Emmanuel Jeandel

MÉTHODES PROBABILISTES EN INFORMATIQUE

TABLE DES MATIÈRES

INTRODUCTION

Voici une liste de livres dont je m'inspire pour le cours
– Johnson, *Probability and Statistics for Computer Science*
– Ross, *Probability Models for Computer Science*
– Olofsson, *Probability, Statistics, and Stochastic Processes*
– Trivedi, *Probability and Statistics with Reliability, Queuing and Computer Science Applications*
– Mari, Schott *Probabilistic and Statistical Methods in Computer Science*
Les deux premiers sont conseillés pour le cours.

Chapitre 1

ESPÉRANCE

1.1 Espérance

Définition 1.1 (*Espérance (Moyenne)*)

Si X est une variable discrète, on note

$$E(X) = \sum_i iP(X = i)$$

Si X est une variable continue de densité f, on note

$$E(X) = \int tf(t)dt$$

♦ **Exemple**

Soit X la variable aléatoire correspondant au résultat du lancer d'un dé. Alors

$$E(X) = \frac{1+2+3+4+5+6}{6} = 3.5$$

Notons que si X est une variable aléatoire, alors $Y = F(X)$ est aussi une variable aléatoire et :

$$E(Y) = E(F(X)) = \sum_i f(i)P(X = i)$$

En effet

$$\begin{aligned} E(Y) &= \sum_i iP(Y = i) = \sum_i \sum_j iP(Y = i|X = j)P(X = j) \\ &= \sum_j \sum_i iP(F(X) = i|X = j)P(X = j) \\ &= \sum_j f(j)P(X = j) \end{aligned}$$

Définition 1.2

$$Var(X) = E(X^2) - E(X)^2$$

L'écart type est $\sqrt{Var(X)}$.

Proposition 1.1

$$E(aX + b) = aE(X) + b$$

$$E(X + Y) = E(X) + E(Y)$$

Sans hypothèse sur X et Y.

$$E(XY) = E(X)E(Y)$$

si X et Y sont indépendantes.

Définition 1.3

X et Y sont indépendantes si

$$P(X = i)P(Y = j) = P(X = i \wedge Y = j)$$

Proposition 1.2

$$Var(aX + b) = a^2 Var(X)$$

$$Var(X + Y) = Var(X) + Var(Y)$$

si X et Y sont indépendantes.

♦ **Exemple**

X est une variable de Bernoulli $P(X = 0) = 1 - p$, $P(X = 1) = p$.
Alors $E(X) = 0 * (1 - p) + 1 * p = p$ et $Var(X) = p - p^2 = p(1 - p)$

A retenir — *Loi de Bernoulli*

Si X suit une loi de Bernoulli de paramètre p, alors

$$E(X) = p, Var(X) = p(1 - p)$$

♦ **Exemple**

Y est une variable binomiale $B(n, p)$.

Théorème 1.3

Si $X_1 \dots X_n$ sont des variables indépendantes de Bernoulli de paramètre p alors $Y = X_1 \dots X_n$ suit une loi binomiale de paramètre $B(n, p)$.

On en déduit $E(Y) = np$ et $Var(Y) = np(1-p)$.

A retenir — *Loi Binomiale*

Si X suit une loi Binomiale de paramètre (n, p), alors

$$E(X) = np, Var(X) = np(1-p)$$

♦ **Exemple**

X est de Poisson $P(X = k) = e^{-\lambda}\frac{\lambda^k}{k!}$.

$$E(X) = \sum ke^{-\lambda}\frac{\lambda^k}{k!} = \lambda\sum e^{-\lambda}\frac{\lambda^{k-1}}{(k-1)!} = \lambda$$

$$Var(X) = \lambda$$

A retenir — *Loi de Poisson*

Si X suit une loi de Poisson de paramètre λ, alors

$$E(X) = \lambda, Var(X) = \lambda$$

Pour l'exponentielle, on va utiliser une astuce.

Proposition 1.4

Soit X une variable discrète à valeurs dans $\mathbb{N}$. Alors

$$E(X) = \sum_i P(X \geq i)$$

De même pour une variable continue de support $\mathbb{R}$.

$$E(X) = \int P(X \geq t)dt$$

Preuve : On fait la preuve dans le cas discret. Le cas continu s'obtient de la même façon.

$$\sum_i iP(X = i) = \sum_i \sum_{j \leq i} P(X = i) = \sum_j \sum_{i \geq j} P(X = i) = \sum_j P(X \geq j)$$

■

♦ **Exemple**

X variable exponentielle. $P(X \leq t) = 1 - e^{-\lambda t}$. Donc $P(X \geq t) = e^{-\lambda t}$ et $E(X) = \int_0^\infty e^{-\lambda t} = \frac{1}{\lambda}$.

A retenir — *Loi exponentielle*

Si X suit une loi exponentielle de paramètre λ, alors

$$E(X) = \frac{1}{\lambda}, Var(X) = \frac{1}{\lambda^2}$$

♦ **Exemple**

X variable géométrique. $P(X = k) = (1-p)^{k-1}p$ et $P(X \geq k) = (1-p)^{k-1}$

D'où

$$E(X) = \sum (1-p)^{k-1} = \frac{1}{1-(1-p)} = \frac{1}{p}$$

Pour la variance, il faut faire le calcul, et on trouve...

A retenir — *Loi géométrique*

Si X suit une loi géométrique de paramètre p, alors

$$E(X) = \frac{1}{p}, Var(X) = \frac{1-p}{p^2}$$

♦ **Exemple**

X de loi uniforme $P[X \leq t] = t$.

Alors $E(X) = \int_0^1 P(X \geq t)dt = \int_0^1 (1-t)dt = \left[t - \frac{t^2}{2}\right]_0^1 = 1/2$

De même

$E(X^2) = \int_0^1 P(X^2 \geq t)dt = \int_0^1 (1-\sqrt{t})dt = \left[t - \frac{2}{3}t\sqrt{t}\right]_0^1 = 1/3$

D'où $Var(X) = E(X^2) - E(X)^2 = 1/3 - 1/4 = 1/12$.

A retenir — *Loi uniforme*

Si X suit une loi uniforme alors

$$E(X) = 1/2, Var(X) = 1/12$$

1.2 Temps d'arrêt

Exemple : On répète une expérience jusqu'à avoir un résultat donné.

Définition 1.4

Soit $X_1 \dots X_n \dots$ des variables aléatoires. Un temps d'arrêt pour X_i est une variable aléatoire N à valeurs dans $\mathbb{N}$ tel que l'évènement $N = i$ est indépendant de $X_{i+1}, X_{i+2} \dots$

♦ **Exemple**

Je lance un dé jusqu'à avoir 6, et je somme les résultats

$$Y = X_1 + X_2 + \cdots + X_N$$

♦ **Exemple**

Je pars avec un capital de 5 euros et je joue sur une machine à sous. Je gagne 1 euro avec probabilité 1/2 et je perds 1 euro avec probabilité 1/2. Je m'arrête quand j'ai gagné 15 euros, ou quand je n'ai plus d'argent.

Théorème 1.5 (*Egalité de Wald*)

Si X_i sont des variables aléatoires indépendantes de même espérance, et si N est un temps d'arrêt pour les X_i, alors

$$E(X_1 + X_2 \cdots + X_N) = E(N)E(X_1)$$

Note : la formule n'est valide que si $E(X_1)$ et $E(N)$ ne sont pas infinis.

Preuve : Soit $Y = X_1 + X_2 + \dots X_N$.

Soit Z_n la variable aléatoire qui vaut 1 si $n \leq N$ et 0 sinon. Alors on peut écrire $Y = X_1Z_1 + X_2Z_2 + \dots$. Plus exactement, $E(Y) = \sum_i E(X_iZ_i)$.

Remarquons ensuite que X_i est indépendant de Z_i. D'où

$$E(Y) = \sum_i E(X_i)E(Z_i) = E(X)\sum_i E(Z_i) = E(X)\sum_i P(N \geq n) = E(X)E(N)$$

■

♦ **Exemple**

Sur le lancé de dé. N suit une loi géométrique de paramètre $1/6$: $P(N = i) = 1/6(1-1/6)^{i-1}$ d'espérance 6.

Donc le résultat moyen est $E(Y) = 6 * 3.5 = 21$.

♦ **Exemple**

Dans ce deuxième exemple $E(X_1) = 0$, d'où $E(Y) = 5$.

Mais $E(Y) = 15 * P(Y = 15)$. D'où $P(Y = 15) = 5/15$.

1.3 Inégalités

Le calcul de l'espérance ou de la variance nous dit quelque chose sur les probabilités

Proposition 1.6

$P(X \geq E(X)) > 0$

Proposition 1.7 (*Markov*)

$P(X \geq p)) \leq \frac{E(X)}{p}$

Dit autrement

$$P(X \geq \lambda E(X)) \leq \frac{1}{\lambda}$$

Proposition 1.8 (*Chebyshev*)

$$P(|X - E(X)| \geq p) \leq \frac{Var(X)}{p^2}$$

Il suffit d'appliquer Markov à $Y = (X - E(X))^2$

On s'intéresse maintenant aux sommes de Bernoulli indépendantes de paramètre p, $S_n = X_1 + \ldots X_n$. Evidemment $E(S_n) = nE(X_1) = np$.

Théorème 1.9 (*Chernoff*)

$$P(|S_n - np| \geq n\lambda) \leq 2e^{-2n\lambda^2}$$

On ne prouvera pas ce théorème.

On a aussi des théorèmes limites :

Théorème 1.10 (*Loi faible des grands nombres*)

Soit X_i des variables aléatoires indépendantes identiquement distribuées, et $E(X_1) \neq \infty$. Alors

$$\frac{S_n}{n} \longrightarrow E(X_1)$$

Plus précisément, pour tout ϵ

$$P\left(\left|\frac{S_n}{n} - E(X_1)\right| \geq \epsilon\right) \longrightarrow_n 0$$

Théorème 1.11 (*Loi forte*)

$$P\left(\frac{S_n}{n} \longrightarrow_n E(X_1)\right) = 1$$

Preuve : On prouve aussi la loi faible sous l'hypothèse supplémentaire que la variance $Var(X_1)$ est finie.

Notons que $Var(S_n) = nVar(X_1)$ et $Var(\frac{S_n}{n}) = \frac{Var(X_1)}{n}$
Et donc Chebyshev appliqué à $\frac{S_n}{n}$ donne :

$$P(|\frac{S_n}{n} - E(X_1)| \geq p) \leq \frac{Var(X_1)}{np^2}$$

qui donne le résultat ■

Exercices

(1 - 1) *(RAID)*

La durée de vie d'un disque dur est souvent modélisée par une variable aléatoire exponentielle.

