

APPLICATIONS LINÉAIRES ET MATRICES

Résumé de cours d'algèbre linéaire L1 de B. Calmès, Université d'Artois
(version du 21 mars 2020)

1. APPLICATIONS LINÉAIRES

Soient E et F des espaces vectoriels sur \mathbb{K} .

1.1. **Définition.** Une application $f : E \rightarrow F$ est dite *linéaire* si

$$f(x + y) = f(x) + f(y) \quad \forall x, y \in E$$

et

$$f(\lambda x) = \lambda f(x) \quad \forall \lambda \in \mathbb{K} \text{ et } \forall x \in E$$

1.2. **Proposition.** L'application $f : E \rightarrow F$ est linéaire si et seulement si

$$f(\lambda x + \mu y) = \lambda f(x) + \mu f(y) \quad \forall \lambda, \mu \in \mathbb{K} \text{ et } \forall x, y \in E$$

si et seulement si

$$f\left(\sum_i \lambda_i x_i\right) = \sum_i \lambda_i f(x_i) \quad \forall \lambda_1, \dots, \lambda_n \in \mathbb{K} \text{ et } \forall x_1, \dots, x_n \in E.$$

De plus, on a alors $f(0_E) = 0_F$.

1.3. **Notation.** L'ensemble des applications linéaires de E dans F se note $\mathcal{L}(E, F)$. Lorsque $E = F$, on utilise également la notation $\mathcal{L}(E)$ au lieu de $\mathcal{L}(E, E)$.

1.4. **Proposition.** La composée de deux applications linéaires est linéaire. Toute combinaison linéaire d'applications linéaires est linéaire. Ce dernier point muni naturellement $\mathcal{L}(E, F)$ d'une structure d'espace vectoriel.

1.5. **Théorème.** Une application linéaire est caractérisée par l'image d'une base : Si $(e_i)_{i \in I}$ est une base de E et $(f_i)_{i \in I}$ sont des vecteurs de F , alors il existe une unique application linéaire $f : E \rightarrow F$ telle que $f(e_i) = f_i$ pour tout $i \in I$.

1.6. **Rappel.** Soit $f : E \rightarrow F$ une application (quelconque). Soit $G \subset E$ un sous-ensemble (quelconque) de E , et soit $H \subset F$ un sous-ensemble (quelconque) de F . Alors on appelle *image de G par f* le sous-ensemble de F

$$f(G) = \{f(x), x \in G\}$$

et on appelle *image réciproque de H par f* le sous-ensemble de E

$$f^{-1}(H) = \{x \in E, f(x) \in H\}.$$

1.7. **Proposition.** Soit $f : E \rightarrow F$ une application linéaire. Alors

- (1) l'image par f d'un sous-espace vectoriel de E est un sous-espace vectoriel de F .
- (2) $f(\text{Vec}(v_i, i \in I)) = \text{Vec}(f(v_i), i \in I)$
- (3) l'image réciproque par f d'un sous-espace vectoriel de F est un sous-espace vectoriel de E .

1.8. Définition. Soit $f : E \rightarrow F$ une application linéaire. On appelle *noyau* de f le sous-espace vectoriel de E $f^{-1}(\{0_F\})$. On appelle *image* de f le sous-espace vectoriel $f(E)$ de F .

Autrement dit, $x \in \ker(f)$ si et seulement si $f(x) = 0$, et $y \in \text{im}(f)$ si et seulement si $\exists x \in E$ tel que $f(x) = y$.

1.9. Remarque. Le point (2) de la proposition 1.7 nous dit que si $(e_i)_{i \in I}$ est une base de E (ou même seulement une famille génératrice de E), alors $(f(e_i))_{i \in I}$ est une famille génératrice de $\text{im}(f)$.

1.10. Théorème. Une application linéaire $f : E \rightarrow F$ est

- (1) injective si et seulement si $\ker(f) = \{0_E\}$;
- (2) surjective si et seulement si $\text{im}(f) = F$;
- (3) bijective si et seulement si $\ker(f) = \{0_E\}$ et $\text{im}(f) = F$. Dans ce cas l'application réciproque f^{-1} est également linéaire.

