TD 5

Exercice 1 On tire les cartes une apres l'autre parmi les 52 cartes et chaque fois on essaie de deviner leur valeur. Utilisez des variables indicatrices pour calculer le nombre esperée des cartes correctement devines si
(a) on ne se souvient pas des cartes déjà sorties ;
(b) on se souviens des cartes déjà sorties.

Correction :

(a) Si on ne se souvient pas des cartes déjà sorties, alors chaque fois qu'on tire une carte, il y a une probabilité de 1/52 de deviner correctement sa valeur, car il y a 52 cartes dans le jeu et elles sont toutes équiprobables. Donc, la variable aléatoire X, qui représente le nombre de cartes correctement devinées, suit une distribution binomiale avec n = 52 (le nombre total de cartes dans le jeu) et p = 1/52 (la probabilité de deviner correctement une carte).

L'espérance de la distribution binomiale est donnée par la formule E(X) = n * p, donc dans ce cas, E(X) = 52 * (1/52) = 1. Cela signifie que, en moyenne, on devinera correctement une carte.

(b) Si on se souvient des cartes déjà sorties, alors la probabilité de deviner correctement la valeur d'une carte dépend des cartes déjà tirées. Au début, la probabilité de deviner correctement la première carte est de 1/52, comme dans le cas précédent. Ensuite, après avoir tiré une carte, on connaît déjà sa valeur, donc la probabilité de deviner correctement la deuxième carte est de 1/51, et ainsi de suite.

En général, la probabilité de deviner correctement la i-ème carte, sachant les i-1 cartes déjà tirées, est de 1/(52 - i + 1), car il reste 52 - i + 1 cartes non tirées dans le jeu. Donc, la variable aléatoire X suit une distribution hypergéométrique, où n = 52 (le nombre total de cartes dans le jeu), k = 52 (le nombre de cartes à deviner) et N = 52 (le nombre total de cartes).

L'espérance de la distribution hypergéométrique est donnée par la formule E(X) = n * (k/N), donc dans ce cas, E(X) = 52 * (52/52) = 52. Cela signifie que, en moyenne, on devine correctement 52 cartes, car on se souvient des cartes déjà sorties.

Exercice 2 (vestiaire à chapeaux) Utilisez des variables indicatrices pour résoudre le problème suivant, connu sous le nom de problème du *vestiaire à chapeaux*. Chaque client parmi n au total donne son chapeau à un employé d'un restaurant. Cet employé redonne les chapeaux aux clients dans un ordre aléatoire. Quel est le nombre attendu de clients qui récupéreront leurs chapeaux ?

Correction :

Dans le problème du vestiaire à chapeaux, il y a un client au total, et chaque client donne son chapeau à un employé du restaurant. Ensuite, les chapeaux sont rendus aux clients dans un ordre aléatoire. On veut calculer le nombre attendu de clients qui récupéreront leurs propres chapeaux.

Pour résoudre ce problème, nous pouvons utiliser des variables indicatrices pour chaque client. Soit X_i une variable indicatrice pour le i-ème client, où X_i = 1 si le i-ème client récupère son propre chapeau et X_i = 0 sinon.

La probabilité que le i-ème client récupère son propre chapeau dépend de l'ordre dans lequel les chapeaux sont rendus aux clients. Au début, la probabilité que le premier client récupère son propre chapeau est de 1/n, car il y a n chapeau et ils sont tous équiprobables d'être rendus au premier client. Ensuite, après que le premier client ait récupéré son chapeau, la probabilité que le deuxième client récupère son propre chapeau est de 1/(n-1), car il reste n-1 chapeaux et ils sont tous équiprobables d'être rendus au deuxième client. Et ainsi de suite pour les clients suivants.

En général, la probabilité que le i-ème client récupère son propre chapeau, sachant que les i-1 clients précédents ont déjà récupéré leurs chapeaux, est de 1/(n-i+1), car il reste n-i+1 chapeaux et ils sont tous équiprobables d'être rendus au i-ème client.

