MATH1 pour Technologies de l'Information

Niklaus Eggenberg & Orestis Malaspinas

Chapitre 1

Rappel sur les espaces vectoriels

1.1 Les vecteurs

Si nous nous plaçons sur une carte à un endroit donné (appelé l'origine) et que nous prenons notre notre voiture pour nous déplacer, notre destination ne peut être uniquement décrite si nous ne connaissons que la distance parcourue (voir par exemple la carte fig. 1.1). En d'autres termes pour notre carte, nous savons que nous irons à la campagne, mais pas si nous allons à Lausanne ou dans le Jura). Un déplacement n'est donc pas uniquement déterminé par le point d'origine et un scalaire. Il est nécessaire d'avoir une information supplémentaire. Dans le cas de la fig. 1.1 le déplacement est décrit uniquement par la flèche, ou *vecteur*, qui a une longueur (la distance parcourue) et une direction (la direction de la flèche).

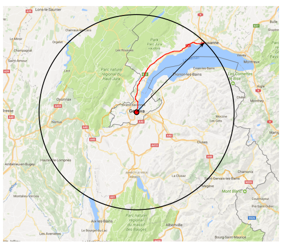

Figure 1.1 – L'ensemble des destinations possibles si nous n'avions que le point de départ et la distance parcourue comme information (cercle noir), la position finale avec la distance et la direction de déplacement (Lausanne).

D'un point de vue mathématique un vecteur en deux dimensions, $\vec{v}$, est noté comme un couple de nombres

$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}, \tag{1.1}$$

avec $v_1, v_2 \in \mathbb{R}$. Il représente la *flèche* allant de l'origine (le point $(0,0)$) au point (v_1, v_2) (voir fig. 1.2).

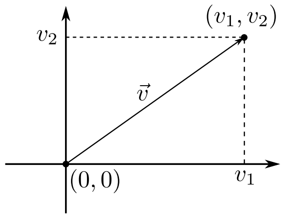

Figure 1.2 – Le vecteur $\vec{v}$ relie le point $(0,0)$ au point (v_1, v_2).

Grâce au théorème de Pythagore (ce Grec mort depuis longtemps) nous pouvons aisément calculer la longueur du vecteur $\vec{v}$, notée $||\vec{v}||$ et également appelée norme

$$||\vec{v}|| = \sqrt{v_1^2 + v_2^2}. \tag{1.2}$$

Et donc on peut également définir le vecteur unitaire, $\hat{v}$, qui est un vecteur qui a la direction de $\vec{c}$, mais qui est de longueur 1

$$\hat{v} = \frac{\vec{v}}{||\vec{v}||}. \tag{1.3}$$

Les composantes du vecteur $\vec{v}$ dépendent du système de coordonnées dans lequel il est représenté. De façon plus générale il dépend de la *base* dans laquelle il est exprimé. Dans la fig. 1.2 nous utilisons les coordonnées cartésiennes, mais cela n'est pas du tout le seul moyen de représenter un vecteur (voir la fig. 1.3)

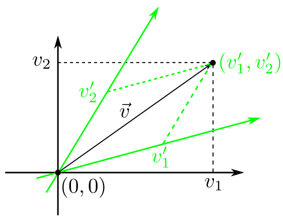

Figure 1.3 – Le vecteur $\vec{v}$ représente toujours la même flèche. Dans le cas des coordonnées cartésiennes (en noir) ses composantes sont v_1 et v_2. Par contre ses composantes sont v_1' et v_2' dans les coordonnées "vertes".

Peu importe la base que nous utilisons, un vecteur dans le plan à deux dimensions se représentera par deux nombres. On notera que $\vec{v} \in \mathbb{R} \times \mathbb{R} = \mathbb{R}^2$, soit "un $\mathbb{R}$ par dimension".

Le concept de vecteur peut donc se généraliser pour plus de dimensions. Une vecteur dans un espace qui a n dimensions, $\vec{v} \in \mathbb{R}^n$, s'écrira

$$\vec{v} = \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix}, \tag{1.4}$$

avec $v_i \in \mathbb{R}, \ \forall i$.

Remarque 1.1

Ici un vecteur se représente comme une liste de taille finie de nombres et la dimension du monde dans lequel vit se vecteur est la taille de la liste. Un vecteur de façon plus générale peut vivre dans un monde de dimension quelconque, y compris infinie.

Remarque 1.2

Ici nous n'avons considéré que des vecteurs vivant dans $\mathbb{R}^n$. On peut en fait considérer n'importe quel ensemble $D \subseteq \mathbb{R}$ et construire des vecteurs de dimension n vivant dans D^n.

Exemple 1.1

Un octet (une séquence de huit bits, comme son nom l'indique) constitue un vecteur qui vit dans $\{0, 1\}^8$.

1.2 L'addition de vecteurs

Comme pour les scalaires, nous voulons pouvoir additionner les vecteurs. Revenons donc aux vecteurs à deux dimensions. Une façon raisonnable de définir l'addition de deux vecteurs est de les mettre "bout à bout" (voir la fig. 1.4).

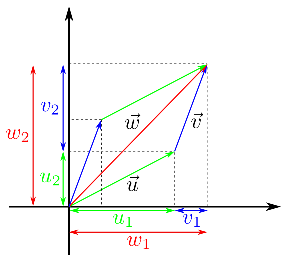

Figure 1.4 – La somme de deux vecteurs $\vec{u}$ et $\vec{v}$ donne le vecteur $\vec{w}$. Elle est obtenue en mettant bout à bout le vecteur $\vec{u}$ et $\vec{v}$.

Cette façon de faire est en fait assez intuitive pour construire la somme de deux vecteurs. Nous pouvons écrire que si le vecteur $\vec{w} \in \mathbb{R}^2$ est la somme de $\vec{u} \in \mathbb{R}^2$

et $\vec{v} \in \mathbb{R}^2$, on peut écrire

$$\begin{aligned} \vec{w} &= \vec{u} + \vec{v}, \\ \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} &= \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}, \\ \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} &= \begin{pmatrix} u_1 + v_1 \\ u_2 + v_2 \end{pmatrix}. \end{aligned} \tag{1.5}$$

En fait il suffit de sommer chacune des composantes du vecteur et on aura donc

$$\begin{aligned} w_1 &= u_1 + v_1, \\ w_2 &= u_2 + v_2. \end{aligned} \tag{1.6}$$

Cette définition de la somme se généralise très facilement à n dimensions. On a que

$$\begin{pmatrix} w_1 \\ \vdots \\ w_n \end{pmatrix} = \begin{pmatrix} u_1 \\ \vdots < \\ u_n \end{pmatrix} + \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} = \begin{pmatrix} u_1 + v_1 \\ \vdots \\ u_n + v_n \end{pmatrix}. \tag{1.7}$$

```
>>> u = [1.0, 2.0, 3.0]
>>> v = [2.0, 3.0, 4.0]
>>> w = [0.0, 0.0, 0.0]
>>> for i in range(3): # loop based syntax for vector addition
...     w[i]=a[i] + b[i]
...
>>> w
[3.0, 5.0, 7.0]
>>> import numpy as np
>>> u_np = np.array([1.0, 2.0, 3.0])
>>> v_np = np.array([2.0, 3.0, 4.0])
>>> w_np = u_np + v_np # numpy syntax for vector addition
>>> w_np
array([3., 5., 7.])
```

Remarque 1.3

Il est important de noter que les dimensions de $\vec{u}$ et $\vec{v}$ doivent être les mêmes. Sinon la somme n'est pas définie. L'opération somme en n dimension prend donc deux vecteurs de dimension n et rend un vecteur de dimension n également.

