Structure et algorithmes aléatoires

Introduction

Probabilités discrètes : espaces au plus dénombrables, et application à différents domaines. Complémentaire au cours d'intégration et probabilités

Références :

- Mitzenmacher & Upfal : *Probability and Computing : Randominzed Algorithms and Probabilitic Algorithms*
- Pierre Brémaud : *Discrete Probabilitiy Models and Methods*

Programme :

- Probabilités discrètes et applications
  - Rappels : variables aléatoires, indépendance, espérance et variance, quelques inégalités, fonctions génératrices, boules et urnes
  - Algorithmes aléatoires
  - Méthode probabiliste
  - Graphes aléatoires
- Modèles markoviens
  - Chaînes de Markov, comportement asymptotique
  - Simulation de Monte Carlo et simulation parfaite
  - Extensions et applications : modèles markoviens cachés, modèles markoviens de décision, champs de Gibbs, automates cellulaires probabilistes

Algorithme probabiliste

Algorithme déterministe est tel que pour chaque entrée, il existe une et une seule valeur de sortie.

On veut une réponse rapide et correcte, ce que l'on ne sait pas toujours faire avec un algorithme de déterministe.

On peut ajouter de l'aléa sous forme de **bits aléatoires** : plus une unique sortie pour chaque entrée. La sortie esrt une variable aléatoire.

On peut modifier l'algorithme de sorte à avoir

- Soit une réponse correcte dans tous les cas et rapide dans la plupart des cas.
- Soit une réponse correcte dans la plupart des cas, mais rapide dans tous les cas.

Exemple :

Problème : deux polynômes de degré d sont-ils égaux ?

Un algo naïf nécessite $\mathcal{O}(d^2)$ opérations pour développer.

Algo probabliliste

Choisir r dans $\{1, ..., 100d\}$ uniformément

Calculer $F(r)$ et $G(r)$ (en $\mathcal{O}(d)$). Si $F(r) = G(r)$ alors $F = G$, sinon $F \neq G$.

On se trompe uniquement si $F \neq G$ et r est une racine de $F - G$.

Au plus d racines dans $\{1, ..., 100d\}$ => probabilité que l'algorithme se trompe est au plus de $1/100$.

Attention, pour plus de précision on pourrait vouloir exécuter plusieurs fois l'algo, mais il ne faut alors pas dépasser la complexité de l'algorithme déterministe pour que cela soit intéressant.

Modèle : une file d'attente

X_n clients après le départ du n-ième paquet. $X_0 = 0$. A_n : nombre de clients qui arrivent pendant le service du n-ième client. Alors : $X_{n+1} = \max(X_n - 1, 0) + A_n$

(X_n) forme un processus stochastique si A_n est décrit de manière probabiliste. Sous certaines conditions, (X_n) est ce qu'on appelle une chaîne de Markov, qui fait l'objet de la seconde partie du cours.

Chaîne de Markov

Soit $\{X_n, n \in \mathbb{N}\}$ une suite de variables aléatoires (un processus stochastique) à valeurs dans une espace d'états $\mathcal{E}$ au plus dénombrable.

Le processus $\{X_n, n \in \mathbb{N}\}$ est appelé une chaîne de Markov à temps discret sur $\mathcal{E}$ si pour tout $n \in \mathbb{N}$ pour tous $i, j, i_1, ..., i_{n-1} \in E$,

$P(X_{n+1} = j | X_n = i, ..., X_0 = i_0) = P(X_{n+1} = j | X_n = i)$

Questions :

- comportement asymptotique (stabilité) ?
- Quelle est la taille moyenne de la file d'attente ?
- Et si le nombre de clients double, que faut-il faire pour garder le même temps moyen d'attente qu'avant ?
- Si on a deux files d'attentes, faut-il avoir des salles d'attentes séparées ou une salle commune ?

Plan

- Introduction : Deux exemples d'aplication
- Événements et probabilités
  - Tribus et événements
  - Espaces de probabilités
  - Indépendance, probabilité conditionnelle

- Variables aléatoires
  - Variables aléartoires et discributions
  - ...