Maintenant, nous pouvons calculer l'espérance du nombre de clients qui récupèrent leurs propres chapeaux en utilisant les variables indicatrices. L'espérance du nombre de clients qui récupèrent leurs chapeaux est donnée par la somme des espérances des variables indicatrices, c'est-à-dire E(X_1 + X_2 + ... + X_n).

En utilisant la propriété d'additivité de l'espérance, nous pouvons réarranger la somme comme suit :

E(X_1 + X_2 + ... + X_n) = E(X_1) + E(X_2) + ... + E(X_n)

Maintenant, pour chaque client i, nous savons que la probabilité que le i-ème client récupère son propre chapeau est de 1/(n-i+1). Donc, E(X_i) = 1/(n-i+1), pour i allant de 1 à n.

En sommant les espérances des variables indicatrices, nous obtenons :

E(X_1 + X_2 + ... + X_n) = E(X_1) + E(X_2) + ... + E(X_n)
= 1/(n-1+1) + 1/(n-2+1) + ... + 1/(n-n+1)
= 1/n + 1/(n-1) + ... + 1/2 + 1/1

Cette somme est connue sous le nom de la série harmonique H_n, et elle est définie comme la somme des inverses des entiers de 1 à n. La série harmonique H_n est bien connue en mathématiques et elle ne possède pas de formule fermée simple

---

Exercice 3 (coupon collecteur) Un collectionneur cherche à avoir toutes les vignettes d'une série de n vignettes distribuées aléatoirement dans des boîtes de céréales. Combien faut-il acheter de boîtes de céréales en moyenne pour avoir la collection complète ? *Indication : On pourra cher- cher à estimer l'espérance de la variable aléatoire T_i décrivant le nombre d'achats nécessaires pour obtenir la i-ème nouvelle vignette, en utilisant le résultat du cours concernant le nombre moyen de tirages d'une expérience de Bernoulli avant le premier succès, c'est-à-dire l'espérance d'une variable suivant une loi géométrique. On utilisera également les nombres harmoniques*

$H_n = \sum_{i=1}^{n} 1/i = \ln n + \gamma + o(1)$ *avec* $\gamma \approx 0,577$ *la constante d'Euler-Mascheroni.*

Correction :

Dans le problème du coupon collecteur, le collectionneur cherche à obtenir toutes les n vignettes d'une série distribuées aléatoirement dans des boîtes de céréales. On veut calculer combien en moyenne il faut acheter de boîtes de céréales pour compléter sa collection.

Pour résoudre ce problème, nous pouvons utiliser la notion de variables aléatoires géométriques. Soit T_i une variable aléatoire décrivant le nombre d'achats nécessaires pour obtenir la i-ème nouvelle vignette, c'est-à-dire le nombre d'achats nécessaires pour obtenir une nouvelle vignette qui n'a pas encore été obtenue.

Selon le résultat du cours concernant le nombre moyen de tirages d'une expérience de Bernoulli avant le premier succès, l'espérance d'une variable aléatoire géométrique suivant une loi géométrique de paramètre p (où p est la probabilité de succès) est donnée par E(T_i) = 1/p.

Dans le cas du coupon collecteur, la probabilité de succès pour obtenir la i-ème nouvelle vignette est de (n - i + 1)/n, car il reste n - i + 1 vignettes qui n'ont pas encore été obtenues sur un total de n vignettes.

Ainsi, nous pouvons écrire l'espérance de T_i comme suit :

E(T_i) = 1/p = 1/((n - i + 1)/n) = n/(n - i + 1)

Maintenant, nous pouvons estimer l'espérance du nombre total d'achats nécessaires pour compléter la collection en sommant les espérances des variables aléatoires T_i pour i allant de 1 à n. Cela peut être exprimé comme suit :

E(T_1 + T_2 + ... + T_n) = E(T_1) + E(T_2) + ... + E(T_n)
= n/(n - 1 + 1) + n/(n - 2 + 1) + ... + n/(n - n + 1)
= n/(n - 0 + 1) + n/(n - 1 + 1) + ... + n/(n - (n - 1) + 1)
= n/(1) + n/(2) + ... + n/(n)