Q 1) Expliquez ce choix.

La durée de vie moyenne est souvent appelée dans ce cadre Mean Time to Failure (MTTF).

On se donne deux disques durs de capacité 1To, de MTTF respectivement α et β, modélisés par deux variables aléatoires exponentielles X et Y.

On assimile ces deux disques dur à un seul disque dur de capacité 2To. On note Z la variable aléatoire correspondant à la durée de vie de ce nouveau système.

Q 2) Exprimez Z en fonction de X et Y

Q 3) Calculer la loi de Z (Aide : Calculer $P(Z > t)$) et en déduire la durée de vie moyenne de ce système.

On simplifie maintenant l'énoncé et on suppose que les deux disques ont la même MTTF qu'on note α.

Pour accroître la durée de vie du système, on décide maintenant de mettre la même chose sur les deux disques durs : Si l'un des disques meurt, on le remplace et on recopie les données. On suppose que cette opération prend un temps t'.

Q 4) On suppose que le premier disque meurt à l'instant t. Quelle est la probabilité que le second meurt pendant qu'on répare le premier ? Simplifier en supposant que $t' \ll \alpha$.

On cherche maintenant à calculer la durée de vie moyenne du système.

Q 5) Expliquer pourquoi on peut modéliser la situation par

$$X_1 + X_2 + \dots X_N$$

où X_i sont des variables aléatoires exponentielles indépendantes de paramètre $2/\alpha$ et N une variable aléatoire géométrique de paramètre t'/α.

Q 6) En utilisant l'équation de Wald, en déduire la durée de vie moyenne de ce système. Application numérique avec$\alpha = 365$ et $t' = 1$.

Source : Patterson, Gibson, Katz. *A case for Redundant Arrays of Inexpensive Disks (RAID)*

(1 - 2) *(Tableaux)*

On se donne un ensemble de n objets x_i, réparti dans un tableau T de n cases. On suppose que la seule façon de retrouver un objet dans le tableau est de partir du début et de parcourir le tableau jusqu'à trouver l'objet.

Q 1) En supposant que l'objet i va être demandé avec probabilité p_i, dans quel ordre doit on ranger les éléments dans le tableau pour minimiser le temps moyen de recherche ?

En pratique, on ne connaît pas les probabilités p_i. Une heuristique fréquemment utilisée est appelée Move-to-Front (MTF) : Lorsqu'on recherche l'élément x, on le replace en tête du tableau (en décalant les éléments). On suppose avoir déjà effectué beaucoup de recherches, et on cherche à estimer le temps que va prendre la recherche suivante.

Q 2) Exprimez en fonction des paramètres $p_1 \dots p_n$ la probabilité que l'objet x_i soit placé avant l'objet x_j dans le tableau.

Q 3) En déduire la position moyenne de l'objet x_i.

Q 4) En déduire le temps moyen de la recherche. Comparer avec le précedent.

On peut démontrer qu'on peut faire mieux, en utilisant une autre heuristique : ne remonter l'élément que d'une position. Mais l'analyse probabiliste est bien plus ardue.

Source : R. Rivest. *On Self-Organizing Sequential Search Heuristics*.

(1 - 3) *(QuickSelect)*

On se donne un tableau de n entiers. Le but est d'obtenir un algorithme qui trouve le $k$ème plus petit.

Q 1) Donner un algorithme de complexité $O(n^2)$ qui répond à la question.

On peut prouver qu'il existe un algorithme de complexité linéaire $O(n)$ qui répond à cette question, mais l'algorithme est relativement complexe.

On donne ici un algorithme probabiliste. L'algorithme fonctionne ainsi : On choisit une case du tableau au hasard, qui contient un nombre x. On regroupe les éléments plus petits que x dans un tableau T_1 et les éléments plus grands que x dans un tableau T_2. Soit n_1 le nombre d'éléments de T_1. Si $n_1 \geq k$, alors on appelle récursivement la procédure sur T_1 avec paramètre k. Si $n_1 = k - 1$, on répond x. Sinon, on appelle récursivement la procédure sur T_2 avec paramètre $k - (n_1 + 1)$.

On cherche à estimer la complexité de cet algorithme, en comptant combien de comparaisons il effectue en moyenne.

On suppose k fixé. On appelle x_i le $i$ème plus petit élément du tableau (on cherche donc x_k).

Soit X la variable aléatoire qui compte le nombre de comparaisons. On note $X_{i,j}$ la variable aléatoire qui vaut 1 si x_i et x_j sont comparés un jour par l'algorithme, et 0 sinon.

Q 2) Comment exprimer X en fonction de $X_{i,j}$?

Q 3) Calculer $E(X_{i,j})$. En déduire l'expression exacte de $E(X)$.

La formule étant complexe, on va procéder autrement pour l'analyser. On note $Y_{k,n}$ la variable aléatoire qui compte le nombre de comparaisons pour chercher le k-ème élément dans un tableau de taille n et $Y_n = \max_k Y_{k,n}$. On cherche à calculer $E(Y_n)$.

Q 4) Etablir une relation de récurrence pour $E(Y_n)$.

Q 5) En déduire que $E(Y_n) \leq 4n$.

Chapitre 2

SIMULATIONS

Dans ce chapitre, on explique comment simuler les différentes lois vues en cours.

2.1 Génération pseudo-aléatoire

Tout d'abord, il est important de savoir qu'il est difficile de générer des nombres aléatoires. D'abord, si jamais on les génère automatiquement, ils ne sont par définition pas aléatoires !

Les méthodes pour obtenir des vrais nombres aléatoires sont nombreuses, et exploitent toutes le fait que certains phénomènes physiques sont supposés totalement aléatoires. On peut par exemple utiliser la désintégration radioactive (`http://www.fourmilab.ch/hotbits/`), le bruit atmosphérique (`http://www.random.org`). Sous Linux, le fichier `/dev/random` fournit des bits aléatoires en utilisant les frappes sur le clavier, les accès disques, etc.

Prenons par exemple le cas du compteur Geiger. Il nous permet d'obtenir une suite aléatoire, mais très biaisée. En appproximation, on peut considérer le résultat comme le résultat de tirages de Bernoulli de paramètre p (avec p petit) indépendants. Comment le transformer en un tirage de paramètre $1/2$?

La solution est dûe a Von Neumann. Il suffit de regarder les bits deux par deux.

– S'ils sont égaux, on les oublie
– Sinon, on recopie le premier bit.

Ainsi sur la suite suivante :

1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 1

Le résultat obtenu sera

. . . . . 1 . . . . . 0 . . . . . 0 . . . . .

Il est clair que ce procéde fonctionne : La probabilité d'avoir 10 est bien égale à la probabilité d'avoir 01. Cependant, comme l'exemple le montre, on rejette très souvent : La probabilité que deux bits soient identiques est de $1 - 2p(1-p)$, qui peut être très grand. Le temps moyen pour lire deux bits différents est donc (temps moyen d'une variable aléatoire géométrique de paramètre $2p(1-p)$) de $\frac{1}{2p(1-p)}$ et dans le pire cas, le temps peut être infini !

Pour certaines applications, il est nécessaire d'obtenir beaucoup de nombres aléatoires, et rapidement. Dans ce cas là, ces générateurs sont trop lents, et on utilise plutôt des générateurs *pseudo-aléatoires*. Ces générateurs n'ont d'aléatoires que le nom, puisqu'ils sont en effet complètement déterministes.

Le modèle de générateur pseudo-aléatoire le plus connu est le générateur congruentiel linéaire. Il fonctionne de la façon suivante. On part d'un entier g , appelé *graine*, initialisé par exemple en fonction de la date. Pour obtenir le nombre aléatoire suivant, il suffit alors de changer la graine en utilisant la formule

$$g := ag + b \mod m$$

Il est à noter que la suite des g ainsi obtenue n'est pas aléatoire, puisque en particulier elle est périodique, de période plus petite que m ! Cependant, si m, a, b sont bien choisis, on peut espérer que cette contrainte ne soit pas trop importante.

Pour obtenir quelque chose qui "ressemble" à de l'aléatoire, il vaut mieux également ne pas utiliser g directement, mais une partie de g. Dans le cas où a, g sont impairs et m pair (cas de la plupart des générateurs), on s'aperçoit que g alterne entre pair et impair ! La solution utilisée fréquemment est de ne regarder que les bits de poids fort de g, pour éviter cette situation désagréable. C'est le cas par exemple de Java 1.7 et de la plupart des compilateurs C.

Lorsqu'on cherche à effectuer des simulations, les générateurs pseudo-aléatoires peuvent également être intéressants, puisqu'il suffit de connaître la graine pour recommencer l'expérience. Cela permet d'avoir à moindre frais quelque chose qui paraît aléatoire, mais qui est reproductible.

2.2 Simulation - Les lois classiques

On suppose maintenant avoir un moyen de simuler une variable aléatoire uniforme U entre 0 et 1. Il peut s'agir de la fonction `random()` de son système d'exploitation qui, comme discuté auparavant, peut n'être qu'une approximation d'une vraie variable aléatoire uniforme.

On va maintenant chercher comment utiliser U pour générer des variables répondant à d'autres lois.

2.2.1 Loi de Bernoulli

L'intuition nous guide naturellement vers l'algorithme suivant :
- Si $U < p$ répondre 1
- Sinon répondre 0

Comme $P(U < p) = p$, cet algorithme répond naturellement à la question.

A retenir — *Loi de Bernoulli*

Pour simuler une loi de Bernoulli de paramètre p à partir d'une loi uniforme U, renvoyer 1 si $U < p$, et 0 sinon.

2.2.2 Loi Binomiale $B(n, p)$

Comme la loi binomiale est une somme de n variables aléatoires de Bernoulli indépendantes, il suffit d'appliquer la formule précédente. Il faut supposer de plus que les n appels consécutifs à la fonction random() sont bien indépendants, ce qui est raisonnable si n n'est pas trop grand.

A retenir — *Loi binomiale*

Pour simuler une loi binomiale de paramètre (n, p), faire la somme de n variables de Bernoulli indépendantes de paramètre p.

2.2.3 Loi exponentielle

Rappelons que si X est de loi exponentielle de paramètre λ alors $P(X \geq t) = e^{-\lambda t}$

Proposition 2.1

$Y = \frac{-1}{\lambda} \ln U$ suit une loi exponentielle de paramètre λ.

Preuve :

$$P(Y > t) = P(\frac{-1}{\lambda} \ln U > t) = P(\ln U < -\lambda t) = P(U < e^{-\lambda t}) = e^{-\lambda t}$$

■

Comment devine-t-on cette formule ?