1.11. Théorème. Soit $f : E \rightarrow F$ une application linéaire.

- (1) f est injective si et seulement si l'image de toute famille libre de E est une famille libre de F , si et seulement si (lorsque E est de type fini) il existe une base de E dont l'image est une famille libre de F .
- (2) f est surjective si et seulement si l'image de toute famille génératrice de E est une famille génératrice de F , si et seulement si il existe une famille génératrice de E dont l'image est génératrice de F .
- (3) f est bijective si et seulement si l'image de toute base de E (de type fini) est une base de F , si et seulement si il existe une base de E (de type fini) dont l'image est une base de F .

2. DIMENSION DU NOYAU ET RANG

Soit $f : E \rightarrow F$ est une application linéaire.

2.1. Définition. Si $\text{im}(f)$ est de type fini, on appelle *rang* de f , noté $\text{rg}(f)$, la dimension de $\text{im}(f)$.

2.2. Remarque. Le sous-espace vectoriel $\text{im}(f)$ est automatiquement de type fini si E ou F est de type fini.

2.3. Théorème (du rang). Si E est de type fini, alors on a

$$\dim(E) = \dim(\ker(f)) + \text{rg}(f).$$

2.4. Corollaire. Si $\dim(E) = \dim(F)$, alors f est injective si et seulement si elle est surjective, si et seulement si elle est bijective.

2.5. Corollaire. Si E est de type fini, alors $\text{rg}(f) \leq \dim(E)$, avec égalité si et seulement si f est injective. Si F est de type fini, alors $\text{rg}(f) \leq \dim(F)$, avec égalité si et seulement si f est surjective.

2.6. Proposition. Le rang d'une application linéaire ne change pas quand on la compose

- (1) à droite par une application linéaire surjective;
- (2) à gauche par une application linéaire injective;
- (3) à droite ou à gauche par une application linéaire bijective.

3. MATRICES

Nous avons vu dans le théorème 1.5 qu'une application linéaire $\phi : E \rightarrow F$ est caractérisée par l'image d'une base de E . Considérons donc le cas où $E = \mathbb{K}^n$ et $F = \mathbb{K}^m$. Ces deux espaces ont chacun une base canonique (voir résumé 1, 2.22). Notons $(e_i)_{i=1,\dots,n}$ celle de E et $(f_i)_{i=1,\dots,m}$ celle de F .

Il est donc équivalent de se donner une application $\phi : \mathbb{K}^n \rightarrow \mathbb{K}^m$ où de se donner des coefficients $(a_{i,j})$ dans \mathbb{K} où $i = 1, \dots, m$ et $j = 1, \dots, n$, avec la convention que $\phi(e_j) = \sum_i a_{i,j} f_i$. On écrit habituellement ces éléments dans un tableau à m lignes et n colonnes, où le coefficient $a_{i,j}$ figure en ligne i et colonne j .

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

Pour cette raison, on pose la définition suivante.

3.1. Définition (matrice). Une *matrice* de taille $m \times n$ à coefficients dans \mathbb{K} est la donnée d'une famille de coefficients $(a_{i,j})$ avec $i \in \{1, \dots, m\}$ et $j \in \{1, \dots, n\}$. L'ensemble des matrices de taille $m \times n$ à coefficients dans \mathbb{K} se note $M_{m,n}(\mathbb{K})$. Lorsque $m = n$, on note $M_n(\mathbb{K})$ au lieu de $M_{n,n}(\mathbb{K})$.

3.2. Exemple. La matrice $\text{Id}_n \in M_n(\mathbb{K})$ est celle dont le coefficient (i, j) est $\delta_{i,j}$ (symbole de Kronecker). Autrement dit, c'est 0 si $i \neq j$ et 1 si $i = j$. On l'appelle la matrice *identité* de taille n .

$$\text{Id}_n = \left. \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{pmatrix} \right\} n$$

3.3. Définition. Si $\phi \in \mathcal{L}(\mathbb{K}^n, \mathbb{K}^m)$, on lui associe la matrice $M_\phi \in M_{m,n}(\mathbb{K})$ définie par l'unique famille $(a_{i,j})$ telle que $\phi(e_j) = \sum_i a_{i,j} f_i$. Dans l'autre sens, si $A \in M_{m,n}(\mathbb{K})$ est une matrice $A = (a_{i,j})$, on lui associe l'unique application linéaire $l_A \in \mathcal{L}(\mathbb{K}^n, \mathbb{K}^m)$ telle que $l_A(e_j) = \sum_i a_{i,j} f_i$.

