ENS LYON
L3 Probabilités

Probabilités, L3 Informatique Fondamentale

Devoir maison numéro 2
A rendre pour le 19 décembre.

Exercice 1 Le chat et la souris

Un chat et une souris se déplacent aléatoirement et indépendamment sur un graphe connecté, non dirigé et biparti $G := (V, E)$, avec $n := |V|$ et $m := |E|$. Iels commencent à $t = 0$ sur deux sommets différents, et chacun.e fait un déplacement (i.e. emprunte une arête) à chaque étape de temps. Le chat mange la souris s'iels se retrouvent sur le même sommet à un instant t.

Donner une borne sup en $\mathcal{O}(m^2 n)$ sur l'espérance du temps avant que le chat mange la souris.

Indice : On peut considérer une chaîne de Markov dont les états sont les paires (u, v) avec u la position du chat et v la position de la souris.

Exercice 2 Méthode probabiliste

Soit $S \subset \mathbf{N}$. Pour $n \in \mathbf{N}$, on définit $R_S(n)$ comme le nombre de manières d'écrire n comme somme de deux éléments distincts de S :

$$R_S(n) = \text{card}\{(i, j) \in S^2 \,:\, i < j,\ i + j = n\}.$$

Le but de cet exercice est de montrer l'existence d'un ensemble $S \subset \mathbf{N}$ et de constantes strictement positives c_1, c_2 tels que, pour tout $n \geqslant 2$,

$$c_1 \log(n) \leqslant R_S(n) \leqslant c_2 \log(n). \tag{1}$$

Pour cela on définit un sous-ensemble $S \subset \mathbf{N}$ aléatoire en décrétant que les événements $\{n \in S\}_{n \in \mathbf{N}}$ sont indépendants et vérifient $\mathbf{P}(n \in S) = \begin{cases} 10\sqrt{\log(n)/n} & \text{si } n \geqslant 1000 \\ 1 & \text{si } n < 1000. \end{cases}$

1. On note $\mu_n = \mathbf{E}[R_S(n)]$. Vérifier que μ_n est équivalent à $50\pi \log n$ quand n tend vers l'infini. On pourra utiliser sans justification la formule

$$\lim_{n \to \infty} \sum_{t=1}^{n-1} \sqrt{\frac{\log(t)\log(n-t)}{t(n-t)}} = \pi \log n$$

2. Soit A_n l'événement $\{\frac{\mu_n}{2} \leqslant R_S(n) \leqslant \frac{3\mu_n}{2}\}$. Montrer que $\sum \mathbf{P}(A_n^c) < +\infty$.
3. En déduire l'existence d'un ensemble $S \subset \mathbf{N}$ vérifiant (1).

Exercice 3 Apprentissage PAC

Dans ce problème, nous étudierons un modèle pour l'apprentissage efficace d'un concept en observant des exemples aléatoires. Intuitivement, l'objectif est d'apprendre une fonction $f : \mathcal{X} \to \{0, 1\}$ étant

donné des exemples de la forme $(x_1, f(x_1)), \ldots, (x_m, f(x_m))$. Nous pouvons avoir par exemple $\mathcal{X} = \mathbb{R}$ et $f(x) = \mathbf{1}_{x \geqslant a}$, où $\mathbf{1}_{x \geqslant a} = 1$ si $x \geqslant a$ et 0 sinon, pour une valeur (inconnue) de $a \in \mathbb{R}$.

Dans un exemple du monde réel, vous pouvez penser à $\mathcal{X}$ comme un ensemble d'images de chats ou de chiens et $f(x) = 0$ si x représente un chat et $f(x) = 1$ si x représente un chien.