Exemple 1.2

Pour en revenir à l'exemple des octets, il est un peu plus compliqué de définir la somme. En effet, la somme entre des bits peut avoir différentes définitions. Une possibilité est de définir la somme comme le XOR, $\oplus$.

Pour “sommer” deux octets, $\vec{u},\ \vec{v} \in \{0,1\}^8$ on peut donc définir la somme comme

$$\vec{u} \oplus \vec{v} = \begin{pmatrix} u_1 \oplus v_1 \\ \vdots \\ u_8 \oplus v_8 \end{pmatrix}. \tag{1.8}$$

1.3 Multiplication entre un vecteur et un scalaire

Soit un scalaire $\lambda \in \mathbb{R}$, nous voulons à présent définir l'opération “multiplier par un scalaire” sur un vecteur. L'interprétation que nous voulons faire de cette opération est une “dilatation”: le vecteur multiplié verra sa longueur changer mais pas sa direction générale (voir la fig. 1.5).

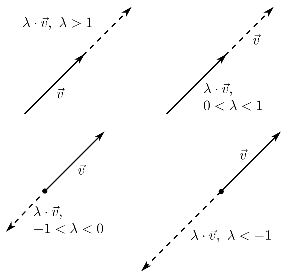

Figure 1.5 – Le produit entre un scalaire λ et un vecteur $\vec{v}$ est une dilatation du vecteur $\vec{v}$. Le cas où λ est négatif change également l'orientation du vecteur.

Comme on le constate sur la fig. 1.5 il y a quatre cas de figure

1. $\lambda > 1$ le vecteur s'allonge et garde la même direction et orientation.
2. $0 < \lambda < 1$ le vecteur raccourcit et garde la même direction et orientation.
3. $0 > \lambda > -1$ le vecteur raccourcit et change d'orientation.
4. $\lambda < -1$ le vecteur s'allonge et change d'orientation.

Avec la définition de la norme d'un vecteur $||\vec{v}||$, il est très aisé de voir que la multiplication par un scalaire revient en fait à multiplier chaque composante du vecteur $\vec{v}$ par un scalaire. En définissant $\vec{u} = \lambda \cdot \vec{v}$, cela revient à faire

$$\vec{u} = \lambda \cdot \vec{v} = \begin{pmatrix} \lambda \cdot v_1 \\ \lambda \cdot v_2 \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}. \tag{1.9}$$

En d'autres termes $u_1 = \lambda \cdot v_1$ et \$u_2 La longueur de $||\vec{u}||$ est donnée par

$$||\vec{u}|| = \sqrt{u_1^2 + u_2^2} = \sqrt{(\lambda \cdot v_1)^2 + (\lambda \cdot v_2)^2} = \sqrt{\lambda^2 (v_1^2 + v_2^2)} = |\lambda| \cdot ||\vec{v}||. \tag{1.10}$$

On voit donc que la longueur de $\vec{v}$ est modifiée d'un facteur λ: il est dilaté ou contracté. Le signe de λ aura uniquement une influence sur l'orientation.

Comme pour la somme la généralisation de la multiplication par un scalaire est triviale. En dimension n, pour $\lambda \in \mathbb{R}$ et $\vec{v} \in \mathbb{R}^n$ on a

$$\vec{u} = \lambda \cdot \vec{v} = \begin{pmatrix} \lambda \cdot v_1 \\ \vdots \\ \lambda \cdot v_n \end{pmatrix}. \tag{1.11}$$

En python, cette opération peut s'écrire à l'air d'une boucle, ou de façon beaucoup plus simple en utilisant la librairie NumPy.

```
>>> u = [1.0, 2.0, 3.0]
>>> v = [0.0, 0.0, 0.0]
>>> alpha = 1.5
>>>  for i in range(3): # loop syntax for scalar multiplication
...     v[i] = alpha * u[i]
...
>>> v
[1.5, 3.0, 4.5]
>>> import numpy as np
>>> v_np = np.array([1, 2, 3])
>>> alpha = 1.5
>>> alpha * v_np  # numpy syntax for scalar multiplication
array([1.5, 3. , 4.5])
```

1.4 Les espaces vectoriels

Les vecteurs en deux dimensions à composantes réelles, $\vec{v} \in \mathbb{R}^2$, et les scalaires réels, $\lambda \in \mathbb{R}$, dotés de la somme de la sec. 1.2 et la multiplication sec. 1.3 forme un espace qui a un certain nombre de propriétés.

1.4.1 Propriétés de la somme

Soient $\vec{u}, \vec{v}, \vec{w} \in \mathbb{R}^2$ deux vecteurs de $\mathbb{R}^2$.

1. La somme est dite *interne*: la somme de deux vecteurs de $\mathbb{R}^2$ reste dans $\mathbb{R}^2$. Ceci se note de la façon suivante
$$+ : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^2. \tag{1.12}$$
2. La somme est *commutative*
$$\vec{v} + \vec{u} = \vec{u} + \vec{v}, \tag{1.13}$$
soit en composantes
$$\begin{pmatrix} v_1 + u_1 \\ v_2 + u_2 \end{pmatrix} = \begin{pmatrix} u_1 + v_1 \\ v_2 + v_2 \end{pmatrix}. \tag{1.14}$$
3. La somme possède un *élément neutre*, noté $\vec{0}$ et définit par
$$\vec{0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{1.15}$$
Lorsqu'on somme l'élément neutre avec un autre vecteur, on ne modifie pas ce vecteur
$$\vec{v} + \vec{0} = \vec{v}, \tag{1.16}$$
soit en composantes
$$\begin{pmatrix} v_1 + 0 \\ v_2 + 0 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}. \tag{1.17}$$

4. La somme est *associative*

$$(\vec{u}+\vec{v})+\vec{w}=\vec{u}+(\vec{v}+\vec{w}), \tag{1.18}$$

soit en composantes

$$\begin{aligned}\left(\begin{pmatrix}u_1\\u_2\end{pmatrix}+\begin{pmatrix}v_1\\v_2\end{pmatrix}\right)+\begin{pmatrix}w_1\\w_2\end{pmatrix}&=\begin{pmatrix}u_1+v_1\\u_2+v_2\end{pmatrix}+\begin{pmatrix}w_1\\w_2\end{pmatrix}=\begin{pmatrix}u_1+v_1+w_1\\u_2+v_2+w_2\end{pmatrix},\\&=\begin{pmatrix}u_1\\u_2\end{pmatrix}+\begin{pmatrix}v_1+w_1\\v_2+w_2\end{pmatrix}=\begin{pmatrix}u_1\\u_2\end{pmatrix}+\left(\begin{pmatrix}v_1\\v_2\end{pmatrix}+\begin{pmatrix}w_1\\w_2\end{pmatrix}\right).\end{aligned} \tag{1.19}$$

5. Tout élément $\vec{v}$ possède un *opposé* qui est noté $-\vec{v}$. Quand on somme $\vec{v}$ et $-\vec{v}$ on obtient l'élément neutre

$$\vec{v}+(-\vec{v})=\vec{0}, \tag{1.20}$$

soit en composantes

$$\begin{pmatrix}v_1\\v_2\end{pmatrix}+\begin{pmatrix}-v_1\\-v_2\end{pmatrix}=\begin{pmatrix}v_1-v_1\\v_2-v_2\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}. \tag{1.21}$$

1.4.2 Propriétés de la multiplication par un scalaire

Soient $\vec{u},\vec{v}\in\mathbb{R}^2$ deux vecteurs de $\mathbb{R}^2$ et $\mu,\lambda\in\mathbb{R}$.