3.4. Remarque (très importante). La j -ème colonne de la matrice M_ϕ est donc constituée des coordonnées de $\phi(e_j)$ (sur la base canonique).

3.5. Proposition. *Les applications*

$$\begin{array}{ccc} \mathcal{L}(\mathbb{K}^n, \mathbb{K}^m) & \rightarrow & M_{m,n}(\mathbb{K}) \\ \phi & \mapsto & M_\phi \end{array} \quad \text{et} \quad \begin{array}{ccc} M_{m,n}(\mathbb{K}) & \rightarrow & \mathcal{L}(\mathbb{K}^n, \mathbb{K}^m) \\ A & \mapsto & l_A \end{array}$$

sont bijectives et inverses l'une de l'autre.

En 1.4, nous avons défini trois opérations sur les applications linéaires : la somme, le produit par un scalaire, et la composition. Nous allons voir qu'elles correspondent par la proposition 3.5 à des opérations sur les matrices.

Dans ce qui suit, la matrice A (resp. B, C , etc.) a pour coefficients $(a_{i,j})$ (resp. $(b_{i,j}), (c_{i,j}),$ etc.).

3.6. Définition. La *somme* de deux matrices $A, B \in M_{m,n}(\mathbb{K})$, est la matrice $C \in M_{m,n}(\mathbb{K})$ définie par $c_{i,j} = a_{i,j} + b_{i,j}$ pour tous i et j . On la note $A + B$.

3.7. Définition. La *multiplication par un scalaire* $\lambda \in \mathbb{K}$ d'une matrice $A \in M_{m,n}(\mathbb{K})$ est la matrice $C \in M_{m,n}(\mathbb{K})$ définie par $c_{i,j} = \lambda a_{i,j}$. On la note λA .

3.8. Définition. Le *produit* de la matrice $A \in M_{m,n}(\mathbb{K})$ par la matrice $B \in M_{n,p}(\mathbb{K})$ est la matrice $C \in M_{m,p}(\mathbb{K})$ définie par $c_{i,j} = \sum_{k=1}^n a_{i,k} b_{k,j}$. On la note AB .

3.9. Théorème. Avec les notations de la proposition 3.5, on a :

- (1) $M_{f+g} = M_f + M_g$ si $f, g \in \mathcal{L}(\mathbb{K}^n, \mathbb{K}^m)$;
- (2) $M_{\lambda f} = \lambda M_f$ si $\lambda \in \mathbb{K}$ et $f \in \mathcal{L}(\mathbb{K}^n, \mathbb{K}^m)$;
- (3) $M_f M_g = M_{f \circ g}$ si $f \in \mathcal{L}(\mathbb{K}^n, \mathbb{K}^m)$ et $g \in \mathcal{L}(\mathbb{K}^p, \mathbb{K}^n)$.

De même

- (1) $l_{A+B} = l_A + l_B$ si $A, B \in M_{m,n}(\mathbb{K})$;
- (2) $l_{\lambda A} = \lambda l_A$ si $\lambda \in \mathbb{K}$ et $A \in M_{m,n}(\mathbb{K})$;
- (3) $l_A \circ l_B = l_{AB}$ si $A \in M_{m,n}$ et $B \in M_{n,p}(\mathbb{K})$.

Autrement dit, quand on passe des applications linéaires à leur description matricielle, la somme reste la somme, la multiplication par un scalaire de même, et la composition devient le produit de matrices.

On peut identifier \mathbb{K}^n aux matrices $M_{n,1}(\mathbb{K})$, appelées *vecteurs colonnes*. La i -ème coordonnée d'un vecteur $v \in \mathbb{K}^n$ est en i -ème ligne.