Théorème 2.2 (*Méthode de l'inverse*)

Soit X tel que $P(X \leq t) = f(t)$. On suppose que $P(X = t) = 0$ partout. Alors $Y = f^{-1}(U)$ suit la même loi que X.

L'hypothèse implique que f est strictement croissante (donc inversible).

Preuve : $P[Y \leq t] = P[f^{-1}(U) \leq t] = P[U \leq f(t)] = f(t)$. ■

Dans le cas de la loi exponentielle ($f(t) = 1 - e^{-\lambda t}$), on ne retombe pas sur la formule vu au dessus mais sur la formule $Y = \frac{-1}{\lambda} \ln(1 - U)$, ce qui ne change pas grand chose. Cependant cette deuxième formule est en pratique meilleure, puisque le résultat de la fonction random() n'est jamais égal à 1 (mais peut être égal à 0).

A retenir — *Loi exponentielle*

Pour simuler une variable de loi exponentielle de paramètre λ à partir de U uniforme, faire $X = \frac{-1}{\lambda} \ln U$

2.2.4 Loi de Poisson

Théorème 2.3

Soit $E_0 \ldots E_n$ des variables aléatoires indépendantes suivant une loi exponentielle de paramètre λ.

On note $B = \min\{n | E_0 + \ldots E_n > 1\}$

Alors B suit une loi de Poisson de paramètre λ

Preuve : $P[B = 0] = P[E_0 > 1] = e^{-\lambda} = e^{-\lambda} \frac{\lambda^0}{0!}$.

$P[B = 1] = P[E_0 + E_1 > 1] - P[E_0 > 1]$ = ?

$$P[B = 1] = \int_0^1 \text{“}P(E_0 = t)\text{”} P(E_1 > 1 - t) dt = \int_0^1 \lambda e^{-\lambda t} e^{-\lambda(1-t)} dt = \lambda e^{-\lambda}$$

La situation se complique pour $P[B = n]$ et n'est pas démontrée ici. ■

Il suffit donc de savoir simuler une loi exponentielle pour simuler une loi de Poisson. Cependant, on peut aussi simplifier les choses. Supposons que E_i soit simulée par $E_i = \frac{-1}{\lambda} \ln U_i$. Dans ce cas, dire que $E_0 + \dots E_n > 1$ revient à dire que $\frac{-1}{\lambda} \ln \prod U_i > 1$, c'est à dire $\prod U_i < e^{-\lambda}$

L'algorithme est donc très simple

A retenir — *Loi de Poisson*

Pour simuler une loi de Poisson de paramètre λ, à partir d'une loi uniforme U donnée par une fonction random, utiliser :

```
x = random()
n = 0
while x > exp(-lambda):
  x = x * random()
  n = n + 1
return n
```

2.2.5 Loi géométrique

On rappelle que la loi géométrique est définie par $P(X = k) = (1 - p)^{k-1}p$. Il s'agit d'une répétition d'une loi de Bernoulli Y de paramètre p jusqu'à tomber sur $Y = 1$, ce qui donne immédiatement un algorithme pour la simuler.

A retenir — *Loi géométrique*

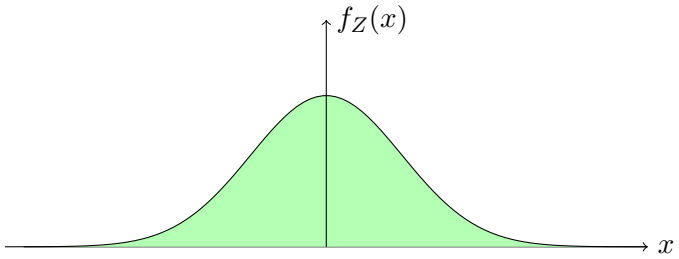
Pour simuler une loi géométrique de paramètre p, simuler une variable de loi de Bernoulli de paramètre p jusqu'à que la réponse soit 1. Le résultat est alors le nombre total de lancers.

2.2.6 Loi normale centrée réduite

La loi normale centrée réduite est la loi Z de densité

$$f_Z(x) == \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$

dont la représentation est



Il n'existe malheureusement pas de formule simple pour représenter $P(Z \geq x)$, et on est obligé de travailler avec la fonction de densité. En particulier, il est difficile d'exprimer son inverse, donc d'utiliser la méthode de l'inverse pour la calculer. De même, le calcul de la variance et de l'espérance est un peu compliqué

A retenir *Loi Normale*

Si X est de loi normale centrée réduite, alors

$$E(X) = 0, Var(X) = 1$$

Une première méthode, approchée, peut-être déduite du théorème central limite :

Théorème 2.4 (*Théorème central limite*)

Soit X_i est une suite de variables aléatoires indépendantes de même loi et d'espérance et de variance finie.
On pose $S_n = \sum_{i=1}^{n} X_i$.
Alors $\frac{S_n - E(S_n)}{\sqrt{Var(S_n)}} \to Z$ où Z est une variable aléatoire de loi normale centrée réduite.

Si n est assez grand, on peut donc utiliser la formule pour approcher (mais pas simuler !) Z.

Supposons par exmeple que les X_i soient uniformes sur $[-1/2, 1/2]$ (c'est à dire $X_i = U_i - 1/2$). Alors $ES_n = 0$ et $Var(S_n) = nVar(X) = \frac{n}{12}$. En particulier si on prend $n = 12$ (qui n'est pas si grand que ça !), on a que $\sum_{i=0} 12U_i - 6$ est une approximation de Z, dont on peut montrer qu'elle n'est pas trop mauvaise. Mais il ne s'agit que d'une approximation.

Prenons X_1 et X_2 chacune de loi normale centrée réduite et indépendante et regardons $X = (X_1, X_2)$ comme un point dans le plan. Sa fonction de densité est

$$\frac{1}{2\pi} e^{-\frac{x_1^2 + x_2^2}{2}}$$

Une façon simple de comprendre ce qui se passe est d'examiner la fonction de densité en coordonnée polaire (r, θ). Elle devient

$$\frac{1}{2\pi} r e^{-\frac{r^2}{2}}$$

On note maintenant R et Θ les variables aléatoires correspondant aux coordonnées polaires du point X.

Comme la densité ne dépend pas de Θ, on en déduit que Θ est uniforme sur $[0, 2\pi]$ (ce qui explique le facteur $\frac{1}{2\pi}$ de la densité). De plus R et Θ sont indépendantes. On peut maintenant calculer la loi de R :

$$P(R \leq t) = \int_0^t r e^{-\frac{r^2}{2}} dt = 1 - e^{-\frac{t^2}{2}}$$

On en déduit que R^2 suit une loi exponentielle de paramètre $1/2$.

Théorème 2.5

Si X_1 et X_2 sont deux variables aléatoires de loi normale centrée réduite, alors $X_1^2 + X_2^2$ est une variable exponentielle de paramètre $1/2$

On en déduit la méthode de simulation suivante :
- Simuler une variable R^2 exponentielle de paramètre $1/2$
- Simuler une variable Θ uniforme sur $[0, 2\pi]$
- Poser $X_1 = R\cos\Theta$ et $X_2 = R\sin\Theta$.

Dit autrement

A retenir — *Loi Normale*

Pour simuler une loi normale centrée réduite à partir de deux variables U_1, U_2 de loi uniforme, calculer
- $X_1 = \sqrt{-2\ln U_1}\cos 2\pi U_2$
- $X_2 = \sqrt{-2\ln U_1}\sin 2\pi U_2$

Alors X_1 et X_2 sont toutes deux indépendantes et de loi normale centrée réduite.

2.3 Simulation - Méthodes générales

On s'intéresse maintenant aux méthodes de simulation de variables aléatoires quelconques.

2.3.1 Méthode de l'inverse

Voir plus haut.

2.3.2 Méthode directe

Soit une variable X discrète pour laquelle on sait calculer $P(X = k)$, et encore mieux $P(X \geq k)$.
Une manière simple à décrire (mais pas forcément à utiliser) est la suivante :
- Si $U \leq P(X = 0)$ renvoyer 0
- Si $U \leq P(X \leq 1)$ renvoyer 1
- Si $U \leq P(X \leq 2)$ renvoyer 2
- $\ldots$

2.3.3 Méthode du rejet

Supposons qu'on sache comment simuler une variable X et qu'on veuille simuler une variable Y.
Supposons de plus que $P(X = k) \geq P(Y = k)$ pour tout k. Une manière de simuler Y est la suivante :
- On simule X.
- Si $X = k$, on accept avec probabilité $\frac{P(Y=k)}{P(X=k)}$. Sinon, on recommence.

L'idée est assez convaincante, sauf qu'il est impossible d'avoir $P(X = k) \geq P(Y = k)$ pour tout k, sans quoi $X = Y$!

On résout le problème en divisant $P(Y = k)$ par une constante C suffisament grande pour que $P(X = k) \geq \frac{P(Y=k)}{C}$.

On peut maintenant procéder de la même manière :
– On simule X.
– Si $X = k$, on accepte avec probabilité $\frac{P(Y=k)}{CP(X=k)}$. Sinon, on recommence.

Théorème 2.6

L'algorithme précédent donne le bon résultat.

Preuve : On note Z le résultat de l'algorithme.

A la première étape, la probabilité que l'algorithme s'arrête et renvoie k est exactement $P(X = k)\frac{P(Y=k)}{CP(X=k)} = \frac{P(Y=k)}{C}$. De plus la probabilité que la première étape réussisse est $\sum_k \frac{P(Y=k)}{C} = \frac{1}{C}$. La probabilité qu'on renvoie k sachant qu'on s'arrête est donc $P(Y = k)$.

Ces formules sont également valables à toutes les étapes, puisqu'on fait toujours la même chose.

La probabilité qu'on s'arrête à la $i$ème étape est donc $\left(\frac{1}{C}\right)\left(1 - \frac{1}{C}\right)^{i-1}$. La probabilité qu'on renvoie k est donc

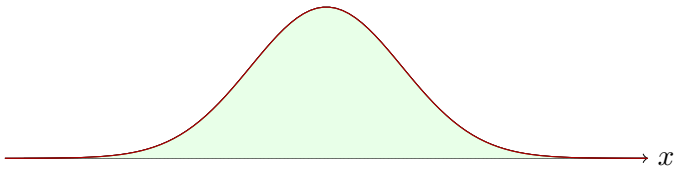
$$P(Z = k) = \sum_i (1 - \frac{1}{C})^{i-1}\frac{1}{C}P(Y = k) = P(Y = k)$$

■

Cas de deux variables continues La même idée peut être utilisée pour des variables X et Y continues.

Notons f_X et f_Y la densité de X et Y et supposons que $f_X \geq Cf_Y$ pour une constante C.

Représentons f_X et $\frac{f_Y}{C}$ sur un dessin.



