Des nombres de Fibonacci à Google

Le calcul des nombres de Fibonacci par la formule de Binet, qui en donne une expression en fonction de l'indice et du nombre d'or, peut être fait en diagonalisant une matrice 2×2. Le même type de calcul donne des expressions explicites des suites de probabilités dans les « graphes probabilistes » à deux états que l'on voit en terminale ES, ainsi bien sûr que l'expression de l'état stable. Dans ce texte, on développe à l'identique les deux calculs pour montrer leur similitude. Puis on étend –sans preuves– les deux problèmes aux suites satisfaisant une récurrence linéaire et à (une version simple de) l'algorithme PageRank utilisé par Google pour hiérarchiser les pages Internet.

1 Calcul des nombres de Fibonacci

La suite de Fibonacci $(f_n)_{n\in\mathbb{N}}$ est, comme nul n'est censé ignorer, définie par :

$$f_0 = F_1 = 1, \qquad \forall n \in \mathbb{N}, \quad f_{n+2} = f_n + f_{n+1}.$$

On peut retrouver la formule de Binet, qui exprime les nombres de Fibonacci en fonction du nombre d'or, de la façon suivante. On commence par introduire une suite de vecteurs de RM^2 :

$$\forall n \in \mathbb{N}, \quad F_n = \begin{pmatrix} f_n \\ f_{n+1} \end{pmatrix},$$

puis on récrit la relation de récurrence sous la forme :

$$\forall n \in \mathbb{N}, \quad F_{n+1} = AF_n, \qquad \text{où } A = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

On reconnaît essentiellement une suite géométrique. Sa raison est une matrice mais ça n'est pas plus difficile de prouver par récurrence que l'on a :

$$\forall n \in \mathbb{N}, \quad F_n = A^n F_0.$$

Pour calculer les puissances de A, on va la diagonaliser. Conceptuellement, la multiplication par A correspond à l'action de l'unique application linéaire φ qui envoie[1] le vecteur $(1,0)$ de $\mathbb{R}^2$ sur $(0,1)$ et le vecteur $(0,1)$ sur $(1,1)$. Il s'agit de trouver une base dans laquelle la matrice de φ est aussi simple que possible. On va trouver deux vecteurs propres, c'est-à-dire deux axes sur lesquels l'application φ est la multiplication par un scalaire. Un tel vecteur $X = (x, y)$ de $\mathbb{R}^2$ est caractérisé par l'équation

$$AX = \lambda X,$$

où λ est un scalaire convenable. Cette équation vectorielle se traduit par un système :

$$\begin{cases} y &= \lambda x \\ x + y &= \lambda y. \end{cases}$$

1. L'application φ envoie le j^{e} vecteur de la base canonique sur le vecteur dont la colonne des coordonnées est la j^{e} colonne de la matrice A, pour tout entier j compris entre 1 et la dimension de l'espace sur lequel on travaille –ici, 2...

Ce sytème équivaut à :

$$\begin{cases} y = \lambda x \\ (\lambda^2 - \lambda - 1)x = 0. \end{cases}$$

On cherche évidemment une solution (x, y) non nulle, ce qui impose à λ d'être une des deux racines du polynôme caractéristique de A, qui est aussi l'équation caractéristique de la relation de récurrence initiale. On la résout : $\lambda = (1 \pm \sqrt{5})/2$. Conformément aux usages, on va poser $\phi = (1 + \sqrt{5})/2$ et $\overline{\phi} = (1 - \sqrt{5})/2$.
Les vecteurs $(1, \phi)$ et $(1, \overline{\phi})$ ne sont pas colinéaires, ils forment une base de $\mathbb{R}^2$. On les regroupe dans une matrice :

$$P = \begin{pmatrix} 1 & 1 \\ \phi & \overline{\phi} \end{pmatrix}.$$

La théorie ou une vérification facile permettent de montrer la relation :

$$A = PDP^{-1}, \qquad \text{où} \quad D = \begin{pmatrix} \phi & 0 \\ 0 & \overline{\phi} \end{pmatrix},$$

qui à son tour donne une expression des puissances de A :

$$\forall n \in \mathbb{N}, \quad A^n = PD^nP^{-1} = P \begin{pmatrix} \phi^n & 0 \\ 0 & \overline{\phi}^n \end{pmatrix}.$$

Le calcul de l'inverse de P peut être proposé à des élèves de seconde, quitte à l'écrire sous forme de système à résoudre :

$$P^{-1} = \frac{1}{\overline{\phi} - \phi} \begin{pmatrix} \overline{\phi} & -1 \\ -\phi & 1 \end{pmatrix} = \frac{1}{5} \begin{pmatrix} 3 - \phi & 2\phi - 1 \\ \phi + 2 & -2\phi + 1 \end{pmatrix}.$$

Quelques lignes de calcul donnent :

$$\forall n \in \mathbb{N}, \quad A^n = \cdots$$

et c'est exactement la formule de Binet.