3.10. Proposition. Avec cette convention, on a $M_\phi v = \phi(v)$ pour tout $v \in \mathbb{K}^n$, ou encore $Av = l_A(v)$.

Pour cette raison, nous identifierons désormais toujours \mathbb{K}^n aux vecteurs colonne, et nous éviterons de les noter comme des lignes, comme nous avons fait jusqu'à présent.

3.11. Proposition. On a

- La multiplication par les scalaires et l'addition munissent $M_{m,n}(\mathbb{K})$ d'une structure d'espace vectoriel (sur \mathbb{K}) ;
- $A(BC) = (AB)C$;
- $A(B+C) = AB + AC$ et $(A+B)C = AC + BC$.



Il est facile de voir que $A + B = B + A$, par contre en général, $BA \neq AB$. Il se peut même que l'un des côtés de cette égalité n'ait aucun sens alors que l'autre oui, pour des raisons de tailles de matrices.

3.12. Définition. Le *noyau* d'une matrice $A \in M_{m,n}(\mathbb{K})$ est le noyau de l_A , i.e. c'est l'ensemble des vecteurs $X \in \mathbb{K}^n$ tels que $AX = 0$.

L'*image* de A est l'image de l_A , i.e. c'est l'ensemble des vecteurs $Y \in \mathbb{K}^m$ tels qu'il existe $X \in \mathbb{K}^n$ avec $AX = Y$.

Le *rang* de A est le rang de l_A , i.e. c'est la dimension de l'image de A .

3.13. Proposition. Si v_1, \dots, v_n sont les vecteurs colonnes de A , alors $\text{im}(A) = \text{Vec}(v_1, \dots, v_n)$.

Introduisons encore une opération classique sur les matrices qui nous servira par la suite.

3.14. Définition. La *transposée* d'une matrice $A \in M_{m,n}(\mathbb{K})$ est notée A^t et est la matrice $B \in M_{n,m}(\mathbb{K})$ de coefficients $b_{i,j} = a_{j,i}$.

3.15. **Proposition.** On a

- (1) $(A + B)^t = A^t + B^t$;
- (2) $(\lambda A)^t = \lambda(A^t)$;
- (3) $(AB)^t = B^t A^t$;
- (4) $(A^t)^t = A$.

3.16. **Définition.** Une matrice A telle que $A = A^t$ est appelée matrice *symétrique*. Si $A = -A^t$, elle est appelée *antisymétrique*. (Dans les deux cas, une telle matrice est nécessairement carrée.)

3.17. **Théorème.** Le rang d'une matrice est égal à celui de sa transposée.

4. APPLICATIONS LINÉAIRES ET MATRICES INVERSIBLES

4.1. **Définition.** Une application linéaire $f : E \rightarrow F$ est *invertible* s'il existe une application linéaire $g : F \rightarrow E$ telle que $g \circ f = \text{Id}_E$ et $f \circ g = \text{Id}_F$.

4.2. **Proposition.** Une application linéaire est invertible $f : E \rightarrow F$ si et seulement si elle est bijective, et il ne peut alors y avoir qu'une g vérifiant les égalités de la définition. On la note f^{-1} . De plus si E est de type fini, on a $\dim(F) = \dim(E)$.



L'application f^{-1} n'existe donc pas toujours.

4.3. **Proposition.** Soit $f : E \rightarrow F$ une application linéaire entre deux espaces vectoriels de type fini et de même dimension. Alors il existe au plus une application $g : F \rightarrow E$ telle que $g \circ f = \text{id}_E$ et, si elle existe, elle vérifie alors automatiquement $f \circ g = \text{id}_F$.

4.4. **Définition.** Une matrice $A \in M_{m,n}(\mathbb{K})$ est *invertible* si et seulement s'il existe une matrice $B \in M_{n,m}(\mathbb{K})$ telle que $AB = \text{Id}_m$ et $BA = \text{Id}_n$.

4.5. **Proposition.** Si $A \in M_{m,n}(\mathbb{K})$ est invertible, alors $m = n$. Autrement dit, une matrice invertible est forcément carrée.

