Université Paris I, Panthéon - Sorbonne

Licence M.I.A.S.H.S. deuxième année 2015 – 2016

Méthodes Numériques S4

Contrôle continu n°1, mars 2016

Examen de 1h30. Tout document ou calculatrice est interdit.

1. **(Sur 17 points)**

   (a) Soit le programme:

   ```
   n=6; B=matrix(0,n,n)
   for (j in c(2:n))
   {i=c((j-1):j)
   B[i,j]=1}
   B[1,1]=1
   C=t(B)%*%B
   det(C)
   ```

   Décrire ce qui a été fait dans ce programme **(0.5pts)**. Ecrire B **(1pt)** et C **(1.5pts)**.

   (b) Le résultat de ce programme est `[1] 1`. Expliquer mathématiquement pourquoi on a obtenu ceci **(1pt)**. Déterminer théoriquement la décomposition LU de B **(1.5pts)**, puis de C **(1.5pts)**.

   (c) On tape ensuite les commandes:

   ```
   b=matrix(1:n,1)
   B1=solve(t(B))
   Y=B1%*%b
   t(Y)
   ```

   Décrire cette suite de commandes (utiliser des notations mathématiques) **(0.5pts)**. On a obtenu comme résultat `[1,] 1 1 2 2 3 3`. Retrouver mathématiquement ce résultat **(2.5pts)**.

   (d) On a ensuite fait tourner les commandes suivantes:

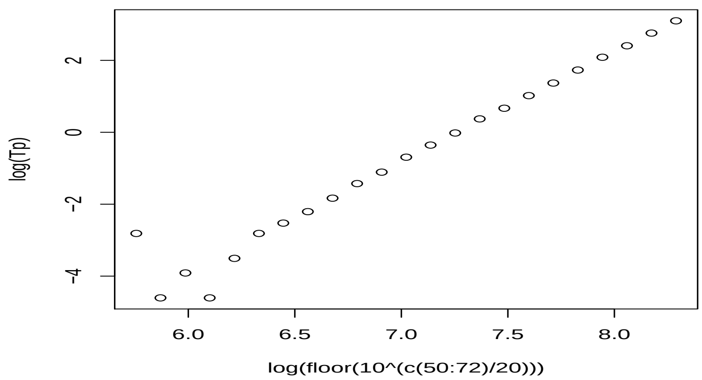
   ```
   X=t(B1)%*%Y
   XX=solve(C)%*%b
   sum((XX-X)^2)
   ```

   Qu'a-t-on fait lors de ces commandes? (utiliser des notations mathématiques) **(0.5pts)**. On a obtenu comme résultat `[1] 0`. Expliquer mathématiquement pourquoi **(2.5pts)**.

   (e) Enfin on effectue la suite de commandes suivante:

   ```
   compt=0; Tp=0
   for (p in c(50:72)/20)
   {compt=compt+1; n=floor(10^p); B=matrix(0,n,n)
   for (j in c(2:n))
   {i=c((j-1):j); B[i,j]=1} B[1,1]=1
   tmp=proc.time()[3]
   solve(B)
   Tp[compt]=proc.time()[3]-tmp}
   plot(log(floor(10^(c(50:72)/20))),log(Tp))
   ```

Expliquer ce que l'on a fait dans ce programme **(0.5pts)**. Le graphe obtenu est celui ci-dessous. En déduire une approximation du temps de calcul pour inverser la matrice lorsque n est grand **(2.5pts)**. Ce résultat était-il attendu? **(1pt)**



Proof. (a) On crée, coordonnée par coordonnée, une matrice B de taille (n,n) avec $n=6$, puis la matrice $C = {}^tB\,B$ et enfin on calcule $\det(C)$.

La matrice B ainsi créée vaut $B = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$ et après multiplication $C = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 0 & 1 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 & 2 \end{pmatrix}$.

