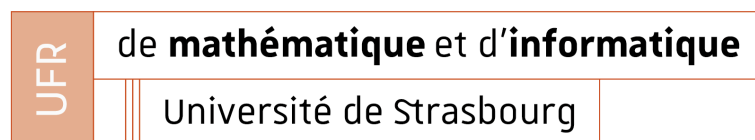


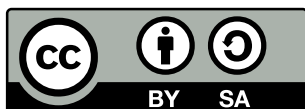
Analyse Numérique Appliquée

Notes de cours

Basile Sauvage

2022





Cette oeuvre, création, site ou texte est sous licence Creative Commons Attribution - Partage dans les Mêmes Conditions 4.0 International. Pour accéder à une copie de cette licence, merci de vous rendre à l'adresse suivante [http ://creativecommons.org/licenses/by-sa/4.0/](http://creativecommons.org/licenses/by-sa/4.0/) ou envoyez un courrier à Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

Table des matières

1	Introduction	5
	Rappels et compléments	9
2	Rappels d'algèbre	11
2.1	Espaces vectoriels	12
2.2	Famille libre et dimension	12
2.3	Famille génératrice et base	13
2.4	Ecriture dans une base	13
2.5	Applications	14
2.6	Espaces particuliers : les espaces euclidiens	14
3	Rappels sur la complexité	17
3.1	Complexité asymptotique	18
3.2	Ordres de grandeur	18
4	Rappels de statistiques	19
	Cours	21
5	Les matrices	23
5.1	Matrices générales	24
5.2	Matrices carrées	24
5.3	Algorithmes standards sur les vecteurs et les matrices	25
5.4	Représentations informatiques des matrices	26
6	Résolution de systèmes linéaires	27
6.1	Position du problème	28
6.2	Exemple	29
6.3	Étape 1 : méthode du pivot de Gauss	29
6.4	Étape 2 : méthode de remontée	31
6.5	Algorithme complet de résolution d'un système linéaire	31
6.6	Stabilité numérique	32
7	Approximation au sens des moindres carrés	33
7.1	Position du problème	35
7.2	Résolution par projection orthogonale	35
7.3	Résolution par minimisation de la distance quadratique	36
7.4	Régression linéaire	36
7.5	Approximation polynômiale	37

7.6	Moindres carrés pondérés, une question de confiance	37
7.7	Changements de variable	38
8	Applications linéaires	39
8.1	Notations	40
8.2	Espaces vectoriels et espaces affines	40
8.3	Applications linéaires	40
8.4	Noyau et image d'une application linéaire	41
8.5	Exemple des projections	42
8.6	Changement de base : matrices de passage	43
8.7	Changement de base dans les application linéaires	44
9	Éléments propres	45
9.1	Définitions	46
9.2	Calcul de l'espace propre associé à une valeur propre réelle connue	46
9.3	Diagonalisation	48
9.4	Calcul approché des valeurs propres et vecteurs propres	50
	Annexes	53
A	Comment éviter les erreurs ?	55
B	Questions de cours	57
B.1	Rappels d'algèbre	57
B.2	Rappels sur la complexité	57
B.3	Les matrices	57
B.4	Résolution de systèmes linéaires	57
B.5	Approximation au sens des moindres carrés	57
B.6	Applications linéaires	58
B.7	Éléments propres	58
C	Annales	59
C.1	Les matrices	59
C.2	Résolution de systèmes linéaires	61
C.3	Approximation au sens des moindres carrés	62
C.4	Applications linéaires	66
C.5	Éléments propres	70
D	Exercices de travaux dirigés	75
E	Exercices de travaux pratiques	79
F	Projet : la valeur des signes	83

Chapitre 1

Introduction

Notes personnelles.

Pourquoi ce cours ?

Certains domaines de l'informatique, dont ceux enseignés dans nos Master, nécessitent de savoir *utiliser* et *programmer* un certain nombre d'outils mathématiques standards. Pour cela il faut les *comprendre* et les *manipuler*, en un mot les démystifier.

Dans ce cours nous reviendrons entre autres sur des notions mathématiques connues. Nous ferons peu de démonstrations, pour nous concentrer sur les algorithmes, les calculs et les applications. L'analyse numérique est parfois perçue comme mathématique par les informaticiens, et comme informatique par les mathématiciens : c'est qu'il s'agit d'écrire et de programmer des algorithmes pour faire des calculs mathématiques, en tenant compte des difficultés liées aux ordinateurs (imprécisions numériques, temps de calcul, et occupation mémoire).

Quel contenu ?

Les sujets abordés sont les suivants :

- Espaces vectoriels et matrices.
- Résolution de systèmes linéaires d'équations (calcul direct, factorisation).
- Algorithmes d'approximation : moindres carrés, régression linéaire, régression polynomiale, et transformation de variable.
- Applications linéaires et changements de base.
- Calcul de valeurs propres, vecteurs propres, diagonalisation.
- Applications sur un logiciel de calcul numérique.

Quelles compétences visées ?

À l'issue de cet enseignement, vous saurez :

- Utiliser des algorithmes pour résoudre des systèmes linéaires et diagonaliser des matrices.
- Effectuer des changements de bases.
- Mener des régressions avec transformation des variables.
- Utiliser un logiciel de calcul numérique.

Quelles modalités pédagogiques ?

Ce module est composé de

- 12h de CM, où nous chercherons à comprendre les notions essentielles, à l'aide d'exemples et de schémas qui compléteront ces notes de cours.
- 6h de TD, où vous réaliserez des calculs et écrirez des algorithmes ;
- 6h de TP, où vous utiliserez les algorithmes pour faire des mesures sur des données réelles.

Ce module sera évalué par

- 2 contrôles sur table, où nous évaluerons les aspects mathématiques, calculatoires et algorithmiques ;
- 1 contrôle sur machine, où nous évaluerons votre capacité à appliquer des algorithmes à l'aide d'un logiciel de calcul numérique, pour analyser et observer des données.

Quels pré-requis ?

Nous nous appuierons d'une part sur des connaissances mathématiques (espaces vectoriels, bases, matrices, systèmes linéaires, applications linéaires) abordées en particulier dans les cours d'algèbre de première année. D'autre part, vous utiliserez vos compétences informatiques (algorithmes itératifs, complexité, tableaux) acquises en première année (algorithmique et programmation) et en deuxième année (structures de données et algorithmes).

Rappels et compléments

Chapitre 2

Rappels d'algèbre

Notes personnelles.

2.1 Espaces vectoriels

Définition Soit K un corps commutatif ; ses éléments sont appelés scalaires. Un espace vectoriel sur le corps K (auss appelé K -espace vectoriel) est un ensemble E non vide d'éléments (appelés vecteurs), muni d'une loi interne (appelée addition) et une loi externe (multiplication par un scalaire), tels que :

- L'addition est une loi interne ($u + v \in E, \forall u, v \in E$) vérifiant les propriétés suivantes :
 1. elle est commutative : $u + v = v + u, \forall u, v \in E$;
 2. elle est associative : $(u + v) + w = u + (v + w), \forall u, v, w \in E$;
 3. elle admet un élément neutre (noté 0) : $u + 0 = u, \forall u \in E$;
 4. chaque vecteur $u \in E$ admet un opposé (noté $-u$) : $u + (-u) = 0$.
- La multiplication par un scalaire est une loi externe ($cu \in E, \forall c \in K, \forall u \in E$) vérifiant les propriétés suivantes :
 1. elle est distributive sur l'addition des vecteurs : $c(u + v) = cu + cv, \forall c \in K, \forall u, v \in E$;
 2. elle est distributive sur l'addition des scalaires : $(c + d)u = cu + du, \forall c, d \in K, \forall u \in E$;
 3. $c(du) = (cd)u, \forall c, d \in K, \forall u \in E$;
 4. $1.u = u, \forall u \in E$;

Il en découle certaines propriétés, telles que, pour $c \in K$ et $u \in E$: $0u = 0$; $c0 = 0$; $-u = (-1)u$.

Exemple 2.1

- $E = \mathbb{R}^d$ est un e.v. sur $K = \mathbb{R}$ (en particulier pour $d = 1$).
- $E = \mathbb{C}^d$ est un e.v. sur \mathbb{R} ou sur \mathbb{C} (en particulier pour $d = 1$).
- L'ensemble $E = \mathbb{R}_d[X]$ des polynômes réels de degré inférieur ou égal à d est un e.v. sur \mathbb{R} .

2.2 Famille libre et dimension

Définition. Soient deux vecteurs u et $v \in E$. Ils sont colinéaires (i.e. ils ont la *même direction*) si et seulement si $\exists c \in K, v = cu$.

Si $c > 0$ ils ont le *même sens*.

Définition. Soient $u_1, u_2, \dots, u_n \in E$ et $c_1, c_2, \dots, c_n \in K$.

$$\sum_{i=1}^n c_i u_i \text{ est appelée combinaison linéaire des } u_i$$

Définition. Des vecteurs $u_1, u_2, \dots, u_n \in E$ sont linéairement indépendants (i.e. ils forment une famille libre) si et seulement si

$$\sum_{i=1}^n c_i u_i = 0 \quad \Rightarrow \quad c_i = 0, \forall i$$

Définition. Un espace vectoriel est de dimension d si on peut trouver d vecteurs linéairement indépendants, mais que $d + 1$ vecteurs sont toujours linéairement dépendants.

Remarque. Un espace vectoriel peut être de dimension infinie.

2.3 Famille génératrice et base

Définition. Soient $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n$ des vecteurs de \mathbf{E} . L'ensemble de toutes les combinaisons linéaires

$$\mathcal{L}\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n\} = \left\{ \sum_{i=1}^n c_i \mathbf{u}_i, \quad \forall c_1, c_2, \dots, c_n \in K \right\}$$

est un sous-espace vectoriel de \mathbf{E} , appelé sous-espace *engendré* par cette famille.

Remarque. Grâce à sa stabilité par combinaison linéaire, ce sous-espace "hérite" de la structure d'espace vectoriel de \mathbf{E} . Il est donc facile de construire des espaces vectoriels de cette manière.

Définition. Une base de \mathbf{E} est une famille libre qui engendre tout \mathbf{E} .

Si \mathbf{E} est de dimension finie d , une base $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_d\}$ contient d vecteurs et $\mathcal{L}\{\mathbf{e}_1, \dots, \mathbf{e}_d\} = \mathbf{E}$

Exemple 2.2

- \mathbb{R}^d est de dimension d , et admet comme base

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}; \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}; \quad \dots; \quad \mathbf{e}_d = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}.$$

- $\mathbb{R}_d[X]$ est de dimension $d + 1$, et admet comme base

$$\{1, X, X^2, \dots, X^d\}.$$

Remarque. Ces bases usuelles sont dites canoniques.

2.4 Ecriture dans une base

Soit $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_d\}$ une base de \mathbf{E} . Tout vecteur $\mathbf{u} \in \mathbf{E}$ se décompose de manière unique dans cette base, i.e. il existe des uniques scalaires u_1, u_2, \dots, u_d (appelés coordonnées de \mathbf{u}) tels que

$$\mathbf{u} = \sum_{i=1}^d u_i \mathbf{e}_i.$$

Exemple 2.3

- Dans \mathbb{R}^3 muni de sa base canonique,

$$\mathbf{u} = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + 3 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + 1 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Donc \mathbf{u} a pour coordonnées $(2, 3, 1)$.

- Dans \mathbb{R}^3 muni de la base

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 1 \\ -5 \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix} + 0 \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} + (-1) \begin{bmatrix} 0 \\ 1 \\ -5 \end{bmatrix}.$$

Donc le même vecteur \mathbf{u} a pour coordonnées $(2, 0, -1)$.

- Dans $\mathbb{R}_3[X]$ muni de sa base canonique $\{1, X, X^2, X^3\}$,

$$P(X) = X^3 + 2X^2 - 3X - 3$$

a pour coordonnées $(-3, -3, 2, 1)$.

- Dans $\mathbb{R}_3[X]$ muni de la base $\{1, (X+1), X^2, (X^3+X^2)\}$, le même polynôme

$$P(X) = X^3 + 2X^2 - 3X - 3 = (X^3 + X^2) + X^2 - 3(X+1) + 0$$

a pour coordonnées $(0, -3, 1, 1)$.

Attention. Un espace vectoriel n'a pas une unique base. Un seul et même vecteur peut être représenté par des coordonnées différentes dans deux bases différentes.

Important. L'application qui à $\mathbf{u} \in \mathbf{E}$ associe ses coordonnées $(u_1, u_2, \dots, u_d) \in K^d$ est linéaire et bijective. Pour cette raison, nous allons nous intéresser aux espaces $\mathbf{E} = K^d$, en particulier \mathbb{R}^d : on peut toujours s'y ramener, à condition de fixer une base de \mathbf{E} , et d'exprimer notre problème dans cette base.

2.5 Applications

Soit f une application d'un espace de départ \mathbf{E} dans un espace d'arrivée \mathbf{F} .