4.6. **Proposition.** Soit $A \in M_n(\mathbb{K})$ une matrice carrée. Il existe au plus une matrice B telle que $AB = \text{Id}_n$ et, si elle existe, elle vérifie alors automatiquement $BA = \text{Id}_n$.

4.7. **Définition.** Lorsque A est invertible, l'unique matrice B telle que $AB = BA = \text{Id}_n$ est appelée matrice *inverse* de A . On la note A^{-1} .



La notation A^{-1} n'a donc de sens que si A est carrée et de plus *invertible* ; Il y a de nombreuses matrices A carrées qui ne sont pas invertibles et pour lesquelles A^{-1} n'existe pas.

4.8. **Proposition.** $f \in \mathcal{L}(\mathbb{K}^n)$ est invertible si et seulement si M_f est invertible. De même, $A \in M_n(\mathbb{K})$ est invertible si et seulement si l_A est invertible.

4.9. **Lemme.** Si $A, B \in M_n(\mathbb{K})$, alors le produit AB est invertible si et seulement si A est invertible et B est invertible. On a alors $(AB)^{-1} = B^{-1}A^{-1}$.



Attention, si A et B ne sont pas carrées, mais que AB l'est, ni A ni B ne peut être invertible, mais il se peut que AB le soit.

Nous allons voir plusieurs manières de déterminer si une telle application linéaire ou une matrice est invertible et comment trouver explicitement l'inverse.

5. SYSTÈMES LINÉAIRES

Le système à m équations et n inconnues x_1, \dots, x_n

$$(5.1) \quad \begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = y_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = y_2 \\ \dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = y_m \end{cases}$$

peut être vu comme l'équation matricielle

$$(5.2) \quad AX = Y$$

où

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}, \quad X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad \text{et} \quad Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}$$

Résoudre le système consiste à exprimer, si possible, les inconnues x_1, \dots, x_n en fonction des seconds membres y_1, \dots, y_m .

5.1. Proposition. *On a*

- (1) *Le système (5.2) a au moins un vecteur X solution si et seulement si le vecteur Y est dans l'image de l'application l_A , ce qui est en particulier toujours le cas quand celle-ci est surjective (i.e. de rang m).*
- (2) *Le système (5.2) a au plus un vecteur X solution si et seulement si l'application l_A est injective.*
- (3) *Le système (5.2) a une unique solution X pour tout Y si et seulement si l'application l_A est inversible. Elle est alors donnée par $X = A^{-1}Y$.*
- (4) *Deux solutions diffèrent d'un élément de $\ker(l_A)$.*

Résoudre un tel système revient donc à

- (1) vérifier premièrement si le vecteur Y est dans l'image de l_A ;
- (2) si oui, lui trouver un antécédent X_0 ;
- (3) déterminer $\ker(l_A)$.

Si la réponse au point (1) est oui, alors le système a une seule solution X_0 si $\ker(l_A) = \{0\}$ ou une infinité de solutions de la forme $X_0 + X'$ où X' est un élément de $\ker(l_A)$ si $\ker(l_A) \neq \{0\}$.

Si l'on détermine une base $(X'_1, X'_2, \dots, X'_k)$ de $\ker(l_A)$, cela fournit une paramétrisation de l'ensemble des solutions de (5.2) :

$$\{X = X_0 + \sum_i \lambda_i X'_i, \quad \lambda_1, \dots, \lambda_k \in \mathbb{K}\}$$

où les λ_i sont appelés les "paramètres".

6. ALGORITHMES

Citons maintenant un certain nombre d'algorithmes utiles.

6.1. **Définition.** Soit $E_n(\lambda, k, l)$ la matrice de $M_n(\mathbb{K})$ définie par

$$e_{i,j} = \begin{cases} \lambda & \text{si } i = k \text{ et } j = l \\ 0 & \text{sinon.} \end{cases}$$

Soit $D_n(\lambda, k)$ la matrice définie par

$$d_{i,j} = \begin{cases} \lambda & \text{si } i = j = k \\ 1 & \text{si } i = j \text{ mais } i \neq k \\ 0 & \text{sinon.} \end{cases}$$

Enfin, on définit également $G_n(k, l) = \text{Id}_n - E_n(1, k, k) - E_n(1, l, l) + E_n(1, k, l) + E_n(1, l, k)$ et si $k \neq l$, $F_n(\lambda, k, l) = \text{Id}_n + E_n(\lambda, k, l)$ et .