2 Graphes probabilistes en terminale ES

On considère un système qui peut être dans deux états, notés A et B. Le temps est discret et représenté par un entier naturel. D'un moment à l'autre, le système peut passer d'un état à l'autre. Il est décrit par deux réels p et q. On note p la probabilité de transition de A à B. Cela signifie que la probabilité que le système passe dans l'état B à une étape, sachant qu'il était dans l'état A à l'étape précédente. De même, on note q la probabilité de transition de B à A. Par commodité, nous allons supposer que p et q ne valent ni 0, ni 1.
À chaque étape n, le système a une probabilité a_n d'être dans l'état A et b_n dans l'état B. La somme de ces deux probabilités est 1 mais nous n'utiliserons pas cette relation avant longtemps. Par disjonction des cas, formule de Bayes et tout ce qu'on voudra, les suites (a_n) et (b_n) satisfont aux relations de récurrence suivantes :

$$\forall n \in \mathbb{N}, \qquad \begin{cases} a_{n+1} &= (1 - p)a_n + qb_n \\ b_{n+1} &= pa_n + (1 - q)b_n \end{cases}$$

Pour une raison obscure, la tradition en terminale ES consiste à introduire pour tout entier n un vecteur-ligne $L_n = (a_n \quad b_n)$ et à écrire ces relations sous la forme

$$\forall n \in \mathbb{N}, \quad L_{n+1} = L_n T, \qquad \text{où } \ T = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}.$$

Dans tous les manuels d'algèbre linéaire, on introduit un vecteur-colonne X_n, transposé du précédent, et la relation devient :

$$\forall n \in \mathbb{N}, \quad X_{n+1} = AX_n, \qquad \text{où } X_n = \begin{pmatrix} a_n \\ b_n \end{pmatrix} \text{ et } A = \begin{pmatrix} 1-p & q \\ p & 1-q \end{pmatrix}.$$

> **Aparté.** Ce sont évidemment deux formulation équivalentes, on passe de l'une à l'autre à l'aide de la transposition des matrices. La transposition est l'application linéaire qui envoie une matrice rectangulaire de format $k \times \ell$ où k et ℓ sont des entiers non nuls, disons $A = (a_{ij})_{(i,j)\in\{1,\dots,k\}\times\{1,\dots,\ell\}}$, sur la matrice de format $\ell \times k$ suivante : ${}^tA = (b_{ji})_{(j,i)\in\{1,\dots,\ell\}\times\{1,\dots,k\}}$, où $b_{ji} = a_{ij}$ pour tout couple (i, j). Les vecteurs lignes ou colonnes sont des matrices rectangulaires ayant respectivement une seule colonne ou ligne. La transposition se comporte bien vis-à-vis du produit : si A est une matrice $k \times \ell$ et B est une matrice $\ell \times m$, alors on peut calculer le produit AB et sa transposée ${}^t(AB)$ est égale au produit ${}^tB{}^tA$.
> Dans notre cas très précis, on a : $X_n = {}^tL_n$ pour tout n et $A = {}^tT$, si bien que les relations $L_{n+1} = L_nT$ et $X_{n+1} = AX_n$ sont équivalentes.
> Une bonne raison de préférer l'écriture $X_{n+1} = AX_n$, c'est que le produit AX d'une matrice A par un vecteur-colonne X correspond (via une application « coordonnées ») à l'application $\varphi(v)$ d'une transformation φ à un vecteur v. Le produit par une matrice à droite correspondrait à une écriture de la forme $(v)\varphi$.

On en déduit comme d'habitude que pour tout entier n, on a :

$$X_n = A^n X_0.$$

Pour calculer les puissances de A, diagonalisons-la comme nous l'avons fait pour les nombres de Fibonacci. Comme ci-dessus, on cherche pour quels scalaires λ le système $AX = \lambda X$ possède une solution non nulle $X = (x, y)$; bien sûr, ce système pourrait être obtenu à partir de la relation $XT = \lambda X$:

$$\begin{cases} (1-p)x & +qy & = & \lambda x \\ px & +(1-q)y & = & \lambda y \end{cases}$$

Il est nécessaire et suffisant que λ soit une racine du polynôme caractéristique de A, $\chi(\lambda) = \det(A - \lambda\mathrm{id})$, où id est la matrice-identité. C'est un polynôme de degré 2 donc il a deux racines, éventuellement complexes ou confondues. On les détermine par un calcul direct ou on utilise l'aparté ci-dessous.