Définition. f est dite injective si et seulement si tout élément de \mathbf{F} a *au plus* un antécédant

$$\Leftrightarrow \forall \mathbf{u}_1, \mathbf{u}_2 \in \mathbf{E}, f(\mathbf{u}_1) = f(\mathbf{u}_2) \Rightarrow \mathbf{u}_1 = \mathbf{u}_2$$

$$\Leftrightarrow \forall \mathbf{u}_1, \mathbf{u}_2 \in \mathbf{E}, \mathbf{u}_1 \neq \mathbf{u}_2 \Rightarrow f(\mathbf{u}_1) \neq f(\mathbf{u}_2).$$

Définition. f est dite surjective si et seulement si tout élément de \mathbf{F} a *au moins* un antécédant

$$\Leftrightarrow \forall \mathbf{v} \in \mathbf{F}, \exists \mathbf{u} \in \mathbf{E} / f(\mathbf{u}) = \mathbf{v}.$$

Définition. f est dite bijective si et seulement si tout élément de \mathbf{F} a *exactement* un antécédant

$$\Leftrightarrow f \text{ est injective et surjective}$$

$$\Leftrightarrow \forall \mathbf{v} \in \mathbf{F}, \exists ! \mathbf{u} \in \mathbf{E} / f(\mathbf{u}) = \mathbf{v}.$$

2.6 Espaces particuliers : les espaces euclidiens

Dans toute cette section, on se place dans l'espace vectoriel \mathbb{R}^n . Soient deux vecteurs

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \text{ et } \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}.$$

On définit un produit scalaire

$$\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^n u_i v_i = \mathbf{u}^T \mathbf{v} = [u_1, u_2, \dots, u_n] \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

On dit que \mathbb{R}^n est *muni* de ce produit scalaire. C'est un espace *euclidien*.

Exemple 2.4

Calculer

$$\left\langle \begin{bmatrix} 2 \\ -5 \\ -1 \end{bmatrix}, \begin{bmatrix} 3 \\ 2 \\ -3 \end{bmatrix} \right\rangle = 2 * 3 + (-5) * 2 + (-1) * (-3) = -1$$

Le produit scalaire vérifie les propriétés suivantes :

- $\langle \mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{v}, \mathbf{u} \rangle$ (commutativité).
- $\langle \mathbf{u} + \mathbf{v}, \mathbf{w} \rangle = \langle \mathbf{u}, \mathbf{w} \rangle + \langle \mathbf{v}, \mathbf{w} \rangle$ (distributivité).
- $\langle c\mathbf{u}, \mathbf{v} \rangle = c \langle \mathbf{u}, \mathbf{v} \rangle$, où $c \in \mathbb{R}$.
- $\langle \mathbf{u}, \mathbf{u} \rangle \geq 0$ et $\langle \mathbf{u}, \mathbf{u} \rangle = 0$ si et seulement si $\mathbf{u} = \mathbf{0}$.

Définition. La norme euclidienne (ou longueur) d'un vecteur \mathbf{u} est $\|\mathbf{u}\| = \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle}$.

On dit que \mathbb{R}^n est *muni* de la norme euclidienne. C'est un *espace vectoriel normé*.

Définition. La distance entre deux vecteurs \mathbf{u} et \mathbf{v} est $\|\mathbf{u} - \mathbf{v}\|$.

Définition. Deux vecteurs \mathbf{u} et \mathbf{v} sont dits orthogonaux si $\langle \mathbf{u}, \mathbf{v} \rangle = 0$.

Définition. Un vecteur \mathbf{u} est dit normé (ou unitaire) si $\|\mathbf{u}\| = 1$.

Définition. Une base $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$ est dite orthogonale si

$$\langle \mathbf{e}_i, \mathbf{e}_j \rangle = 0 \quad \text{pour tout } i \neq j.$$

Définition. Une base $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$ est dite orthonormale si pour tout $1 \leq i, j \leq n$

$$\langle \mathbf{e}_i, \mathbf{e}_j \rangle = \delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}$$

Une base orthonormale est donc une base orthogonale de vecteurs normés.

Chapitre 3

Rappels sur la complexité

Notes personnelles.

Définition. Un algorithme est une suite finie d'opérations élémentaires constituant un schéma de calcul ou de résolution d'un problème.

Définition. La complexité en temps d'un algorithme est la mesure du nombre d'opérations élémentaires qu'il effectue sur un jeu de données. Elle est exprimée comme une fonction (notée T) de la taille du jeu de données (notée n).

L'unité de mesure sera ici l'opération flottante (flop). On sépare parfois les additions/soustractions des multiplications/divisions, ces dernières coûtant plus cher.

3.1 Complexité asymptotique

Quand les flop ne sont pas dénombrées exactement, on calcule une complexité asymptotique, qui caractérise le comportement de la complexité *quand la taille des données devient grande*. Rappelons les notations de Landau (vues en SDA1 et SDA2) :

- $T(n) = O(g(n))$ signifie $\exists c, n_0$ positives telles que

$$\forall n \geq n_0, 0 \leq T(n) \leq cg(n)$$

On dit que g est une borne asymptotique supérieure.

- $T(n) = \Omega(g(n))$ signifie $\exists c, n_0$ positives telles que

$$\forall n \geq n_0, 0 \leq cg(n) \leq T(n)$$

On dit que g est une borne asymptotique inférieure.

- $T(n) = \Theta(g(n))$ signifie $\exists c_1, c_2, n_0$ positives telles que

$$\forall n \geq n_0, 0 \leq c_1g(n) \leq T(n) \leq c_2g(n)$$

On dit que g borne asymptotiquement T .

3.2 Ordres de grandeur

$T(n)$	$\log_2(n)$	n	$n \log_2(n)$	n^2	n^3	2^n
$T(10^1)$	3	10	33	100	1 000	1 000
$T(10^2)$	7	100	660	10 000	10^6	[d] $1,3 \cdot 10^{30}$
$T(10^3)$	10	1 000	10 000	10^6	10^9	∞
$T(10^4)$	13	10 000	130 000	10^8	[b] 10^{12}	∞
$T(10^5)$	17	100 000	$1,7 \cdot 10^6$	10^{10}	10^{15}	∞
$T(10^6)$	20	1 000 000	[a] $20 \cdot 10^6$	[b] 10^{12}	[c] 10^{18}	∞

Sur une machine qui fait 1 milliard d'opération par seconde (1 giga-flops), les temps d'exécution :

- [a] 20 ms.
- [b] 17 minutes.
- [c] 32 années.
- [d] 40 milliards d'années.

Ordres de grandeur pour n :

- Téléphonie mobile (France) : plusieurs 10^4 antennes émettrices, plusieurs 10^7 abonnés.
- Web mondial : 10^{14} pages web référencées.

Chapitre 4

Rappels de statistiques

Notes personnelles.

Une variable. Soit $\mathbf{x} = \{x_1, \dots, x_n\} \in \mathbb{R}$ un ensemble de n valeurs (appelé *échantillon*).

La moyenne de \mathbf{x} vaut $\bar{x} = \frac{1}{N} \sum_{i=1}^n x_i$.

La variance de \mathbf{x} vaut $Var(\mathbf{x}) = \frac{1}{N-1} \sum_{i=1}^n (x_i - \bar{x})^2$.

La variance mesure la dispersion de l'échantillon autour de la moyenne : $Var(\mathbf{x}) = 0$ si tous les x_i sont égaux, et elle augmente quand les x_i s'éloignent les uns des autres. On utilise aussi l'écart type de \mathbf{x} , qui vaut $\sqrt{Var(\mathbf{x})}$.

Deux variables. Soient $\mathbf{x} = \{x_1, \dots, x_n\}$ et $\mathbf{y} = \{y_1, \dots, y_n\}$ deux échantillons réels.

La covariance de \mathbf{x} et \mathbf{y} vaut $Cov(\mathbf{x}, \mathbf{y}) = \frac{1}{N-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})$.

La covariance mesure la façon dont \mathbf{x} et \mathbf{y} varient l'un par rapport à l'autre : la covariance est positive s'ils ont tendance à croître et décroître ensemble ; elle est négative si l'un a tendance à croître quand l'autre décroît ; elle est nulle s'il n'y a pas de lien net entre \mathbf{x} et \mathbf{y} .

Le coefficient de corrélation entre \mathbf{x} et \mathbf{y} vaut $Cor(\mathbf{x}, \mathbf{y}) = \frac{Cov(\mathbf{x}, \mathbf{y})}{\sqrt{Var(\mathbf{x})}\sqrt{Var(\mathbf{y})}}$.

Le coefficient de corrélation donne une information similaire à la covariance, sauf que sa valeur est comprise dans $[-1; 1]$. La valeur 1 indique une corrélation linéaire parfaite et positive ; -1 indique une corrélation linéaire parfaite et négative ; 0 indique une absence de corrélation.

Estimateurs et biais. Les formules ci-dessus sont appelés des *estimateurs*, car elles permettent d'estimer des grandeurs (moyenne, variance, et covariance) à partir d'un échantillon. Il peut exister plusieurs estimateurs pour une même grandeur. Dans les formules de variance et covariance, attention au numérateur, qui est bien $(N-1)$ et non N , ce qui rend ces estimateurs non *biaisés*.

Cours

Chapitre 5

Les matrices

Notes personnelles.

5.1 Matrices générales

Une matrice réelle A à n lignes et p colonnes s'écrit

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & \dots & a_{1p} \\ a_{21} & a_{22} & \dots & \dots & a_{2p} \\ \vdots & & & & \vdots \\ a_{n1} & \dots & \dots & \dots & a_{np} \end{bmatrix}, \quad \text{avec } a_{ij} \in \mathbb{R}.$$

Transposition. La matrice notée A^T , transposée de A , a p lignes égales aux colonnes de A et n colonnes égales aux lignes de A .

$$A^T = \begin{bmatrix} a_{11} & a_{21} & \dots & a_{n1} \\ a_{12} & a_{22} & \dots & a_{n2} \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ a_{1p} & \dots & \dots & a_{np} \end{bmatrix}$$

En notant $B = A^T$, on a donc $b_{ji} = a_{ij}$.

La transposition vérifie $(A + B)^T = A^T + B^T$ et $(AB)^T = B^T A^T$.

On remarque la cohérence des dimensions :

- A : n lignes et p colonnes ;
- B : p lignes et q colonnes ;
- AB : n lignes et q colonnes ;
- A^T : p lignes et n colonnes ;
- B^T : q lignes et p colonnes ;
- $B^T A^T$: q lignes et n colonnes.

Exemple 5.1

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}^T = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$$

5.2 Matrices carrées

Une matrice est dite carrée si et seulement si $n = p$. Une matrice carrée peut avoir des propriétés particulières :

- Elle est **diagonale** si seuls ses éléments diagonaux sont non nuls. Exemple :

$$\begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Elle est **symétrique** si $A = A^T$, c'est à dire si $a_{i,j} = a_{j,i}$ pour tout $1 \leq i, j \leq n$. Exemple :

$$\begin{bmatrix} 2 & -1 & 5 & 0 \\ -1 & 3 & 0 & -2 \\ 5 & 0 & 2 & 0 \\ 0 & -2 & 0 & 1 \end{bmatrix}$$

- Elle est **triangulaire** supérieure (resp. inférieure) si tous les éléments en dessous (resp. dessus) de la diagonale sont nuls. Exemple :

$$\begin{bmatrix} 2 & -2 & 5 & 3 \\ 0 & 3 & 2 & -1 \\ 0 & 0 & 2 & -4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Elle est **inversible** s'il existe une matrice notée A^{-1} , appelée "inverse de A ", qui vérifie $AA^{-1} = A^{-1}A = I$.
- Elle est **définie positive** (positive définitive) si $x^T Ax > 0$ pour tout $x \in \mathbb{R}^n$, $x \neq 0$.
- Elle est **semi-définie positive** si $x^T Ax \geq 0$ pour tout $x \in \mathbb{R}^n$.

Exemple 5.2

$$A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 \end{bmatrix} \text{ est une matrice carrée, diagonale, semi-définie positive mais pas inversible.}$$

$$A_2 = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ est une matrice carrée, triangulaire supérieure et inversible.}$$

$$A_3 = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 1 & 0 & 3 \end{bmatrix} \text{ est une matrice carrée, triangulaire inférieure et inversible.}$$

$$A_4 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \text{ est une matrice non carrée.}$$

5.3 Algorithmes standards sur les vecteurs et les matrices

- La multiplication d'un vecteur de \mathbb{R}^n par un scalaire coûte $n = \Theta(n)$ multiplications.

$$y = \lambda x = \lambda \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \lambda x_1 \\ \lambda x_2 \\ \vdots \\ \lambda x_n \end{bmatrix} \quad \begin{array}{l} \text{pour } i = 1 \text{ à } n \text{ faire} \\ \quad y(i) \leftarrow \lambda * x(i) \\ \text{fin pour} \end{array}$$

Exemple : opérations sur les lignes dans la résolution d'un système.

- La multiplication d'une matrice $n \times p$ par un scalaire coûte $np = \Theta(np)$ multiplications, donc $\Theta(n^2)$ si la matrice est carrée.

$$B = \lambda A = \lambda \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1p} \\ a_{21} & a_{22} & \dots & a_{2p} \\ \vdots & & & \vdots \\ a_{n1} & \dots & \dots & a_{np} \end{bmatrix} \quad \begin{array}{l} \text{pour } i = 1 \text{ à } n \text{ faire} \\ \quad \text{pour } j = 1 \text{ à } p \text{ faire} \\ \quad \quad b(i, j) \leftarrow \lambda * a(i, j) \\ \quad \text{fin pour} \\ \text{fin pour} \end{array}$$

- Une addition de vecteurs de \mathbb{R}^n coûte n additions, i.e. $\Theta(n)$ flops.

$$z = x + y = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_n + y_n \end{bmatrix} \quad \begin{array}{l} \text{pour } i = 1 \text{ à } n \text{ faire} \\ \quad z(i) \leftarrow x(i) + y(i) \\ \text{fin pour} \end{array}$$

Exemples : opérations sur les lignes dans la résolution d'un système.