Lorsque n est clair par le contexte, on ne le mentionne plus et on écrit $E(\lambda, k, l)$, $D(\lambda, k)$, $F(\lambda, k, l)$, et $G(k, l)$.

$$E(\lambda, k, l) = \begin{pmatrix} & \overset{l}{\vdots} & & \\ & \lambda & \text{---} & k \\ & & & \\ (0) & & & \end{pmatrix} \quad F(\lambda, k, l) = \begin{pmatrix} & \overset{l}{\vdots} & & \\ 1 & \lambda & \text{---} & k \\ & \ddots & & \\ & & (0) & \\ (0) & & & 1 \end{pmatrix}$$

$$D(\lambda, k) = \begin{pmatrix} & \overset{k}{\vdots} & & \\ 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & \lambda & \text{---} & k \\ & & & & \\ (0) & & & & 1 \end{pmatrix} \quad G(k, l) = \begin{pmatrix} & \overset{k}{\vdots} & \overset{l}{\vdots} & & \\ 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & 0 & \text{---} & k \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \\ & & 1 & \text{---} & l \\ & & & & 1 \end{pmatrix}$$

6.2. **Lemme.** On a

$$E(\lambda, k, l)E(\mu, o, p) = \delta_{l,o}E(\lambda\mu, k, p)$$

6.3. **Proposition.** La matrice $D(\lambda, k)$ est inversible si et seulement si $\lambda \neq 0$ et alors $D(\lambda, k)^{-1} = D(\lambda^{-1}, k)$.

La matrice $F(\lambda, k, l)$ est inversible pour tout λ et $F(\lambda, k, l)^{-1} = F(-\lambda, k, l)$.

La matrice $G(k, l)$ est inversible et est égale à son inverse.

La multiplication à gauche par une matrice élémentaire correspond à une opérations sur les lignes de la matrice qu'on multiplie :

6.4. **Théorème** (opérations sur les lignes). Si $A \in M_{m,n}(\mathbb{K})$, alors

(A) $E(\lambda, k, l)A$ est une matrice de $M_{m,n}(\mathbb{K})$ constituée de λ fois la l -ième ligne de A mise en ligne k , et qui est nulle partout ailleurs.

(B) $F(\lambda, k, l)A$ est obtenue à partir de A en lui rajoutant λ fois sa l -ième ligne à sa k -ième ligne.

(C) $D(\lambda, k)A$ est obtenue à partir de A en multipliant sa k -ième ligne par λ .

(D) $G(k, l)A$ est obtenue à partir de A en échangeant ses k -ième et l -ième lignes.

Le théorème symétrique existe bien entendu pour les colonnes au lieu des lignes, et correspond à des multiplications à droite. Nous ne les utiliserons pas.

Passons maintenant aux matrices échelonnées, et à la manière de transformer une matrice en matrice échelonnée.

6.5. Définition. Une matrice $B \in M_{m,n}(\mathbb{K})$ est dite *échelonnée* si elle vérifie les conditions suivantes :

Pour tout i , représentant un numéro d'une ligne de B , soit $p(i)$ le plus petit indice j tel que $b_{i,j} \neq 0$, s'il existe. S'il n'existe pas ($b_{i,j} = 0$ pour tout j), on pose $p(i) = +\infty$.

(1) Si $p(i) \neq +\infty$, alors $b_{i,p(i)} = 1$.

(2) Si $i_1 < i_2$, alors $p(i_1) < p(i_2)$, sauf si $p(i_1) = +\infty$, auquel cas $p(i_2) = +\infty$ aussi.

(3) Si $i_1 < i_2$ et $p(i_2) \neq +\infty$, alors $b_{i_1,p(i_2)} = 0$.