1. La multiplication par un scalaire d'un vecteur est dite *externe*: le produit entre un scalaire et d'un vecteur de $\mathbb{R}^2$ reste un vecteur de $\mathbb{R}^2$.

$$\cdot:\mathbb{R}\times\mathbb{R}^2\to\mathbb{R}^2. \tag{1.22}$$

2. La multiplcation est *distributive* par rapport à la somme

$$\lambda\cdot(\vec{u}+\vec{v})=\lambda\vec{u}+\lambda\vec{v}, \tag{1.23}$$

soit en composantes

$$\begin{aligned}\lambda\cdot\left(\begin{pmatrix}v_1\\v_2\end{pmatrix}+\begin{pmatrix}u_1\\u_2\end{pmatrix}\right)&=\lambda\cdot\begin{pmatrix}v_1+u_1\\v_2+u_2\end{pmatrix}=\begin{pmatrix}\lambda\cdot(v_1+u_1)\\\lambda\cdot(v_2+u_2)\end{pmatrix},\\&=\begin{pmatrix}\lambda\cdot v_1+\lambda\cdot u_1\\\lambda\cdot v_2+\lambda\cdot u_2)\end{pmatrix}=\begin{pmatrix}\lambda\cdot v_1\\\lambda\cdot v_2\end{pmatrix}+\begin{pmatrix}\lambda\cdot u_1\\\lambda\cdot u_2\end{pmatrix},\\&=\lambda\cdot\begin{pmatrix}v_1\\v_2\end{pmatrix}+\lambda\cdot\begin{pmatrix}u_1\\u_2\end{pmatrix}.\end{aligned} \tag{1.24}$$

3. Elle est *distributive* par rapport à la somme de $\mathbb{R}$

$$(\lambda+\mu)\cdot\vec{u}=\lambda\vec{u}+\mu\vec{u}, \tag{1.25}$$

soit en composantes

$$\begin{aligned}(\lambda+\mu)\cdot\begin{pmatrix}u_1\\u_2\end{pmatrix}&=\begin{pmatrix}(\lambda+\mu)\cdot u_1\\(\lambda+\mu)\cdot u_2\end{pmatrix}=\begin{pmatrix}\lambda\cdot u_1+\mu\cdot u_1\\\lambda\cdot u_2+\mu\cdot u_2\end{pmatrix},\\&=\lambda\cdot\begin{pmatrix}u_1\\u_2\end{pmatrix}+\mu\cdot\begin{pmatrix}u_1\\u_2\end{pmatrix}\end{aligned} \tag{1.26}$$

4. Elle vérifie *l'associativité* par rapport à la multiplication de $\mathbb{R}$

$$(\lambda \cdot \mu) \cdot \vec{u} = \lambda \cdot (\mu \cdot \vec{u}), \tag{1.27}$$

soit en composantes

$$\begin{aligned} (\lambda \cdot \mu) \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} &= \begin{pmatrix} (\lambda \cdot \mu) \cdot u_1 \\ (\lambda \cdot \mu) \cdot u_2 \end{pmatrix} = \begin{pmatrix} \lambda \cdot (\mu \cdot u_1) \\ \lambda \cdot (\mu \cdot u_2) \end{pmatrix}, \\ &= \lambda \cdot \begin{pmatrix} \mu \cdot u_1 \\ \mu \cdot u_2 \end{pmatrix} = \lambda \cdot \left(\mu \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \right). \end{aligned} \tag{1.28}$$

5. L'élément neutre de la multiplication de $\mathbb{R}$ est neutre à gauche

$$1 \cdot \vec{u} = \vec{u}, \tag{1.29}$$

soit en composantes

$$1 \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} 1 \cdot u_1 \\ 1 \cdot u_2 \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}. \tag{1.30}$$

1.4.3 Définition générale d'un espace vectoriel

Soit un ensemble K muni d'une "addition" et d'une "multiplication"[1] (par exemple les nombre réel, $\mathbb{R}$, les rationnels, $\mathbb{Q}$, ...). Les éléments de K sont appelés *scalaires*. Un espace vectoriel sur K est un ensemble V, dont les éléments sont appelés vecteurs, qui est muni de deux lois:

1. Une somme, notée $+$, qui est dite interne. C'est-à-dire que la somme de deux vecteurs de V est aussi un vecteur de V

$$+ : V \times V \to V. \tag{1.31}$$

2. La multiplication par un scalaire *à gauche*, qui est dite *externe*. C'est-à-dire que la multiplication à gauche associe à un scalaire dans K et un vecteur dans V un vecteur qui se trouve toujours dans V

$$\cdot : K \times V \to V. \tag{1.32}$$

Soit $\vec{u}, \vec{v}, \vec{w} \in V$, la somme possède les propriétés suivantes:

1. Elle est *associative*

$$(\vec{u} + \vec{v}) + \vec{w} = \vec{u} + (\vec{v} + \vec{w}). \tag{1.33}$$

2. Elle est *commutative*

$$\vec{u} + \vec{v} = \vec{v} + \vec{u}. \tag{1.34}$$

3. Elle admet un *élément neutre*, noté 0 (ou vecteur nul), tel que

$$\vec{u} + 0 = \vec{u}. \tag{1.35}$$

1. On dit que K est un corps commutatif.

4. Tout vecteur $\vec{v}$ admet un *opposé*, noté $-\vec{v}$, tel que

$$\vec{v} + (-\vec{v}) = 0. \tag{1.36}$$

Soit $\lambda, \mu \in K$ et $\vec{u}, \vec{v} \in V$, le produit satisfait les propriétés suivantes:

1. Elle est *distributive à gauche* par rapport à la somme de V

$$\lambda \cdot (\vec{u} + \vec{v}) = \lambda\vec{u} + \lambda\vec{v}. \tag{1.37}$$

2. Elle est *distributive à droite* par rapport à la somme de K

$$(\lambda + \mu) \cdot \vec{u} = \lambda\vec{u} + \mu\vec{u}. \tag{1.38}$$

3. Elle vérifie *l'associativité mixte* par rapport à la multiplication de K

$$(\lambda \cdot \mu) \cdot \vec{u} = \lambda \cdot (\mu \cdot \vec{u}). \tag{1.39}$$

4. L'élément neutre de la multiplication de K, noté 1, est neutre à gauche pour "$\cdot$"

$$1 \cdot \vec{u} = \vec{u}. \tag{1.40}$$

Avec cette définition, on voit que les n-uplets dotés des opérations d'addition et de multiplication qu'on a vues dans les sec. 1.2 et sec. 1.3 forme un espace vectoriel.