- Un produit scalaire dans \mathbb{R}^n coûte n multiplications et $n - 1$ additions, i.e. $\Theta(n)$ flops.

$$x \cdot y = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \sum_{i=1}^n x_i y_i$$

```

s ← 0
pour i = 1 à n faire
    s ← s + x(i) * y(i)
fin pour

```

Exemples : calcul de coordonnées dans une base, calcul de distances euclidiennes, de normes.

- Le produit d'une matrice $n \times p$ par un vecteur de \mathbb{R}^p coûte $\Theta(np)$ flops.

$$y = Ax, \quad y_i = \sum_{j=1}^p a_{ij} x_j$$

```

pour i = 1 à n faire
    y(i) ← 0
    pour j = 1 à p faire
        y(i) ← y(i) + a(i, j) * x(j)
    fin pour
fin pour

```

- Le produit d'une matrice $n \times p$ par une matrice $p \times q$ coûte $\Theta(npq)$ flops, donc $\Theta(n^3)$ si les matrices sont carrées.

$$C = AB, \quad c_{ij} = \sum_{k=1}^p a_{ik} b_{kj}$$

```

pour i = 1 à n faire
    pour j = 1 à q faire
        c(i, j) ← 0
        pour k = 1 à p faire
            c(i, j) ← c(i, j) + a(i, k) * b(k, j)
        fin pour
    fin pour
fin pour

```

5.4 Représentations informatiques des matrices

Dans la section précédente, pour calculer la complexité des algorithmes sur les matrices, nous avons négligé le temps nécessaire pour accéder aux données, en ne comptant que les opérations arithmétiques. Cette approximation est valide tant que deux hypothèses sont vérifiées :

1. La structure de données est à accès aléatoire, c'est à dire que l'accès aux données se fait en temps constant. Par exemple le temps d'accès à un élément a_{ij} d'une matrice A de taille $n \times n$ ne doit pas dépendre de la valeur de n , ni de i , ni de j .
2. Les données tiennent en mémoire centrale (mémoire vive, RAM = Random Access Memory). Ceci dépend essentiellement de la taille des données. Nous supposons qu'elles sont suffisamment petites pour que cette deuxième hypothèse soit vérifiée.

Quand les matrices sont pleines (*dense* en anglais), les matrices sont habituellement codées dans des tableaux 1D ou 2D. Ceci assure un accès aléatoire, car l'adresse de a_{ij} se calcule directement à partir de i , j , et n .

Dans de nombreux problèmes les matrices contiennent beaucoup d'éléments nuls. On dit qu'elles sont creuses (*sparse* en anglais). On peut alors économiser de la place en ne stockant pas les éléments nuls (on parle de "formats creux"), ce qui conduit à des algorithmes plus rapides mais souvent plus complexes. Supposons que A , de taille $n \times n$ n'ait que m éléments non nuls, m étant petit devant n^2 . L'idée est de stocker m éléments (seulement les $a_{ij} \neq 0$) au lieu de n^2 . en contrepartie, il peut être nécessaire de stocker les indices i et j , et il arrive que l'on perde la propriété d'accès aléatoire.

Chapitre 6

Résolution de systèmes linéaires

Notes personnelles.

Les systèmes linéaires apparaissent souvent en informatique, en mathématiques, et en physique. Les problèmes d'optimisation, par exemple, utilisent beaucoup de résolutions de systèmes linéaires.

Il existe des méthodes variées pour résoudre un système linéaire, selon la forme de la matrice (pleine, creuse, symétrique, etc.), selon qu'on veut résoudre le problème une seule ou plusieurs fois, selon qu'on a besoin d'un résultat précis, rapide ou stable. C'est un sujet important d'analyse numérique. Nous revoyons ici la méthode de Gauss, vue en Algèbre 1.

6.1 Position du problème

On veut résoudre le système d'équations linéaires de n équations à n inconnues suivant :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases} \quad (6.1)$$

Il s'écrit sous forme matricielle $AX = B$ avec

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix}, \quad X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad B = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

- A est une matrice (n lignes et n colonnes) contenant les coefficients du système linéaire.
- X est le vecteur des inconnues.
- B est le second membre, ou membre de droite.

On suppose que le système est inversible, c'est à dire que $\det(A) \neq 0$, c'est à dire que $\text{rang}(A) = n$, c'est à dire qu'il possède une seule solution.

Remarque. Si A n'est pas inversible, il peut y avoir plusieurs solutions. Par exemple, si l'on cherche X tel que $AX = b$ avec

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 \end{bmatrix} \quad b = \begin{bmatrix} 3 \\ 3 \\ 3 \\ 0 \\ 3 \end{bmatrix}.$$

Tous les vecteurs $X = \begin{bmatrix} 3 \\ 3/2 \\ 1 \\ x \\ 3/5 \end{bmatrix}$ avec $x \in \mathbb{R}$ sont solution. Il se peut aussi qu'il n'y ait aucune solution.

Par exemple, si l'on cherche X tel que $AX = b$ avec

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 \end{bmatrix} \quad b = \begin{bmatrix} 3 \\ 3 \\ 3 \\ 3 \\ 3 \end{bmatrix}.$$

Dans tous les algorithmes, on supposera que A , X et B sont des tableaux passés en paramètre par variable (donc modifiés par effet de bord).

On note la i -ème équation $L_i : \sum_{j=1}^n a_{ij}x_j = b_j$.

6.2 Exemple

On résout le système

$$\begin{aligned} \begin{bmatrix} 1 & -1 & 2 \\ 3 & 2 & 1 \\ 2 & -3 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= \begin{bmatrix} 5 \\ 10 \\ -10 \end{bmatrix} \Leftrightarrow \begin{cases} x_1 - x_2 + 2x_3 = 5 \\ 3x_1 + 2x_2 + x_3 = 10 \\ 2x_1 - 3x_2 - 2x_3 = -10 \end{cases} \\ \Leftrightarrow \begin{bmatrix} 1 & -1 & 2 \\ 0 & 5 & -5 \\ 0 & -1 & -6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= \begin{bmatrix} 5 \\ -5 \\ -20 \end{bmatrix} \quad \text{par } \begin{cases} L_2 \leftarrow L_2 - 3L_1 \\ L_3 \leftarrow L_3 - 2L_1 \end{cases} \\ \Leftrightarrow \begin{bmatrix} 1 & -1 & 2 \\ 0 & 5 & -5 \\ 0 & 0 & -7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= \begin{bmatrix} 5 \\ -5 \\ -21 \end{bmatrix} \quad \text{par } L_3 \leftarrow L_3 + \frac{1}{5}L_2 \end{aligned}$$

On en déduit les valeurs de x_3 puis x_2 puis x_1 :

$$\begin{aligned} x_3 &= \frac{-21}{-7} = 3 \\ x_2 &= \frac{-5 + 5x_3}{5} = 2 \\ x_1 &= \frac{5 + x_2 - 2x_3}{1} = 1 \end{aligned}$$

On a mis le système sous forme triangulaire (avec des 0 sous la diagonale), puis on a calculé les x_i en “remontant”.

Exercice 6.1

En suivant la même méthode, résoudre le système

$$\begin{bmatrix} 1 & 2 & -1 \\ 3 & 1 & 3 \\ 2 & 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ -1/5 \end{bmatrix}$$

6.3 Étape 1 : méthode du pivot de Gauss

L'objectif est de transformer un système *quelconque* en système *triangulaire supérieur* équivalent, puis de faire une remontée. Pour cela partons du système $AX = B$ (6.1).

Nous aurons besoin d'un algorithme qui additionne à une équation du multiple d'une autre : $L_i \leftarrow L_i + c * L_j$ (ALGO. (6.1)).

ALGO. 6.1 *addmultiple*(A, B, i, j, c) Algorithme d'addition d'un multiple $L_i \leftarrow L_i + c * L_j$.

pour k de 1 à n **faire**
 $A(i, k) \leftarrow A(i, k) + c * A(j, k)$
fin pour
 $B(i) \leftarrow B(i) + c * B(j)$

Pour obtenir un système triangulaire supérieur, on procède en $n - 1$ étapes qui mettent des 0 sous la diagonale grâce à des combinaisons de lignes :

- Première étape : on fait apparaître des 0 dans la première colonne de A sous la diagonale. En supposant que $a_{11} \neq 0$, il faut soustraire $\frac{a_{i1}}{a_{11}}L_1$ à L_j , pour $2 \leq j \leq n$. Ceci donne l'algorithme

pour j de 2 à n **faire**

$addmultiple(A, B, j, 1, -A(j, 1)/A(1, 1))$

fin pour

- Deuxième étape : on fait apparaître des 0 dans la deuxième colonne de A sous la diagonale. Ceci donne l'algorithme

pour j de 3 à n **faire**

$addmultiple(A, B, j, 2, -A(j, 2)/A(2, 2))$

fin pour

- Et ainsi de suite, jusqu'à la colonne $n - 1$. L'étape i fait apparaître des 0 sous la diagonale dans la colonne i :

pour j de $i + 1$ à n **faire**

$addmultiple(A, B, j, i, -A(j, i)/A(i, i))$

fin pour

À la i -ème étape on divise par $A(i, i)$, appelé "pivot". Donc il faut que $A(i, i) \neq 0$. Ce n'est pas toujours le cas, mais il existe toujours une valeur non nulle parmi $A(i, i), A(i + 1, i), \dots, A(n, i)$ (car A est inversible). Donc si $A(i, i)$ est nul, on échange d'abord L_i avec L_j tel que $j > i$ et $A(j, i) \neq 0$, de façon à ce que le pivot soit non nul. Il nous faut donc deux autres algorithmes élémentaires :

1. La recherche du premier $j \geq i$ tel que $A(j, i) \neq 0$ (ALGO. (6.2)).
2. L'échange de deux équations L_i et L_j (ALGO. (6.3)).

On obtient donc l'algorithme qui rend le système triangulaire (ALGO. (6.4)).

ALGO. 6.2 *choixPivot*(A, i) Algorithme de recherche du premier $j \geq i$ tel que $A(j, i) \neq 0$.

pour j de i à n **faire**

si $A(j, i) \neq 0$ **alors**

renvoyer j

fin si

fin pour

renvoyer (-1)

ALGO. 6.3 *echangeLigne*(A, B, i, j) Algorithme d'échange de L_i et L_j .

pour k de 1 à n **faire**

$tmp \leftarrow A(i, k)$

$A(i, k) \leftarrow A(j, k)$

$A(j, k) \leftarrow tmp$

fin pour

$tmp \leftarrow B(i)$

$B(i) \leftarrow B(j)$

$B(j) \leftarrow tmp$

ALGO. 6.4 *triangulaire*(A, B) Algorithme qui rend le système triangulaire.

pour i de 1 à $n - 1$ **faire**

$j \leftarrow \text{choixPivot}(A, i)$

echangeLigne(A, B, i, j)

pour j de $i + 1$ à n **faire**

$addmultiple(A, B, j, i, -A(j, i)/A(i, i))$

fin pour

fin pour

6.4 Étape 2 : méthode de remontée

On se donne le système triangulaire supérieur

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22} & \dots & a_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

La solution s'obtient facilement par une remontée : calcul de x_n avec la ligne n , puis de x_{n-1} avec la ligne $n-1$, etc :

$$\begin{aligned} x_n &= \frac{b_n}{a_n} \\ x_{n-1} &= \frac{b_{n-1} - a_{n-1,n}x_n}{a_{n-1,n-1}} \\ x_{n-2} &= \frac{b_{n-2} - a_{n-2,n}x_n - a_{n-2,n-1}x_{n-1}}{a_{n-2,n-2}} \\ &\vdots \\ x_i &= \left(b_i - \sum_{j=i+1}^n a_{ij}x_j \right) / a_{i,i} \\ &\vdots \\ x_1 &= \left(b_1 - \sum_{j=2}^n a_{1j}x_j \right) / a_{1,1} \end{aligned}$$

ALGO. 6.5 *remontee*(A, B, X). Algorithme de remontée.

```

pour  $i$  descendant de  $n$  à 1 faire
   $X(i) \leftarrow B(i)$ 
  pour  $j$  de  $i+1$  à  $n$  faire
     $X(i) \leftarrow X(i) - A(i,j)X(j)$ 
  fin pour
   $X(i) \leftarrow X(i)/A(i,i)$ 
fin pour

```

Exercice 6.2

Écrire l'algorithme de descente pour résoudre un système triangulaire inférieur.

Remarque. Cet algorithme suppose que tous les éléments diagonaux (a_{ii}) sont non nuls. Ceci est vrai car $\det(A) = \prod_{i=1}^n a_{ii} \neq 0$.

6.5 Algorithme complet de résolution d'un système linéaire

L'algorithme complet de résolution est ALGO. (6.6). L'analyse de complexité donne un nombre total d'opérations arithmétiques $\Theta(n^3)$.

ALGO. 6.6 *gauss*(A, B, X) Algorithme résolution de $AX = B$.

triangulaire(A, B)

remontee(A, B, X)

6.6 Stabilité numérique

Dans la pratique, il se peut que le pivot $A(i, i)$ soit non nul mais très petit. Comme on divise par ce pivot pour rendre le système triangulaire, cela peut provoquer des erreurs numériques importantes.