6.6. Remarque. Les positions $(i, p(i))$ jouent un rôle clé et sont appelées positions *pivots*. Il y a au plus un pivot par ligne, et il n'y a que des zéros à gauche des pivots. S'il n'y a pas de pivot, toute la ligne est nulle.

On peut alors intuitivement énoncer la description d'une matrice échelonnée sous la forme suivante :

(1) Le coefficient en position pivot est 1.

(2) Les pivots sont placés dans le même ordre sur les lignes et les colonnes.

(3) Il n'y a que des zéros au-dessus des pivots.

Si B est échelonnée, soit p_B le nombre de pivots de B , i.e. le nombre de i tels que $p(i) \neq \infty$.

De manière imagée, une telle matrice a donc la forme

$$\begin{array}{cccc} & p(1) & p(2) & p(p_B) \\ \left(\begin{array}{cccc} 1 & \star \cdots \star & 0 & \star \cdots \star & 0 & \star \cdots \\ & & 1 & \star \cdots \star & 0 & \star \cdots \\ & & & \ddots & \vdots & \\ & & & & 0 & \star \cdots \\ & (0) & & & 1 & \star \cdots \end{array} \right) \begin{array}{l} 1 \\ 2 \\ \\ \\ p_B \end{array} \end{array}$$

où les \star représentent des coefficients quelconques et les 1 sont en positions pivots.

6.7. Remarque. Si un système d'équations linéaires $BX = Y$ est donné par une matrice échelonnée B , alors sa résolution est immédiate. C'est la raison pour laquelle ces matrices sont aussi importantes d'un point de vue pratique.

6.8. Lemme. p_B est égal au nombre de lignes non nulles de B ; pour tout $i > p_B$, $b_{i,j} = 0$ pour tout j .

6.9. Proposition. *L'image de B est constituée des vecteurs dont les coordonnées sont nulles au-delà de l'indice p_B (non compris). On a donc $\text{rg}(B) = p_B$.*

6.10. Proposition. *Le noyau de B est de dimension égale à $n - p_B$. Une base de ce noyau est constitué des vecteurs*

$$e_j - \sum_{i=1}^{l(j)} b_{i,j} e_{p(i)}$$

où la colonne j est sans pivot et $l(j)$ désigne le plus petit indice tel que $b_{i,j} = 0$ si $i > l(j)$.

Étant donnée une matrice $A \in M_{m,n}(\mathbb{K})$, il est très utile de trouver une matrice $M \in M_m(\mathbb{K})$ inversible telle que la matrice $B = MA$ est échelonnée, en particulier pour résoudre le système associé, mais également pour d'autres raisons que nous allons voir plus bas.

6.11. Théorème. *Étant donné une matrice A , il existe une et une seule matrice $B = MA$ échelonnée telle que M soit inversible.*

6.12. Remarque. La matrice M n'est pas unique, comme on s'en convainc aisément lorsque A est nulle : toute matrice M inversible convient alors (et B est bien évidemment la matrice nulle).

6.13. Théorème. *La matrice A est inversible si et seulement si $m = n$ et $B = \text{Id}_n$. Son inverse est alors la matrice M .*

On constate que si une matrice est sous forme échelonnée, ses sous-matrices obtenues en ne conservant que les lignes et colonnes en-dessous et à droite d'une position pivot sont également sous forme échelonnée. On peut donc appliquer une forme de récurrence pour transformer une matrice en une matrice échelonnée.

6.14. Algorithme (Échelonnage par opérations sur les lignes). Les opérations (B), (C) et (D) mentionnées ci-dessous sont celles du théorème 6.4.

étape pivot. Si une colonne d'une matrice n'est pas nulle, par les opérations sur les lignes, transformer cette colonne en un 1 sur la première ligne suivi de 0 sur toutes les autres : on trouve le premier coefficient non nul de la colonne, on le permute en première ligne par l'opération (D) puis on le change en 1 en le multipliant par son inverse par l'opération (C). Puis on annule le coefficient en première colonne de chaque autre ligne en additionnant à cette ligne un multiple de la première ligne.

L'algorithme se déroule alors de la manière suivante. Au départ, la matrice en cours est la matrice $A \in M_{m,n}(\mathbb{K})$ que l'on veut échelonner.