Tribus et événements

Univers : Ω - un ensemble qui décrit toutes les possibilités d'une expérience. Par eemple, pour un dé, on a $\Omega = \{1, 2, 3, 4, 5, 6\}$. Les éléments de Ω sont appelés les élévements élémentaires (les éventualités)

Définition : une tribue sur Ω est une famille $\mathcal{F}$ de sous ensemble de Ω qui contiennent $\emptyset$, Ω, est stable par passage au complémentaire, par réunion dénombrable (intersection dénombrable).

La **tribu grossière** est la plus petite tribu sur Ω.

La **tribu fine** est la plus grosse tribu sur Ω.

Exemples d'événements plus complexes (espaces non dénombrables) :

- $\Omega = \{0, 1\}^{\mathbb{N}}$: une suite infinie de lancers de pièce.
- A=l'événement concerne uniquement les k premiers lancers.
- Les événements de type A ne forment pas une tribu, on considère donc la tribu engendrée (la plus petite tribu contenant les événements de type A). (Ici il s'agit de la tribu engendrée par les cylindres finis)

Espace de probabilités

Définition : une probabilité sur une espace probabilisable (Ω, $\mathcal{F}$) est une application $P : F- > [0, 1]$ telle que $P(\Omega) = 1$, et P est σ-additive.

Propriétés :

- Passage au complémentaire
- Monotonie
- Inégalité de Boole - Union bound $P(\bigcup_i A_i) \leq \sum_i P(A_i)$
- Continuité séquentielle : Si (A_n) est croissante, $P(\bigcup_n A_n) = \lim_n P(A_n)$
- Idem pour les intersections décroissantes

Exemple (retour sur les polynômes) : Augmenter la précision, comment

- Augmenter l'espace. Problème : précision sur les grands entiers
- Répéter l'algorithme plusieurs fois
  - On peut choisir avec ou sans remplacement des valeurs déjà tirées. Dans le premier cas, les tirages sont indépendants.

Indépendance

Définition : deux événements A et B sont indépendants si $P(A \cap B) = P(A)P(B)$. Une famille d'événements est mutuellement indépendants si

(. . . todo. . .)

Probabilité conditionnelle

Définition : Probabilité d'un événement A conditionné par B est : $P(A|B) = \frac{P(A \cap B)}{P(B)}$ (uniquement définie si $P(B) > 0$)

Théorème : Formule des probabiltiés composées : $P(E_1 \cap ... \cap E_n) = P(E_1)P(E_2|E_1)...P(E_n|E_{n-1}....E_1)$

Exemple : Tirage des racines sans remise : un calcul montre que c'est une erreur plus petite qu'avec remplacement.

En pratique il est parfois plus judicieux d'implémenter la version avec remplacement.

Théorème : loi des probabiltiés totales : $P(A) = \sum_i P(A|E_i)P(E_i)$ pour $\{E_i\}$ partition de Ω

Théorème : loi de Bayes : $P(A|B) = \frac{P(A)}{P(B)} * P(B|A)$

Exercice : vérification d'une multiplication matricielle : Meilleur qu'un algo naïf ($\mathcal{O}(n^3)$) : algo probabiliste. Soit $r \in \{0,1\}^n$

Complexité de $ABr = Cr$: $\mathcal{O}(n^2)$ (cf diapo) Proba d'erreur : $\leq 1/2$ (cf diapo)

Variables aléatoires

Définition : $(\Omega, \mathcal{F}, P)$ espace de probabilité. E un ensemble au plus dénombrable. Une fonction $X : \Omega -> E$ telle que pour tout $x \in E$, $\omega | X(\omega) = x \ \in \mathcal{F}$ est une variable aléatoire discète sur E. (techniquement il faut aussi la condition de mesurabilité de X)

Propriété : toute image d'un n-uplet de v.a. est une v.a.