Exemple 6.3

Supposons qu'on est en train de mettre 0 dans la colonne i à la ligne j : $L_j \leftarrow L_j - a_{ji}/a_{ii}L_i$.
Donc à la colonne k :

$$a_{jk} \leftarrow a_{jk} - \frac{a_{ji}}{a_{ii}}a_{ik}$$

Supposons que a_{ii} soit 1000 fois plus petit que a_{ji} et que a_{ik} soit entâché d'une erreur ϵ , alors

$$a_{jk} - \frac{a_{ji}}{a_{ii}}(a_{ik} + \epsilon) = a_{jk} - \frac{a_{ji}}{a_{ii}}a_{ik} - 1000\epsilon$$

L'erreur sur a_{jk} a été multipliée par 1000.

ALGO. 6.7 *choixPivotPartiel*(A, i) Algorithme de recherche du pivot partiel.

$p \leftarrow i$

$v \leftarrow \text{abs}(A(i, i))$

pour j de $i + 1$ à n **faire**

si $\text{abs}(A(j, i)) > v$ **alors**

$p \leftarrow j$

$v \leftarrow \text{abs}(A(j, i))$

fin si

fin pour

renvoyer p

La méthode dite du *pivot partiel* permet de limiter ces erreurs (améliorer la stabilité numérique), en choisissant le plus grand pivot (en valeur absolue) de la colonne sous la diagonale (ALGO. (6.7)).

Il existe un algorithme dit du "pivot total" encore plus stable. Le pivot est choisi parmi les lignes et les colonnes entre i et n . Cet algorithme nécessite des manipulations sur les colonnes, ce qui modifie l'ordre des inconnues.

Dans la pratique on implémente toujours au moins le pivot partiel.

Exercice 6.4

Écrire l'algorithme du pivot total. Faire attention à l'ordre des inconnues.

Chapitre 7

Approximation au sens des moindres carrés

Notes personnelles.

7.1 Position du problème

Au chapitre 6 nous avons étudié la résolution d'un système linéaire $Ax = b$ carré et inversible. Il se peut que le système ne soit pas carré (par exemple plus/moins d'équations que d'inconnues) ou non inversible (pas de solution, ou plusieurs solutions). Nous nous intéressons ici au cas où le système n'a pas de solution exacte. Cette situation est fréquente quand il y a plus d'équations que d'inconnues.

Le système a m équations à n inconnues, c'est à dire :

- A est de taille $m \times n$
- le vecteur d'inconnues x est dans \mathbb{R}^n
- le membre de droite b est dans \mathbb{R}^m

Comme il n'y a pas de x tel que $Ax = b$, nous allons chercher une solution \hat{x} telle que $A\hat{x}$ se rapproche le plus possible de b , c'est à dire telle que la distance $\|A\hat{x} - b\|$ est la plus petite possible.

On travaille dans \mathbb{R}^n et \mathbb{R}^m munis du produit scalaire classique et de la norme euclidienne.

Définition. Une *solution au sens des moindres carrés* à $Ax = b$ est un \hat{x} tel que pour tout $x \in \mathbb{R}^n$

$$\|A\hat{x} - b\| \leq \|Ax - b\|$$

Remarques.

- On parle de *moindres carrés* car la norme utilisée est la norme quadratique, qui somme des carrés des erreurs.
- Il s'agit plus précisément de moindres carrés *linéaires* car le problème est linéaire en x (les inconnues).

7.2 Résolution par projection orthogonale

Notons a_i la i -ième colonne de $A = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix}$

On remarque que, quel que soit x , le vecteur Ax appartient toujours au sous-espace de \mathbb{R}^m engendré par les colonnes de A :

$$Ax = x_1 a_1 + x_2 a_2 + \dots + x_n a_n \quad \in \mathcal{L}\{a_1, \dots, a_n\} = \text{Col}(A)$$

On veut que $A\hat{x}$ soit dans $\text{Col}(A)$ le vecteur le plus proche de b : c'est la projection orthogonale de b sur $\text{Col}(A)$. Cela implique que le vecteur $A\hat{x} - b$ est orthogonal à tous les vecteurs de $\text{Col}(A)$, c'est à dire à tous les a_i :

$$\langle a_i, A\hat{x} - b \rangle = 0 \quad \text{pour tout } i \quad \Leftrightarrow \quad a_i^T (A\hat{x} - b) = 0 \quad \text{pour tout } i$$

Comme a_i^T est une ligne de A^T , ceci équivaut à

$$A^T (A\hat{x} - b) = 0 \quad \Leftrightarrow \quad A^T A\hat{x} = A^T b$$

Le système linéaire $A^T A\hat{x} = A^T b$ est appelé *équations normales* de $Ax = b$. On montre que les solutions des équations normales coïncident avec l'ensemble des solutions au sens des moindres carrés de $Ax = b$.

La matrice $A^T A$ est symétrique et ses valeurs propres sont positives ou nulles. Si elle est inversible (pas de valeur propre nulle), alors il y a une unique solution au sens des moindres carrés : $\hat{x} = (A^T A)^{-1} A^T b$. Dans la pratique, plutôt que d'inverser $A^T A$, on résoudra le système linéaire des équations normales.

7.3 Résolution par minimisation de la distance quadratique

Voici un autre éclairage. Minimiser $\|A\mathbf{x} - \mathbf{b}\|$ est la même chose que minimiser son carré :

$$E(\mathbf{x}) = \|A\mathbf{x} - \mathbf{b}\|^2 = (A\mathbf{x} - \mathbf{b})^T (A\mathbf{x} - \mathbf{b}) = (\mathbf{x}^T A^T - \mathbf{b}^T)(A\mathbf{x} - \mathbf{b}) = \mathbf{x}^T A^T A \mathbf{x} - 2\mathbf{x}^T A^T \mathbf{b} + \mathbf{b}^T \mathbf{b}$$

La fonction E (de \mathbb{R}^n dans \mathbb{R}) est convexe et son gradient est

$$\nabla E(\mathbf{x}) = 2A^T A \mathbf{x} - 2A^T \mathbf{b}$$

Les valeurs minimales sont atteintes quand le gradient s'annule, c'est à dire $\nabla E(\mathbf{x}) = \mathbf{0}$. On retrouve les équations normales.

7.4 Régression linéaire

Supposons qu'on mesure expérimentalement des couples de valeurs (x_i, y_i) . On observe que ces points sont à peu près alignés sur une droite. Quelle droite est la plus proche des mesures ?

Donnons nous la droite d'équation $y = \alpha x + \beta$. Cette droite est-elle proche des observations ?

Pour chaque x_i , on dispose d'une valeur *observée* y_i , et d'une valeur *prédite* $\alpha x_i + \beta$. La différence entre ces deux valeurs est appelée *résidu*. Quelle est l'erreur entre les valeurs prédites et les valeurs observées ?

Une façon courante de calculer cette erreur est de faire la somme des carrés des résidus :

$$E(\alpha, \beta) = \sum_{i=1}^m ((\alpha x_i + \beta) - y_i)^2$$

La droite qui minimise cette erreur est appelée *droite des moindres carrés*, ou *droite de régression*. Comment la calculer ? C'est-à-dire comment calculer α et β ?

Idéalement, on voudrait que les données prédites soient égales aux données observées :

$$\begin{cases} \alpha x_1 + \beta = y_1 \\ \alpha x_2 + \beta = y_2 \\ \vdots \\ \alpha x_m + \beta = y_m \end{cases} \Leftrightarrow \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_m & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \Leftrightarrow A \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \mathbf{y} \quad (7.1)$$

Ce système n'a généralement pas de solution exacte, mais il a une solution au sens des moindres carrés. Cette solution coïncide avec la droite des moindres carrés, qui minimise

$$E \left(\begin{bmatrix} \alpha \\ \beta \end{bmatrix} \right) = \left\| A \begin{bmatrix} \alpha \\ \beta \end{bmatrix} - \mathbf{y} \right\|^2$$

Bilan. Pour trouver une droite des moindres carrés il faut :

- Écrire le système (7.1).
- Écrire les équations normales.
- Résoudre les équations normales pour trouver α et β .

Remarque. On parle de régression *linéaire* car l'équation d'une droite est linéaire en x . Il ne faut pas confondre avec la linéarité en α et β (les inconnues) : des régressions *non-linéaires* peuvent se résoudre par des moindres carrés *linéaires*.

Exercice 7.1

Calculer la droite des moindres carrés pour les observations $(2, 1)$, $(5, 2)$, $(7, 3)$ et $(8, 3)$.

Exercice 7.2

Reprendre l'exemple 9.6 : simuler l'évolution des populations pour un u_0 quelconque, puis faire une régression sur R_k en fonction de C_k . On doit retrouver le ratio rats/chouettes d'un vecteur propre. Observer le résultat de la régression pour des périodes plus ou moins longues. Essayer aussi en négligeant les premiers mois.

Faire deux régressions linéaires séparées sur C_k (fonction de C_{k-1}) et sur R_k (fonction de R_{k-1}). On doit retrouver le taux de croissance de 2%.

7.5 Approximation polynômiale

Dans la section précédente, nous avons approximé les données (x, y) par une droite. Si les données ne sont pas alignées sur une droite, mais sur une courbe plus complexe, il est possible de les approximer par une fonction plus complexe, par exemple un polynôme $P(x)$, en faisant le même raisonnement. Cherchons un polynôme P de degré $k < n$ tel que $P(x_i) \approx y_i$ pour tout i . La question essentielle est de définir ce que signifie " $P(x_i) \approx y_i$ pour tout i ". On définit l'erreur quadratique comme

$$E = \sum_{i=1}^n (P(x_i) - y_i)^2$$

On cherche le polynôme P qui minimise E . Pour cela, écrivons P dans la base des monômes :

$$P(x) = c_0 + c_1x + \dots + c_jx^j + \dots + c_kx^k = [1, x, x^2, \dots, x^k] \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_k \end{bmatrix}$$

Donc

$$Ac - \mathbf{y} = \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^k \\ 1 & x_2 & x_2^2 & \dots & x_2^k \\ \vdots & & & & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^k \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_k \end{bmatrix} - \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} P(x_1) - y_1 \\ P(x_2) - y_2 \\ \vdots \\ P(x_n) - y_n \end{bmatrix}$$

D'où la reformulation de l'erreur

$$E(\mathbf{c}) = \|A\mathbf{c} - \mathbf{y}\|^2,$$

dont le minimum est atteint pour \mathbf{c} vérifiant les équations normales

$$A^T A \mathbf{c} = A^T \mathbf{y}.$$

Les équations normales sont un système linéaire, de taille $(k+1) \times (k+1)$ qui est symétrique semi-défini positif.

7.6 Moindres carrés pondérés, une question de confiance

Supposons maintenant qu'on a plus confiance dans certains points que dans d'autres. On donne un poids $0 \leq w_i \leq 1$ à chaque point (x_i, y_i) . Ce poids est un indice de confiance :

- $w_i = 0$ signifie qu'on n'a pas du tout confiance en (x_i, y_i) .
- $w_i = 1$ signifie qu'on a une grande confiance en (x_i, y_i) .

On ré-écrit l'erreur en accordant plus d'importance aux points avec un fort indice de confiance :

$$E = \sum_{i=1}^n w_i^2 (P(x_i) - y_i)^2 = \|WAC - WY\|^2$$

avec W matrice $n \times n$ diagonale :

$$W = \begin{bmatrix} w_1 & & 0 \\ & w_2 & \\ 0 & & \ddots \\ & & & w_n \end{bmatrix}$$

Les équations normales s'écrivent donc :

$$(WA)^T(WA)C = (AW)^T WY \Leftrightarrow A^T W W A C = A^T W W Y$$

7.7 Changements de variable

Pour analyser le lien entre x et y , il peut être utile de faire un changement de variable : on observe le lien entre x' et y' , obtenues depuis x et y par un changement de variable.

Exemple 7.3

Si l'on pense que nos variables sont liées par un loi exponentielle $y = \alpha^x$, il est utile d'observer

$$y' = \log(y) \text{ en fonction de } x' = x$$

car la relation devient linéaire :

$$y' = \log(\alpha^x) = \log(\alpha)x = \log(\alpha)x'.$$

Une régression linéaire sur (x', y') permet d'estimer le coefficient d'exponentiation α .

Cette technique revient à observer (x, y) en échelle semi-logarithmique, très classique dans les outils de visualisation. On pourrait dire que cette échelle “transforme les exponentielles en droites”.

Exemple 7.4

Si l'on pense que nos variables sont liées par un loi de puissance $y = x^p$, il est utile d'observer

$$y' = \log(y) \text{ en fonction de } x' = \log(x)$$

car la relation devient linéaire :

$$y' = \log(x^p) = p \log(x) = p x'.$$

Une régression linéaire sur (x', y') permet d'estimer le puissance p .

Cette technique revient à observer (x, y) en échelle logarithmique, dite “log-log”, elle aussi très classique dans les outils de visualisation.

Chapitre 8

Applications linéaires

Notes personnelles.