Tirer un point selon X revient à choisir un point au hasard dans la partie verte. S'il n'est pas dans la partie hachurée, on accepte, sinon en relance.

Exercices

(2 - 1) *(randU)*

Un des premiers générateurs aléatoires, noté randU, était conçu de la façon suivante. On part d'une graine g, un entier sur 31 bits (entre 0 et $2^{31}-1$). Si on demande un nombre aléatoire entre 0 et 1 à randU, les opérations suivantes sont effectuées :
– g est mis à jour par $g := 65539 \times g \mod 2^{31}$.
– On renvoie $g/2^{31}$
Notons que $65539 = 2^{16} + 3$.

Q 1) On note g_1, g_2, g_3 trois graines successives. Montrer qu'on peut exprimer g_3 par $g_3 = 6g_2 - 9g_1 \mod 2^{31}$. En déduire que ce générateur est très mauvais.

(2 - 2) *(Java)*

Le générateur pseudo aléatoire de Java fonctionne de la manière suivante. On part d'une graine g, qui est un entier sur 48 bits (donc un entier entre 0 et $2^{48}-1$). La graine est en général initialisée à partir de la date.

Lorsqu'on demande un nombre aléatoire avec la fonction `nextInt`, le programme effectue les opérations suivantes :
– Il recalcule la graine g avec la formule $g := ag + b \mod 2^{48}$ (générateur congruentiel linéaire)
– Il renvoie les 32 premiers bits (les bits de poids fort) de g. (Cela revient à calculer $g/2^{16}$)

Q 1) On remplace 48 par 4, 32 par 2 et on prend $a = 3$, $b = 1$. On suppose que la graine g vaut 8. Calculer les 4 premières valeurs renvoyées par `nextInt`.

Q 2) Expliquer pourquoi on prend les 32 bits de poids fort de g et non pas les 32 bits de poids faible.

On revient au cas de Java. On suppose (comme c'est souvent le cas dans un générateur congruentiel linéaire), que a est *inversible* (on peut diviser par a) c'est à dire qu'il existe un entier a^{-1} tel que $aa^{-1} = 1 \mod 2^{48}$.

Q 3) On suppose que `nextInt` renvoie 147201. Combien de graines différentes possibles peuvent donner ce résultat ?

Q 4) On suppose que la graine est une variable aléatoire G uniforme entre 0 et $2^{48}-1$. On note X_1 le premier nombre aléatoire renvoyé par `nextInt`. Prouver que X_1 est uniforme.

Q 5) On note X_2 le deuxième nombre aléatoire renvoyé par `nextInt`. Prouver que X_1 et X_2 ne sont pas indépendants.

Q 6) Expliquer comment retrouver rapidement la graine G connaissant X_1 et X_2. (Aide : 2^{16} est un petit nombre).

(2 - 3) *(Génération uniforme)* On suppose que le générateur qu'on utilise est parfait, et qu'il renvoie un nombre entier entre 0 et $2^{32}-1 = 4294967295$

On cherche à simuler un entier choisi uniformément entre 0 et 999.

Q 1) Montrer que $rand() \mod 1000$ ne donne pas le bon résultat.

Q 2) Expliquer comment obtenir une bonne simulation

Q 3) Expliquer le code java suivant, qui réalise la simulation (code adapté des source de `nextInt`):

```
do {
     bits = rand();
     val = bits % n;
     } while (bits - val + 999 >= 4294967295);
```

(2 - 4) *(Variance)* Dans cet exercice, on cherche à estimer un paramètre m d'un système.

Q 1) On dispose de deux simulations différentes permettant d'estimer m. La première correspond à une variable aléatoire X et la deuxième à une variable Y, pour lesquelles on a donc $E(X) = E(Y) = m$. Quelle méthode doit-on préférer ?

Q 2) Une troisième méthode consiste à utiliser $\frac{X+Y}{2}$. Dans quel cas est-ce que cette méthode est meilleure ? Discutez en introduisant le terme $CoV(X, Y) = E(XY) - E(X)E(Y)$

Q 3) Que vaut $CoV(X, Y)$ si X et Y sont indépendants ? Si $X = Y$?

Q 4) Soit f, g deux fonctions croissantes. Montrer $E(f(X)g(X)) \geq E(f(X))E(g(X))$. (Aide : introduire une autre variable Z, indépendante de X et de même loi, et calculer $E\left[(f(X) - f(Z))(g(X) - g(Z))\right]$)

Q 5) On suppose que $X = f(U)$ pour U uniforme sur $[0, 1]$. Montrer que $\frac{f(U)+f(1-U)}{2}$ est un meilleur estimateur de m

(2 - 5) *(Loi du χ^2)*

La loi du χ^2 avec k degrés de liberté est la loi correspondant à la somme du carré de k variables aléatoires normales centrées réduites.

Q 1) Expliquer comment simuler la loi du χ^2. Distinguer les cas k pair et k impair.

Chapitre 3

CHAÎNES DE MARKOV

3.1 Définitions

On s'intéresse dans ce chapitre à une suite de variables aléatoires X_i qui correspondent à l'évolution temporelle d'un processus aléatoire. L'idée est qu'à chaque moment de temps i, le processus est dans l'*état* X_i et l'évolution du processus au temps X_{i+1} ne dépend que de X_i.

♦ **Exemple**

On joue au casino. A chaque étape, on peut gagner 4 euros avec probabilité $1/20$ ou en perdre 1. X_i représente alors le capital au moment i. X_0 est le capital initial. On a donc $P[X_{i+1} = k + 5|X_i = k] = 1/20$ et $P[X_{i+1} = k - 1|X_i = k] = 19/20$.

♦ **Exemple**

(Bouche à oreille) On considère le transfert d'un bit d'information dans n canaux bruités successifs, chacun changeant la valeur du bit avec probabilité p.

On peut modéliser la situation par une chaîne de Markov. X_0 représente le bit d'information à l'origine.

$P[X_{i+1} = 1|X_i = 0] = p$ et $P[X_{i+1} = 0|X_i = 0] = 1 - p$, etc.

Définition 3.1

Une chaîne de Markov est une suite de variable aléatoires X_i tels que

$$P\left[X_{n+1} = x|X_1 = x_1 \wedge \ldots X_n = x_n\right] = P\left[X_{n+1} = x|X_n = x_n\right]$$

Définition 3.2

Une chaîne de Markov homogène est une chaîne de Markov telle que

$$P\left[X_{n+1} = j|X_n = i\right] = p_{ij}$$

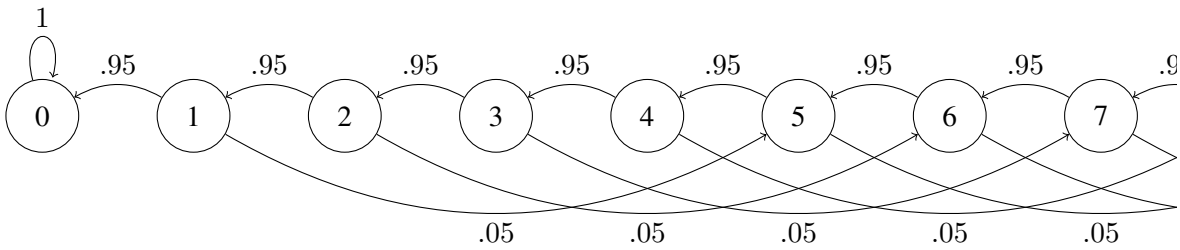
(La probabilité ne varie pas avec le temps n)

3.1.1 Représentations

La manière la plus classique de représenter une chaîne de Markov est par un graphe (automate). On représente chaque état i par un sommet, et on met une arête de i à j étiquetée avec la probabilité d'aller de i à j (donc p_{ij}).

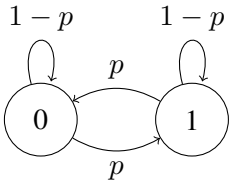
♦ **Exemple**

L'exemple du casino



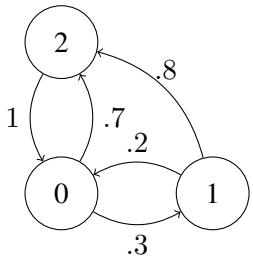
♦ **Exemple**

Le bouche à oreille



♦ **Exemple**

Un autre exemple qui servira dans la suite



La représentation la plus utile est par une *matrice* P. On met en P_{ij} la probabilité d'aller de i à j.

♦ **Exemple**

Le bouche à oreille

$$\begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$$

♦ **Exemple**

L'autre exemple

$$\begin{pmatrix} 0 & .3 & .7 \\ .2 & 0 & .8 \\ 1 & 0 & 0 \end{pmatrix}$$

Il faut noter que :
- Pour que la représentation sous forme de graphe soit correcte, la somme des coefficients en sortie d'un noeud doit valoir 1.
- Pour que la représentation sous forme de matrice soit correcte, la somme de chaque ligne doit valoir 1
- De plus, dans les deux cas, les nombres doivent être compris entre 0 et 1.

Une matrice qui vérifie les deux conditions est appelée une matrice *stochastique*.

Définition 3.3

La loi initiale de la chaîne de Markov est la loi $p[X_0 = i]$. S'il existe un état i pour lequel $p[X_0 = i] = 1$, on appelle i l'état initial de la chaîne de Markov.

On représente la loi initiale par un *vecteur* dans le cas de la représentation matricielle. Dans le cas du graphe, il n'y a pas de notation standard, mais on peut par exemple écrire la probabilité de commencer au noeud i à l'intérieur du noeud i.

3.2 Calcul

Ce qui nous intéresse dans une chaîne de Markov, c'est le comportement à la limite. Que vaut $P[X_n = i]$? Que vaut cette probabilité quand n tend vers l'infini ?

Proposition 3.1

$$P[X_0 = i_0 \wedge X_1 = i_1 \wedge \ldots X_n = i_n] = P[X_0 = i_0] \times p_{i_0,i_1} \times p_{i_1,i_2} \ldots p_{i_{n-1},i_n}$$

Commençons par le cas particulier $P[X_{n+2} = i | X_n = j]$

$$\begin{array}{rcl} P[X_{n+2} = i | X_n = j] & = & \sum_k P[X_{n+2} = i \wedge X_{n+1} = k | X_n = j] \\ & = & \sum_k P[X_{n+2} = i | X_{n+1} = k \wedge X_n = j] P[X_{n+1} = k | X_n = j] \\ & = & \sum_k P[X_{n+2} = i | X_{n+1} = k] P[X_{n+1} = k | X_n = j] \\ & = & \sum_k p_{i,k} p_{k,j} \end{array}$$

(Par application successive de la probabilité de l'union disjointe, puis de la loi de composition des probabilités conditionnelles, puis de l'hypothèse de Markov.)