étape 1. On applique l'étape pivot à la première colonne non nulle de la matrice en cours. Soit la matrice est nulle et l'on s'arrête, soit cela fait apparaître un pivot de plus. On passe ensuite à la sous-matrice constituée des colonnes et des lignes strictement après le pivot. Si elle est vide on s'arrête, sinon, on reprends cette étape 1 dessus.

Après avoir répété un nombre fini de fois l'étape 1 (au plus $\min(m, n)$), on obtient une matrice B' qui satisfait à toutes les conditions de la définition d'une matrice échelonnée, sauf peut-être la condition (3).

étape 2. À l'aide de l'opération (B), on supprime successivement tous les coefficients non nuls au-dessus des pivots, en commençant par le dernier pivot et en remontant jusqu'au premier.

L'algorithme précédent ou une de ses proches variantes est également connu sous le nom de "pivot de Gauss".

6.15. Algorithme (obtenir la matrice de passage). Partant de A , pour obtenir une matrice M telle que $B = MA$ avec B échelonnée, lors de l'algorithme précédent, il faut conserver le produit des matrices élémentaires utilisées. Une manière de le faire est de poser, à côté de la matrice A sur laquelle on travaille, une autre matrice qui au départ est Id_m . Puis, à chaque fois qu'on fait une opération sur les lignes de A , on fait la même sur cette matrice. À la fin de l'algorithme, elle s'est changée en la matrice M recherchée.

6.16. Algorithme (Extraire une base d'une liste de vecteurs). On regroupe ces vecteurs comme les colonnes d'une matrice A , qu'on échelonne en une matrice B . Les colonnes de A qui se trouvent aux positions des colonnes de B à pivots forment une base de l'espace engendré par les vecteurs de départ, qui n'est autre que $\text{im}(A)$.

6.17. Algorithme (Vérifier si un vecteur est engendré par d'autres vecteurs). On veut vérifier si un vecteur v est engendré par les vecteurs c_1, \dots, c_n . On forme une matrice A dont c_1, \dots, c_n sont les colonnes. On échelonne A , en $B = MA$. Alors v est dans l'image de A si et seulement si Mv est dans l'image de B , ce qui est évident à vérifier par la proposition 6.9.

6.18. Algorithme (Trouver une base du noyau d'une matrice). Si A est cette matrice, elle a le même noyau que la matrice échelonnée $B = MA$ obtenue par l'algorithme 6.14. On en obtient donc une base par la proposition 6.10.

6.19. Remarque. Passer de la description d'un sous-espace vectoriel comme solution d'équations linéaires vers une description paramétrique, c'est-à-dire décrire ses vecteurs comme les combinaisons linéaires d'une base revient exactement à appliquer l'algorithme précédent.

6.20. Algorithme (Inverser une matrice). Si A n'est pas carrée, inutile d'aller plus loin, elle n'est pas inversible. Si elle est carrée, on applique l'algorithme 6.14 pour échelonner A . Par le théorème 6.13 La matrice A est inversible si et seulement si B est l'identité, et dans ce cas $A^{-1} = M$, obtenue par l'algorithme 6.15.

6.21. Algorithme (Forme paramétrique vers forme cartésienne). Soit V un sous-espace vectoriel de \mathbb{K}^n décrit comme $V = \text{Vec}(v_1, \dots, v_l)$. On veut trouver des équations linéairement indépendantes (i.e. non redondantes) de manière à ce que V soit exactement la solution de ces équations. Autrement dit, on veut trouver une matrice $A \in M_{k,n}$ pour un certain k , telle que $Av = 0$ si et seulement si $v \in V$ et telle que ses lignes soient linéairement indépendantes (k soit minimum). Soit $C \in M_{n,l}$ la matrice formée des colonnes v_1, \dots, v_n . Alors AC doit être nulle, autrement dit, on doit avoir $C^t A^t = 0$. Le problème revient ainsi à trouver une base de $\ker(C^t)$, ce que l'on sait faire grâce à l'algorithme 6.18, puis à s'en servir comme colonnes de A^t , autrement dit comme lignes de A .