8.1 Notations

Soient \mathbf{E} un espace vectoriel sur \mathbb{R} , muni d'une base $e = \{e_1, \dots, e_n\}$. Tout vecteur \mathbf{u} est représenté de manière unique dans cette base par son vecteur de coordonnées dans \mathbb{R}^n noté

$$\mathbf{u}^{(e)} = \begin{bmatrix} u_1^{(e)} \\ \vdots \\ u_n^{(e)} \end{bmatrix} \quad \text{tel que} \quad \mathbf{u} = \sum_{i=1}^n u_i^{(e)} \mathbf{e}_i$$

Attention : dans beaucoup d'exemples $\mathbf{E} = \mathbb{R}^n$ et les vecteurs se confondent avec leurs coordonnées dans la base canonique.

Exemple 8.1

Soit $\mathbf{E} = \mathbb{R}^2$ muni de sa base canonique $e = \left\{ \mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\}$.

Le vecteur $\mathbf{u} = \begin{bmatrix} 2 \\ -1 \end{bmatrix} \in \mathbf{E}$ s'écrit $\mathbf{u} = 2\mathbf{e}_1 + (-1)\mathbf{e}_2$ et donc les coordonnées $\mathbf{u}^{(e)} = \begin{bmatrix} u_1^{(e)} \\ u_2^{(e)} \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \end{bmatrix}^{(e)}$ sont confondues avec le vecteur \mathbf{u} lui-même.

8.2 Espaces vectoriels et espaces affines

On fait habituellement de la géométrie dans un espace *affine*, qui dispose de points en plus des vecteurs. On y définit des *repères*, constitués d'une base de vecteurs et d'une origine. Il est alors possible de définir des applications affines (par exemple les translations), des droites passant par des points, etc.

Dans un espace *vectoriel*, on ne dispose que de vecteurs. On peut y définir des applications linéaires, mais pas affines. Il existe des droites et des plans dits *vectoriels*, qui sont des sous-espaces vectoriels. On les représente passant par l'origine.

Exemple 8.2

Soit \mathcal{D} une droite vectorielle, de vecteur directeur \mathbf{d} .
Par définition, c'est le sous-espace engendré par \mathbf{d} : $\mathcal{D} = \mathcal{L}\{\mathbf{d}\}$.

8.3 Applications linéaires

Définition. Soient \mathbf{E} et \mathbf{F} deux espaces vectoriels sur \mathbb{R} . Une application l de \mathbf{E} dans \mathbf{F} est dite linéaire si

- $l(\mathbf{u} + \mathbf{v}) = l(\mathbf{u}) + l(\mathbf{v})$ pour tout $\mathbf{u}, \mathbf{v} \in \mathbf{E}$.
- $l(c\mathbf{u}) = cl(\mathbf{u})$ pour tout $\mathbf{u} \in \mathbf{E}$ et $c \in \mathbb{R}$.

Remarque. On en déduit que $l(\mathbf{0}) = \mathbf{0}$

Représentation dans une base.

Soient n et p les dimensions respectives de \mathbf{E} et \mathbf{F} . Soient $e = \{e_1, \dots, e_n\}$ une base de \mathbf{E} , et $f = \{f_1, \dots, f_p\}$ une base de \mathbf{F} . Une application linéaire $l : \mathbf{E} \rightarrow \mathbf{F}$ s'écrit

$$l(\mathbf{u}) = l\left(\sum_{i=1}^n u_i^{(e)} \mathbf{e}_i\right) = \sum_{i=1}^n u_i^{(e)} l(\mathbf{e}_i) = \sum_{i=1}^n u_i^{(e)} \left(\sum_{j=1}^p l_{ji} \mathbf{f}_j\right) = \sum_{j=1}^p \left(\sum_{i=1}^n u_i^{(e)} l_{ji}\right) \mathbf{f}_j = \sum_{j=1}^p v_j^{(f)} \mathbf{f}_j$$

Notons L la matrice des l_{ji} (de taille $p \times n$). Si on représente $\mathbf{u} \in \mathbf{E}$ par ses coordonnées $\mathbf{u}^{(e)} \in \mathbb{R}^n$ dans la base e , et $\mathbf{v} = l(\mathbf{u}) \in \mathbf{F}$ par ses coordonnées $\mathbf{v}^{(f)} \in \mathbb{R}^p$ dans la base f , l'application l s'écrit $l^{(e,f)} : \mathbb{R}^n \rightarrow \mathbb{R}^p$

$$\mathbf{v}^{(f)} = l^{(e,f)}(\mathbf{u}^{(e)}) = L\mathbf{u}^{(e)} = \begin{bmatrix} l_{11} & l_{12} & \dots & l_{1n} \\ l_{21} & l_{22} & \dots & l_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ l_{p1} & \dots & \dots & l_{pn} \end{bmatrix} \begin{bmatrix} u_1^{(e)} \\ \vdots \\ u_n^{(e)} \end{bmatrix} = \begin{bmatrix} v_1^{(f)} \\ \vdots \\ v_p^{(f)} \end{bmatrix}$$

La i -ème colonne de L contient les coordonnées (dans la base f) l'image $l(\mathbf{e}_i)$ du vecteur \mathbf{e}_i :

$$l^{(e,f)}(\mathbf{e}_i^{(e)}) = l^{(e,f)} \left(\begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}^{(e)} \right) = \begin{bmatrix} l_{11} & l_{12} & \dots & l_{1n} \\ l_{21} & l_{22} & \dots & l_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ l_{p1} & \dots & \dots & l_{pn} \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}^{(e)} = \begin{bmatrix} l_{1i} \\ l_{2i} \\ \vdots \\ l_{pi} \end{bmatrix}^{(f)}$$

Donc $l(\mathbf{u})$ dans la base f est une combinaison linéaire des colonnes de L :

$$l^{(e,f)}(\mathbf{u}^{(e)}) = \sum_{i=1}^n u_i^{(e)} l(\mathbf{e}_i^{(e)}) = \sum_{i=1}^n u_i^{(e)} \begin{bmatrix} l_{1i} \\ l_{2i} \\ \vdots \\ l_{pi} \end{bmatrix}^{(f)}. \quad (8.1)$$

Remarques :

- Ne pas confondre l et $l^{(e,f)}$.
- Attention, la matrice L dépend des bases e et f .

8.4 Noyau et image d'une application linéaire

Noyau.

Le noyau (kernel) de l , noté $\text{Ker}(l)$, est le sous-espace vectoriel de \mathbf{E} des vecteurs d'image nulle :

$$\text{Ker}(l) = \{\mathbf{u} \in \mathbf{E} / l(\mathbf{u}) = \mathbf{0}\}$$

Image.

L'image de l , noté $\text{Im}(l)$ est le sous-espace vectoriel de \mathbf{F} des vecteurs image :

$$\text{Im}(l) = \{\mathbf{v} \in \mathbf{F} / \exists \mathbf{u} \in \mathbf{E}, \mathbf{v} = l(\mathbf{u})\}$$

Remarques :

- Théorème du rang : $\dim(\text{Ker}(l)) + \dim(\text{Im}(l)) = \dim(\mathbf{E}) = n$.
- $\text{Im}(l)$ est l'espace engendré par les colonnes de L , vues comme des vecteurs de coordonnées dans la base f (confer Eq. (8.1)) :

$$\text{Im}(l) = \mathcal{L} \left\{ \begin{bmatrix} l_{1,1} \\ \vdots \\ l_{p,1} \end{bmatrix}^{(f)}, \dots, \begin{bmatrix} l_{1,n} \\ \vdots \\ l_{p,n} \end{bmatrix}^{(f)} \right\}$$

Lien avec la résolution de systèmes linéaires.

Dans le chapitre 6 nous cherchions à résoudre un système $AX = B$, avec X et B dans \mathbb{R}^n . Notons $l : \mathbb{R}^n \rightarrow \mathbb{R}^n$ l'application linéaire qui à X associe $l(X) = AX$. Le système s'écrit alors $l(X) = B$. Donc résoudre le système revient à chercher le(s) vecteur(s) dont l'image par l est B .

8.5 Exemple des projections

Exemple 8.3 projection sur une droite dans \mathbb{R}^2 .

On se place dans \mathbb{R}^2 muni du produit scalaire euclidien. Soit $\mathcal{D} = \mathcal{L}\{\mathbf{d}\}$ une droite vectorielle, de vecteur directeur \mathbf{d} unitaire ($\|\mathbf{d}\| = 1$). La projection orthogonale sur \mathcal{D} est l'application :

$$p : \mathbb{R}^2 \rightarrow \mathbb{R}^2$$

$$\mathbf{u} \mapsto p(\mathbf{u}) = \langle \mathbf{u}, \mathbf{d} \rangle \mathbf{d}$$

Prenons $\mathbf{d} = \frac{1}{\sqrt{5}} \begin{bmatrix} 2 \\ 1 \end{bmatrix}$. Dans la base canonique :

$$p(\mathbf{u}) = \left\langle \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \frac{1}{\sqrt{5}} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\rangle \frac{1}{\sqrt{5}} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \frac{1}{5} (2u_1 + u_2) \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \frac{1}{5} \begin{bmatrix} 4u_1 + 2u_2 \\ 2u_1 + u_2 \end{bmatrix} = \frac{1}{5} \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

Donc $L = \frac{1}{5} \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix}$.

La projection de $\begin{bmatrix} 1 \\ 3 \end{bmatrix}$ est $p\left(\begin{bmatrix} 1 \\ 3 \end{bmatrix}\right) = \frac{1}{5} \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \end{bmatrix} = \frac{1}{5} \begin{bmatrix} 10 \\ 5 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$

Exemple 8.4 projection sur une droite dans \mathbb{R}^3 .

Identique à la projection dans \mathbb{R}^2 .

Exemple 8.5 projection sur un plan dans \mathbb{R}^3 .

On se place dans \mathbb{R}^3 muni du produit scalaire euclidien. Soit \mathcal{P} un plan vectoriel de normale \mathbf{n} unitaire. Donc $\langle \mathbf{u}, \mathbf{n} \rangle = 0$ pour tout $\mathbf{u} \in \mathcal{P}$, et réciproquement. La projection orthogonale sur \mathcal{P} est l'application $p : \mathbb{R}^3 \rightarrow \mathbb{R}^3$:

$$p(\mathbf{u}) = \mathbf{u} - \langle \mathbf{u}, \mathbf{n} \rangle \mathbf{n}$$

Le projeté $p(\mathbf{u})$ appartient à \mathcal{P} , car $\langle p(\mathbf{u}), \mathbf{n} \rangle = 0$ (exercice).

La projection est une application linéaire (exercice) :

- Pour tout $\mathbf{u}, \mathbf{v} \in \mathbb{R}^3$, $p(\mathbf{u} + \mathbf{v}) = p(\mathbf{u}) + p(\mathbf{v})$.
- Pour tout $\mathbf{u} \in \mathbb{R}^3$ et $c \in \mathbb{R}$, $p(c\mathbf{u}) = cp(\mathbf{u})$.

Exemple 8.6 projection sur une droite dans \mathbb{R}^2 .

Quels sont le noyau et l'image de la projection de l'exemple 8.3 ?

Attention ici $\mathbf{E} = \mathbf{F} = \mathbb{R}^2$ sont confondus.

$$\mathbf{u} \in \text{Ker}(p) \Leftrightarrow p(\mathbf{u}) = \mathbf{0} \Leftrightarrow \langle \mathbf{u}, \mathbf{d} \rangle = 0$$

Donc $\text{Ker}(p)$ est l'ensemble des vecteurs orthogonaux à \mathbf{d} . C'est une droite vectorielle dans \mathbf{E} : la droite orthogonale à \mathcal{D} . Sur l'exemple numérique

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \in \text{Ker}(p) \Leftrightarrow \left\langle \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\rangle = 0 \Leftrightarrow 2u_1 + u_2 = 0$$

L'image est l'espace (dans \mathbf{F}) engendré par les colonnes de L :

$$\text{Im}(l) = \mathcal{L} \left\{ \frac{1}{5} \begin{bmatrix} 4 \\ 2 \end{bmatrix}, \frac{1}{5} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\} = \mathcal{L} \left\{ \begin{bmatrix} 4 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\} = \mathcal{L} \left\{ \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\} = \mathcal{L} \left\{ \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\}$$

C'est la droite \mathcal{D} elle-même (considérée dans \mathbf{F}).

Exemple 8.7 projection sur un plan dans \mathbb{R}^3 .

Quels sont le noyau et l'image de la projection de l'exemple 8.5 ?

On a $\mathbf{u} \in \text{Ker}(p) \Leftrightarrow p(\mathbf{u}) = \mathbf{0} \Leftrightarrow \mathbf{u} = \langle \mathbf{u}, \mathbf{n} \rangle \mathbf{n}$. Donc il faut que \mathbf{u} et \mathbf{n} soient colinéaires (condition nécessaire). C'est aussi une condition suffisante (exercice), donc $\text{Ker}(p) = \mathcal{L}\{\mathbf{n}\}$ est le sous-espace engendré par \mathbf{n} .

8.6 Changement de base : matrices de passage

Soient $e = \{e_1, \dots, e_n\}$ et $e' = \{e'_1, \dots, e'_n\}$ deux bases d'un espace vectoriel \mathbf{E} . Tout vecteur $u \in \mathbf{E}$ a des vecteurs coordonnées $u^{(e)}$ et $u^{(e')}$ dans ces bases. Il existe une unique matrice $P_{e \rightarrow e'}$, appelée matrice de passage de la base e dans la base e' , telle que

$$u^{(e')} = P_{e \rightarrow e'} u^{(e)}$$

Propriétés.

