

Chapitre 15

Sommes doubles

Sommaire

15.1 Sommes doubles “rectangulaires”	1
15.1.1 Sommations par lignes et par colonnes	1
15.1.2 Cas des termes séparables	4
15.1.3 Et en Python?	4
15.2 Sommes doubles “triangulaires”	6
15.2.1 Sommations par lignes et par colonnes	6
15.2.2 Méthodes	8
15.2.3 Et en Python?	8
15.3 Exercices	9

Dans tout ce document, $n, p, n_0, n_1, p_0, p_1, \dots$ désignent des entiers ad-hoc, et on considère pour tous $i, j \in \mathbb{Z}$ des nombres complexes $a_{i,j} \in \mathbb{C}$.

15.1 Sommes doubles “rectangulaires”

Dans ce paragraphe, on cherche à calculer des sommes du type $\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} a_{i,j}$. C’est-à-dire qu’on cherche à déterminer la somme de tous les $a_{i,j}$ pour $i \in \llbracket 1, n \rrbracket$ et $j \in \llbracket 1, p \rrbracket$.

Notation : lorsque $n = p$ on note $\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} a_{i,j} = \sum_{1 \leq i, j \leq n} a_{i,j}$

15.1.1 Sommations par lignes et par colonnes

Rangeons les nombres $a_{i,j}$ dans un tableau à n lignes et p colonnes dans lequel $a_{i,j}$ est placé à la ligne i , colonne j .

Pour calculer $\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} a_{i,j}$, on peut d’abord calculer la somme sur chacune des lignes :

$$\begin{array}{ccccccc} a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,p} & \longrightarrow & a_{1,1} + \dots + a_{1,j} + \dots + a_{1,p} = \sum \\ \vdots & & \vdots & & \vdots & & \\ a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,p} & \longrightarrow & a_{i,1} + \dots + a_{i,j} + \dots + a_{i,p} = \sum \\ \vdots & & \vdots & & \vdots & & \\ a_{n,1} & \cdots & a_{n,j} & \cdots & a_{n,p} & \longrightarrow & a_{n,1} + \dots + a_{n,j} + \dots + a_{n,p} = \sum \end{array}$$

Conclusion : en sommant ligne par ligne, on obtient

$$\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} a_{i,j} = \sum_{j=1}^p a_{1,j} + \dots + \sum_{j=1}^p a_{i,j} + \dots + \sum_{j=1}^p a_{n,j} = \sum \left(\sum_{j=1}^p a_{i,j} \right)$$

Pour calculer $\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} a_{i,j}$, on peut aussi calculer la somme sur chacune des colonnes :

$$\begin{array}{cccccc}
 a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,p} \\
 \vdots & & \vdots & & \vdots \\
 a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,p} \\
 \vdots & & \vdots & & \vdots \\
 a_{n,1} & \cdots & a_{n,j} & \cdots & a_{n,p} \\
 \downarrow & & \downarrow & & \downarrow \\
 \Sigma & & \Sigma & & \Sigma
 \end{array}$$

Conclusion : en sommant colonne par colonne, on obtient

$$\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} a_{i,j} = \sum_{i=1}^n a_{i,1} + \dots + \sum_{i=1}^n a_{i,j} + \dots + \sum_{i=1}^n a_{i,p} = \sum \left(\sum_{i=1}^n a_{i,j} \right).$$

Finalement :

$$\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} a_{i,j} =$$

Remarque : Les indices ne commencent pas forcément à 1. Plus généralement, on a :

$$\sum_{\substack{n_0 \leq i \leq n_1 \\ p_0 \leq j \leq p_1}} a_{i,j} = \sum_{i=n_0}^{n_1} \left(\sum_{j=p_0}^{p_1} a_{i,j} \right) = \sum_{j=p_0}^{p_1} \left(\sum_{i=n_0}^{n_1} a_{i,j} \right)$$

Plus généralement encore, on peut sommer sur des indices (i, j) appartenant à n'importe quel produit cartésien : $I \times J$ (c'est-à-dire pour tous les couples (i, j) avec $i \in I$ et $j \in J$) et alors

$$\sum_{(i,j) \in I \times J} a_{i,j} = \sum_{i \in I} \left(\sum_{j \in J} a_{i,j} \right) = \sum_{j \in J} \left(\sum_{i \in I} a_{i,j} \right)$$