1.4.4 Exemples d'espace vectoriel

Il existe un très grand nombre d'espace vectoriel que vous utilisez quotidiennement sans savoir que s'en est un. Nous allons en voir un certain nombre.

1. L'espace des réels est un espace vectoriel sur les réels.
2. L'espace des entiers est un espace vectoriel sur les entiers.
3. L'espace des réels est un espace vectoriel sur les réels.
4. L'espace $\{0\}$ sur $\mathbb{R}$.
5. L'espace des fonctions définies telles que si

$$f : \mathbb{R} \to \mathbb{R}, \quad g : \mathbb{R} \to \mathbb{R}, \tag{1.41}$$

   et $\lambda \in \mathbb{R}$, on a

$$\begin{aligned} (f+g)(x) &= f(x) + g(x), \quad \forall x \in \mathbb{R}, \\ (\alpha \cdot f)(x) &= \alpha \cdot f(x), \quad \forall x \in \mathbb{R}. \end{aligned} \tag{1.42}$$

6. L'espace des polynômes sur les réels.

Exercice 1.1

Montrer que les points 4, 5 et 6 ci-dessus sont bien des espaces vectoriels.

1.4.5 Combinaisons linéaires

Une des raisons pour laquelle, nous définissons les espaces vectoriels est la possibilité d'effectuer des combinaisons linéaires dans ces espaces.

Soit un espace vectoriel E sur K, muni des opérations "+"" et "·". Soient $\vec{v}_1, \vec{v}_2, ..., \vec{v}_n \in E$ et $\lambda_1, ..., \lambda_n \in K$, une combinaison linéaire de $\vec{v}_1, ..., \vec{v}_n$

$$\sum_{i=1}^{n} \lambda_i \cdot \vec{v}_i = \lambda_1 \cdot \vec{v}_1 + ... + \lambda_n \cdot \vec{v}_n. \tag{1.43}$$

A l'aide de la définition de l'espace vectoriel, nous savons que le vecteur résultant de n'importe quelle combinaison linéaire de l'ensemble E sera toujours un élément de E. Une question intéressante à se poser à présent, c'est de savoir quel est l'ensemble qu'on peut générer en faisant des combinaisons linéaires d'un ensemble de $\{\vec{v}_i\}_{i=1}^{n}$ (cet ensemble de vecteurs s'appelle un *famille de vecteurs*?

Exercice 1.2

1. Considérons deux vecteurs $\vec{v}_1$, $\vec{v}_2$ donnés par

$$\vec{v}_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad \vec{v}_2 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}. \tag{1.44}$$

   Quel est l'espace généré par les combinaisons linéaires de ces deux vecteurs?
2. Considérons deux $\vec{v}_1$, $\vec{v}_2$ donnés par

$$\vec{v}_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad \vec{v}_2 = \begin{pmatrix} -1 \\ -2 \end{pmatrix}. \tag{1.45}$$

   Quel est l'espace généré par les combinaisons linéaires de ces deux vecteurs?
3. Considérons le vecteur $\vec{v}$ donnés par

$$\vec{v} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{1.46}$$

   Quel est l'espace généré par les combinaisons linéaires de $\vec{v}$?
4. Considérons trois vecteurs $\vec{v}_1$, $\vec{v}_2$, et $\vec{v}_3$ donnés par

$$\vec{v}_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad \vec{v}_2 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \quad \vec{v}_3 = \begin{pmatrix} 3 \\ 4 \end{pmatrix}. \tag{1.47}$$

   Quel est l'espace généré par les combinaisons linéaires de ces trois vecteurs?

1.4.6 Familles libres et liées

Dans les cas 1 et 3, on dit que la famille de vecteurs est *libre*. En d'autres termes, on ne peut pas obtenir un des vecteurs de famille en faisant des combinaisons linéaires des autres vecteurs de la famille. Ils sont *linéairement indépendants*.

A l'inverse 2 et 4, sont des familles liées: on peut obtenir au moins un des vecteurs de la famille par combinaison linéaire des vecteurs de la famille. Ils sont *linéairement dépendants*.

En notation mathématique, si on considère un ensemble $\{\vec{v}_i\}_{i=1}^n \in E$ et $\{\lambda_i\}_{i=1}^n \in K$ (E est un espace vectoriel sur K). Alors on dit que les vecteurs $\vec{v}_i$ sont *linéairement dépendants* si et seulement si

$$\sum_{i=1}^{n} \lambda_i \vec{v}_i = 0, \tag{1.48}$$

avec au moins un $\lambda_i \neq 0$. On peut réécrire cette condition comme

$$\vec{v}_n = \sum_{i=1}^{n-1} \mu_i \vec{v}_i, \tag{1.49}$$

où au moins un des $\mu_i \neq 0$.

Inversement s'il n'existe pas de λ_i non nul tel que

$$\sum_{i=1}^{n} \lambda_i \vec{v}_i = 0, \tag{1.50}$$

alors les vecteurs sont indépendants.

Exercice 1.3

Déterminer si les vecteurs des familles de l'exercice précédent sont linéairement dépendants ou indépendants.

1.4.7 Base d'espace vectoriel

Soit à présent un ensemble de vecteur $B = \{\vec{b}_i\}_{i=1}^n$ qui sont linéairement indépendants (c'est une famille libre) et l'ensemble E, un espace vectoriel, généré par toutes les combinaisons linéaires de B. Alors on dit que l'ensemble B est une *base* de E. N'importe quel vecteur de E peut être obtenu comme une combinaison linéaire des vecteurs de la base B.

En revanche, si $V = \{\vec{v}_i\}_{i=1}^n$$ est une famille liée (ses vecteurs sont linéairement dépendant) et même s'ils génèrent le même ensemble E. L'ensemble V ne forme pas une base. Une base est le plus "petit" ensemble de vecteur générant E.

Exercice 1.4

Quelles familles de l'exercice 2 forment un base de l'espace généré par les familles?

De façon générale il n'existe pas une base unique d'un espace vectoriel. Dans l'espace $\mathbb{R}^2$, n'importe quelle paire de vecteurs linéairement indépendants forment une base. A contrario, on sait que n'importe quelle famille contenant au moins 3 vecteurs ne formeront pas une base de $\mathbb{R}^2$.

Chapitre 2

Les applications linéaires

2.1 Rappel sur les fonctions

Une fonction, notée f (quelle originalité), est une relation entre deux ensembles. Soient deux ensembles X et Y, cette fonction va associer tous les éléments de X à un élément de Y (voir la fig. 2.1)

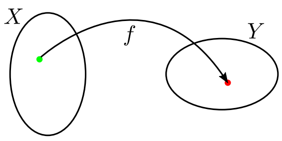

Figure 2.1 – Une fonction est une relation entre deux ensemble, X et Y, qui va relier un point de X avec un point de Y.

On note de façon formelle

$$f : X \to Y. \tag{2.1}$$

Ici X est le *domaine de définition* de f et Y est le *domaine d'arrivée.* Cette notation nous donne juste le domaine de définition et le domaine d'arrivée de f mais ne nous dit pas quelle est la règle d'association entre les éléments de X et les éléments de Y.