En utilisant les nombres harmoniques H_n définis comme la somme des inverses des entiers de 1 à n, nous pouvons réécrire la somme comme suit :

n/(1) + n/(2) + ... + n/(n) = n * (1/1 + 1/2 + ... + 1/n) = n * H_n

Selon l'indication donnée dans l'énoncé, nous pouvons utiliser l'approximation des nombres harmoniques H_n par ln n + γ + o(1), où ln n est le logarithme naturel de n et γ est la constante d'Euler-Mascheroni approximée à 0,577.

Ainsi, nous pouvons écrire l'estimation de l'espérance du nombre total d'achats nécessaires pour compléter la collection comme suit :

E(T_1 + T_2 + ... + T_n) ≈ n * (ln n + γ)

Donc, en moyenne, il faut environ n * (ln n + γ) achats de boîtes de céréales pour compléter la collection de n vignettes, en utilisant l'approximation des nombres harmoniques.

Il est important de noter que cette estimation est basée sur des approximations et peut ne pas être exacte pour de petites valeurs de n. Cependant, elle donne une estimation raisonnable de la quantité moyenne d'achats nécessaires pour compléter la collection dans le cas où n est grand.

En résumé, le nombre moyen d'achats de boîtes de céréales nécessaires pour compléter la collection de n vignettes dans le problème du coupon collecteur est approximativement donné par n * (ln n + γ), où ln n est le logarithme naturel de n et γ est la constante d'Euler-Mascheroni.

Exercice 4 (test de deux polynômes) Considérons un algorithme pour tester si deux polynômes P et Q de meme degré d sont identiques. Par exemple, on cherche à savoir si $P = (X+1)(X-2)(X+3)(X-4)(X+5)(X-6)$ est egal à $Q = X^6 - 7X^3 + 25$. L'algorithme choisit un entier aleatoire r dans l'intervalle $\{1, \ldots, 100d\}$ et calcule $P(r)$ et $Q(r)$. Si $P(r) = Q(r)$, alors l'algorithme décide que les deux polynômes ne sont pas identiques. Si $P(r) = Q(r)$, alors l'algorithme décide que les deux polynômes sont identiques.

(a) Quelle est la probabilité que l'algorithme retourne une mauvaise réponse (c'est-à-dire qu'il décide que P et Q sont identiques, alors même qu'ils ne le sont en fait pas) ? (*Indication : Utiliser le fait qu'un polynôme de degré d possède au plus d racines.*) Comment faire pour diminuer la probabilité d'échec ?

(b) Supposons qu'on répète le tirage k fois (sans prendre soin de ne pas tester plusieurs fois la même valeur test r) et on retourne P et Q sont identiques si et seulement si $P(r_i) = Q(r_i)$ pour $i = 1, \ldots, k$. Quelle est la probabilité de retourner une mauvaise réponse ?

(c) Que devient cette probabilité si on prend soin de ne pas tester plusieurs fois la même valeur test r ?

Correction :

(a) La probabilité que l'algorithme retourne une mauvaise réponse, c'est-à-dire qu'il décide que les polynômes P et Q sont identiques alors qu'ils ne le sont pas en réalité, dépend du nombre de racines communes que P et Q peuvent avoir. Un polynôme de degré d possède au plus d racines distinctes. Donc, si P et Q ont d racines communes, alors la probabilité que P(r) = Q(r) pour un r aléatoire est de 1/d, et la probabilité que P(r) ≠ Q(r) est de (d-1)/d.

Pour diminuer la probabilité d'échec de l'algorithme, on peut augmenter le nombre de valeurs testées k. Plus k est grand, plus la probabilité d'obtenir une mauvaise réponse diminue. On peut également augmenter la plage des valeurs aléatoires choisies pour r, par exemple en choisissant r dans l'intervalle {1, ..., 1000d} au lieu de {1, ..., 100d}, pour augmenter les chances de trouver une racine commune.