> **Aparté.** Le fait que les lignes de T ont une somme égale à 1 s'interprète comme le fait que le vecteur $V = {}^t(1 \quad 1)$ est un vecteur propre « à gauche » de T : $TV = V$. Cette remarque signifie que le système $TV = \lambda V$ a une solution non nulle lorsque $\lambda = 1$, si bien que 1 est une racine du polynôme $\chi(\lambda) = \det(T - \lambda\mathrm{id})$ (oui, le même χ que pour A...). Ainsi, 1 est une valeur propre de A.

Bien que la valeur propre 1 nous ait en quelque sorte sauté aux yeux, le vecteur propre correspondant, c'est-à-dire la solution de $AX = X$, lui, n'est pas donné par la manipulation. En effet, le système $TV = V$ donne une solution du système $LA = A$ qui est différent de $AX = X$...

L'autre racine de χ est le quotient du terme constant par la racine évidente, c'est donc

$$\delta = \det(A)/1 = 1 - p - q.$$

Notons que puisque l'on a supposé que p et q étaient dans l'intervalle $]0,1[$, cette deuxième racine δ appartient à $]-1,1[$.

Ayant les valeurs propres, on trouve les vecteurs propres en résolvant les systèmes $AX = X$ et $AX = \delta X$. On trouve deux solutions $X_1 = (q,p)$ pour le premier système et $X_\delta = (1,-1)$ pour le second. On regroupe ces solutions dans une matrice :

$$P = \begin{pmatrix} q & 1 \\ p & -1 \end{pmatrix},$$

dont l'inverse est :

$$P^{-1} = \frac{1}{p+q}\begin{pmatrix} 1 & 1 \\ p & -q \end{pmatrix}$$

et on vérifie que la relation suivante est satisfaite :

$$A = PDP^{-1}, \qquad \text{où } D = \begin{pmatrix} 1 & 0 \\ 0 & \delta \end{pmatrix} \text{ et } \delta = 1 - p - q.$$

On en déduit après calculs que pour tout entier n :

$$A^n = PD^nP^{-1} = P\begin{pmatrix} 1 & 0 \\ 0 & \delta^n \end{pmatrix}P^{-1} = \frac{1}{p+q}\begin{pmatrix} q + p\delta^n & q - q\delta^n \\ p - p\delta^n & p + q\delta^n \end{pmatrix}.$$

Remarquons que le vecteur X_∞ est, à une constante multiplicative près, la première colonne de P, c'est-à-dire le vecteur propre de A correspondant à la plus grande valeur propre. Ce phénomène sera exploité dans un cadre plus général.

On peut en déduire une expression explicite des suites (a_n) et (b_n). Puisque δ est strictement compris entre -1 et 1, lorsque n tend vers l'infini, la suite (δ^n) tend vers 0 et (A^nX_0) tend vers le vecteur :

$$X_\infty = \frac{1}{p+q}\begin{pmatrix} q(a_0 + b_0) \\ p(a_0 + b_0) \end{pmatrix}.$$

Vu l'hypothèse initiale, $a_0 + b_0 = 1$, cette expression se simplifie et prouve que l'état stable est indépendant des conditions initiales et vaut :

$$L = (a_\infty \quad b_\infty) = \left(\frac{q}{p+q} \quad \frac{p}{p+q}\right).$$

3 Suites récurrentes linéaires et algorithme PageRank

Suites récurrentes linéaires
On se donne un entier naturel non nul d et des complexes $a_0, \dots, a_{d-1}$. On s'intéresse aux suites complexes $(x_n)_{n\in\mathbb{N}}$ satisfaisant à la relation :

$$\forall n \in \mathbb{N}, \quad x_{n+d} = a_{d-1}x_{n+d-1} + \cdots + a_1 x_{n+1} + a_0 x_0.$$

Pour obtenir une expression explicite de (x_n) en fonction de n, on définit une suite de vecteurs (X_n) de $\mathbb{C}^d$ par :

$$\forall n \in \mathbb{N}, \quad X_n = {}^t\!\begin{pmatrix} x_n & x_{n+1} & \cdots & x_{n+d-2} & x_{n+d-1} \end{pmatrix}.$$