(b) On obtient numériquement $\det(C) = 1$. Or $C = {}^tB\,B$, donc $\det(C) = \det(B)^2$ et $\det(B) = 1$ comme déterminant d'une matrice triangulaire dont tous les termes diagonaux valent 1.
Il est clair que la décomposition LU de B est $B = I_6 * B$, où I_6 est la matrice identité de taille 6, car B est une matrice triangulaire supérieure.
De même, comme ${}^tB$ est une matrice triangulaire inférieure dont les termes diagonaux sont des 1, donc la décomposition LU de C est $C = {}^tB\,B$.

(c) On définit le vecteur colonne b de taille n et tel que $b = {}^t(1,2,3,\cdots,n)$ (avec $n=6$), puis on inverse la matrice ${}^tB$ (ce qui donne $B1 = ({}^tB)^{-1}$) et on affecte à Y le vecteur $Y = ({}^tB)^{-1}b$ (ce qui équivaut à résoudre le système ${}^tB\,Y = b$).
Or si l'on résout le système, en posant $Y = {}^t(y_1,\cdots,y_6)$ on obtient le système d'équations:

$$\left\{\begin{array}{llllllcl} y_1 & & & & & & = & 1 \\ y_1 & +y_2 & & & & & = & 2 \\ & y_2 & +y_3 & & & & = & 3 \\ & & y_3 & +y_4 & & & = & 4 \\ & & & y_4 & +y_5 & & = & 5 \\ & & & & y_5 & +y_6 & = & 6 \end{array}\right. \iff \left\{\begin{array}{llllllcc} y_1 & & & & & & = & 1 \\ & y_2 & & & & & = & 2-1=1 \\ & & y_3 & & & & = & 3-1=2 \\ & & & y_4 & & & = & 4-2=2 \\ & & & & y_5 & & = & 5-2=3 \\ & & & & & y_6 & = & 6-3=3 \end{array}\right.$$

(d) On effectue l'opération $X = {}^t(B1) * Y$, soit encore $X = B^{-1}\,Y$, puis $XX = C^{-1}\,b$. Enfin on calcule $\|X - XX\|_2^2$. X est donc solution du système $B\,X = Y$ et XX solution du système $C\,XX = b$.
On obtient $\|X - XX\|_2^2 = 0$ ce qui n'est pas surprenant: en effet, on a $B\,X = Y$ et comme ${}^tB\,Y = b$, on a donc ${}^tB\,(B\,X) = {}^tB\,Y = b$. Mais comme $C = {}^tB\,B$, au final on a $C\,X = b$. La matrice C étant inversible on a donc $X = XX$.

(e) Ce programme permet de calculer le temps de calcul Tp nécessaire pour inverser la matrice B lorsque n varie. On fait prendre à n les $\{[10^{2.50}]\,,\,[10^{2.55}]\,,\,[10^{2.60}],\cdots,[10^{3.60}]$, puis on trace la courbe des $\ln(Tp)$ en fonction des $\log(n)$.
On obtient approximativement une droite, ce qui signifie que Tp varie comme n^α où α est le coefficient directeur de la droite. On trouve que pour $\ln(n) \simeq 6$, $\ln(Tp) \simeq -4$ et pour $\ln(n) \simeq 8$ alors $\ln(Tp) \simeq 2$, donc le coefficient est $\alpha \simeq (2-(-4))/(8-6) \simeq 3$. On a ainsi $\ln(Tp) \simeq C + 3\ln(n)$, donc $Tp \simeq e^C\,n^3$. Ceci n'est pas étonnant car on sait qu'asymptotiquement le temps de calcul de l'inverse en utilisant la décomposition LU est en $2n^3/3$.