(b) Si on répète le tirage k fois sans prendre soin de ne pas tester plusieurs fois la même valeur r, la probabilité de retourner une mauvaise réponse dépendra du degré de corrélation entre les valeurs de r choisies. Si les valeurs de r sont indépendantes les unes des autres, alors la probabilité de retourner une mauvaise réponse sera égale à la probabilité qu'une seule valeur de r donne une mauvaise réponse, soit (d-1)/d.

(c) Si on prend soin de ne pas tester plusieurs fois la même valeur r, la probabilité de retourner une mauvaise réponse sera diminuée. En évitant de tester plusieurs fois la même valeur r, on élimine les cas où une racine commune aurait été trouvée plusieurs fois, ce qui augmenterait la probabilité de fausse identification des polynômes comme identiques. La probabilité dépendra alors du nombre de racines communes entre P et Q et du nombre total de valeurs r testées.a

Exercice 5 (tri rapide) Montrer que le nombre espéré de comparaisons du l'algorithme de tri rapide sur un tableau de n éléments est de $O(n \log n)$. *Indication : Soit $b_1 < b_2 <$ $< b_n$ les éléments du tableau, dans l'ordre croissant. On considère une distribution aléatoire uniforme sur l'ensemble des permutations possibles de ces n éléments. Soit X_{ij} la variable aleatoire définie par $X_{ij} = 1$ si les éléments b_i et b_j sont comparés par le tri rapide dans le tableau correspondant, et $X_{ij} = 0$ sinon. Calculer $\mathbf{E}(X_{ij})$. Noter que si $X = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}$, alors $\mathbf{E}(X)$ est le resul´ tat souhaité, puis conclure.*

Correction :

L'algorithme de tri rapide (quicksort) est un algorithme de tri récursif qui choisit un pivot dans un tableau d'éléments, partitionne le tableau en deux sous-tableaux autour du pivot, puis trie récursivement les sous-tableaux. La complexité de l'algorithme de tri rapide dépend du choix du pivot et de la distribution des éléments dans le tableau.

Pour montrer que le nombre espéré de comparaisons dans le tri rapide est de O(n log n), on peut utiliser une approche probabiliste. On considère une distribution aléatoire uniforme sur l'ensemble des permutations possibles des n éléments du tableau. Soit b1 < b2 < ... < bn les éléments du tableau dans l'ordre croissant.

On définit une variable aléatoire Xij pour chaque paire d'indices i et j, où $1 \leq i < j \leq n$, telle que Xij = 1 si les éléments bi et bj sont comparés par le tri rapide dans le tableau correspondant, et Xij = 0 sinon.

Pour calculer l'espérance E(Xij), on peut observer que bi et bj sont comparés par le tri rapide si et seulement si l'un des deux est choisi comme pivot à un certain point de l'algorithme. Le pivot est choisi de manière aléatoire et uniforme parmi les éléments du sous-tableau en cours de traitement. Donc, la probabilité que bi et bj soient comparés est de 1/(j - i + 1), car il y a j - i + 1 éléments possibles parmi lesquels choisir le pivot.

Maintenant, on peut calculer E(X) comme la somme des espérances de toutes les variables aléatoires Xij, où i varie de 1 à n-1 et j varie de i+1 à n:

E(X) = Σ [1/(j - i + 1)] (pour i allant de 1 à n-1, et j allant de i+1 à n)

En utilisant cette formule, on peut montrer que E(X) est de l'ordre de O(n log n). En effet, en faisant des calculs plus détaillés, on peut montrer que $E(X) \leq 2n \log n$, ce qui signifie que le nombre espéré de comparaisons dans le tri rapide est de O(n log n). Cela montre que, en moyenne, l'algorithme de tri rapide effectue un nombre de comparaisons proportionnel à n log n pour trier un tableau de n éléments dans le cas d'une distribution aléatoire uniforme des éléments dans le tableau.