Examen Final - MAPSI

Durée : 2 heures

Seuls documents autorisés : Calculatrice, antisèche recto-verso, tables de lois.
– Barême indicatif –

Exercice 1 (3 pts) – Indépendence

Soit quatre variables aléatoires X, Y, Z, W, de modalités respectives $\{x_1, x_2\}$, $\{y_1, y_2\}$, $\{z_1, z_2\}$ et $\{w_1, w_2\}$. La probabilité jointe de ces quatre variables est fournie par le tableau suivant :

	y_1				y_2			
	z_1		z_2		z_1		z_1	
	w_1	w_2	w_1	w_2	w_1	w_2	w_1	w_2
x_1	0,048	0,032	0,048	0,032	0,012	0,008	0,012	0,008
x_2	0,192	0,128	0,192	0,128	0,048	0,032	0,048	0,032

Q 1.1 (1 pt) Indépendence conditionnelle Calculer le tableau $P(X, W|Y, Z)$ et justifier le fait que X et W sont indépendantes du couple (Y, Z).

Q 1.2 (1 pt) Indépendence. Déterminer si Z est indépendante du couple (X, Y).

Q 1.3 (1 pt) Indépendence et conjonction Soit trois variables aléatoires X, Y, Z. Montrer que si X est indépendante du couple (Y, Z), et Y est indépendante de Z, alors Z est indépendante du couple (X, Y).

Exercice 2 (4 pts) – À l'attaque !

L'autorité maritime d'un certain pays souhaite évaluer un logiciel qui analyse les données de navigation de vaisseaux (satellitaires, enregistrées dans les ports, etc.) pour détecter des attaques de piraterie. D'après son constructeur, une attaque est détectée dans 90% des cas. Malheureusement, il y a aussi 20% de chances que le logiciel identifie une attaque lorsqu'il n'y en a pas. Pour l'évaluation, l'autorité se concentre sur un trajet en particulier qui, dans la dernière année, a enregistré 200 attaques des pirates sur 4000 passages de vaisseaux. On note L la variable aléatoire pour la prédiction du logiciel et A pour l'attaque.

Q 2.1 (1 pt) Probabilité à posteriori Le logiciel signale une attaque en ce moment. En utilisant le nombre d'attaques observé dans la dernière année pour estimer la probabilité a priori, calculer la probabilité qu'il y ait effectivement une attaque. Écrire la formule correspondant à cette probabilité, puis calculer sa valeur.

Q 2.2 (1.5 pt) Information complémentaire L'autorité a à disposition des outils supplémentaires : depuis quelques années elle a mis en place un réseau d'observateurs permanents (choisi parmi des pécheurs et d'autres navires civils), dotés d'un appareil spécial pour notifier en temps réels l'occurrence de mouvements suspects. Selon une statistique interne, ce réseau a permis de reconnaitre 40%

des attaques en avance. Malheureusement, cette méthode donne aussi 30% de faux positifs (mouvements suspects sans attaque). En supposant les deux notifications (logiciel et réseaux d'observateurs) indépendantes, calculer la probabilité qu'il y ait une attaque lorsque les deux notifications sont actives (utiliser R pour la variable aléatoire de prédiction venant du réseau).

Q 2.3 (1.5 pt) Recalibration ? Après une recherche qualitative, nous nous sommes aperçus que la statistique des attaques utilisée pour estimer la valeur des probabilités a priori était non optimale. En effet, les pirates n'attaquent pas tous les navires, mais principalement ceux qui ont un certain tonnage. Supposons que les navires se distribuent en 3 classes (I, II, III). D'après les statistiques historiques, la probabilité d'attaque en fonction de la classe est la suivante : $P(\text{I}) = 0.1$, $P(\text{II}) = 0.5$ $P(\text{III}) = 0.4$. D'un autre coté, les données détaillées de l'année passée donne :

Classe de navire	I	II	III
Nombre d'attaques	40	120	40

Faire un test d'ajustement avec un niveau de confiance de 90% pour déterminer si les observations correspondent toujours à la distribution historique sur les classes de navires.

