathbf{x} \cap \mathbf{x}'|}{|\mathbf{x} \cup \mathbf{x}'|}$

K-ppv : Méthode simplisssssssssiiiiiime

Questions

- Complexité de l'algorithme lorsque $k = 1$?
- Idem lorsque $k > 1$?

K-ppv : Méthode simplisssssssssiiiiiime

Avantages/inconvénients

+ Simplicité
+ Efficacité (performance)
+ Propriétés théoriques de généralisation
+ Implémenté dans Weka
- Temps de calcul si base d'apprentissage très grande
- Pas de représentation compacte du classifieur

Classification multi-classe

Modèles génératifs

- Modèle inchangé
- Apprentissage inchangé
- Processus de décision inchangé

Modèles discriminants

- Pour certains, rien à changer (k-ppv, réseaux de neurones)
- Pour d'autres, découpage en sous-problèmes :
  - 1 vs 1
  - 1 vs all

Outline

Intérêt de la sélection d'attributs

Objectif : vocabulaire pertinent pour la classification/catégorisation

- Beaucoup de termes du dictionnaire ne sont pas informatifs en vue de la classification (même en TFIDF)
- Intérêt des mots facilitant la discrimination/extraction d'information
- Compacité des modèles appris

Approches

- Critère d'information
- Méthodes *ad hoc*

Critère DF (Document frequency)

Fréquences d'apparitions des termes dans la collection

- Idée : les mots qui apparaissent dans très peu (par rapport à un seuil fixé au préalable) de documents de la collection ne sont pas informatifs. On peut donc les supprimer de l'index et avoir un index réduit
- Méthode qui ne prend pas en compte les classes
- Choix du seuil ?

χ^2

	w	$\neg w$
k	A	B
$\neg k$	C	D

Caractéristiques

- Le critère se calcule pour un terme w et une classe k de la manière suivante

$$\chi^2(w,k) = \frac{\ell(AD - BC)^2}{(A+C)(B+D)(A+B)(C+D)}$$

- Pour w seul : $\chi^2(w) = \sum_{k \in \mathcal{Y}} P(k)\chi^2(w,k)$
- Mesure non robuste pour les termes rares

Information mutuelle

Caractéristiques

- Un des critères les plus utilisés pour les langages statistiques
- Le critère se calcule pour un terme w et une classe k de la manière suivante

$$IM(w,k) = \log \frac{P(w,k)}{P(w) \times P(k)}$$

  avec
  - $P(w,k)$: proportion de documents dans la collection étant de classe k et où w est présent
  - $P(w)$: DF de **w**
  - $P(k)$: proportion de documents de classe k
- $IM(w,k) = \log P(k|w) - \log P(k)$
- Pour w seul : $IM(w) = \sum_{k \in \mathcal{Y}} P(k) IM(w,k)$

Gain d'information

Raffinement du critère d'information mutuelle

- Méthode la plus utilisée pour la sélection de termes informatifs
- Le critère se calcule pour un terme *w* de la manière suivante

$$GI(w) = \sum_{x \in \{w, \neg w\}} \sum_{y \in \mathcal{Y}} P(x,y) \log \frac{P(x,y)}{P(x)P(y)}$$

- Lien avec l'information mutuelle

$$GI(w) = \sum_{y \in \mathcal{Y}} P(w,y) IM(w,y) + \sum_{y \in \mathcal{Y}} P(\neg w,y) IM(w,y)$$

Outline

A retenir

Bilan

- Différentes représentations en sac de mots
- Avantages/inconvénients de chaque représentation
- Naive Bayes
- Régressions Logisitique
- k-plus-proches-voisins
- critères de sélection d'attributs

Acknowledgment

Quelques transparents (sur la sélection d'attributs) sont inspirés de ceux de Massih Reza Amini, de l'Université Paris 6.

Bonnes vacances
Joyeuses fêtes