Elle satisfait la récurrence d'ordre 1 suivante :

$$\forall n \in \mathbb{N}, \quad X_{n+1} = AX_n, \qquad \text{où} \;\; A = \begin{pmatrix} 0 & & & & a_0 \\ 1 & 0 & & & a_1 \\ & \ddots & \ddots & & \vdots \\ & & \ddots & 0 & a_{d-2} \\ & & & 1 & a_{d-1} \end{pmatrix}.$$

(Dans cette matrice carrée de format $d \times d$, les coefficients qui ne sont pas écrits sont nuls.)
On en déduit comme avant que $X_n = A^n X_0$ pour tout entier n, ce qui conduit à chercher une expression simple des puissances de A. Pour cela, on diagonalise A.
La recherche de valeurs propres de A, des scalaires λ tels que le système $AX = \lambda X$ possède une solution non nulle X dans $\mathbb{C}^d$, est essentiellement équivalente à la recherche de suites géométriques satisfaisant à la relation de récurrence. Les seules valeurs possibles sont les racines de l'équation caractéristique :

$$\lambda^n = a_{d-1}\lambda^{d-1} + \cdots + a_1\lambda + a_0.]$$

Pour chacune de ces racines, on trouve une suite géométrique solution (λ^n) dont les premiers termes forment un vecteur propre $X_\lambda = (\lambda^{k-1})_{k=1,\dots,d} \in \mathbb{C}^d$. La situation est particulièrement agréable lorsque les d racines complexes $\lambda_1, \dots, \lambda_d$ de l'équation caractéristique sont distinctes. Dans ce cas, les vecteurs propres correspondants forment une base de $\mathbb{C}^d$ et on peut construire une matrice inversible :

$$P = (\lambda_j^{i-1})_{(i,j)\in\{1,\dots,d\}\times\{1,\dots,d\}} = \begin{pmatrix} 1 & \cdots & \cdots & 1 \\ \lambda_1 & \cdots & \cdots & \lambda_d \\ \vdots & & & \vdots \\ \lambda_1^{d-1} & \cdots & \cdots & \lambda_d^{d-1} \end{pmatrix}.$$

> **Aparté :** On reconnaît la matrice de Vandermonde dont le déterminant est le produit des différences $\lambda_i - \lambda_j$ pour $i > j$. Notons au passage que pour des bonnes valeurs des λ_i, cette matrice est la matrice de la transformée de Fourier discrète, cruciale dans l'algorithme FFT de multiplication rapide.

Si D désigne la matrice diagonale dont les coefficients diagonaux sont les λ_i, on a la relation :

$$\forall n \in \mathbb{N}, \quad A^n = PD^nP^{-1},$$

qui donne une expression de x_n en fonction des λ_i, de n et de $(x_0, \dots, x_{d-1})$.

Aparté : La deuxième épreuve de l'écrit du CAPES 2010 propose une tout autre approche de ce calcul, fondée sur les séries entières. Il est amusant de faire le lien entre les deux approches.

Algorithme PageRank de Google

Comme chacun sait, les pages correspondant à une requête faite sur le moteur de recherche Google arrivent dans un ordre qui n'est pas aléatoire. C'est ce qui a rendu le moteur si célèbre. Cet ordre reflète une hiérarchie définie par le résultat de l'algorithme PageRank. Cet algorithme prend en entrée le web à un moment donné et produit une liste de « scores », un réel compris entre 0 et 1 pour chaque page du web. À chaque requête, les pages sont présentées par ordre décroissant de score.
L'idée de l'algorithme, c'est de marcher au hasard sur le web et de compter combien de fois on passe sur chaque page. Le *credo*, c'est que plus on revient souvent sur une page, plus elle est pertinente. L'approche fréquentiste des probabilités suggère que les fréquences de passage sur chaque page après une longue marche sont les probabilités de s'y trouver. Pour cela, la théorie des graphes probabilistes, que l'on peut appeler plus pompeusement la théorie des chaînes de Markov finies et qui est fondée sur les probabilités et un peu d'algèbre linéaire, donne une façon concrète de calculer ces probabilités.