Exercice 3 (4 pts) – Rejection sampling

Supposons qu'un phénomène réel peut être modélisé par une variable aléatoire $X \in [0, 1]$ qui suit une loi normale tronquée proportionnelle à $\mathcal{N}(\frac{3}{4}, 1)$. La fonction de densité d'une loi normale tronquée proportionnelle à $\mathcal{N}(\mu, \sigma^2)$ définie sur $[a, b]$ peut être écrite comme :

$$f(x) = \begin{cases} C \cdot \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} & \text{pour } a \leq x \leq b \\ 0 & \text{pour } x < a \text{ et } x > b \end{cases}$$

où C est un facteur de normalisation.

Q 3.1 (1.5 pt) Déterminer la fonction de densité $f(x)$ de X en calculant C.
Note : utiliser la table de la loi normale.

Q 3.2 (1.5 pt) Imaginons que la fonction $f(x)$ est difficile à échantillonner. Nous allons donc utiliser la méthode Monte Carlo appelée *rejection sampling*.
Rappel de cours : Choisir une distribution $q(\cdot)$, facile à échantillonner, telle qu'il existe un facteur k satisfaisant $\forall x, k \cdot q(x) \geq f(x)$. L'algorithme d'échantillonnage est constitué de quatre étapes : (1) tirer un nombre z selon $q(\cdot)$ (*pre-échantillonnage*) ; (2) calculer $m_q = k \cdot q(z)$; (3) tirer un nombre u selon la distribution uniforme sur $[0, m_q]$; (4) accepter z comme échantillon si $u \leq f(x)$.

Calculez le *taux d'acceptation* (la proportion de pre-échantillons acceptés) lorsque $q(\cdot)$ est une loi uniforme sur $[0, 1]$, et $k \cdot q(\cdot) = \max_{x\in[0,1]} f(x)$.

Q 3.3 (1 pt) Supposons que le calcul de $f(x)$ est relativement coûteux. Une méthode pour augmenter l'efficacité (appelée *compression*) est de proposer une fonction $r(x)$ simple—e.g. une droite—qui est une limite inférieure de $f(x)$. L'algorithme modifié prend en compte un *pré-filtrage* des éléments qui sont certainement acceptés, quand $u \leq r(z)$. Calculez le taux des préfiltrage si on prend $r(x) = \min_{x\in[0,1]} f(x)$.

Exercice 4 (4 pts) – **Modélisation markovienne**

Nous souhaitons construire un générateur automatique de morceaux de piano (séquence de notes), en s'inspirant du travail d'un compositeur contemporain n'utilisant que 20 notes. Nous disposons donc de plusieurs morceaux $\mathbf{x} = \{n_1, \ldots, n_T\}, n_t \in \{1, \ldots, 20\}$.

Q 4.1 (0.5 pt) Proposer un système simple capable de générer des notes en s'inspirant des 3 morceaux dont nous disposons $X = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$.
Donner les dimensions des matrices de paramètres et détailler les étapes successives à suivre.

Q 4.2 (1 pt) Le compositeur n'est pas du tout satisfait du résultat. Il vous explique que chacun de ses morceaux est structuré de la manière suivante : introduction (style A), couplet (style B), couplet (style C), conclusion (style A de nouveau). Il y a des variations dans chaque partie, mais globalement, les portions A, B, C sont plutôt homogènes. Toutes les parties ont des longueurs assez proche.
Proposer un système tenant compte de ces contraintes ; pour simplifier l'apprentissage, nous faisons l'hypothèse que les parties sont *exactement* de la même longueur. (Détailler les étapes et citer les algorithmes utilisés et donner une idée de la forme des matrices quand c'est possible).

Q 4.3 (0.5 pt) Quels ajustements sont nécessaires pour tenir compte des variations de longueurs possibles dans les différentes parties ?

Q 4.4 (1 pt) Est-il facile d'insérer un refrain R entre A et B, entre B et C et entre C et A ? Pourquoi ? Donner une idée pour insérer ce refrain.