Exemples de distributions :

- Loi constante
- Loi de Bernouilli
- Loi binomiale
- Loi géométrique (sans mémoire)
- Loi de Poisson
- . . .

Espérance (cf diapo)

Propriété : linéarité

Propriété : monotonie si $f(X) < g(X)$ p.s. alors $E(f(X)) < E(g(X))$

Propriété : indépendance : $\mathrm{E}(XY) = \mathrm{E}(X)\mathrm{E}(Y)$ si X et Y sont indépendantes

Variance

Propriété : $\mathrm{Var}(X,Y) = \mathrm{Var}(X) + \mathrm{Var}(Y) + 2\mathrm{Cov}(X,Y)$

Propriété : Si X et Y sont indépendantes, $\mathrm{Cov}(X,Y) = 0$

Propriété : Si X et Y sont indépendantes, $\mathrm{Var}(X+Y) = \mathrm{Var}(X) + \mathrm{Var}(Y)$

Inégalités

Inégalité de Jensen : Si ϕ est une fonction convexe, X et $\phi(X)$ ont une espérance, alors $\mathrm{E}(\phi(X)) \geq \phi(E(X))$

Inégalité de Markov : X v.a. ≥ 0. Alors $P(X \geq a) \leq \mathrm{E}(X)/a$

Exemple : lancer de pièces non biaisées (n fois). Borner la proba d'obtenir au moins $3n/4$ fois face. Markov donne $P(X \geq 3n/4) \leq 2/3$.

Inégalité de Tchebychev : $P(|X - \mathrm{E}(X)| \geq a) \leq \frac{\mathrm{Var}(X)}{a^2}$ Sur le lancer de n pièces, $P(X \geq 3n/4) \leq 4/n$

Exemple : collectionneur de coupons Chaque boîte contient un et un seul coupon, tiré de manière uniforme parmi toutes les possibilités et indépendemment des autres boîtes. Il y a n coupons au total. Combien de boîtes faut-il acheter en moyenne pour obtenir tous les coupons ? Soit X le nombre de boîtes pour obtenir n coupons. On cherche $\mathrm{E}(X)$. Soit X_i le nombre de boîtes ouvertes depuis que le collectionneur a obtenu $i-1$ coupons jusqu'à l'obtention du i-ème coupon. On a $X = X_1 + ... + X_i$ Distribution des X_i : géométrique de paramètre $\frac{n-(i-1)}{n}$. $\mathrm{E}(X_i) = \frac{n}{n-i+1}$ Donc $\mathrm{E}(X) = nH(n)$ avec $H(n)$ somem harmonique.

Question : quelle est la probabilité que le temps pour collectionner les n coupons soit au moins le double de cette espérance ? Markov : 1/2 Tchebychev : indépendance donc variance linéaire. Tchebychev donne : Indégalité de Boole (union bound) donne $1/n$

Classification des algorithmes probabilites

Algorithmes de Monte Carlo : complexité déterministe mais peut se tromper avec erreur contrôlée. **Erreur unilatérale :** renvoie `vrai` ou `faux` mais ne se trompe que sur une des valeurs **Erreur bilatérale :** peut se tromper sur les deux valeurs Si la proba d'erreur est $< 1/2$, alors on peut diminuer cette proba en exécutant plusieurs fois l'algorithme et en renvoyant la réponse majoritaire.

Algorithmes de Las Vegas : résout un problème exactement, avec une complexité moyenne finie (que l'on cherche à minimiser) **Exemple :** choix uniforme du pivot dans l'algorithme du tri rapide

Un algorithme Monte Carlo de complexité C_{MC} qui renvoie soit la réponse correct, soit `erreur` (avec proba au plus p), on peut transformer cet algo en un algorithme de type Las Vega en le répétant tant que la réponse renvoyée est `erreur`. Alors la complexité de l'algorithme Las Vegas est alors $\mathrm{E}(C_{LV}) \leq C_{MC}/p$.

Terminaison : 3 cas : - Un algo termine avec proba α - Un algorithme termine presque sûrement - Un algorithme termine sûrement