- Les colonnes de cette matrice sont les coordonnées dans e' des vecteurs de la base e :

$$P_{e \rightarrow e'} = [e_1^{(e')}, e_2^{(e')}, \dots, e_n^{(e')}]$$

- La matrice $P_{e \rightarrow e'}$ est carrée et inversible.
- La matrice de passage de e' dans e est son inverse :

$$P_{e' \rightarrow e} = (P_{e \rightarrow e'})^{-1}$$

- La matrice $P_{e \rightarrow e'}$ représente l'identité (application linéaire) de \mathbf{E} muni de e dans \mathbf{E} muni de e' .

Exemple 8.8

Plaçons nous dans \mathbb{R}^2 muni des bases $e = \left\{ e_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, e_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\}$ et $e' = \left\{ e'_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, e'_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \right\}$. Ces bases peuvent être vues comme des grilles pour repérer les vecteurs de \mathbb{R}^2 .

Les matrices de passage sont

$$P_{e' \rightarrow e} = [e_1^{(e')}, e_2^{(e')}] = \begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix} \quad \text{et} \quad P_{e \rightarrow e'} = [e_1^{(e)}, e_2^{(e)}] = \frac{1}{3} \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}$$

Le vecteur u a pour coordonnées $u^{(e)} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$ donc $u^{(e')} = P_{e \rightarrow e'} u^{(e)} = \frac{1}{3} \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} 3 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \end{bmatrix}$.

Le vecteur v a pour coordonnées $v^{(e')} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ donc $v^{(e)} = P_{e' \rightarrow e} v^{(e')} = \begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$.

Exemple 8.9

Reprenons l'exemple du chapitre 2. Dans $\mathbf{E} = \mathbb{R}^3$ muni de la base e

$$e_1 = \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix}, e_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, e_3 = \begin{bmatrix} 0 \\ 1 \\ -5 \end{bmatrix}$$

et de sa base canonique e' .

$$P_{e \rightarrow e'} = \begin{bmatrix} 1 & 1 & 0 \\ 2 & 1 & 1 \\ -2 & 0 & -5 \end{bmatrix}$$

Le vecteur $u = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix}$ a pour coordonnées $u^{(e)} = \begin{bmatrix} 2 \\ 0 \\ -1 \end{bmatrix}$ et $u^{(e')} = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix}$.

On vérifie que

$$u^{(e')} = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} = P_{e \rightarrow e'} u^{(e)} = \begin{bmatrix} 1 & 1 & 0 \\ 2 & 1 & 1 \\ -2 & 0 & -5 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ -1 \end{bmatrix}$$

8.7 Changement de base dans les application linéaires

On peut maintenant exprimer une même application linéaire dans différentes bases. Soit $l : \mathbf{E} \rightarrow \mathbf{F}$, avec les bases e, e', f et f' . On peut exprimer l de e' dans f' par

- $l^{(e',f')} = Id^{(f,f')} \circ l^{(e,f)} \circ Id^{(e',e)}$
- $L^{(e',f')} = P_{f \rightarrow f'} L^{(e,f)} P_{e' \rightarrow e}$

Exemple 8.10 projection sur une droite dans \mathbb{R}^2 .

Prolongeons l'exemple 8.3 et 8.6. Soit e la base canonique de \mathbb{R}^2 et $e' = \left\{ e'_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, e'_2 = \begin{bmatrix} -1 \\ 2 \end{bmatrix} \right\}$. On a

$$P_{(e' \rightarrow e)} = \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix} \quad \text{et} \quad P_{(e \rightarrow e')} = \frac{1}{5} \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix}$$

l est la projection sur la droite vectorielle $\mathcal{D} = \mathcal{L} \left\{ d = \frac{1}{\sqrt{5}} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\}$. Soit $v = l(u)$. Écrivons l dans différentes bases, en l'illustrant avec

$$u = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \quad \text{donc} \quad u^{(e)} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}^{(e)} \quad \text{et} \quad u^{(e')} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}^{(e')}$$

- De e dans e : on sait que $v^{(e)} = l^{(e,e)}(u^{(e)}) = L^{(e,e)}u^{(e)}$, avec

$$L^{(e,e)} = \frac{1}{5} \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix}.$$

$$\text{Par exemple } l^{(e,e)} \left(\begin{bmatrix} 1 \\ 3 \end{bmatrix}^{(e)} \right) = \frac{1}{5} \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \end{bmatrix}^{(e)} = \frac{1}{5} \begin{bmatrix} 10 \\ 5 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}^{(e)}.$$

- De e' dans e' : $v^{(e')} = l^{(e',e')}(u^{(e')}) = L^{(e',e')}u^{(e')}$ avec

$$L^{(e',e')} = P_{e \rightarrow e'} L^{(e,e)} P_{e' \rightarrow e} = \frac{1}{25} \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}.$$

$$\text{Par exemple } l^{(e',e')} \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}^{(e')} \right) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}^{(e')} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^{(e')}.$$

- De e dans e' : $v^{(e')} = l^{(e,e')}(u^{(e)}) = L^{(e,e')}u^{(e)}$ avec

$$L^{(e,e')} = P_{e \rightarrow e'} L^{(e,e)} = \frac{1}{25} \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix} = \frac{1}{5} \begin{bmatrix} 2 & 1 \\ 0 & 0 \end{bmatrix}$$

Qui s'écrit aussi

$$L^{(e,e')} = L^{(e',e')} P_{e \rightarrow e'} = \frac{1}{5} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix}.$$

$$\text{Par exemple } l^{(e,e')} \left(\begin{bmatrix} 1 \\ 3 \end{bmatrix}^{(e)} \right) = \frac{1}{5} \begin{bmatrix} 2 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \end{bmatrix}^{(e)} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^{(e')}.$$

- De e' dans e : $v^{(e)} = l^{(e',e)}(u^{(e')}) = L^{(e',e)}u^{(e')}$ avec

$$L^{(e',e)} = P_{e' \rightarrow e} L^{(e',e')} = \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 1 & 0 \end{bmatrix}.$$

$$\text{Par exemple } l^{(e',e)} \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}^{(e')} \right) = \begin{bmatrix} 2 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}^{(e')} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}^{(e)}.$$

Chapitre 9

Éléments propres

Notes personnelles.

9.1 Définitions

Vecteurs propres et valeurs propres.

Soit A une matrice réelle carrée de taille $n \times n$.

Un vecteur propre \mathbf{v} associé à une valeur propre λ (scalaire) est un vecteur non nul tel que $A\mathbf{v} = \lambda\mathbf{v}$.

Cette équation se réécrit $(A - \lambda I)\mathbf{v} = \mathbf{0}$.

Par définition ce système admet une solution $\mathbf{v} \neq \mathbf{0}$.

Comme $\mathbf{0}$ est aussi solution, ce système admet plusieurs solutions.

Donc $(A - \lambda I)$ n'est pas inversible.

Donc $\det(A - \lambda I) = 0$.

Polynôme caractéristique.

$P_A(\lambda) = \det(A - \lambda I)$ est un polynôme en λ de degré n , appelé polynôme caractéristique de A .

Les valeurs propres de A sont les racines du polynôme caractéristique.

Attention : même si A est réelle, il se peut que certaines valeurs propres soient complexes.

Exemple 9.1

Prenons $A = \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix}$, on a

$$\det(A - \lambda I) = \begin{vmatrix} 3 - \lambda & -2 \\ 1 & -\lambda \end{vmatrix} = (3 - \lambda)(-\lambda) + 2 = \lambda^2 - 3\lambda + 2 = (\lambda - 1)(\lambda - 2)$$

Donc A a deux valeurs propres $\lambda = 2$ et $\lambda = 1$.

Remarque.

Cette méthode de calcul des valeurs propres (calculer le polynôme caractéristique et en trouver les racines) ne fonctionne que pour les petites matrices, car il n'existe pas de formule pour trouver les racines d'un polynôme de degré n supérieur ou égal à 5.

9.2 Calcul de l'espace propre associé à une valeur propre réelle connue

Espace propre.

Soit λ une valeur propre de A . L'espace propre associé à λ est l'ensemble E_λ de tous les vecteurs propres associés à cette valeur propre :

$$E_\lambda = \{\mathbf{v} \in \mathbb{R}^n / A\mathbf{v} = \lambda\mathbf{v}\}$$

C'est un sous-espace vectoriel de \mathbb{R}^n .

Soit λ une valeur propre réelle de A . Posons $B_\lambda = A - \lambda I$. L'espace propre E_λ est donc l'ensemble des solutions du système linéaire $B_\lambda \mathbf{v} = \mathbf{0}$. Nous allons calculer E_λ en deux étapes.

La **première étape** consiste à mettre B_λ sous forme échelonnée \tilde{B}_λ , c'est à dire de la forme

$$\tilde{B}_\lambda = \begin{bmatrix} \mathbf{x} & \times & \times & \times & \times & \times \\ 0 & 0 & \mathbf{x} & \times & \times & \times \\ \vdots & & 0 & \mathbf{x} & \times & \times \\ \vdots & & & 0 & 0 & 0 \\ \vdots & & & & & \vdots \\ 0 & \dots & \dots & \dots & \dots & 0 \end{bmatrix} \quad \text{telle que} \quad \tilde{B}_\lambda \mathbf{v} = \mathbf{0} \Leftrightarrow B_\lambda \mathbf{v} = \mathbf{0}$$

Le premier élément de chaque échelon (en gras) est appelé pivot.

Cette transformation se fait grâce à des opérations sur les lignes, par un algorithme similaire à celui de mise sous forme triangulaire des systèmes inversibles.

Définition. Une matrice est dite échelonnée (en ligne) si le nombre de zéros précédant la première valeur non nulle (appelée pivot) augmente ligne par ligne jusqu'à ce qu'il ne reste plus que des 0 sur la ligne. Ainsi si le pivot de la ligne i est en position (i, j) , le pivot de la ligne suivante sera en position $(i + 1, k)$ avec $k \geq j + 1$.

Exemple 9.2

Reprenons $A = \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix}$ avec $\lambda = 1$, donc $B_1 = \begin{bmatrix} 2 & -2 \\ 1 & -1 \end{bmatrix}$. Alors

$$B_1 \mathbf{v} = \mathbf{0} \Leftrightarrow \begin{bmatrix} 2 & -2 \\ 1 & -1 \end{bmatrix} \mathbf{v} = \mathbf{0} \Leftrightarrow \begin{bmatrix} 2 & -2 \\ 0 & 0 \end{bmatrix} \mathbf{v} = \mathbf{0} \quad \text{par } L_2 \leftarrow L_2 - \frac{1}{2}L_1$$

Donc $\tilde{B}_1 = \begin{bmatrix} 2 & -2 \\ 0 & 0 \end{bmatrix}$.

Exemple 9.3

$A = \begin{bmatrix} 4 & -1 & 6 \\ 2 & 1 & 6 \\ 2 & -1 & 8 \end{bmatrix}$ admet $\lambda_1 = 9$ et $\lambda_2 = 2$ comme valeurs propres.

$$B_9 = \begin{bmatrix} -5 & -1 & 6 \\ 2 & -8 & 6 \\ 2 & -1 & -1 \end{bmatrix} \quad \text{devient} \quad \begin{bmatrix} -5 & -1 & 6 \\ 0 & -42 & 42 \\ 0 & -7 & 7 \end{bmatrix} \quad \text{devient} \quad \begin{bmatrix} -5 & -1 & 6 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} = \tilde{B}_9$$

$$B_2 = \begin{bmatrix} 2 & -1 & 6 \\ 2 & -1 & 6 \\ 2 & -1 & 6 \end{bmatrix} \quad \text{devient} \quad \begin{bmatrix} 2 & -1 & 6 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \tilde{B}_2$$

La **deuxième étape** consiste à exprimer l'ensemble de des solutions de $\tilde{B}_\lambda \mathbf{v} = \mathbf{0}$ sous forme d'une combinaison linéaire, c'est-à-dire à trouver une base de E_λ .

On observe que $\tilde{B}_\lambda \mathbf{v} = \mathbf{0}$ possède un nombre $m_\lambda < n$ d'équations linéaires indépendantes (m_λ est le nombre de lignes non nulles de \tilde{B}_λ). On choisit alors m_λ variables dites liées (une par équation, par exemple correspondant au pivot) et $(n - m_\lambda)$ variables dites libres (les autres). On exprime alors les variables liées en fonction des variables libres (par un algorithme similaire à l'algorithme de remontée dans la résolution de systèmes), puis on exprime les solutions de $\tilde{B}_\lambda \mathbf{v} = \mathbf{0}$ comme une combinaison linéaire dont les coefficients sont les variables libres.

Exemple 9.4

Reprenons l'exemple 9.3.

Pour $\lambda_1 = 9$ on a $m_9 = 2$ équations indépendantes. On choisit les variables liées v_1 et v_2 , et la variable libre v_3 . Alors

$$\tilde{B}_9 \mathbf{v} = \mathbf{0} \Leftrightarrow \begin{cases} -5v_1 - v_2 + 6v_3 = 0 \\ -v_2 + v_3 = 0 \end{cases} \Leftrightarrow \begin{cases} -5v_1 = v_2 - 6v_3 \\ v_2 = v_3 \end{cases} \Leftrightarrow \begin{cases} v_1 = v_3 \\ v_2 = v_3 \end{cases}$$

Donc toute solution s'écrit $\mathbf{v} = v_3 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ et $E_9 = \mathcal{L} \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}$.