Pour un exemple concret, on peut prendre

$$f : \mathbb{R} \to \mathbb{R} \tag{2.2}$$

, où f est définie par

$$f : x \to x^2, \quad \text{ou} \quad f(x) = x^2. \tag{2.3}$$

On a donc ici que $\mathbb{R}$ est le domaine de définition et d'arrivée de f. En revanche tout $\mathbb{R}$ n'est pas "couvert" par les éléments de $f(x)$. On appelle l'image de f le sous-ensemble du domaine d'arrivée qui est effectivement atteint par $f(x)$. Dans le cas ci-dessus le carré d'un nombre ne peut être négatif, donc tous les réels ne peuvent pas être associés à la fonction x^2 et donc l'image de f est un sous-ensemble de f. Ici, l'image de f est $\mathbb{R}^+$ (les nombres réels $x \in \mathbb{R}$ tels que $x \geq 0$).

On a vu ici une fonction unidimensionnelle, mais le concept de fonction s'étend à n'importe quel ensemble. Dans ce qui va nous intéresser pour la suite, on va avoir des fonctions de n-uplets ou de vecteurs à n dimensions. En fait de façon aussi générale que possible, on va associer un n-uplet à un m-uplet (ou un vecteur de dimension n à un vecteur de dimension m)

$$f : \mathbb{R}^n \to \mathbb{R}^m. \tag{2.4}$$

Exemple 2.1 *(Translation)*

Une telle fonction peut être une translation horizontale en deux dimensions

$$\begin{aligned} &f : \mathbb{R}^2 \to \mathbb{R}^2, \\ &f : \vec{x} \to \vec{x} + \begin{pmatrix} 2 \\ 0 \end{pmatrix}. \end{aligned} \tag{2.5}$$

2.2 Les fonctions linéaires

On dit qu'une fonction, f, (ou application) est linéaire si pour E, V des espaces vectoriels sur V avec

$$f : E \to V. \tag{2.6}$$

Et soient $\vec{x}, \vec{y} \in E$, et $\lambda \in K$, on a

1. La fonction appliquées à $\vec{x} + \vec{y}$ est égal à la somme de f appliqué à $\vec{x}$ avec f appliqué à y
$$f(\vec{x} + \vec{y}) = f(\vec{x}) + f(\vec{y}). \tag{2.7}$$
2. Le produit de λ avec f appliqué à $\vec{x}$ est égal à f appliqué à $\lambda \cdot \vec{x}$
$$\lambda \cdot f(\vec{x}) = f(\lambda \cdot \vec{x}). \tag{2.8}$$

Exemple 2.2

Déterminer si la fonction

$$\begin{aligned} &f : \mathbb{R}^2 \to \mathbb{R}^2, \\ &f : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to \begin{pmatrix} x_1 + x_2 \\ 2 \cdot x_1 \end{pmatrix}. \end{aligned} \tag{2.9}$$

est linéaire?

Il faut vérifier les conditions 1 et 2 ci-dessus. Commençons par (1)

$$f(\vec{x} + \vec{y}) = f\begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \end{pmatrix} = \begin{pmatrix} x_1 + y_1 + x_2 + y_2 \\ 2 \cdot (x_1 + y_1) \end{pmatrix}, \tag{2.10}$$

et

$$f(\vec{x})+f(\vec{y})=\begin{pmatrix}x_1+x_2\\2\cdot x_1\end{pmatrix}+\begin{pmatrix}y_1+y_2\\2\cdot y_1\end{pmatrix}=\begin{pmatrix}x_1+y_1+x_2+y_2\\2\cdot(x_1+y_1)\end{pmatrix}. \tag{2.11}$$

On voit que l'éq. 2.10 et l'éq. 2.11 sont égales et donc la fonction f est linéaire.

Exercice 2.1 *(Translation)*

Est-ce que la translation horizontale de $\begin{pmatrix}2\\0\end{pmatrix}$ est linéaire?

$$\begin{aligned}&f:\mathbb{R}^2\to\mathbb{R}^2,\\&f:\begin{pmatrix}x_1\\x_2\end{pmatrix}\to\begin{pmatrix}x_1\\x_2\end{pmatrix}+\begin{pmatrix}2\\0\end{pmatrix}.\end{aligned} \tag{2.12}$$

2.2.1 Exemple de transformation linéaire sur un triangle: la rotation

En informatique, une grande partie des applications 3D sont basées sur la manipulation de triangles (pour représenter des surfaces). Définissons un triangle relie les trois points

$$P_0=(0,0),\quad P_1=(1,0),\quad P_2=(0,1). \tag{2.13}$$

Les positions P_0, P_1 et P_2 suffisent pour décrire le triangle. Cependant afin de les dessiner, il peut être utile de connaître également les équations des segments reliant les points. Cela peut se faire via les vecteurs, $\vec{x}_0$, $\vec{x}_1$ et $\vec{x}_2$, reliant l'origine aux points \$(voir la fig. 2.2)

$$\vec{x}_0=\begin{pmatrix}0\\0\end{pmatrix},\quad \vec{x}_1=\begin{pmatrix}1\\0\end{pmatrix},\quad \vec{x}_2=\begin{pmatrix}0\\1\end{pmatrix}. \tag{2.14}$$

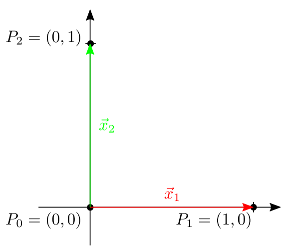

Figure 2.2 – Les points P_0, P_1, et P_2. Les vecteurs $\vec{x}_1$ et $\vec{x}_2$.

Les équations paramétriques des segments reliant les points peuvent s'écrire en

$$\begin{aligned}&\vec{s}_0=\lambda\cdot\vec{x}_1, &&\lambda\in[0,1],\\&\vec{s}_1=\lambda\cdot\vec{x}_2, &&\lambda\in[0,1],\\&\vec{s}_2=\lambda\cdot\vec{x}_1+(1-\lambda)\cdot\vec{x}_2, &&\lambda\in[0,1].\end{aligned} \tag{2.15}$$

Ces trois vecteurs sont des *combinaisons linéaires* des vecteurs $\vec{x}_1$ et $\vec{x}_2$. Finalement, le triangle peut donc s'écrire comme l'ensemble des trois segments

$$\triangle = \{\vec{s}_0, \vec{s}_1, \vec{s}_2\}. \tag{2.16}$$

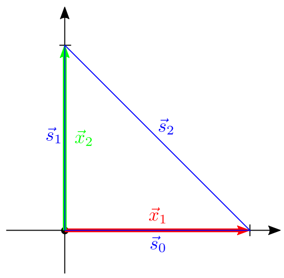

Figure 2.3 – Les segments $\vec{s}_0$, $\vec{s}_1$, et $\vec{s}_2$ relient les sommets des triangles. On peut exprimer leur équations paramétriques à l'aide des vecteurs $\vec{x}_1$, $\vec{x}_2$.

Définissons à présent l'application suivante

$$R : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix}. \tag{2.17}$$

Question 2.1

Cette application est-elle linéaire?

