Mastere M2 MVA 2011/2012 - Modèles graphiques

Exercices à rendre pour le 19 décembre 2012.

Ces exercices doivent être réalisés individuellement.

Si vous souhaitez nous envoyer votre devoir au format pdf, merci de le nommer `MVA_DM3_<votre nom>.pdf`

1 Inférence dans les modèles gaussiens

Soit un modèle graphique Gaussien non-orienté de *paramètres canoniques* $\eta \in \mathbb{R}^n$ (le vecteur potentiel) et $\Lambda \in \mathbb{R}^{n \times n}$ (la matrice de précision). On considère le cas où la variable aléatoire correspondant à chaque noeud est une variable Gaussienne scalaire et où le modèle graphique est un arbre $G = (V, E)$. On s'intéresse au calcul des lois marginales par l'algorithme de propagation de messages (PM). L'algorithme de propagation des messages tel que vu en cours échange des messages qui sont des fonctions-potentiels de $\mathbb{R}$ dans $\mathbb{R}$.

1. Montrer que de façon abstraite la PM consiste à échanger des messages $M_{s \to t}$ qui sont des exponentielles de fonctions quadratiques, et qu'en pratique il suffit de passer d'un noeud s vers un noeud t un message constitué de deux scalaires $\lambda^{s \to t}$ et $\eta^{s \to t}$. Donner la forme de la récurrence permettant de calculer ces messages $\lambda^{s \to t}$ et $\eta^{s \to t}$ au noeud s.
2. Montrer qu'au terme de l'algorithme, le calcul des marginales sur les noeuds et les arêtes permet de calculer $\mu = \mathbb{E}[X]$ et $(\mathbb{E}[X_s X_t])_{\{s,t\} \in E}$.
3. Quelle est la complexité de l'algorithme PM ? Quelle est la complexité en fonction de n du calcul de μ à partir de Λ et η dans le cas général du modèle Gaussien ? Dans le cas où le modèle graphique est un arbre ? Commentez.
4. Soit $A \in \mathbb{R}^{n \times n}$ une matrice définie positive telle que pour tout $s \neq t$ et $(s, t) \notin E$ on a $A_{st} = 0$; soit $b \in \mathbb{R}^n$. Déduire des résultats précédents un algorithme efficace pour résoudre le système linéaire $Ax = b$. Quelle est sa complexité ?

2 Apprentissage de la structure d'un arbre

Dans cet exercice, un algorithme pour apprendre la structure d'un modèle graphique à partir de données i.i.d sera dérivé pour les arbres.

1. Préliminaire I (entropie marginale) : Soit une variable aléatoire discrète X à valeurs dans un ensemble fini $\mathcal{X}$. Soit $\eta(x) = p(X = x)$ le vecteur de paramètres. Soit un échantillon i.i.d (x^n), $n = 1, \ldots, N$ de taille N de cette variable. On note

$\hat{p}(x)$ la densité empirique, définie par $\hat{p}(x) = \frac{1}{N}\sum_{n=1}^{N} \delta(x^n = x)$. Montrer que la log-vraisemblance conditionnelle maximale $\max_\eta \sum_{n=1}^{N} \log p(x^n|\eta)$ est égale à :

$$\max_\eta \sum_{n=1}^{N} \log p(x^n|\eta) = -NH(X),$$

où $H(X)$ est l'entropie empirique de X, définie par $H(X) = -\sum_{x\in\mathcal{X}} \hat{p}(x) \log \hat{p}(x)$.

2. Préliminaire II (entropies jointe et conditionnelle) : Soient deux variables aléatoires discrètes X, Y à valeurs dans des ensembles finis $\mathcal{X}$ et $\mathcal{Y}$. Soit $\eta(x, y) = p(Y = x|X = x)$ la matrice de paramètres de la loi conditionnelle. Soit un échantillon i.i.d (x^n, y^n), $n = 1, \ldots, N$ de taille N de ces deux variables. On note $\hat{p}(x, y)$ la densité empirique, définie par $\hat{p}(x, y) = \frac{1}{N}\sum_{n=1}^{N} \delta(x^n = x)\delta(y^n = y)$. Montrer que la log-vraisemblance conditionnelle maximale $\max_\eta \sum_{n=1}^{N} \log p(y^n|x^n, \eta)$ est égale à :

$$\max_\eta \sum_{n=1}^{N} \log p(y^n|x^n, \eta) = N(H(X) - H(X, Y))$$

où $H(X, Y)$ l'entropie (jointe) empirique de (X, Y) est définie par

$$H(X, Y) = -\sum_{y\in\mathcal{Y}} \sum_{x\in\mathcal{X}} \hat{p}(x, y) \log \hat{p}(x, y).$$