Ce dernier terme est exactement le coefficient en (i, j) de la matrice P^2 !

Théorème 3.2 (*Chaiman-Kolmogorov*)

$P[X_n = i | X_0 = j]$ est le coefficient en (i, j) de la matrice P^n.

Pour obtenir la loi de X_n, il suffit donc de multiplier P^n par le vecteur initial.

♦ **Exemple**

Traité au tableau.

Le calcul de P^n est relativement laborieux. On cherche dans la suite des propriétés sur P pour connaître le comportement limite de la chaîne.

3.3 Classification des états

Définition 3.4

L'état j est *accessible* à partir de i s'il existe n tel que $P^n_{i,j} \neq 0$, autrement dit s'il existe n tel que $P[X_n = i|X_0 = j] > 0$.

Une chaîne de Markov est *irréductible* si pour tous états i, j, l'état j est accessible à partir de i.

Les deux derniers exemples vus sont irréductibles. Ce n'est pas le cas du casino : Aucun état autre que 0 n'est accessible de 0.

Définition 3.5

i est récurrent si $P[\exists n, X_n = i|X_0 = i] = f_i = 1$. ($i$ est récurrent s'il est presque sûr qu'on va retourner en i une fois qu'on l'a quitté).

Si i n'est pas récurrent on dit que i est transient.

Dans l'exemple du casino, seul l'état 0 est récurrent. En effet, dans tous les autres états i, on a une probabilité p_i (qui dépend de l'état i) d'arriver en 0, donc la probabilité de retourner en i est au plus $1 - p_i$.

Dans les deux autres exemples, on peut prouver que tous les états sont récurrents.

Si un état est récurrent, on est presque sûr de toujours y retourner, donc on y retourne une infinité de fois.

Si un état est transient, on a une probabilité p, une fois quitté i, de ne pas y retourner. La probabilité d'y aller k fois est donc exactement $p(1-p)^{k-1}$ et donc on y va en moyenne $1/p$ fois.

Proposition 3.3

Un état i est transient si l'espérance du temps de passage en i est fini. Il est récurrent sinon.

Proposition 3.4

i est récurrent si $\sum_n P^n_{i,i} = \infty$.

Corollaire 3.5

Dans une chaîne de Markov finie irréductible, tous les états sont récurrents.

Preuve : D'après la proposition précédente, il n'est pas possible pour tous les états d'être transients. Donc l'un est récurrent. Mais ca implique que tous le sont (pourquoi ?) ■

Définition 3.6

Le temps de retour en i est défini par $Y_i = \min\{n > 0|X_n = i\}$ (en supposant $X_0 = i$). On note $E(Y_i)$ l'espérance de Y_i.

- i est récurrent positif si $E(Y_i) < \infty$
- Sinon i est récurrent nul.

Dans une chaîne de Markov finie, les récurrents sont récurrents positifs, mais ce n'est pas toujours le cas dans une chaîne infinie.

3.4 Comportements limites

Regardons la matrice P^n quand P correspond à l'exemple du bouche à oreilles pour $p = 0.01$ et $n = 200$.

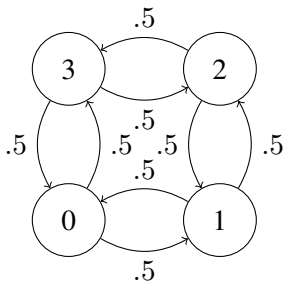
$$\begin{pmatrix} 0.509 & 0.491 \\ 0.491 & 0.509 \end{pmatrix}$$

Dans l'autre exemple et pour $n = 15$

$$\begin{pmatrix} 0.446 & 0.134 & 0.42 \\ 0.445 & 0.135 & 0.42 \\ 0.447 & 0.134 & 0.419 \end{pmatrix}$$

Cela veut dire que quand n est grand, l'endroit d'où on vient n'a pas d'importance, et donc (pour le deuxième exemple), on a une probabilité de 0.446 d'être dans l'état 1 quel que soit l'état initial !

Il y a cependant des exemples où la situation n'est pas si simple :



correspondant à la matrice :

$$\begin{pmatrix} 0 & 0.5 & 0 & 0.5 \\ 0.5 & 0 & 0.5 & 0 \\ 0 & 0.5 & 0 & 0.5 \\ 0.5 & 0 & 0.5 & 0 \end{pmatrix}$$

Si on itère 20 fois la matrice, on retombe sur la même. MAIS, si on l'itère 21 fois, on tombe sur

$$\begin{pmatrix} 0.5 & 0 & 0.5 & 0 \\ 0 & 0.5 & 0 & 0.5 \\ 0.5 & 0 & 0.5 & 0 \\ 0 & 0.5 & 0 & 0.5 \end{pmatrix}$$

La raison est que la chaîne de Markov est *périodique*

Définition 3.7

La période d'un état i, noté d_i est le pgcd de tous les n tel que $P^n_{i,i} > 0$. Un état est périodique si $d_i > 1$, et apériodique sinon. Une chaîne de Markov est apériodique si tous les états sont apériodiques

Théorème 3.6

Si une chaîne de Markov finie est apériodique, alors $\pi_j = \lim_n P[X_n = j]$ existe. Si elle est de plus irreductible, alors π_j ne dépend pas de $P[X_0 = j]$.

Dans ce cas le vecteur π est l'unique solution de l'équation $\pi P = \pi$. avec π de somme 1.

Définition 3.8

Une distribution π tel que $\pi P = \pi$ et de somme 1, autrement dit telle que $\pi_i = sum_j p_j P_i j$ est appelée distribution (ou loi, ou probabilité) stationnaire.

Proposition 3.7

Pour une chaîne infinie irréductible, π existe si et seulement si il existe un état positivement récurrent. Dans ce cas, $\pi_j = \lim_n P[X_n = j]$.

On ne prouvera pas ce théorème pourtant fondamental.
Reprenons l'exemple. On obtient le système

$$\begin{aligned} x &= 0,2.y + z \\ y &= .3x \\ z &= 0.7x + 0.8y \\ x + y + z &= 1 \end{aligned}$$

d'où on tire $z = 0.94x$ puis

$$x = \frac{1}{2.24}, y = \frac{3}{22.4}, z = \frac{94}{224}$$

On peut donner d'autres caractérisations de π qui peuvent être utiles :

Proposition 3.8

Soit X_n le temps passé en i en n étapes de temps. Alors $E(X_n)/n$ converge vers π_i. (π_i est le temps moyen passé en i.)

Preuve : $X_n = Y_1 + \ldots Y_n$ où Y_i vaut 1 si on est en i et 0 sinon. Par définition $P[Y_i = 1]$ converge vers π_i, donc $E(X_n)/n$ aussi (c'est la moyenne de Césaro). ■

Le résultat n'est pas vrai qu'en moyenne : X_n/n converge presque surement vers π_i.

Proposition 3.9

Soit f une fonction (bornée) et soit $(X_n)_{n\in\mathbb{N}}$ une chaîne de Markov irreductible finie. Alors

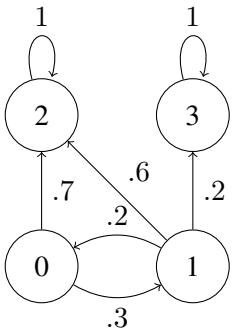
$$\frac{\sum_{i=1}^n f(X_i)}{n} \to_{nx\to\infty} \sum f(j)\pi_j$$

3.5 Chaînes de Markov absorbantes

Définition 3.9

Un état est absorbant si $p_{ii} = 1$ (on reste dans i une fois qu'on y est). Une chaîne de Markov est absorbante si tout état accède à un état absorbant.

♦ **Exemple**



de matrice

$$\begin{pmatrix} 0 & .3 & .7 & 0 \\ .2 & 0 & .6 & .2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Proposition 3.10

Quitte à renuméroter les états, la matrice correspondant à une chaîne de Markov absorbante est :

$$\begin{pmatrix} Q & R \\ 0 & I \end{pmatrix}$$

Où I_a est la matrice identité de dimension a

La matrice $N = (I_{n-a} - Q)^{-1}$ est bien définie ($I - Q$ est inversible) et est appelée *matrice fondamentale* de la chaîne.

Dans l'exemple on a donc

$$N = \begin{pmatrix} 1 & -.3 \\ -.2 & 1 \end{pmatrix}^{-1} = \frac{1}{.94} \begin{pmatrix} 1 & .3 \\ .2 & 1 \end{pmatrix}$$

Notons que $(I - Q)^{-1} = \sum_k Q^k$. On en déduit

Proposition 3.11

$N_{i,j}$ est le nombre moyen de passages en j en partant de i. $M_i = \sum_j N_{i,j}$ est le temps moyen (partant de i) avant absorption.

Calculons maintenant la probabilité $a_{i,k}$ que, partant de i le résultat soit l'état absorbant k. Alors

$$a_{i,k} = p_{i,k} = \sum_j p_{i,j} a_{j,k}$$

où la somme est sur les j non absorbants. En terme matriciel :

$$A = R + QA$$

ce qui donne $A - QA = R$, soit $(I - Q)A = R$, d'où $A = (I - Q)^{-1}R = NR$.

Proposition 3.12

La matrice A telle que $A_{i,j}$ soit la probabilité que partant de i on termine dans l'état absorbant j est

$$A = NR$$

Revenons sur l'exemple pratique :

$$NR = \frac{1}{.94}\begin{pmatrix} 1 & .3 \\ .2 & 1 \end{pmatrix}\begin{pmatrix} .7 & 0 \\ .6 & .2 \end{pmatrix} = \frac{1}{.94}\begin{pmatrix} 0.88 & 0.06 \\ 0.74 & 0.2 \end{pmatrix}$$

La somme de chaque ligne fait bien 1, ce qui représente le fait qu'on est presque certain de tomber sur un état absorbant.

Exercices

(3 - 1) *(Taxi)*

Une société de Taxi opère entre les villes de Charmes, Lunéville et Nancy, qui sont à peu près équidistantes. Un chauffeur a remarqué les statistiques suivantes :
– 70% des clients à Nancy veulent aller à Lunéville, les autres voulant aller à Charmes
– 90% des clients à Charmes veulent aller à Nancy, les autres à Lunéville
– 20% des clients de Lunéville veulent aller à Charmes, les autres voulent aller à Nancy.

On suppose que chaque course prend exactement 1h au taxi. Lorsque le taxi est à Nancy, il trouve toujours un client rapidement. Quand il est à Lunéville ou à Charmes, il a une chance sur deux de trouver un client rapidement, et sinon il part au bar prendre un café pendant 1h.

Examinez le temps que passe le chauffeur en moyenne à Charmes. (On modélisera la situation par une chaîne de Markov).