L'effet de notre notre application R sur les trois vecteurs $\vec{x}_i$ est la suivante

$$\begin{aligned} R(\vec{x}_0) &= \vec{x}_0' = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \\ R(\vec{x}_1) &= \vec{x}_1' = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \\ R(\vec{x}_2) &= \vec{x}_2' = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \end{aligned} \tag{2.18}$$

Comme on peut le voir sur la fig. 2.4 notre application linéaire R, effectue une rotation de $\pi/2$ dans le sens des aiguilles d'une montre.

Les équations du triangle peuvent être obtenu de deux façons différentes. On veut calculer

$$\triangle' = \{\vec{s}_0', \vec{s}_1', \vec{s}_2'\}. \tag{2.19}$$

Soit on part des $\vec{x}_i'$ et on refait le même raisonnement que précédemmen§t pour relier les sommets des vecteurs et on obtient, $\vec{s}_0'$, $\vec{s}_1'$, et $\vec{s}_2'$ qui sont donnés par

$$\begin{aligned} \vec{s}_0' &= \lambda \cdot \vec{x}_1', && \lambda \in [0,1], \\ \vec{s}_1' &= \lambda \cdot \vec{x}_2', && \lambda \in [0,1], \\ \vec{s}_2' &= \lambda \cdot \vec{x}_1' + (1-\lambda) \cdot \vec{x}_2', && \lambda \in [0,1]. \end{aligned} \tag{2.20}$$

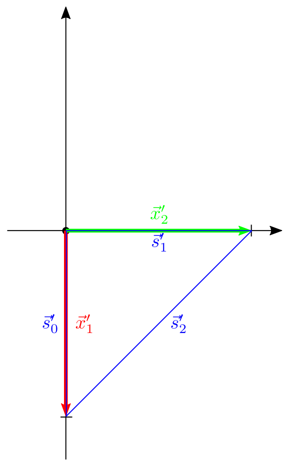

Figure 2.4 – Après application de la fonction linéaire R, les segments $\vec{s}_0'$, $\vec{s}_1'$, et $\vec{s}_2'$ relient les sommets des triangles qui peuvent être représents par les vecteurs $\vec{x}_0'$, $\vec{x}_1'$ et $\vec{x}_2'$

Soit on peut appliquer R à $\vec{s}_0$, $\vec{s}_1$, et $\vec{s}_2$. Commençons par le calcul de $R(\vec{s}_0)$

$$\vec{s}_0 = R(\vec{s}_0) = R(\lambda \cdot \vec{x}_1) = \lambda \cdot R(\vec{x}_1) = \lambda \vec{x}_1', \tag{2.21}$$

où à la deuxième égalité on a utilisé la propriété (2) de l'application linéaire. De même on peut écrire $\vec{s}_1'$

$$\vec{s}_1' = R(\vec{s}_1) = R(\lambda \cdot \vec{x}_2) = \lambda \cdot R(\vec{x}_2) = \lambda \vec{x}_2'. \tag{2.22}$$

Pour le cas de $\vec{s}_2'$, les choses sont un tout petit peu plus compliquées

$$\begin{aligned}\vec{s}_2' = R(\vec{s}_2) &= R(\lambda \cdot \vec{x}_1 + (1-\lambda) \cdot \vec{x}_2) = \lambda \cdot R(\vec{x}_1) + (1-\lambda) \cdot R(\vec{x}_2) \\ &= \lambda \cdot \vec{x}_1' + (1-\lambda) \cdot \vec{x}_2',\end{aligned} \tag{2.23}$$

où cette fois nous avons utilisé les propriétés (1) et (2) de l'application linéaire.

On voit qu'on a donc deux méthodes équivalentes pour calculer les segments constituant un traingle. Soit on calcule la rotation des segments en leur appliquant la rotation R, soit on effectue la rotation des vecteurs $\vec{x}_i$ et on calcule les segments à partir des $\vec{x}_i$ tournés.

On constate également que cette application linéaire garde intacte la structure de notre triangle: les droites restent des droites et elles restent conectées entre elles. Cette propriété est générale pour toutes les applications linéaires. Les droites restent des droites, et la structure des objet reste la même.

Ici nous avons considéré une rotation qui est une application linéaire spéciale. Sous rotation les longueur et les angles d'un objet restent les mêmes.

2.2.2 Exemple de transformation linéaire sur un triangle: la dilatation

Si nous reprenons l'exemple de notre triangle qui relie les points P_0, P_1, et P_2 et que nous appliquons cette fois l'application linéaire D définie par

$$D: \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow \begin{pmatrix} 2 \cdot x_1 \\ 3 \cdot x_2 \end{pmatrix}. \tag{2.24}$$

On va obtenir que les vecteurs $\vec{x}_i$ $(i = 0, \ldots, 2)$ seront transformés comme (voir la fig. 2.5)

$$\begin{aligned} \vec{x}'_0 &= D(\vec{x}_0) = \vec{x}_0, \\ \vec{x}'_1 &= D(\vec{x}_1) = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \\ \vec{x}'_2 &= D(\vec{x}_2) = \begin{pmatrix} 0 \\ 3 \end{pmatrix}. \end{aligned} \tag{2.25}$$

Les segments seront donc simplement obtenus par $\triangle' = D(\triangle)$

$$\begin{aligned} \vec{s}'_0 &= \lambda \cdot \vec{x}'_1, & \lambda &\in [0, 1], \\ \vec{s}'_1 &= \lambda \cdot \vec{x}'_2, & \lambda &\in [0, 1], \\ \vec{s}'_2 &= \lambda \cdot \vec{x}'_1 + (1 - \lambda) \cdot \vec{x}'_2, & \lambda &\in [0, 1]. \end{aligned} \tag{2.26}$$

Un première chose à réaliser est que ces équations ont la même structure que les équations éq. 2.18 bien que l'effet l'application linéaire soit très différentes.

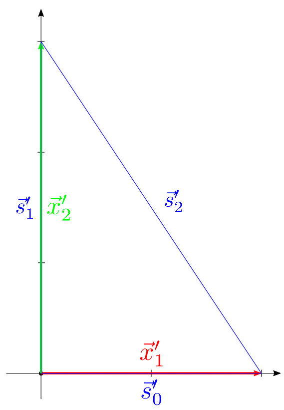

Figure 2.5 – Après application de la fonction linéaire D, les segments $\vec{s}'_0$, $\vec{s}'_1$, et $\vec{s}'_2$ relient les sommets des triangles qui peuvent être représents par les vecteurs $\vec{x}'_0$, $\vec{x}'_1$ et $\vec{x}'_2$

Sur la fig. 2.5, nous constatons que les vecteurs $\vec{x}_1$ et $\vec{x}_2$ d'un facteur deux et d'un facteur trois respectivement. En revanche, la figure géométrique transformée reste toujours un triangle. Comme pour la rotation, l'application de cette fonction linéaire n'a pas transformé les segment de droite en une autre figure géométrique: ils sont restés des segments de droites.