Formalisons cette idée. On représente Internet par un graphe orienté dont les sommets sont les pages internet et les flèches, les liens hypertexte. Plus précisément, on numérote les pages web de 1 à N, où N est le (très grand) nombre de pages du web (recensées par Google) ; les sommets du graphe sont les éléments de $\{1, \ldots, N\}$. On met autant de flèches du sommet j vers le sommet i qu'il y a de liens de la page correspondant à j vers celle qui correspond à i. On construit une matrice A_0, de taille $N \times N$, dont le coefficient d'indices (i, j) est le nombre de liens de la page j vers la page i divisé par le nombre total de liens issus de la page j. Ainsi, la somme des coefficients de chaque colonne de A_0 vaut 1.
Notons $p_i^{(n)}$ la probabilité d'être à la page i à l'instant n. Les $p_i^{(n)}$ forment un vecteur X_n de $\mathbb{R}^N$ et la propriété de marche au hasard se traduit par la récurrence :

$$\forall n \in \mathbb{N}, \quad X_{n+1} = A_0 X_n.$$

Cette récurrence n'est autre que la formule de Bayes appliquée autant de fois qu'il y a de pages web. En quelque sorte, les graphes probabilistes de terminale ES modélisent un réseau Internet avec deux ou trois sites. C'est peu réaliste, mais les idées importantes (probabilité de transition, état stable) y apparaissent déjà clairement.
En fait, pour éviter les impasses dans le graphe, pour prendre en compte le comportement consistant à sauter d'une page à une autre en tapant l'adresse dans la barre d'adresse du navigateur, mais aussi pour des raisons de convergence numérique de l'algorithme, on modifie un peu la matrice A_0 en la remplaçant par

$$A = tA_0 + (1 - t)J,$$

où t vaut environ $0,8$ et J est la matrice dont tous les coefficients valent $1/N$. Cela signifie que l'on a 80% de chance de passer d'une page à la suivante selon la règle initiale et 20% de chances de sauter sur une page choisie uniformément au hasard dans le web. Dans cette nouvelle version, le vecteur des probabilités (X_n) satisfait à la récurrence :

$$\forall n \in \mathbb{N}, \quad X_{n+1} = AX_n.$$

Par construction, la matrice A est stochastique : la somme des coefficients de chaque colonne vaut 1. Comme dans la partie précédente, on en déduit que 1 est une valeur propre de A. De plus, tous ses coefficients sont strictement positifs. On en déduit assez facilement que le module des autres valeurs propres est au plus 1.
Le *théorème de Perron* assure que dans ces conditions, la matrice A possède une valeur propre de module strictement plus grand que toutes les autres, qu'elle est réelle et que le vecteur propre correspondant X_∞ a tous ses coefficients réels, non nuls et de même signe. Quitte à le diviser par la somme des coefficients, on peut donc supposer que les coefficients de X_∞ sont strictement positifs et que leur somme vaut 1. Dans notre cas, la valeur propre maximale est $\lambda_1 = 1$.
Par ailleurs, il est presque certain que l'on peut diagonaliser A : parmi les matrices qui satisfont les propriétés que l'on vient de mettre en évidence, celles qui ne sont pas diagonalisables forment un ensemble de mesure nulle, c'est-à-dire qu'on a une probabilité nulle de tomber dessus. Cela signifie qu'il existe une matrice inversible P et des complexes $\lambda_2, \ldots, \lambda_n$ tels que $A = PDP^{-1}$, où D est la matrice diagonale dont les coefficients diagonaux sont $(1, \lambda_2, \ldots, \lambda_n)$. Comme dans la partie précédente, lorsque n tend vers l'infini, la suite de matrices D^n converge vers la matrice diagonale dont les coefficients diagonaux sont $(1, 0, \ldots, 0)$.
On en déduit facilement que A^n converge vers une matrice dont toutes les colonnes sont proportionnelles à la première colonne de P, c'est-à-dire au vecteur propre de Perron X_∞. Il en résulte alors que la suite $(X_n) = (A^n X_0)$ converge vers un vecteur proportionnel à X_∞. Comme la somme des coefficients de X_n vaut 1, la limite est exactement X_∞.
Ainsi, *la probabilité d'être à la page numéro i converge rapidement vers la* i^{e} *coordonnée du vecteur propre de Perron, qui représente l'état stable du graphe du web.*
Reste un problème difficile : calculer effectivement ces probabilités. Apparemment, il est plus rapide de calculer A^n pour n grand, c'est-à-dire $n = 15$ ou 16..., que de calculer P. Mais la taille de la matrice est gigantesque et le produit de deux matrices $N \times N$, si on le fait naïvement, met en jeu N^3 opérations : c'est beaucoup trop lorsque N est de l'ordre de plusieurs centaines de millions ou même quelques milliards. Heureusement, la matrice A_0 est creuse, c'est-à-dire que de nombreux coefficients sont nuls : cela permet d'imaginer des calculs plus rusés. Les méthodes utilisées pour ces problèmes, sur lesquelles Google est peu disert, prouvent à la fois une grande expertise informatique et une énorme puissance de calcul.