(3 - 2) *(Parapluies)* Caroline possède deux parapluies, qu'elle utilise pour aller de chez elle au bureau et vice versa. Caroline ne prend un parapluie pour se déplacer que si il pleut (Donc si il pleut sur le chemin du bureau, mais pas au retour, le parapluie restera au bureau).

On suppose qu'au départ les deux parapluies sont chez elle et que chaque jour il y a une probabilité p de pleuvoir. Calculez le nombre de trajets en moyenne que Caroline effectuera sans parapluie alors qu'il pleut.

(3 - 3) *(Move-to-Front, le retour)*

On se donne un tableau contenant 3 objets x_1, x_2, x_3, chacun ayant respectivement une probabilité p_1, p_2, p_3, dont on suppose $p_1 > p_2 > p_3 > 0$.

On rappelle l'algorithme Move-to-Front : Lorsqu'on recherche l'élément x_i on le replace en tête du tableau.

Modéliser la situation par une chaîne de Markov. Vérifiez qu'elle est irréductible et apériodique. Calculez la probabilité qu'en régime continu les objets soient triés par ordre de probabilité décroissante.

(3 - 4) *(Tableaux)*

On considère un tableau au départ vide. A chaque instant, on effectue une insertion (ajout) dans le tableau avec probabilité a, une suppression avec probabilité d, et une recherche avec probabilité $1 - a - d = r$.

On cherche à modéliser le taille moyenne du tableau (on suppose que supprimer dans un tableau vide ne fait rien).

Q 1) Représenter le problème par une chaîne de Markov (infinie). Sous quelle condition (simple) sur a, d, r est-elle irréductible et apériodique ?

Q 2) Ecrire les équations régissant sa distribution stationnaire p_i.

Q 3) Montrer que $ap_i - dp_{i+1} = ap_{i-1} - dp_i$. En déduire $ap_i - dp_{i+1} = 0$.

Q 4) En déduire une expression de p_i en fonction de p_0.

Q 5) En utilisant $\sum p_i = 1$, en déduire une expression de p_i. Que se passe-t-il dans les cas
– $a = d$
– $a < d$
– $a > d$?

Q 6) On suppose maintenant que le tableau n'est pas infini, mais de taille m. Lorsque le tableau est plein et qu'on cherche à insérer, un message d'erreur est affiché mais le système continue à fonctionner. Résoudre à nouveau les équations. En déduire le nomre moyen de messages affichés.

Q 7) On considère un multiplexeur ATM, relié à deux réseaux en entrée. A chaque étape de temps, il y a une probabilité p qu'un paquet arrive sur l'un des réseaux. A chaque étape de temps, le multiplexeur ne peut traiter qu'un seul des paquets. Le multiplexeur a un tampon dans lequel il peut mettre les paquest en attente de traitement. Donner une estimation de la taille que doit avoir ce tampon en fonction de p.

(3 - 5) *(Dés et Pièces)*

Q 1) Calculez le temps moyen avant d'obtenir Pile-Face-Pile

Q 2) Montrer que le temps moyen avant d'obtenir k fois le même résultat en lançant un dé est

$$\frac{6^k - 1}{6 - 1}$$

Exercices

(3 - 6) *(Routage)*

On considère 3 machines A,B,C. Les machines communiquent entre elles par une technologie sans fil, et A et C sont trop loin l'une de l'autre pour qu'un paquet puisse aller directement de A à C. On cherche à envoyer 3 paquets de A à C. Un paquet entre A et B (resp. B et C) a une probabilité p d'être perdu. Si un paquet est perdu, il est retransmis par la machine.

Calculer le temps moyen avant que les 3 paquets arrivent en C.

On considère maintenant qu'une machine arrête totalement d'émettre si jamais un de ses paquets n'est pas arrivé. Donner (en fonction de k) la probabilité qu'à la limite la machine C ait reçu exactement k paquets.

(3 - 7) *(Loi géométrique)* En utilisant le formalisme des chaînes de Markov, montrer que l'espérance d'une loi géométrique de paramètre p est $1/p$.

(3 - 8) *(Protocoles de population)*

Alice, Bob et Carole sont passionés par les élections américaines et en discutent dès qu'ils se rencontrent. Chaque jour, deux des trois amis se rencontrent et en discutent. A la fin de la discussion, ils sont toujours d'accord sur le résultat.

Q 1) Au début, Alice est pour Barack Obama et Bob et Carole sont pour Mitt Romney. Calculer la probabilité qu'à la fin les trois votent pour Obama.

Q 2) Dessiner la chaîne de Markov dans le cas de 5 personnes. Ecrire les équations permettant de calculer les probabilités.

(3 - 9) *(Dés et Pièces)*

Q 1) Calculez le temps moyen avant d'obtenir Pile-Face-Pile

Q 2) Montrer que le temps moyen avant d'obtenir k fois le même résultat en lançant un dé est

$$\frac{6^k - 1}{6 - 1}$$

(3 - 10) *(Mémoire partagée)* On considère un ordinateur constitué de deux processeurs et d'une mémoire RAM, composée de deux cases mémoires. Il est impossible pour les deux processeurs d'accéder simultanément à la même case mémoire, et le processeur doit donc attendre avant de lire la case mémoire si elle est déjà occupée par l'autre processeur. Chacun des processeurs demande sans arrêt à lire une des deux cases mémoires, et ne fait rien d'autre. Il demande la première case avec probabilité p et la deuxième avec probabilité q ($p + q = 1$). A chaque cycle de l'ordinateur, chacun des deux processeurs est donc dans l'un des états suivants :

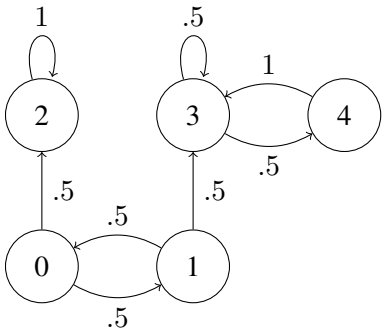
– Il lit la case mémoire 1 (resp. 2)

– Il attend pour lire la case mémoire 1 (resp. 2)

Calculer le nombre moyen de lectures par unité de temps. Calculer son maximum.

(3 - 11) *(Pathologie)*

Soit X_n la chaîne de Markov donné par le graphe suivant. Sachant qu'on part de l'état 1, calculer $\lim_{n\to\infty} P[X_n = 4]$



Exercices

(3 - 12) *(Moteurs de Recherche)*

Le graphe du Web est le graphe dont les sommets sont les pages Web et où on relie une page web x à une page web y s'il y a un lien vers la page y à partir de la page x.

La plupart des moteurs de recherche donnent une "valeur" (son *PageRank*™pour l'algorithme le plus connu) à chaque page web de la façon suivante : on considère un utilisateur qui, partant d'une page web au hasard, clique sur les liens au hasard. Le pagerank d'une page web x est alors la probabilité qu'une personne se trouve sur la page web x.

Q 1) Comment modéliser la situation par une chaîne de Markov ? Donner la formule permettant de calculer la distribution stationnaire

Q 2) Les théorèmes vus en cours sont-ils applicables ? Que faire ?

Q 3) Comment calculer le pagerank ?

Q 4) Comment augmenter son pagerank ?

(3 - 13) *(Marches aléatoires)* On considère un graphe non orienté (s'il y a une arête de x à y, il y a une arête de y à x), qu'on parcourt aléatoirement (c'est donc le même modèle que dans l'exercice précédent, sauf que le graphe est non orienté).

On note T le temps moyen qu'il faut pour passer par tous les sommets du graphe. On peut démontrer que $T \leq n^3$, où n est le nombre de sommets (la preuve n'est pas difficile et pourrait constituer un bon sujet d'exercice, mais ce sera pour un autre jour).

Q 1) Devinez un graphe avec n sommets pour lequel T est très petit, puis pour lequel T est très grand.

On suppose maintenant que tous les sommets ont le même nombre d'arêtes sortantes, et qu'elles sont numérotées de 1 à d. Le graphe G est fixé dans les questions qui suivent.

Q 2) On choisit un mot X sur l'alphabet $\{1, 2, \ldots, d\}$ de longueur $2n^3$. Montrer que la probabilité qu'on passe par tous les sommets en lisant X est au moins $1/2$.

Q 3) On suppose maintenant que X est de taille $2tn^3$. Que devient la probabilité ?

Maintenant le graphe G n'est plus fixé. On note S_G l'ensemble des mots u de longueur $2tn^3$ tels qu'on passe par tous les sommets en lisant u.

Q 4) Combien y a-t-il de graphes à n sommets et à d arêtes (numérotées) par sommets ?

Q 5) Montrer, si $t = dn \log n$, qu'il existe un mot u de longueur $2tn^3$ tel que, quel que soit le graphe à n sommets d'on part, on est certain de passer par tous les sommets du graphe.

(3 - 14) *(Inégalité de Kraft)*

Un ensemble S de mots est un *code préfixe* si aucun mot de S n'est préfixe d'un mot de S. On suppose dans la suite que les mots sont sur l'alphabet $\{a, b\}$

On note $S = \{u_1, u_2 \dots u_n\}$, où le mot u_i est de longueur l_i.

Soit $L = \max l_i$.

Q 1) Soit U la variable aléatoire correspondant au tirage d'un mot au hasard parmi tous les mots de taille L (pas seulement les mots de S). Calculer, en fonction de i, la probabilité que U commence par u_i.

Q 2) Montrer que $\sum_i \frac{1}{2^{l_i}} \leq 1$.

(3 - 15) *(3-SAT)* Un littéral est soit une variable x_i soit la négation d'une variable $\neg x_i$ qu'on note plus souvent $\overline{x_i}$.

Une formule est dite en forme normale conjonctive (CNF) si elle s'écrit comme un "et" ($\wedge$) de "ou" ($\vee$) de littéraux, c'est à dire s'il existe des littéraux $V_{i,j}$ tels que $\phi = \bigwedge_i (\bigvee_j V_{i,j})$. On note $C_i = (\bigvee_j V_{i,j})$. C_i est appelé une clause, et correspond au i-ème terme du $\wedge$. On ne s'intéresse dans la suite qu'aux formules où chaque clause contient exactement 3 littéraux.

Voici deux exemples de telles formules (la première avec deux clauses, la deuxième avec quatre clauses) :

$$(x \vee y \vee \overline{w}) \wedge (w \vee z \vee \overline{x})$$

$$(w \vee x \vee \overline{y}) \wedge (y \vee z \vee \overline{x}) \wedge (a \vee b \vee \overline{w}) \wedge (b \vee w \vee \overline{z})$$

Le problème 3SAT est de savoir décider, étant donné une formule en forme normale conjonctive avec exactement 3 littéraux par clause, s'il est possible de rendre la formule vraie par un bon choix des variables.