On vérifie par exemple pour $v_3 = 2$ que $A \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 18 \\ 18 \\ 18 \end{bmatrix}$.

Les vecteurs de E_9 subissent une mise à l'échelle d'un facteur 9.

Pour $\lambda_2 = 2$ on a $m_2 = 1$ équations indépendantes. On choisit la variable liée v_1 , et les variables libres v_2 et v_3 . Alors

$$\tilde{B}_2 \mathbf{v} = \mathbf{0} \Leftrightarrow 2v_1 - v_2 + 6v_3 = 0 \Leftrightarrow v_1 = \frac{1}{2}v_2 - 3v_3$$

Donc toute solution s'écrit $\mathbf{v} = v_2 \begin{bmatrix} 1/2 \\ 1 \\ 0 \end{bmatrix} + v_3 \begin{bmatrix} -3 \\ 0 \\ 1 \end{bmatrix}$ et $E_2 = \mathcal{L} \left\{ \begin{bmatrix} 1/2 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -3 \\ 0 \\ 1 \end{bmatrix} \right\}$.

On vérifie par exemple pour $v_2 = 2$ et $v_3 = 1$ que $A \begin{bmatrix} -2 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} -4 \\ 4 \\ 2 \end{bmatrix}$.

Les vecteurs de E_2 subissent une mise à l'échelle d'un facteur 2.

On remarque que $\dim(E_9) = 1 = n - m_9$ et $\dim(E_2) = 2 = n - m_2$. De manière générale $\dim(E_\lambda) = n - m_\lambda$.

Remarque.

Cette méthode de calcul des espaces propres fonctionne pour les calculs à la main, mais elle n'est pas fiable numériquement. Les erreurs d'arrondi peuvent mener à une forme échelonnée erronée.

9.3 Diagonalisation

Comme nous le verrons dans l'exemple 9.6, il est parfois utile de savoir calculer la puissance d'une matrice. Dans le cas où cette matrice est diagonale, ce calcul est simple :

$$D = \begin{bmatrix} 5 & 0 \\ 0 & 3 \end{bmatrix} \Rightarrow D^2 = \begin{bmatrix} 5^2 & 0 \\ 0 & 3^2 \end{bmatrix} \Rightarrow D^3 = \begin{bmatrix} 5^3 & 0 \\ 0 & 3^3 \end{bmatrix} \Rightarrow D^k = \begin{bmatrix} 5^k & 0 \\ 0 & 3^k \end{bmatrix}$$

Prenons maintenant $A = \begin{bmatrix} 7 & 2 \\ -4 & 1 \end{bmatrix}$ qui s'écrit $A = PDP^{-1}$ avec $P = \begin{bmatrix} 1 & 1 \\ -1 & -2 \end{bmatrix}$ et $D = \begin{bmatrix} 5 & 0 \\ 0 & 3 \end{bmatrix}$. Alors

$$A^2 = (PDP^{-1})(PDP^{-1}) = PD(P^{-1}P)DP^{-1} = PD^2P^{-1}$$

$$A^3 = AA^2 = (PDP^{-1})(PD^2P^{-1}) = PD^3P^{-1}$$

En fait pour tout $k \geq 1$,

$$A^k = PD^kP^{-1} = \begin{bmatrix} 1 & 1 \\ -1 & -2 \end{bmatrix} \begin{bmatrix} 5^k & 0 \\ 0 & 3^k \end{bmatrix} \begin{bmatrix} 2 & 1 \\ -1 & -1 \end{bmatrix}$$

Donc si on sait écrire A sous la forme PDP^{-1} avec D une diagonale et P inversible, alors on peut facilement calculer ses puissances.

Matrices semblables. On dit que deux matrices A et B sont semblables si et seulement si il existe une matrice P inversible telle que $B = PAP^{-1}$.

Diagonalisation. Une matrice est dite diagonalisable si elle est semblable à une matrice diagonale, ce qui est vrai si et seulement si elle a n vecteurs propres linéairement indépendants.

Dans l'écriture $A = PDP^{-1}$, les éléments diagonaux de D sont les valeurs propres de A et les colonnes de P sont les vecteurs propres correspondants.

Remarques.

- Toutes les matrices ne sont pas diagonalisables.

- Une matrice qui a n valeurs propres réelles distinctes est toujours diagonalisable.

Exemple 9.5

Reprenons les exemples 9.3 et 9.4 : $A = \begin{bmatrix} 4 & -1 & 6 \\ 2 & 1 & 6 \\ 2 & -1 & 8 \end{bmatrix}$ et $P = \begin{bmatrix} 1 & 1 & -3 \\ 1 & 2 & 0 \\ 1 & 0 & 1 \end{bmatrix}$

On a

$$A = P \begin{bmatrix} 9 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} P^{-1}$$

donc

$$A^k = P \begin{bmatrix} 9^k & 0 & 0 \\ 0 & 2^k & 0 \\ 0 & 0 & 2^k \end{bmatrix} P^{-1}$$

Exemple 9.6 Système proie/prédateur.

Dans les forêts de séquoias californiennes, les chouettes se nourrissent essentiellement de rats. Notons C_k le nombre de chouettes, et R_k le nombre milliers de rats au k -ième mois. On suppose que

$$\begin{cases} C_k &= 0,5 C_{k-1} + 0,4 R_{k-1} \\ R_k &= -0,104 C_{k-1} + 1,1 R_{k-1} \end{cases}$$

Ce système signifie que :

- En l'absence de rats, seules la moitié des chouettes survivent chaque mois.
- Si les rats sont abondants, ils permettent aux chouettes de vivre et de se reproduire à raison de $1000/0,4 = 2500$ rats pour 1 chouette.
- En l'absence de chouettes, les rats prolifèrent de 10% chaque mois.
- Si les chouettes sont là, elles mangent 104 rats chacune par mois.

En posant $\mathbf{u}_k = \begin{bmatrix} C_k \\ R_k \end{bmatrix}$ et $A = \begin{bmatrix} 0,5 & 0,4 \\ -0,104 & 1,1 \end{bmatrix}$, ce système s'écrit

$$\mathbf{u}_k = A\mathbf{u}_{k-1} = A^2\mathbf{u}_{k-2} = \dots = A^k\mathbf{u}_0$$

Or la matrice A admet les éléments propres $\lambda_1 = 1,02$ associé à $\mathbf{v}_1 = \begin{bmatrix} 10 \\ 13 \end{bmatrix}$ et $\lambda_2 = 0,58$ associé à $\mathbf{v}_2 = \begin{bmatrix} 5 \\ 1 \end{bmatrix}$. Donc on peut calculer la population au mois k en fonction du mois 0 :

$$\mathbf{u}_k = P \begin{bmatrix} 1,02^k & 0 \\ 0 & 0,58^k \end{bmatrix} P^{-1}\mathbf{u}_0$$

On cherche maintenant à savoir comment se comporte "globalement" les populations à long terme. Pour cela, écrivons $\mathbf{u}_0 = u_1\mathbf{v}_1 + u_2\mathbf{v}_2$ dans la base de vecteurs propres. On a alors

$$\begin{aligned} \mathbf{u}_k &= A^k(u_1\mathbf{v}_1 + u_2\mathbf{v}_2) \\ &= u_1A^k\mathbf{v}_1 + u_2A^k\mathbf{v}_2 \\ &= u_1(1,02)^k\mathbf{v}_1 + u_2(0,58)^k\mathbf{v}_2 \end{aligned}$$

Le temps passant, k devient grand, $(0,58)^k$ se rapproche rapidement de 0, et

$$\mathbf{u}_k \approx u_1(1,02)^k \begin{bmatrix} 10 \\ 13 \end{bmatrix} \approx 1,02 \mathbf{u}_{k-1}.$$

Donc les populations tendent à croître au rythme de 2% par mois, avec un ratio de 10 chouettes pour 13 000 rats.

Écriture dans une base de vecteurs propres et lien avec les applications linéaires.

Supposons que la matrice A , représente une application linéaire l dans la base canonique e de \mathbb{R}^n :

$$\begin{aligned} l^{(e,e)} : \mathbb{R}^n &\longrightarrow \mathbb{R}^n \\ \mathbf{u} &\longmapsto A\mathbf{u} \end{aligned}$$

Si A est diagonalisable, elle admet une base $v = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ de vecteurs propres associés à des valeurs propres $\{\lambda_1, \dots, \lambda_n\}$. Soit $P = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n]$ la matrice dont les colonnes sont les vecteurs de v exprimés dans la base canonique.

Donc P est la matrice de passage de la base v dans la base e : $P = P_{v \rightarrow e}$. Comme $A = PDP^{-1}$, la matrice $D = P^{-1}AP = P_{e \rightarrow v}AP_{v \rightarrow e}$ représente l dans la base v :

$$l^{(v,v)}(\mathbf{v}^{(v)}) = D\mathbf{v}^{(v)} \quad \text{avec} \quad D = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$$

Cette expression montre que dans la base de vecteurs propres, l'application linéaire multiplie simplement la coordonnée sur chaque vecteur propre \mathbf{v}_i par la valeur propre λ_i .

Exemple 9.7 projection sur une droite dans \mathbb{R}^2 .

Nous avons vu dans le chapitre précédent (exemples 8.3, 8.6, 8.10) que la projection sur la droite vectorielle $\mathcal{D} = \mathcal{L}\left\{\begin{bmatrix} 2 \\ 1 \end{bmatrix}\right\}$ est représentée :

- Dans la base canonique par la matrice $A = \frac{1}{5} \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix}$.
- Dans la base $\left\{\begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 2 \end{bmatrix}\right\}$ par la matrice $D = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$. Il s'agit de l'écriture dans une base de vecteurs propres : la composante parallèle à la droite est inchangée (valeur propre 1) alors que la composante perpendiculaire est annulée (valeur propre 0).

9.4 Calcul approché des valeurs propres et vecteurs propres

Nous avons signalé à la section 9.1 que le calcul des valeurs propres par le polynôme caractéristique est impossible pour les grandes matrices. Nous allons voir ici comment approximer itérativement la plus grande valeur propre ainsi qu'un vecteur propre associé, à l'aide de la "méthode de la puissance".

Soit A une matrice $n \times n$ diagonalisable avec une valeur propre strictement dominante (plus grande que toutes les autres en valeur absolue). On range les valeurs propres $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|$ avec une base de vecteurs propres $v = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$. Soit \mathbf{u} un vecteur qui s'écrit $\mathbf{u} = u_1\mathbf{v}_1 + \dots + u_n\mathbf{v}_n$ dans la base v . Comme dans l'exemple du système proie-prédateur, on peut écrire :

$$A^k \mathbf{u} = u_1(\lambda_1)^k \mathbf{v}_1 + u_2(\lambda_2)^k \mathbf{v}_2 + \dots + u_n(\lambda_n)^k \mathbf{v}_n$$

Donc en divisant par λ_1^k on a

$$\frac{1}{\lambda_1^k} A^k \mathbf{u} = u_1 \mathbf{v}_1 + u_2 \left(\frac{\lambda_2}{\lambda_1}\right)^k \mathbf{v}_2 + \dots + u_n \left(\frac{\lambda_n}{\lambda_1}\right)^k \mathbf{v}_n$$

Comme les fractions $\frac{\lambda_2}{\lambda_1}, \dots, \frac{\lambda_n}{\lambda_1}$ sont strictement inférieures à 1 en module, leurs puissances tendent vers 0, et donc

$$\mathbf{u}_k = \frac{1}{\lambda_1^k} A^k \mathbf{u} = \frac{1}{\lambda_1} A \mathbf{u}_{k-1} \rightarrow u_1 \mathbf{v}_1 \text{ quand } k \rightarrow \infty$$

Donc \mathbf{u}_k se rapproche de $u_1 \mathbf{v}_1$ qui est lui même un vecteur propre associé à λ_1 , à condition que $u_1 \neq 0$. Le problème pour faire ce calcul est qu'il faudrait déjà connaître λ_1 . Au lieu de normaliser par λ_1 , nous allons normaliser par μ_k la plus grande composante (en valeur absolue) de $A\mathbf{u}_{k-1}$:

$$\mathbf{u}_k = \frac{1}{\mu_k} A\mathbf{u}_{k-1}$$

Il se trouve que cette suite converge vers un multiple de \mathbf{v}_1 , dont la plus grande composante est égale à 1, et donc μ_k est proche de λ_1 . L'algorithme ALGO. (9.1) calcule cette suite. Il permet donc d'approximer en même temps λ_1 (par la valeur de μ) et un vecteur propre (par la valeur de \mathbf{u}).

ALGO. 9.1 *approximation_vp(A)*.

choisir \mathbf{u} dont la plus grande composante est 1

tant que l'approximation n'est pas assez bonne **faire**

$\mathbf{u} \leftarrow A\mathbf{u}$

$\mu \leftarrow$ la composante de \mathbf{u} de plus grande valeur absolue

$\mathbf{u} \leftarrow \mathbf{u}/\mu$

fin tant que

renvoyer μ et \mathbf{u}

Remarques.

- Condition d'arrêt de la boucle : on choisit d'arrêter l'itération quand μ et \mathbf{u} n'évoluent presque plus.
- Cette méthode peut être adaptée pour calculer la plus petite valeur propre, ou bien pour affiner d'autres valeurs propres dont on connaît une approximation grossière.
- Il existe des méthodes plus complexes pour approximer toutes les valeurs propres à la fois.