Pour modéliser cette situation, une *classe de concepts* $\mathcal{C}$ est un ensemble de fonctions. Pour l'instance décrite ci-dessus, la classe de concept est donnée par

$$\mathcal{C}_{\text{half-line}} = \{f : \mathbb{R} \to \{0, 1\} : f(x) = \mathbf{1}_{x \geqslant a} \text{ pour un } a \in \mathbb{R}\}.$$

Quand disons-nous qu'une classe de concepts $\mathcal{C}$ est apprenable ? Intuitivement, nous voulons qu'un algorithme $\mathcal{C}$ puisse déterminer à partir d'un nombre raisonnable d'exemples $(x_i, f(x_i))$ la bonne fonction f parmi toutes les fonctions possibles dans $\mathcal{C}$. Cependant, apprendre la fonction exacte pourrait être impossible, donc notre objectif sera seulement de produire une fonction qui se rapproche de f. De plus, nous exigerons uniquement que la sortie de l'algorithme soit correcte avec grande probabilité. Ce modèle s'appelle le modèle Probablement Approximativement Correct (ou PAC).

Plus précisement, nous disons qu'une classe de concepts $\mathcal{C}$ est PAC-apprenable s'il existe un algorithme $\mathcal{C}$ et un polynôme $p(s, t)$ en deux variables tels que pour $f \in \mathcal{C}$, toute mesure de probabilité $\mathcal{D}$ sur $\mathcal{X}$, tout $\epsilon \in]0, 1/2[$, tout $\delta \in]0, 1[$, l'algorithme $\mathcal{C}$ prend en entrée $m = p(\frac{1}{\epsilon}, \frac{1}{\delta})$ échantillons indépendants $x_1, \ldots, x_m$ selon $\mathcal{D}$, ainsi que les évaluations de f à ces points $f(x_1), \ldots, f(x_m)$ et il produit $\tilde{f}$ (qui est aléatoire car elle dépend de $x_1, \ldots, x_m$). La sortie $\tilde{f}$ doit satisfaire

$$\mathbb{P}_{x_1, \ldots, x_m \sim \mathcal{D}^{\times m}} \left(\text{Err}_{\mathcal{D}}(\tilde{f}, f) \leqslant \epsilon \right) \geqslant 1 - \delta$$

où $\text{Err}_{\mathcal{D}}(\tilde{f}, f) = \mathbb{P}_{x \sim \mathcal{D}} \left(f(x) \neq \tilde{f}(x) \right)$.

1. Nous montrons maintenant que la classe de concept $\mathcal{C}_{\text{half-line}} = \{f : f(x) = \mathbf{1}_{x \geqslant a} \text{ pour un } a \in \mathbb{R}\}$ est PAC-apprenable. Pour cela, nous fixons un $f \in \mathcal{C}_{\text{half-line}}$ comme $f(x) = \mathbf{1}_{x \geqslant a}$, une mesure de probabilité $\mathcal{D}$ sur $\mathbb{R}$, et ϵ et δ.
   (a) Considérons l'algorithme simple suivant qui prend m examples de la form $(x_i, f(x_i))$ et émet la fonction $\tilde{f}$ définie par $\tilde{f}(x) = \mathbf{1}_{x \geqslant \hat{a}}$ où $\hat{a} = \min\{x_i : i \in \{1, \ldots, m\}, f(x_i) = 1\}$. Prouver que $\text{Err}_{\mathcal{D}}(\tilde{f}, f) = \mathcal{D}([a, \hat{a}[)$.
   (b) Définir $a_\epsilon = \sup\{a' : \mathcal{D}([a, a'[) \leqslant \epsilon\}$. Montrer que $\mathcal{D}([a, a_\epsilon[) \leqslant \epsilon$ et $\mathcal{D}([a, a_\epsilon]) \geqslant \epsilon$. *Note :* Des points seront accordés si la question est traitée dans un cas spécial, par exemple, $\mathcal{D}$ fini.
   (c) Trouver une borne supérieure sur la probabilité $\mathbb{P}(\hat{a} > a_\epsilon)$ (ici la probabilité est sur le choix de $x_1, \ldots, x_m$).
   (d) Conclure sur la PAC-apprenabilité de la classe de concepts $\mathcal{C}_{\text{half-line}}$. Quelle valeur choisir pour m en fonction de ϵ et δ ?