Il s'agit d'un des problèmes les plus difficiles rencontrés en informatique : on ne connaît aucune solution pour résoudre ce problème rapidement, si ce n'est tester tous les choix possibles pour les variables.

Dans cet exercice, on va chercher un algorithme qui répond partiellement à la solution, en donnant un choix des variables qui vérifie une grande partie des clauses

Q 1) On choisit les variables au hasard. On note S le nombre de clauses satisfaites par ce choix aléatoire. Calculer $E(S)$.

Q 2) Montrer qu'il existe un choix des variables qui permet de réaliser les $7/8$-èmes des clauses.

Q 3) On note S_0 (resp. S_1) la variable aléatoire correspondant au nombre de clauses satisfaites sous l'hypothèse où la première variable vaut 0 (resp. 1). Quel est le lien entre $E(S), E(S_0)$ et $E(S_1)$?

Q 4) En déduire un algorithme déterministe rapide pour obtenir un choix de variable permettant de réaliser au moins $7/8$ des clauses. Tester l'algorithme sur l'exemple suivant

$$(c \vee a \vee b) \wedge (d \vee f \vee c) \wedge (c \vee a \vee f) \wedge (c \vee b \vee e) \wedge (a \vee \overline{c} \vee b) \wedge (c \vee \overline{a} \vee f) \wedge$$
$$(e \vee c \vee a) \wedge (d \vee \overline{a} \vee \overline{c}) \wedge (f \vee \overline{e} \vee \overline{a}) \wedge (b \vee a \vee e) \wedge (\overline{c} \vee b \vee d) \wedge (b \vee \overline{d} \vee a)$$

Annexe A

RÉSUMÉ DU COURS

A.1 Principales lois

Nom	Paramètre	Domaine	Formule	Espérance	Variance
Bernoulli	p	$\{0,1\}$	$P(X=0)=1-p, P(X=1)=p$	p	$p(1-p)$
Binomiale	(n,p)	$\{0,\dots,n\}$	$P(X=k)=\binom{n}{k}p^k(1-p)^{n-k}$	np	$np(1-p)$
Géométrique	p	$\mathbb{N}^\star$	$P(X=k)=(1-p)^{k-1}p$	$1/p$	$(1-p)/p^2$
Poisson	λ	$\mathbb{N}$	$P(X=k)=e^{-\lambda}\lambda^k/k!$	λ	λ
Uniforme		$[0,1]$	$P(X\leq t)=t$	1/2	1/12
Exponentielle	λ	$\mathbb{R}^+$	$P(X\geq t)=e^{-\lambda t}$	$1/\lambda$	$1/\lambda^2$
Normale		$\mathbb{R}$	*densité* $f_X(t)=\frac{1}{\sqrt{2\pi}}e^{-t^2/2}$	0	1

A.2 Principales inégalités

Markov : Si $X \geq 0$ et $p \geq 0$,

$$P(X \geq p) \leq \frac{E(X)}{p}$$

Chebyshev : Si $E(X) \neq \infty$ et $p \geq 0$

$$P\left(|X - E(X)| \geq p\right) \leq \frac{Var(X)}{p^2}$$

Chernov : Si S_n est une somme de n Bernoulli de paramètres p indépendantes :

$$P(|S_n - np| \geq n\lambda) \leq 2e^{-2n\lambda^2}$$

Théorème centrale limite : Si $S_n = \sum X_i$, les X_i indépendantes de même loi. Alors

$$\frac{S_n - E(S_n)}{\sqrt{Var(S_n)}} \xrightarrow{n\to\infty} Z$$

où Z est une variable aléatoire de loi normale centrée réduite.

Annexe B

DÉTAIL DE PREUVES

B.1 Espérances et Variances

B.1.1 Loi géométrique

$P(X = k) = (1 - p)^{k-1}p$.

$$E(X^2) = \sum_{k>0} k^2(1-p)^{k-1}p$$

On note $x = E(X^2)$. Alors en changeant les indices on obtient $x = \sum_{k>1}(k-1)^2(1-p)^{k-2}p$ donc $(1-p)x = \sum_{k>1}(k-1)^2(1-p)^{k-1}p = \sum_{k>0}(k-1)^2(1-p)^{k-1}p$ (le terme pour $k = 1$ est nul).

On en déduit

$$\begin{aligned} x - (1-p)x &= \sum_{k>0}(k^2 - (k-1)^2)(1-p)^{k-1}p \\ px &= \sum_{k>0}(2k-1)(1-p)^{k-1}p \\ &= 2E(X) - 1 \end{aligned}$$

D'où $px = 2/p - 1$ et finalement $E(X^2) = x = 2/p^2 - 1/p$, d'où enfin

$$Var(X) = E(X^2) - E(X)^2 = \frac{2}{p^2} - \frac{1}{p} - \frac{1}{p^2} = \frac{1-p}{p^2}$$

B.1.2 Loi de Poisson

$P[X = k] = e^{-\lambda}\frac{\lambda^k}{k!}$

$$E(X^2) = \sum_{k\geq 0} k^2 e^{-\lambda}\frac{\lambda^k}{k!}$$

On utilise la même astuce qu'au dessus

$$\begin{aligned} E(X^2) &= \sum_{k\geq 1} k e^{-\lambda} \frac{\lambda^k}{(k-1)!} \\ &= \sum_{k\geq 0} (k+1) e^{-\lambda} \frac{\lambda^{k+1}}{k!} \\ &= \lambda \left(\sum_{k\geq 0} k e^{-\lambda} \frac{\lambda^k}{k!} + \sum_{k\geq 0} e^{-\lambda} \frac{\lambda^k}{k!} \right) \\ &= \lambda(1 + E(X)) \\ &= \lambda + \lambda^2 \end{aligned}$$

D'où $Var(X) = E(X^2) - E(X)^2 = \lambda + \lambda^2 - \lambda^2 = \lambda$.

B.1.3 Exponentielle

$P(X \leq t) = 1 - e^{-\lambda t}$ de densité $f_X(t) = \lambda e^{-\lambda t}$. Par intégration par partie (on intégère $\lambda e^{-\lambda t}$ et on dérive t^2) :

$$\begin{aligned} E(X^2) &= \int_0^\infty t^2 \lambda e^{-\lambda t} dt \\ &= \left[-t^2 e^{-\lambda t} \right]_0^\infty + \int_0^\infty 2t e^{-\lambda t} dt \\ &= \frac{2}{\lambda} \int_0^\infty t e^{-\lambda t} dt \\ &= \frac{2}{\lambda} E(X) \\ &= \frac{2}{\lambda^2} \end{aligned}$$

D'où $Var(X) = E(X^2) - E(X)^2 = 1/\lambda^2$.

B.2 Loi Normale

La loi normale est de densité $f_X(t) = \frac{1}{\sqrt{2\pi}} e^{-t^2/2}$.

$$\begin{aligned} E(X) &= \int_{-\infty}^{\infty} t \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt \\ &= \left[\frac{1}{\sqrt{2\pi}} - e^{-t^2/2} \right]_{-\infty}^{+\infty} \\ &= 0 \end{aligned}$$

Vu autrement, comme X est symétrique autour de 0, son espérance, si elle existe, est nécessairement nulle. Le calcul ci-dessus montre qu'elle existe. Pour $E(X^2)$, on procède par intégration par partie, en dérivant t et en intégrant $te^{-t^2/2}$

$$\begin{aligned} E(X^2) &= \int_{-\infty}^{\infty} tt\frac{1}{\sqrt{2\pi}}e^{-t^2/2}dt \\ &= \left[\frac{1}{\sqrt{2\pi}} - te^{-t^2/2}\right]_{-\infty}^{+\infty} + \int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}}e^{-t^2/2}dt \\ &= 0+1 \end{aligned}$$

La dernière intégrale vaut 1 puisque c'est l'intégrale de la densité sur son intervalle de définition. Dit autrement

$$P(X \leq z) = \int_{-\infty}^{z}\frac{1}{\sqrt{2\pi}}e^{-t^2/2}dt$$

Et évidemment $P(X \leq +\infty) = 1$. On en déduit $Var(X) = E(X^2) - E(X)^2 = 1$

B.3 Sommes de variables exponentielles

Soit $E_0 \ldots E_n$ des variables aléatoires indépendantes exponentielles de paramètre λ.

On note $B = \min\{n|E_0 + \ldots E_n > 1\}$. On veut montrer que B suit une loi de Poisson.

On note enfin pour tout k, $S_k = E_0 + \ldots E_k$. S_k est aussi appelée distribution de Erlang de paramètre (k, λ).

On montre par récurrence sur k que

$$P(S_k \geq t) = e^{-\lambda t}\sum_{i=0}^{k}\frac{(\lambda t)^i}{i!}$$

Comme $P[B = n] = P(S_n \geq 1) - P(S_{n-1} \geq 1)$ on obtiendra directement le résultat sur la distribution de B.

C'est vrai pour $k = 1$ (la formule se simplifie et nous donne la formule bien connue pour l'exponentielle) Supposons le résultat vrai pour k.

Alors (la première ligne correspond à l'intuition, la deuxième à la preuve formelle)

$$\begin{aligned}
P(S_{k+1} \geq t) &= \int_0^\infty \text{“}P(E_{k+1} = x)'' P(S_k > t - x)dx \\
&= \int_0^\infty \lambda e^{-\lambda x} P(S_k > t - x)dx \\
&= \int_0^t \lambda e^{-\lambda x} e^{-\lambda(t-x)} \sum_{i=0}^{k} \frac{(\lambda(t-x))^i}{i!} dx + \int_t^\infty \lambda e^{-\lambda x} dx \\
&= \lambda e^{-\lambda t} \int_0^t \sum_{i=0}^{k} \frac{(\lambda(t-x))^i}{i!} dx + \left[-e^{-\lambda x}\right]_t^\infty \\
&= \lambda e^{-\lambda t} \sum_{i=0}^{k} \frac{\lambda^i}{i!} \int_0^t (t-x)^i dx + e^{-\lambda t} \\
&= \lambda e^{-\lambda t} \sum_{i=0}^{k} \frac{\lambda^i}{i!} \left[\frac{-(t-x)^{i+1}}{i+1}\right]_0^t + e^{-\lambda t} \\
&= \lambda e^{-\lambda t} \sum_{i=0}^{k} \frac{\lambda^i}{i!} \frac{t^{i+1}}{i+1} + e^{-\lambda t} \\
&= e^{-\lambda t} \sum_{i=0}^{k} \frac{\lambda^{i+1} t^{i+1}}{(i+1)!} + e^{-\lambda t} \\
&= e^{-\lambda t} \sum_{i=0}^{k+1} \frac{(\lambda t)^i}{i!}
\end{aligned}$$

Le passage de la deuxième à la troisième ligne vient du fait qu'il faut séparer le cas $t - x > 0$ (auquel cas $P(S_k \geq t - x)$ est donné par la formule de récurrence) du cas $t - x < 0$ (auquel cas $P(S_k \geq t - x) = 1$).