□

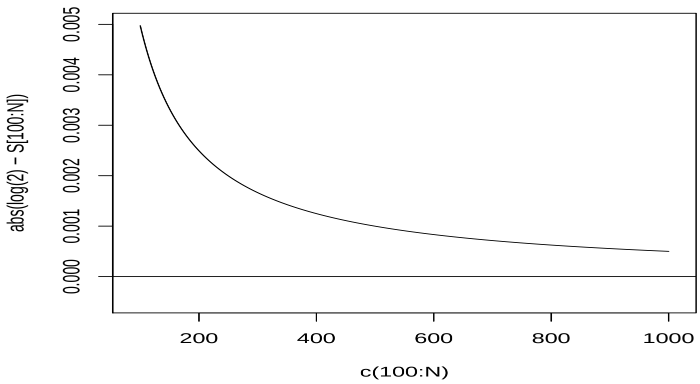
2. **(16 points)** On rappelle que pour tout $x \in]-1,1]$, on a: $\ln(1+x) = \displaystyle\sum_{n=0}^{\infty} (-1)^n \frac{x^{n+1}}{n+1}$.

(a) On a tapé les commandes suivantes:

```
S=1; N=1000;
k=c(1:N); u=(-1)^(k+1)/k
for (n in c(2:N)) S[n]=S[n-1]+u[n]
plot(c(100:N), abs(log(2)-S[100:N]),'l')
```

Décrire ce qui a été fait (en termes mathématiques!) en donnant notamment la formule mathématique de $S[N]$ **(1pt)**. Le graphe ci-dessous est tracé. Expliquer mathématiquement la courbe obtenue **(0.5pts)**. Quelle est approximativement l'ordre de grandeur de $\ln(2)$ obtenu avec $S[1000]$? **(0.5pts)** Montrer mathématiquement que l'erreur commise avec $S[N]$ est majorée par $1/(N+1)$ **(1pt)**. Comparer avec le graphe et expliquer la différence **(0.5pts)**.

(b) Montrer que $\ln(2) - S[2n] = \sum_{k=n}^{\infty} \frac{1}{(2n+1)(2n+2)}$ **(2pts)**. En déduire (penser à une intégrale!) que pour n grand, $\ln(2) - S[2n] \sim \frac{1}{4n}$ **(2pts)**. Ce résultat est-il plus proche des résultats du graphe **(1pt)**? En déduire la valeur de n nécessaire pour que $S[n]$ approche $\ln(2)$ à 10^{-16} près **(0.5pts)**. Concrètement est-ce jouable? **(0.5pts)**



(c) On réalise maintenant les commandes suivantes:

```
N=10; k=c(1:N); u=1/(2^k*k)
T=sum(u); abs(log(2)-T)
```

Le résultat de la dernière commande est `[1] 8.232441e-05`. Pourquoi T permet d'approcher $\ln(2)$ (se ramener au développement en série entière de $\ln(1+x)$...)? **(1.5pts)** Comparer le résultat obtenu avec celui obtenu en (a) et (b) **(0.5pts)**. Montrer que $|T[n]-\ln(2)| \leq \frac{2^{-n}}{n+1}$ **(2pts)**. Combien aurait-il fallu à peu près calculer de termes pour être sûr d'avoir une approximation à 10^{-16} près de $\ln(2)$? **(1pt)** Ecrire un code R qui permettrait de trouver ce N **(1.5pts)**.

Proof. (a) On commence par définir la somme partielle $S[N] = \sum_{n=1}^{N} (-1)^{n+1} \frac{1}{n}$, en définissant d'abord le vecteur des $((-1)^{n+1} \frac{1}{n})_{1 \leq n \leq N}$, que l'on somme, ceci pour $N = 1000$. On sait que $S[N] \to \ln(2)$ quand $N \to \infty$ (formule du développement en série entière de $\ln(1+x)$ prise en $x = 1$), donc $S[N]$ est une approximation de $\ln(2)$. On trace alors la valeur absolue du reste $R[n] = \ln(2) - S[n]$ en fonction de n pour n variant de 100 à $N = 1000$.
Sur la courbe, on note que pour $N = 1000$ alors $|R[N]| \simeq 0.0005$, donc l'approximation de $\ln(2)$ par $S[1000]$ est à $5\,10^{-4}$ près.
La majoration de $|R[n]|$ s'obtient par le théorème de convergence des séries alternées $\sum (-1)^n a_n$ et par le fait qu'ici, $(a_n) = (1/(n+1))_n$ est positive et décroissante vers 0. D'où $|R[N]| \leq a_{N+1} = 1/(N+1)$.
Pour $N = 1000$, on a donc $|R[N]| \leq 0.001$. Ceci est 2 fois plus que ce que l'on observe sur le graphe, ce qui marque l'écart entre une majoration et une vraie valeur.