Remarque : Il faut ensuite utiliser les outils usuels des sommes simples, en particulier :

Si $a_{i,j} = \tilde{a}_j$ ne dépend pas de i on peut simplifier la somme $\sum_{i=1}^n a_{i,j} = \sum_{i=1}^n \tilde{a}_j =$

De même, Si $a_{i,j} = \tilde{a}_i$ ne dépend pas de j on peut simplifier la somme $\sum_{j=1}^p a_{i,j} = \sum_{j=1}^p \tilde{a}_i =$

Exercice 15.1. Calculer $S_1 = \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} 1$, $S_2 = \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} i$, $S_3 = \sum_{1 \leq i, j \leq n} i + j$ et $S_4 = \sum_{1 \leq i, j \leq n} ij$.

15.1.2 Cas des termes séparables

On s'intéresse au cas particulier où $a_{i,j}$ s'écrit $a_{i,j} = a_i b_j$.

Propriété 15.1. Soient $a_1, \dots, a_n, b_1, \dots, b_p \in \mathbb{C}$. On a :

$$\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} a_i b_j = \left(\sum_{i=1}^n a_i \right) \times \left(\sum_{j=1}^p b_j \right)$$

Démonstration :

Exercice 15.2. Écrire sous forme d'une somme double : $\left(\sum_{k=1}^n k \right)^2$

Exercice 15.3. Calculer pour $n, p \in \mathbb{N}^*$ et $a \in \mathbb{C} \setminus \{1\}$: $S_5 = \sum_{0 \leq i, j \leq n} a^{i+j}$, $S_6 = \sum_{\substack{1 \leq i \leq n \\ 0 \leq j \leq p}} i a^j$, et $P = \prod_{1 \leq i, j \leq n} ij$.

15.1.3 Et en Python?

Pour calculer une somme double du type $\sum_{\substack{a \leq i \leq b \\ c \leq j \leq d}} a_{i,j}$

on utilise 2 boucles `for` imbriquées :

```
s = 0
for i in range(a, b+1):
    for j in range(c, d+1):
        s += a(i,j) # bien sur il faut remplacer a(i,j) par sa
                   # valeur en fonction de i et j
```

Exercice 15.4. Écrire une fonction Python prenant en argument deux entiers n et m et renvoyant la valeur de :

1.
$$\sum_{\substack{1 \leq i \leq n \\ 0 \leq j \leq m}} \frac{i}{i+j}$$

2.
$$\prod_{\substack{2 \leq i \leq n+1 \\ 0 \leq j \leq m}} (i+2^j)$$

15.2 Sommes doubles “triangulaires”

On se place dans le cas “carré” $n = p$. On range à nouveau les nombres $a_{i,j}$ dans un tableau (à n lignes et n colonnes). Le nombre $a_{i,j}$ est placé ligne i colonne j .

$$\begin{pmatrix} a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,n} \\ \vdots & & \vdots & & \vdots \\ a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,n} \\ \vdots & & \vdots & & \vdots \\ a_{n,1} & \cdots & a_{n,j} & \cdots & a_{n,n} \end{pmatrix}.$$

On souhaite calculer la somme des coefficients du tableau placés au-dessus ou en-dessous de la diagonale (diagonale incluse), c’est-à-dire les sommes :

- $\sum a_{i,j}$ (termes au-dessus de la diagonale), ou
- $\sum a_{i,j}$ (termes en-dessous de la diagonale).

Dans la suite, on s’intéresse à la deuxième somme, les résultats sont bien sûr similaires dans le cas de la première.

15.2.1 Sommations par lignes et par colonnes

Pour calculer $\sum_{1 \leq j \leq i \leq n} a_{i,j}$, on peut d’abord calculer la somme sur chacune des lignes :

$$\begin{array}{l} a_{1,1} \qquad \qquad \qquad \longrightarrow a_{1,1} = \sum \\ \vdots \quad \ddots \\ a_{i,1} \quad \cdots \quad a_{i,i} \qquad \longrightarrow a_{i,1} + \dots + a_{i,i} = \sum \\ \vdots \qquad \qquad \vdots \quad \ddots \\ a_{n,1} \quad \cdots \quad a_{n,j} \quad \cdots \quad a_{n,n} \longrightarrow a_{n,1} + \dots + a_{n,j} + \dots + a_{n,n} = \sum \end{array}$$

Conclusion : en sommant ligne par ligne, on obtient

$$\sum_{1 \leq j \leq i \leq n} a_{i,j} = \sum_{j=1}^1 a_{1,j} + \cdots + \sum_{j=1}^i a_{i,j} + \cdots + \sum_{j=1}^n a_{n,j} = \sum_{i=1}^n \left(\sum_{j=1}^i a_{i,j} \right)$$

Exercice 15.6. Calculer $T = \sum_{1 \leq j < i \leq n} 1$.