2.2.3 Exemple de transformation non-linéaire

Nous avons déjà vu (voir Exercice 2.1 (Translation)) que la translation n'est pas une application linéaire. Considérons à présent une autre application non-linéaire, g, définie comme

$$g : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to \begin{pmatrix} x_1^2 \\ x_2^2 \end{pmatrix}. \tag{2.27}$$

L'effet de g sur les vecteurs $\vec{x}_i$ est le suivant

$$\begin{aligned} \vec{x}_0' &= g(\vec{x}_0) = \vec{x}_0, \\ \vec{x}_1' &= g(\vec{x}_1) = \vec{x}_1, \\ \vec{x}_2' &= g(\vec{x}_2) = \vec{x}_2. \end{aligned} \tag{2.28}$$

L'effet sur les segment en revanche est très différent

$$\begin{aligned} \vec{s}_0' &= g(\vec{s}_0) = \lambda^2 \vec{x}_1, \\ \vec{s}_1' &= g(\vec{s}_1) = \lambda^2 \vec{x}_2, \\ \vec{s}_2' &= g(\vec{s}_2) = \lambda^2 \vec{x}_1 + (1-\lambda)^2 \vec{x}_2. \end{aligned} \tag{2.29}$$

On voit donc que comme prévu on a que

$$\begin{aligned} \vec{s}_0' &\neq \lambda \vec{x}_1, \\ \vec{s}_1' &\neq \lambda \vec{x}_2, \\ \vec{s}_2' &\neq \lambda \vec{x}_1 + (1-\lambda) \vec{x}_2, \end{aligned} \tag{2.30}$$

comme cela serait le cas pour une application linéaire.

L'effet de cette application non-linéaire peut se voir sur la fig. 2.6. On voit que les deux segments alignés avec les axes principaux restent des droites. En revanche l'hypoténuse du triangle n'est plus une droite mais un bout de parabole. Contrairement aux applications linéaires, on voit que les applications non-linéaires peuvent transformer les droites en d'autres figures géométriques.

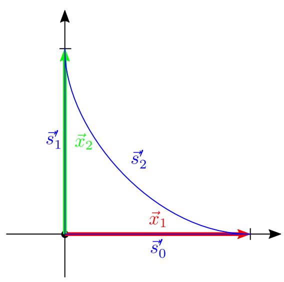

Figure 2.6 – L'effet de l'application non-linéaire g sur le triangle constitué des points P_0, P_1, P_2.

2.3 Les applications linéaires comme un produit matrice-vecteur

Dans cette section, nous allons montrer qu'une application linéaire peut s'écrire sous la forme du produit entre une matrice et un vecteur.

2.3.1 Introduction aux matrices

Une matrice est un tableau de nombres à double entrée qui de façon générale a un nombre m de lignes et n de colonnes. On note l'espace des matrices de taille $m \times n$ à coefficient réels

$$M_{m,n}(\mathbb{R})[^3]. \tag{2.31}$$

Par exemple la matrice $\underline{\underline{A}}$[1] suivante est de taille 2×3

$$\underline{\underline{A}} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix}, \tag{2.32}$$

elle possède deux lignes et trois colonnes. Une matrice, $\underline{\underline{A}}$ de taille m $\times\, n$, se note

$$\underline{\underline{A}} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}, \tag{2.33}$$

où les a_{ij} sont les indice de la matrices $\underline{\underline{A}}$, avec $i = 1..m$ et $j = 1..n$.

Si $m = 1$, alors la matrice n'est rien d'autre qu'un vecteur *ligne*

$$\underline{\underline{A}} = \begin{pmatrix} a_1 & a_2 & \cdots & a_n \end{pmatrix}. \tag{2.34}$$

Alors que si $n = 1$, alors la matrice est un vecteur *colonne*

$$\underline{\underline{A}} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix}. \tag{2.35}$$

On peut aussi voir une matrice $m \times n$ comme un vecteur “ligne” à n composantes de vecteurs “colonne”

$$\underline{\underline{A}} = \begin{pmatrix} \vec{a}_1 & \vec{a}_2 & \cdots & \vec{a}_n \end{pmatrix}, \tag{2.36}$$

où tous les $\vec{a}_i$ ont une longueur m. C'est souvent la structure de donnée utilisée en informatique. En python on écrirait une matrice 2×3 comme ci-dessous.

```
>>> [[1, 2, 3], [2, 3, 4]] # a list of list
[[1, 2, 3], [2, 3, 4]]
>>> import numpy as np
>>> np.matrix([[1, 2, 3], [2, 3, 4]]) # the matrix structure un numpy
matrix([[1, 2, 3],
        [2, 3, 4]])
```

1. Une autre notation est $\boldsymbol{A}$.

On constate que la convention utilisée par NumPy est que nous construisons une matrice comme un vecteur “colonne” de taille n, composé de m vecteurs “ligne”.

Afin d’accéder à l’élément i, j de la matrice, on peut utiliser la notation $\{\underline{\underline{A}}\}_{ij} = a_{ij}$. Dans le cas de la matrice de l’éq. 2.32, l’élément A_{12} est en rouge ci-dessous

$$\underline{\underline{A}} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix}, \tag{2.37}$$

ou encore

$$\{\underline{\underline{A}}\}_{12} = a_{12} = 2. \tag{2.38}$$

En fait il se trouve que l’espace des matrices, de taille $m \times n$, à coefficients réels est un espace vectoriel [2] dans $\mathbb{R}$, muni de la somme et de la multiplication par un scalaire qui est “la même” que pour les vecteurs. Cette propriété a pour effet que la somme de deux matrices sera toujours une matrice et que la multiplication par un scalaire d’une matrice sera toujours une matrice. Voyons à présent comment on définit ces deux opérations.

2.3.2 La somme de matrices

Soient $\underline{\underline{A}}$ et $\underline{\underline{B}}$ deux matrices à coefficients réels. On a que l’opération “+” est simplement la somme de chaque élément de la matrice $\underline{\underline{A}}$ avec l’élément correspondant de la matrice $\underline{\underline{B}}$

$$\{\underline{\underline{A}} + \underline{\underline{B}}\}_{ij} = a_{ij} + b_{ij}. \tag{2.39}$$

Remarque 2.1

Il faut noter que pour que cette opération soit bien définie la taille de la matrice $\underline{\underline{A}}$ et de la matrice $\underline{\underline{B}}$ doivent être la même.

Exemple 2.3 *(Somme de matrices)*

La somme des matrices $\underline{\underline{A}}$ et $\underline{\underline{B}}$ données par

$$\underline{\underline{A}} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix}, \underline{\underline{B}} = \begin{pmatrix} -1 & 2 & 1 \\ -1 & 1 & -3 \end{pmatrix}, \tag{2.40}$$

est donnée par

$$\underline{\underline{A}} + \underline{\underline{B}} = \begin{pmatrix} 0 & 4 & 4 \\ 1 & 4 & 1 \end{pmatrix}. \tag{2.41}$$

En Python la somme peut se faire en parcourant les indices dans deux boucles ou à l’aide de la librairie NumPy

2. On peut remplacer $\mathbb{R}$ par un autre ensemble, comme $\mathbb{N}$, $\mathbb{Q}$, ou encore $\mathbb{C}$.

```
>>> A=[[ 1, 2, 3], [ 2, 3,  4]]
>>> B=[[-1, 2, 1], [-1, 1, -3]]
>>> C=[[0,0,0],[0,0,0]]
>>> for i in range(2):
...     for j in range(3):
...             C[i][j]=A[i][j]+B[i][j]
...
>>> C
[[0, 4, 4], [1, 4, 1]]
>>> import numpy as np
>>> A=np.matrix([[ 1, 2, 3], [ 2, 3,  4]])
>>> b=np.matrix([[-1, 2, 1], [-1, 1, -3]])
>>> A+B
matrix([[0, 4, 4],
        [1, 4, 1]])
```