3. On considère maintenant P variables aléatoires $X_1, \ldots, X_P$ à supports finis $\mathcal{X}_1, \ldots, \mathcal{X}_P$. On considère N observations i.i.d. de ces P variables, (x_p^n), $p = 1, \ldots, P$, $n = 1, \ldots, N$. On note $\hat{p}(x_1, \ldots, x_P)$ la densité empirique, définie comme suit :

$$\hat{p}(x_1, \ldots, x_P) = \frac{1}{N}\sum_{n=1}^{N} \delta(x_1^n = x_1) \cdots \delta(x_P^n = x_P).$$

Cette densité jointe permet de définir des densités marginales $\hat{p}(x_p, x_q)$ et $\hat{p}(x_q)$ par marginalisation.

Soit un arbre couvrant orienté T à P sommets (i.e., un DAG connexe avec au plus un parent par sommet). Quelle est la paramétrisation la plus générale pour une loi $p(x_1, \ldots, x_P)$ se factorisant dans un tel DAG ? Montrer qu'une fois maximisée par rapport à ces paramètres, la log-vraisemblance des données est égale à :

$$\ell(T) = N\sum_{p=1}^{P} \left\{H(X_{\pi_p(T)}) - H(X_p, X_{\pi_p(T)})\right\}$$

(avec les conventions $H(X_p, X_\varnothing) = H(X_p)$ et $H(X_\varnothing) = 0$)

4. Pour tout p, q, on appelle information mutuelle empirique la quantité $I(X_p, X_q) = -H(X_p, X_q) + H(X_p) + H(X_q)$. Exprimer cette quantité comme une divergence de Kullback-Leibler et montrer qu'elle est positive ou nulle.
5. Exprimer $\ell(T)$ à l'aide des informations mutuelles. Comment maximiser $\ell(T)$ par rapport à l'arbre T ? Retrouver un problème classique de théorie des graphes, et décrire un algorithme permettant d'apprendre la structure de l'arbre ayant le maximum de vraisemblance. En faisant l'hypothèse que $|\mathcal{X}_1| = \ldots = |\mathcal{X}_p| = K$ quelle est la complexité de l'algorithme en fonction de K et n ?

3 Implémentation - HMM

On considère les mêmes données d'apprentissage que le devoir précédent, dans le fichier "EMGaussienne.dat", mais cette fois-ci en considérant la structure temporelle, i.e., les données sont de la forme $u_t = (x_t, y_t)$ où $u_t = (x_t, y_t) \in \mathbb{R}^2$, pour $t = 1, \ldots, T$. Le but de cet exercice est d'implémenter l'inférence dans les HMM ainsi que l'algorithme EM pour l'apprentissage des paramètres. Il est conseillé d'utiliser le code du devoir précédent.

On considère le modèle HMM suivant avec une chaîne (q_t) à K=4 états et matrice de transition $a \in \mathbb{R}^{4\times4}$, et des "probabilités d'émission" Gaussiennes : $u_t|q_t = i \sim \mathcal{N}(\mu_i, \Sigma_i)$.

1. Implémenter les récursions α et β vues en cours et dans le polycopié pour estimer $p(q_t|u_1, \ldots, u_T)$ et $p(q_t, q_{t+1}|u_1, \ldots, u_T)$.
2. Calculer les équations d'estimation de EM.
3. Implémenter l'algorithme EM pour l'apprentissage (on pourra initialiser les moyennes et les covariances avec celles trouvées dans le devoir précédent).
4. Implémenter l'inférence pour estimer la séquence d'états la plus probables, i.e. $\arg\max_q p(q_1, \ldots, q_T|y_1, \ldots, y_T)$, et représenter le résultat obtenu avec les données (pour le jeu de paramètres appris par EM).
5. Commenter les différents résultats obtenus avec ceux du devoir précédent. En particulier, comparer les log-vraisemblances, sur les données d'apprentissage, ainsi que sur les données de test (dans "EMGaussienne.test").