Q 4.5 (1 pt) Un statisticien pense que les différentes parties ne sont pas très importante ; pour lui, tout se joue sur la mémoire : il pense qu'il faudrait prendre en compte entre les 4 et 6 dernières notes pour prédire efficacement la suivante.
Comment construire et apprendre un tel système ? Cela vous semble-t-il raisonnable avec les 3 morceaux de la base d'apprentissage ?

Exercice 5 (6.5 pts) – **Recommandation par filtrage collaboratif**

Nous voulons construire un système de recommandation de livres ; nous avons des utilisateurs $U = \{\ldots, \mathbf{u}_i, \ldots\}$ et des livres $X = \{\ldots, \mathbf{x}_j, \ldots\}$, nous proposons de modéliser l'intérêt de $\mathbf{u}_i$ pour $\mathbf{x}_j$ avec une variable de Bernoulli Y : $y_{ij} = 1$ si l'utilisateur aime le livre et 0 sinon. Nous utilisons la fonction logistique pour modéliser Y : $p(Y_{ij} = 1|\mathbf{u}_i, \mathbf{x}_j) = f(\mathbf{u}_i, \mathbf{x}_j) = \frac{1}{1+\exp(-\mathbf{u}_i \cdot \mathbf{x}_j)}$, où $\cdot$ désigne le produit scalaire. Dans l'approche que nous envisageons, chaque utilisateur $\mathbf{u}_i$ (respectivement livre $\mathbf{x}_j$) est un vecteur *abstrait* de $\mathbb{R}^d$. Le concepteur du système fixe d et initialise aléatoirement U et X. L'idée est de concevoir un algorithme d'optimisation qui modifient les $\mathbf{u}_i$ et $\mathbf{x}_j$ pour rendre la fonction d'évaluation de l'intérêt pertinente.

Q 5.1 (1 pt) En imaginant que les normes des $\mathbf{x}_j$ et $\mathbf{u}_i$ sont contraintes à une valeur α, dans quels cas particuliers obtenons-nous un interêt minimum/maximum ?

Q 5.2 (1.5 pt) Exprimer la probabilité de la variable de Bernoulli $P(Y = y_{ij}|\mathbf{u}_i, \mathbf{x}_j)$ en fonction de $y_{ij}, \mathbf{u}_i, \mathbf{x}_j$. Puis exprimer la vraisemblance conditionnelle $\mathcal{L}$ d'une base de donnée contenant N couples $\mathbf{u}_i, \mathbf{x}_j$ associés à des valeurs d'intérêt binaires $y_{ij} \in \{0, 1\}$ (comme pour la régression logistique). Vous ferez les hypothèses d'indépendance usuelles.
Montrer que la log-vraisemblance est de la forme $\sum_{couples(i,j)} \beta(\mathbf{u}_i \cdot \mathbf{x}_j) + \gamma \log(f(\mathbf{u}_i, \mathbf{x}_j))$

Q 5.3 (1.5 pt) Afin de simplifier la suite, nous nous intéressons à un seul couple $(\mathbf{u}_i, \mathbf{x}_j)$, ce qui simplifie l'expression de la vraisemblance $\log \mathcal{L}_{ij}$. Donner la dimension de $\nabla_{\mathbf{u}_i} \log \mathcal{L}_{ij}$ puis montrer que le gradient $\nabla_{\mathbf{u}_i} \log \mathcal{L}_{ij}$ est de la forme $\mathbf{x}_j \, (y_{ij} - f(\mathbf{u}_i, \mathbf{x}_j))$.

Q 5.4 (0.5 pt) En déduire $\nabla_{\mathbf{x}_j} \log \mathcal{L}_{ij}$.

Q 5.5 (1.5 pt) Pour un couple ($\mathbf{x}_j$, $\mathbf{u}_i$) correspondant à $y_{ij} = 1$. Rappeler les formules de mise à jour des paramètres pour une montée de gradient. Démontrer que la mise à jour des paramètres par montée de gradient provoque une hausse de la fonction d'intérêt f.

Q 5.6 (0.5 pt) Expliquer le titre de l'exercice (notamment l'aspect collaboratif de la recommandation).