Annexe C

DEVOIR

Ce devoir doit être rendu avant le 6 novembre 2012, en version électronique (PDF seulement) à l'adresse `emmanuel.jeandel@loria.fr` *ou délivrée en mains propres à E. Jeandel. La question 8 du devoir nécessite d'écrire un programme. Celui-ci devra être envoyé exclusivement par mail pour la même date.*

Le devoir doit être effectué par groupe de 1 *à* 2 *personnes, avec au plus un groupe d'une seule personne, chaque personne appartenant à exactement un groupe. Chaque devoir contiendra dans une première partie au moins un paragraphe expliquant comment la répartition du travail s'est effectuée entre les différents membres d'un même groupe. Cette partie n'est facultative que pour les groupes d'une personne et sera prise en compte dans la notation.*

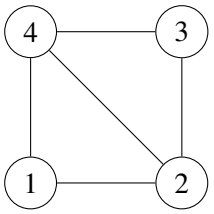
Avant de commencer, quelques résultats mathématiques qui peuvent servir :
- Si np est suffisament petit, alors $(1-1/p)^n \approx np$
- Pour tout n, $(1-1/n)^n < 1/e = 0.36$
- $1 + 1/2 + 1/3 + \cdots + 1/n \approx n \ln n$.

PARTIE 1 - Méthode MCMC

La méthode MCMC (Markov Chain Monte Carlo) est une méthode utilisée en simulation qui cherche à simuler une variable aléatoire X, à valeur disons dans un ensemble S, qu'on ne connaît pas totalement en cherchant une chaîne de Markov qui a comme ensembles d'état S et de sorte que la probabilité que X soit égal à i est exactement la même chose que la probabilité que la chaîne de Markov soit, à l'infini, dans l'état i (c'est à dire formellement que $P[X = i]$ est égal à π_i où π est la distribution stationnaire de la chaîne de Markov).

Dans ce cas, il suffira d'être capable de simuler la distribution stationnaire de la chaîne de Markov pour simuler X.

Un exemple important est le modèle hard core. Ce modèle est très important en physique statistique, et apparaît aussi dans la modélisation des réseaux. Dans ce modèle, on se donne un graphe G, potentiellement très grand. On utilisera comme exemple dans la suite le graphe G_1 suivant :



On cherche toutes les façons de colorier les sommets en noir et blanc de sorte que deux sommets noirs ne sont pas reliés entre eux. On note S l'ensemble des coloriages valides. La variable aléatoire X qui nous intéresse est celle où tous ces coloriages de S ont la même probabilité d'apparaître.

Q 1) Donner S dans le cas du graphe G_1.

En général, il est très difficile de simuler directement X si G est grand, ou même d'estimer la taille de S.

Cependant, on peut facilement le simuler par une chaîne de Markov. L'idée est la suivante. Les états de la chaîne de Markov est S. Si on est dans un coloriage C

- On choisit un sommet v du graphe au hasard uniformément.
- Avec probabilité $1/2$, on colorie v en blanc. Avec probabilité $1/2$, on le colorie en noir si c'est possible (si ca ne crée pas deux sommets voisins noirs)

Q 2) Représenter la chaîne de Markov dans le cas du graphe G_1.

Q 3) Montrer que la chaîne de Markov est irréductible et apériodique. Montrer que c'est le cas dans le cas général (et pas uniquement l'exemple particulier du graphe G_1)

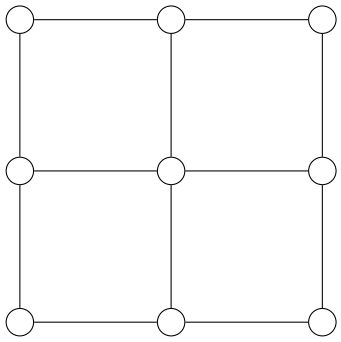
Q 4) Montrer, toujours dans le cas général, que la matrice P de la chaîne vérifie $P_{C,C'} = P_{C',C}$ (la probabilité d'aller de C à C' est la même que la probabilité d'aller de C' à C).

Q 5) Montrer qu'une chaîne de Markov finie apériodique irréductible dont la matrice P est symétrique ($P_{i,i'} = P_{i',i}$ pour tous états i, i') avec n états a une distribution stationnaire π telle que $\pi_i = 1/n$ pour tout i.

Q 6) En déduire que la distribution stationnaire de la chaîne de Markov correspondant au graphe G est bien la distribution uniforme sur S.

Q 7) On simule la chaîne de Markov, partant d'un état donné, pendant 1000000 étapes. Expliquer comment obtenir une approximation de $|S|$ (le nombre de coloriages valide)

Q 8) Ecrire un programme qui simule la chaîne de Markov pour le graphe G_2 suivant. Donner une estimation de $|S|$. On note $f(X)$ la fonction qui compte le nombre de sommets de X coloriés en noir. Donner une estimation de l'espérance de $f(X)$ (Aide : le résultat est entre 2 et 3)



Comme vous pourrez le constater, la convergence en utilisant cette méthode est très lente. Il n'y a en effet aucune raison pour que la chaîne de Markov converge rapidement vers la distribution stationnaire.

PARTIE 2 - Algorithme de Monte Carlo

Les étudiants veulent savoir si les notes de cours seront autorisées à l'examen. Lorsqu'ils demandent à l'enseignant, celui-ci répond de la façon suivante :

- Il commence par lancer une pièce (non truquée). Si elle tombe sur pile, il répond la vérité (oui s'il les notes sont autorisées, non si elles ne le sont pas)
- Si la pièce est tombé sur face, il répondu oui ou non aléatoirement avec probabilité 1/2.

On modèlise la réponse du professeur par une variable X.

Q 1) Donner la loi de X dans le cas où les notes sont autorisées, puis dans le cas où elles ne le sont pas.

Q 2) Pour préciser la réponse, un étudiant interroge n fois le professeur. Expliquer comment l'étudiant obtient ainsi la réponse, et calculer la probabilité que l"étudiant se trompe. Si l'étudiant veut avoir la bonne réponse avec probabilité 0.99, combien de fois doit-il interroger le professeur ? Justifier.

PARTIE 3 - Mélange de cartes

Dans cette partie, on cherche à mélanger un jeu de n cartes, ou encore un tableau de n cases. Mélanger un tableau est nécessaire dans la confection de beaucoup d'algorithmes probabilistes, il est donc important de savoir le faire correctement. La question semble facile, et la réponse est effectivement facile à comprendre, mais il est très facile de proposer des algorithmes faux.

On part donc d'un tableau T de n cases qui contient les nombres de 1 à n. On cherche à mélanger le tableau, c'est à dire à obtenir une variable aléatoire X sur tous les $n!$ tableaux possibles, ou tous les tableaux ont la même probabilité d'apparaître (soit $1/n!$).

Q 1) Montrer que l'algorithme suivante ne mélange pas correctement un tableau (Aide : il suffit de le prouver pour $n = 3$)

```
Pour i allant de 1 à n
  Echanger (T[i], T[nextInt(n)])
```

On considère maintenant l'algorithme suivant, qui renvoie dans S un mélange du tableau T. On utilise un tableau auxiliaire U initialisé à 0.

```
j = 0;
Tant que j != n:
   k = nextInt(n)
   si U[k] == 0:
     S[j] = T[k]
     j = j + 1
     U[k] = 1
   sinon:
     rien
```

Q 2) Expliquer brièvement ce que fait l'algorithme et pourquoi il fonctionne.

Q 3) Dans le meilleur des cas, combien de temps (nombre de passage dans la boucle) met l'algorithme avant de répondre ? Dans le pire des cas ?

Q 4) On note T la variable aléatoire correspondant au temps d'exécution de cet algorithme. Donner $E(T)$. On pourra pour cela introduire des variables aléatoires auxiliaires T_i tel que T_i représente le temps passé entre le moment où $j = i$ et le moment où $j = i + 1$.

Q 5) On se donne maintenant un entier m et on note maintenant A_i l'évènement "Aucun des m premiers tirages de nextInt() n'a renvoyé i". Calculer $P(A_i)$. En déduire que

$$P(T > m) \leq ne^{-m/n}$$

On considère maintenant un nouvel algorithme, qu'on exprime en terme de cartes plutôt que de tableau. L'idée est la suivante : A chaque étape, on prend la première carte et on l'insère au hasard dans le paquet.

On suppose qu'au départ les cartes sont triées, de sorte que la première carte soit l'as de coeur et la dernière le roi de pique.

Q 6) Montrer qu'au moment où le roi de pique est en tête du paquet, le reste du paquet est correctement mélangé.

Q 7) On note T le moment où le roi de pique est en tête du paquet. A l'instant $T + 1$, le tableau est donc correctement mélangé.. Calculer $E(T+1)$ (Aide : s'inspirer fortement d'une question précédente).

Q 8) Expliquer brièvement pourquoi si on s'arrête à un nombre d'étapes n fixé, le jeu de cartes n'est pas mélangé correctement

Le bon algorithme à utiliser pour mélanger un tableau est l'algorithme de Fisher-Yates, décrit comme suit :

```
Pour i de n - 1 à 1
  j = nextInt(i+1)
  echange(T[i],T[j])
```

Q 9) Montrer que l'algorithme est correct. Pour cela, montrer (a) que toutes les permutations sont possibles (b) que l'algorithme ne peut se dérouler que de $n!$ façons différentes. (c) que (a) et (b) vous permettent de conclure.

PARTIE 4 - Pour finir

Q 1) Montrer que la somme de deux Poisson de paramètre α et β est aussi Poisson. Trouver son paramètre.

Q 2) Montrer qu'on peut simuler la loi géométrique de paramètre p par $\lfloor -\ln U / \ln(1-p) \rfloor$ où $\lfloor x \rfloor$ désigne la partie entière de x et U est une variable aléatoire uniforme sur $[0, 1]$. Expliquer d'où provient cette formule.

Q 3) Soit $X_1 \dots X_n$ des variables aléatoires indépendantes de loi géométrique de paramètre p. On pose $N = \max\{k | (X_1 + 1) + \dots (X_k + 1) \leq n\}$. Calculer la loi de N. (Aide : commencer par calculer $P(N = n)$, $P(N = n - 1)$, $P(N = n - 2)$ et essayer de reconnaître une loi connue.)