2.3.3 La multiplication avec un scalaire d'une matrice

Soit $\underline{\underline{A}}$ une matrices à coefficients réels et $\lambda\mathbb{R}$. On a que l'opération "$\cdot$" avec un scalaire est simplement le produit de chaque élément de la matrice $\underline{\underline{A}}$ avec λ

$$\{\lambda \cdot \underline{\underline{A}}\}_{ij} = \lambda \cdot a_{ij}. \tag{2.42}$$

Exemple 2.4 *(Produit de matrice avec un scalaire)*

Le produit de la matrice $\underline{\underline{A}}$ donnée par

$$\underline{\underline{A}} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix}, \tag{2.43}$$

et $\lambda = 2$ est donné par

$$\lambda \cdot \underline{\underline{A}} = \begin{pmatrix} 2 & 4 & 6 \\ 4 & 6 & 8 \end{pmatrix}. \tag{2.44}$$

En Python ce produit peut se faire en parcourant les indices dans deux boucles ou à l'aide de la librairie NumPy

```
>>> A=[[ 1, 2, 3], [ 2, 3,  4]]
>>> C=[[0,0,0],[0,0,0]]
>>> alpha = 2
>>> for i in range(2):
...     for j in range(3):
...             C[i][j]=alpha*A[i][j]
...
>>> C
[[2, 4, 6], [4, 6, 8]]
>>> import numpy as np
>>> A=np.matrix([[ 1, 2, 3], [ 2, 3,  4]])
```

```
>>> alpha = 2
>>> alpha * A
matrix([[2, 4, 6],
        [4, 6, 8]])
```

2.3.4 Le produit matrice-vecteur

Après avoir défini le produit d'une matrice avec un scalaire et l'addition de matrice. Il peut être très pratique de disposer d'un produit entre une matrice et un vecteur.

Soient une matrice, $\underline{\underline{A}}$, de taille $m \times n$ et un vecteur, $\vec{v}$, de taille n. Le produit de la matrice et du vecteur, qui aura comme résultat un vecteur de taille m, et sera noté $\vec{u} = \underline{\underline{A}} \cdot \vec{v}$, se définit comme

$$\{\underline{\underline{A}} \cdot \vec{v}\}_i = u_i = \sum_{k=1}^{n} a_i j \cdot v_j = a_{i1} \cdot a_1 + \cdots + a_{in} \cdot a_n, \text{ avec } i = 1, ..., m. \quad (2.45)$$

On constate d'après cette définition qu'il est très important que le nombre de colonnes du vecteur soit le même que le nombre de lignes du vecteur, sinon ce produit est mal défini.

Exemple 2.5 *(Matrice identité)*

Soit la matrice $\underline{\underline{I}}$, qui est une matrice 2×2, qui est définie par

$$\underline{\underline{I}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \quad (2.46)$$

On définit également un vecteur $\vec{v} \in \mathbb{R}^2$

$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}. \quad (2.47)$$

Calculer le produit $\underline{\underline{I}} \cdot \vec{v}$.

Solution 2.1 *(Matrice identité)*

En utilisant la définition de l'éq. 2.45, on a que

$$\{\underline{\underline{I}} \cdot \vec{v}\}_i = \sum_{i=1}^{2} i_{ij} \cdot v_j, \quad (2.48)$$

ou réécrit par composante

$$\begin{aligned} i = 1 : \quad & \{\underline{\underline{I}} \cdot \vec{v}\}_1 = \sum_{i=1}^{2} i_{1j} \cdot v_j = i_{11} \cdot v_1 + i_{12} \cdot v_2 = v_1, \\ i = 2 : \quad & \{\underline{\underline{I}} \cdot \vec{v}\}_2 = \sum_{i=1}^{2} i_{2j} \cdot v_j = i_{21} \cdot v_1 + i_{22} \cdot v_2 = v_2. \end{aligned} \quad (2.49)$$

Grâce à cet exemple on se rend compte que la matrice identité est l'élément neutre pour le produit matrice-vecteur: elle laisse le vecteur inchangé.

Le produit matrice-vecteur s'écrit en Python

```
>>> A=[[ 1, 2, 3], [ 2, 3,  4]]
>>> x=[-1,1,-1]
>>> b=[0,0]
>>> for i in range(2):
...     for j in range(3):
...             b[i] += A[i][j]*x[j]
...
>>> b
[-2, -3]
>>> import numpy as np
>>> A=np.matrix(A)
>>> x=np.array(x)
>>> A.dot(x)
matrix([[-2, -3]])
```

Remarque 2.1

Le produit d'une matrice avec un vecteur n'est pas commutatif.

Le produit d'une matrice avec un vecteur est une application linéaire. On peut montrer que si $\underline{\underline{A}} \in M_{m,n}(\mathbb{R})$ est une matrice à coefficients réels de dimension $m \times n$, $\lambda \in \mathbb{R}$ un scalaire, et $\vec{u}\vec{v} \in \mathbb{R}^n$ deux vecteurs colonnes de taille n, alors

$$\begin{aligned} \underline{\underline{M}} \cdot (\vec{u} + \vec{v}) = \underline{\underline{M}} \cdot \vec{u} + \underline{\underline{M}}\vec{v}, \\ \lambda \cdot \underline{\underline{M}}\vec{u} = \underline{\underline{M}} \cdot (\lambda \cdot \vec{u}). \end{aligned} \tag{2.50}$$

Exerice 2.1

Démontrer que le produit matrice vecteur est une application linéaire.

2.3.5 Le produit matrice-vecteur dans le cas de la rotation

L'application linéaire pour la rotation bidimensionnelle de $\pi/2$ dans le sens des aiguilles d'une montre est définie comme

$$R : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix}. \tag{2.51}$$

En fait cette application peut s'écrire comme un produit de matrice avec un vecteur

$$R : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to \underline{\underline{R}} \cdot \vec{x}, \tag{2.52}$$

où

$$\underline{\underline{R}} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}. \tag{2.53}$$

On peut donc écrire

$$\underline{\underline{R}} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix}. \tag{2.54}$$

Nous pouvons à présent calculer la rotation des trois vecteurs que nous avions défini à l'éq. 2.14

$$\vec{x}_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \vec{x}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \vec{x}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

On a donc

$$\vec{x}_0' = \underline{\underline{R}} \cdot \vec{x}_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$
$$\vec{x}_1' = \underline{\underline{R}} \cdot \vec{x}_1 = \begin{pmatrix} 0 \\ -1 \end{pmatrix},$$
$$\vec{x}_2' = \underline{\underline{R}} \cdot \vec{x}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Qui est exactement le même résultat que ce qu'on avait obtenu dans l'éq. 2.18.

Chapitre 3

Remerciements

Je voudrais remercier l'étudiant du cours qui a contribué à améliorer ce polycopié.
Merci à M. Chételat.