Département d'informatique et de recherche opérationnelle Automne 2006

IFT 6266
Algorithmes d'apprentissage

Yoshua Bengio

Bureau : PAA #3339, courriel: pift6266@iro

Devoir #2

Donné le 21 septembre 2006, Dû le 5 octobre 2006

1. Considérez un probléme de classification à N classes où l'on veut minimiser non pas la probabilité d'erreur mais plutôt un coût spécificié par une matrice C de dimension $N \times N$ avec l'élément C_{ij} indiquant le coût de la décision $f(x) = i$ quand $y = j$, pour un exemple (x, y) et une fonction de décision f. Si on vous fournit un estimateur $P(Y = i|X = x)$ de la probabilité conditionnelle pour les N classes étant donné x, quelle serait la fonction de décision f qui minimise le coût espéré selon ce P?

2. La divergence de Kullback-Liebler $\mathrm{KL}(p||q)$ entre deux densités p et q est

$$\mathrm{KL}(p||q) = \int p(x) \log \frac{p(x)}{q(x)} dx.$$

   Calculer cette divergence quand p et q sont deux normales univariées de paramètres (μ, σ) et (m, s) respectivement. Vérifiez que le résultat est toujours positif sauf quand $m = \mu$ et $s = \sigma$ (truc utile: $\log(x) \leq x - 1$).

3. Vous allez démontrer le résultat de la régression linéaire Bayesienne, c'est à dire avec au départ une distribution à priori sur les paramètres w et à l'arrivée une distribution à postériori sur ces mêmes paramètres, après avoir vu les données $D = \{(x_t, y_t)\}_{t=1}^n$, $x_t \in \mathbb{R}^d$, $y \in \mathbb{R}$. On suppose $Y|X = x$ normal de variance σ^2 (connue pour simplifier) et d'espérance $w'x$. Soit $p(w) = \mathcal{N}(w|m_0, S_0)$ la loi à priori sur w, et notons $\mathbf{X}$ la matrice dont les rangées sont les x_t et $\mathbf{Y}$ le vecteur colonne dont les éléments sont les y_t. Le résultat à démontrer est:

$$p(w|\mathbf{X}, \mathbf{Y}) = \mathcal{N}(w|m_n, S_n)$$

   avec

$$\begin{aligned} m_n &= S_n(S_0^{-1} m_0 + \mathbf{X}'\mathbf{Y}/\sigma^2) \\ S_n^{-1} &= S_0^{-1} + \mathbf{X}'\mathbf{X}/\sigma^2. \end{aligned}$$

Contrairement à la régression linéaire ordinaire, on obtient non seulement une prédiction mais aussi une incertitude sur les prédictions qui sera d'autant plus petite qu'il y a beaucoup d'exemples. Utilisez la formule ci-haut pour obtenir la distribution à postériori sur Y pour un nouvel exemple $X = x$, i.e., donnez une formule pour $p(Y|X = x, \mathbf{X}, \mathbf{Y})$.

4. (BONUS) Vous allez montrer que si l'enveloppe convexe de deux ensembles de points ont une intersection, alors on ne peut les séparer avec un classifieur linéaire. Soit $\{u_1, \ldots, u_n\}$ des exemples de la classe 1 et $\{v_1, \ldots, v_m\}$ des exemples de la classe 2, avec $u_i \in \mathbb{R}^d$, $v_j \in \mathbb{R}^d$. Pour chaque ensemble d'exemples on définit l'enveloppe convexe associée. Par exemple pour les u_i c'est $\{x \in \mathbb{R}^d | x = \sum_i \alpha_i u_i, \alpha_i \geq 0, \sum_i \alpha_i = 1\}$. Vérifier d'abord que si on pouvait trouver un classifieur linéaire qui sépare les u_i des v_j cela voudrait dire qu'il existe $w \in \mathbb{R}^d$ tel que $w'u_i > w'v_j$, $\forall i, j$. Je vous suggère ensuite de procéder en montrant que si on **peut** trouver un tel w alors on **ne peut** avoir une intersection des enveloppes convexes.

5. Vous allez faire une implantation de l'algorithme de régression à noyau aussi appelé fenêtres de Parzen ou encore modèle de Nadaraya-Watson. Pour un ensemble d'apprentissage $D = \{(x_t, y_t)\}_{t=1}^n$ cet estimateur de régression est

$$f_D(x) = \sum_t y_t K_D(x, x_t)$$

avec K_D un *noyau adaptatif* de la forme

$$K_D(x, x') = \frac{K(x, x')}{\sum_t K(x, x_t)}$$

avec $K(x, x')$ ici de forme Gaussienne

$$K(x, x') = e^{-0.5||x-x'||^2/\sigma^2}.$$

Il y a un seul hyper-paramètre σ. Vous allez télécharger l'ensemble d'apprentissage D du site du cours (voir la section des devoirs), ainsi qu'un ensemble de validation V. On utilise D pour former l'estimateur f_D et on calcule l'erreur de cet estimateur sur les exemples de V. Vous allez faire un graphique montrant la relation entre σ et l'erreur quadratique moyenne sur les exemples de validation. Choisissez un intervalle de σ qui met en évidence l'existence d'un minimum (i.e. une courbe grosso-modo en forme de U), et notez ce que cette courbe suggère approximativement comme σ optimal. Faites ensuite un graphique de $f_D(x)$ en fonction de x avec le σ optimal ainsi trouvé (en affichant les x dans l'intervalle couvrant leurs valeurs dans $D \cup V$). Sur le même graphique montrez les points (x_t, y_t) de l'ensemble d'apprentissage D et ceux de l'ensemble de validation V (avec une couleur ou une forme différente). Imprimez votre code et les graphiques pour la remise du devoir.