(b) On a $\ln(2) - S[2n] = \sum_{k=2n+1}^{\infty} (-1)^{k+1} \frac{1}{k} = \sum_{k=n}^{\infty} \frac{1}{2k+1} - \sum_{k=n}^{\infty} \frac{1}{2k+2} = \sum_{k=n}^{\infty} \frac{1}{(2k+1)(2k+2)}$.
D'après le Théorème de comparaison avec une intégrale, si f est positive, décroissante et tendant vers 0, et si $\int_1^{\infty} f(t)dt < \infty$ alors $\int_n^{\infty} f(t)dt \leq \sum_{k=n}^{\infty} f(k) \leq \int_{n-1}^{\infty} f(t)dt$. Or ici $f(t) = \frac{1}{2t+1} - \frac{1}{2t+2}$, d'où $\int_x^{\infty} f(t)dt = \frac{1}{2} \ln\left(1 + \frac{1}{2x+1}\right)$. Ainsi quand $x \to \infty$, $\ln\left(1 + \frac{1}{2x+1}\right) \simeq \frac{1}{2x}$, d'où $R[2n] \simeq \frac{1}{4n}$.
De ceci on déduit aussi que $R[2n+1] \simeq \frac{1}{4n} - \frac{1}{2n+2} \simeq -\frac{1}{4n}$, donc $|R[n]| \simeq \frac{1}{2n}$ pour tout n. Ceci correspond au $5\,10^{-4}$ relevé sur le graphe.
On en déduit que pour avoir une approximation à 10^{-16} de $\ln(2)$, on doit calculé $S[n]$ avec $\frac{1}{2n} \leq 10^{-16}$, soit $n \geq 5\,10^{15}$. Ce n'est clairement pas possible numériquement.

(c) On a $T=\sum_{k=1}^{N}\frac{1}{2^k\,k}$ avec $N=10$. En utilisant le développement en série entière de $\ln(1+x)$ en $x=-0.5$, on obtient: $\ln(0.5)=-\ln(2)=\sum_{k=1}^{\infty}(-1)^{k+1}\frac{1}{k}\left(-\frac{1}{2}\right)^k=\sum_{k=1}^{\infty}(-1)^{2k+1}\frac{1}{2^k\,k}=-\sum_{k=1}^{\infty}\frac{1}{2^k\,k}$, d'où le résultat.
On voit ici que le reste à l'ordre 10, $R[10]=|\ln(2)-T[10]|$ est de l'ordre de 10^{-4}, ce que l'on aurait obtenu en calculant 5000 termes avec la première série considérée. On va donc beaucoup plus vite pour converger vers $\ln(2)$.
On a $|T[n]-\ln(2)|=\sum_{k=n+1}^{\infty}\frac{1}{2^k\,k}\leq\frac{1}{n+1}\sum_{k=n+1}^{\infty}\frac{1}{2^k}\leq\frac{1}{n+1}\frac{1}{2^{n+1}}\sum_{k=0}^{\infty}\frac{1}{2^k}\leq\frac{1}{n+1}\frac{1}{2^{n+1}}\frac{1}{1-\frac{1}{2}}\leq\frac{1}{n+1}\frac{1}{2^n}$.
On sait que $2^{10}\sim10^3$, donc avec $\frac{1}{51}2^{-50}<10^{-16}$. Il est donc sûr qu'avec $N=50$ cela aurait été suffisamment... (la vraie réponse est 48...).
Un programme possible:

```
er=1; k=0;  T=0
while (er>3e-16)
{k=k+1; T=T+1/(2^k*k); er=abs(log(2)-T) }
k
```

□