15.2.2 Méthodes

Il faut ensuite utiliser les outils usuels des sommes simples, en particulier :

— Si $a_{i,j} = \tilde{a}_i$ ne dépend pas de j on peut simplifier la somme $\sum_{j=1}^i a_{i,j} = \sum_{j=1}^i \tilde{a}_i =$

— Si $a_{i,j} = \tilde{a}_j$ ne dépend pas de i on peut simplifier la somme $\sum_{i=j}^n a_{i,j} = \sum_{i=j}^n \tilde{a}_j =$

Cette utilisation est un peu plus subtile que dans le cas “rectangulaire” car l’indice dont dépend $a_{i,j}$ peut lui-même être une borne...

Important : Le choix de l’ordre de sommation devient donc ici une étape essentielle du calcul : choisir le bon ordre simplifie (voire rend possible) le calcul de la somme.

Exercice 15.7. Écrire les deux ordres de sommations possibles pour calculer les sommes suivantes, choisir le bon, et terminer le calcul : $S_1 = \sum_{1 \leq i \leq j \leq n} \frac{1}{j}$, et $S_3 = \sum_{0 \leq j \leq k \leq n} \binom{n}{k} \binom{k}{j} a^j b^{k-j}$.

Pour conclure, certaines sommes “rectangulaires” se prêtent particulièrement à un “découpage” en sommes “triangulaires”. On écrit alors par exemple que :

$$\sum_{1 \leq i, j \leq n} a_{i,j} = \sum_{1 \leq i \leq j \leq n} a_{i,j} + \sum_{1 \leq j < i \leq n} a_{i,j}$$

Exercice 15.8. Calculer $S_4 = \sum_{1 \leq i, j \leq n} \min(i, j)$ et $S_5 = \sum_{1 \leq i, j \leq n} \max(i, j)$

Exercice 15.9. Calculer pour $n \in \mathbb{N}^*$: $S_6 = \sum_{0 \leq i < j \leq n} \frac{i^2}{j^2 - j}$, $S_7 = \sum_{0 \leq i \leq j \leq n} 2^{i-j}$

15.2.3 Et en Python ?

Pour calculer une somme double du type $\sum_{a \leq i \leq j \leq b} a_{i,j}$ on utilise 2 boucles for imbriquées :

```
s = 0
for i in range(a, b+1):
    for j in range(i, b+1):
        s += a(i,j) # bien sur il faut remplacer a(i,j) par sa
                    valeur en fonction de i et j
```

Exercice 15.10. Écrire une fonction Python prenant en argument deux entiers n et m et renvoyant la valeur de :

$$1. \sum_{1 \leq j \leq i \leq n} \frac{i}{i+j}$$

$$2. \prod_{m \leq j < i \leq n} ij$$

15.3 Exercices

Exercice 15.11. Calculer les sommes et produits suivants pour $n, p \in \mathbb{N}^*$:

$$1. S_1 = \sum_{1 \leq i, j \leq n} i^2 j$$

$$3. P_1 = \prod_{1 \leq i, j \leq n} i^2 j$$

$$2. S_2 = \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} 2^{i-j}$$

$$4. P_2 = \prod_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} e^{i+j}$$

Exercice 15.12. Calculer les sommes et produits suivants pour $n \in \mathbb{N}^*$:

$$1. S_1 = \sum_{1 \leq j \leq i \leq n} i + j$$

$$3. S_4 = \sum_{1 \leq i < j \leq n} \frac{1}{n-i}$$

$$2. S_2 = \sum_{1 \leq i \leq j \leq n} \frac{i}{j}$$

$$4. S_5 = \sum_{1 \leq i, j \leq n} |i-j|$$

Exercice 15.13. Pour $n \geq 1$ on note $H_n = \sum_{k=1}^n \frac{1}{k}$ et $u_n = \sum_{k=1}^n H_k$.

Montrer que pour tout $n \geq 1$, $u_n = (n+1)H_n - n$.