Annexes

Annexe A

Comment éviter les erreurs ?

Qu'il s'agisse de faire un calcul, écrire un algorithme ou un programme, la solution est souvent difficile à trouver mais facile à vérifier. Votre objectif doit être avant tout de ne pas faire d'erreurs. Donc prenez le temps de vérifier vos résultats ! Voici quelques conseils, faciles et rapides à mettre en œuvre.

Calculs. Prenons l'exemple la résolution d'un système dans la section 6.2. Il y a des risques de se tromper dans le calcul, mais vérifier que $\begin{bmatrix} 1 & -1 & 2 \\ 3 & 2 & 1 \\ 2 & -3 & -2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$ est bien égal au membre de droite, c'est rapide et presque sans risque. Il en va de même pour les changements de base (exemple 8.9), les calculs d'éléments propres (exemple 9.3), etc. Un conseil pour que ça fonctionne : ne calculez pas "de tête" en vous disant "ça a l'air bon", mais faites *vraiment* le calcul au brouillon, *sans regarder le résultat*.

Algorithmes. Plusieurs alternatives sont possibles :

- Déroulez un exemple à la main. Un conseil pour que ce soit efficace : déroulez les instructions avec minutie, et *écrivez* les résultats au fur et à mesure.
- Observez des indices comme :
 - Est-ce que la complexité est celle attendue ?
 - Est-ce que toutes les données sont parcourues ?
 - Est-ce que les dimensions de matrices et vecteurs sont cohérentes ?
- Programmez l'algorithme et testez le code.

Programmes. Faites des tests, c'est à dire exécutez votre programme avec plusieurs entrées et vérifiez les sorties. Pour qu'un test soit fiable il ne suffit pas d'observer quelques sorties en se disant "ça a l'air bon" :

- Prévoyez des entrées variées qui couvrent un maximum de situations.
- Écrivez les résultats attendus.
- Seulement *ensuite*, exécuter le programme.

Annexe B

Questions de cours

Pour vous entraîner et auto-évaluer votre maîtrise du cours, voici des questions de cours classiques, organisées par chapitres.

B.1 Rappels d'algèbre

- Qu'est-ce qu'une base ?
- Qu'est-ce que la dimension d'un espace vectoriel ?
- Pourquoi peut-on généralement travailler dans \mathbb{R}^n ou \mathbb{C}^n ?

B.2 Rappels sur la complexité

- Que mesure la complexité asymptotique ?
- Que (ne) peut-on (pas) faire avec ?
- Quelles complexités sont faibles ? fortes ? ingérables ?

B.3 Les matrices

- La transposée est-elle distributive sur le produit matriciel ?
- L'inversion de matrice est-elle distributive sur le produit matriciel ?
- Quelle est la complexité du produit matrice-matrice ?
- Selon quels critères choisir une structure de données pour les matrices ?

B.4 Résolution de systèmes linéaires

- Quelles sont les conditions pour pouvoir appliquer la méthode du pivot de Gauss ?
- Quelles sont les principales étapes de cette méthode ?
- Quelle est la complexité de cette méthode ?
- Qu'est-ce que la méthode du pivot partiel ? Quel est son intérêt ?

B.5 Approximation au sens des moindres carrés

- D'où vient le terme *moindres carrés* ?
- Que sont les équations normales ? Comment les calculer ?

- Comment connaître la précision d'une régression ?
- Comment gérer des données peu fiables ?

B.6 Applications linéaires

- À quoi sert une matrice de passage ?
- Soient e et e' , deux bases de \mathbb{R}^n et soit $P_{e \rightarrow e'}$, la matrice de passage de e vers e' . Que contient la i^{eme} colonne de $P_{e \rightarrow e'}$?

B.7 Éléments propres

- Qu'est-ce que le polynôme caractéristique d'une matrice ?
- Qu'est-ce qu'un espace propre ? Comment le calculer ?
- Quelles matrices sont diagonalisables ?
- Quel est le lien entre le noyau d'une application linéaire et les éléments propres ?
- Comment calculer les puissances d'une matrice ?

Annexe C

Annales

Pour vous entraîner et préparer les évaluations, voici des exercices posés les années précédentes, organisés par chapitres. Les durées indicatives contiennent le temps de rédaction ou de recopie au propre.

C.1 Les matrices

Exercice C.1 Matrices circulantes (durée 50 minutes).

Une matrice A carrée est circulante si elle a la forme suivante :

$$A = \begin{bmatrix} a_1 & a_2 & a_3 & \dots & a_n \\ a_n & a_1 & a_2 & \dots & a_{n-1} \\ a_{n-1} & a_n & a_1 & \dots & a_{n-2} \\ \vdots & & & & \vdots \\ a_2 & a_3 & \dots & a_n & a_1 \end{bmatrix}. \quad \text{On note } \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \text{ le vecteur associé à } A.$$

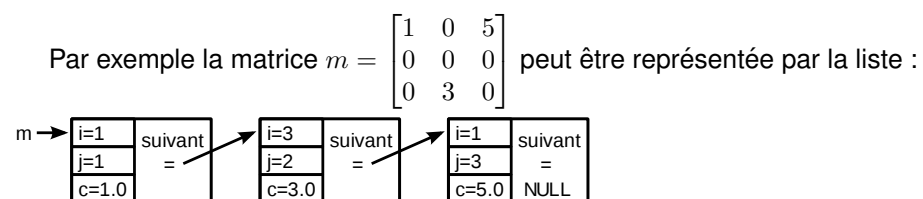
Le vecteur \mathbf{a} suffit à reconstruire la matrice A , ce qui permet un stockage compact : au lieu de stocker A entièrement, on stocke uniquement \mathbf{a} . Nous allons faire différents calculs matriciels avec \mathbf{a} au lieu de A .

1. Calculez le produit de A par le vecteur ne contenant que des 1. Qu'en déduisez-vous ?
2. La matrice $B = A^T$ est circulante. Que vaut le vecteur \mathbf{b} associé à B ?
Écrivez un algorithme qui calcule \mathbf{b} en fonction de \mathbf{a} .
3. Écrivez un algorithme qui construit la matrice A à partir du vecteur \mathbf{a} .
4. Écrivez un algorithme qui calcule le vecteur $\mathbf{v} = A\mathbf{u}$, en fonction de \mathbf{a} et \mathbf{u} (sans construire A).

Exercice C.2 Matrices creuses (durée 20 minutes).

On représente des matrices creuses (*i.e.* contenant beaucoup de 0) par une liste chaînée des coefficients non nuls avec leurs indices $1 \leq i \leq N$ et $1 \leq j \leq N$:

```
struct elem {int i; int j; float c; struct elem * suivant;};  
typedef struct elem * matrice;
```



Les matrices sont carrées, et leur taille est définie par une constante N globale. Les vecteurs sont représentés par des tableaux de flottants.

Codez une fonction de profil `float* produit (matrice m, float* v)` qui alloue, calcule et renvoie le résultat du produit mv .

Exercice C.3 Algorithme d'orthonormalisation de Gram-Schmidt (durée 1h).

À partir d'une base quelconque $a = \{a_1, a_2, \dots, a_n\}$ de \mathbb{R}^n , l'algorithme de Gram-Schmidt calcule une base orthonormale $e = \{e_1, e_2, \dots, e_n\}$ définie ainsi :

$$\begin{aligned} e_1 &= \frac{b_1}{\|b_1\|} && \text{avec } b_1 = a_1 \\ e_2 &= \frac{b_2}{\|b_2\|} && \text{avec } b_2 = a_2 - p(e_1, a_2) \\ e_3 &= \frac{b_3}{\|b_3\|} && \text{avec } b_3 = a_3 - p(e_1, a_3) - p(e_2, a_3) \\ &\vdots && \\ e_k &= \frac{b_k}{\|b_k\|} && \text{avec } b_k = a_k - \sum_{j=1}^{k-1} p(e_j, a_k) \\ &\vdots && \end{aligned}$$

où p représente la projection $p(u, v) = \langle u; v \rangle u$.

Pour les algorithmes, les bases d'entrée et de sortie seront stockées comme colonnes de matrices

$$A = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} \text{ et } E = \begin{bmatrix} e_1 & e_2 & \cdots & e_n \end{bmatrix}$$

La taille n est supposée fixée globalement.

1. Dérouler à la main l'algorithme pour la base de \mathbb{R}^3 suivante :

$$a_1 = \begin{bmatrix} 3 \\ 0 \\ 4 \end{bmatrix}, \quad a_2 = \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix}, \quad a_3 = \begin{bmatrix} 0 \\ 0 \\ 5 \end{bmatrix}$$

2. Écrire l'algorithme de Gram-Schmidt.

Indication : on pourra définir des algorithmes intermédiaires

`ProduitScalaire(A, j, B, k)` qui renvoie le produit scalaire des colonnes $A(:, j)$ et $B(:, k)$;

`Normalise(A, k)` qui normalise la colonne $A(:, k)$.

3. Quelle est la complexité de cet algorithme ?
4. Une matrice dont les colonnes sont orthonormales (comme E) est appelée *matrice orthogonale*.
Que vaut $E^T E$? Que peut valoir le déterminant de E ? Justifiez vos réponses.

Exercice C.4 Normalisation d'une famille (durée 20 minutes).

Soit $a = \{a_1, a_2, \dots, a_p\}$ une famille quelconque de \mathbb{R}^n . On souhaite normaliser tous les vecteurs, c'est à dire construire une famille $e = \{e_1, e_2, \dots, e_p\}$, telle que chaque e_i est de norme 1, colinéaire à a_i et de même sens. Les familles seront stockées comme colonnes de matrices

$$A = \begin{bmatrix} a_1 & a_2 & \cdots & a_p \end{bmatrix} \text{ et } E = \begin{bmatrix} e_1 & e_2 & \cdots & e_p \end{bmatrix}$$

1. Quelles sont les dimensions de A et E ?
2. Écrire l'algorithme qui calcule E à partir de A .
3. Combien de multiplications / divisions sont faites ?
4. Combien d'additions / soustractions sont faites ?
5. Que valent les éléments diagonaux de $E^T E$?

C.2 Résolution de systèmes linéaires

Exercice C.5 Systèmes linéaires (durée 20 minutes).

Calculer $\det(A)$, puis résoudre $Ax = b$ pour

$$A = \begin{bmatrix} 0 & 3 & 3 \\ -2 & 2 & 1 \\ 3 & -1 & -1 \end{bmatrix} \quad \text{et} \quad b = \begin{bmatrix} 9 \\ 9 \\ -9 \end{bmatrix}$$

Exercice C.6 Systèmes linéaires (durée 20 minutes).

Calculer $\det(A)$, puis résoudre $Ax = b$ pour

$$A = \begin{bmatrix} 0 & 2 & -1 \\ 2 & 1 & 3 \\ -4 & 2 & -6 \end{bmatrix} \quad \text{et} \quad b = \begin{bmatrix} -7 \\ 13 \\ -34 \end{bmatrix}$$

Exercice C.7 Factorisation LU (durée 25 minutes).

Supposons qu'on sache factoriser une matrice carrée A de taille $n \times n$ sous la forme $A = LU$, où L (respectivement U) est une matrice triangulaire inférieure (respectivement supérieure). Cette factorisation est calculée en temps $\Theta(n^3)$.

1. Proposez une méthode de résolution du système $AX = B$ s'appuyant sur cette décomposition.
2. Quelle est sa complexité ?
3. On souhaite résoudre le système pour plusieurs membres de droite B . Quel est l'avantage de cette méthode par rapport au pivot de Gauss ?

Exercice C.8 Pivot de Gauss (durée 45 minutes).

On veut ré-écrire la méthode du pivot de Gauss partiel en utilisant une matrice triangulaire inférieure au lieu de supérieure.

1. Écrivez un algorithme qui transforme un système quelconque en système triangulaire inférieur équivalent.
2. Écrivez un algorithme "de descente" qui résout un système triangulaire inférieur.

Exercice C.9 Résolution d'un système linéaire (durée 20 minutes).

On cherche à résoudre le système suivant :

$$\begin{pmatrix} 3 & 2 & 0 \\ 0 & 1 & 0 \\ 4 & 1 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}, \quad (\text{C.1})$$

c'est-à-dire à trouver le vecteur $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$.

1. (**Méthode du pivot de Gauss**) Transformer ce système en un système triangulaire supérieur équivalent, à l'aide de la méthode du pivot de Gauss. (On détaillera chaque étape).

2. **(Résolution)** Résoudre ce système triangulaire supérieur au moyen d'un algorithme vu en classe. Vérifier que la solution trouvée satisfait bien le système de départ.
3. **(Complexité algorithmique)** Comment s'appelle cet algorithme ? Quelle est sa complexité lorsque le système initial est de taille n (on ne demande pas de justifier le résultat) ?

Exercice C.10 Résolution d'un système triangulaire (durée 25 minutes).

1. **(Algorithme)** Écrire l'algorithme de descente pour résoudre une matrice triangulaire inférieure, de taille $n \times n$. Quelle est sa complexité (on justifiera le résultat) ?
2. **(Résolution pratique)** L'appliquer sur l'un des deux systèmes suivants (en justifiant votre choix de système) :

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 0 & 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 5 \\ 4 \\ 3 \\ 2 \\ 1 \end{pmatrix} \quad \text{ou} \quad \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 \\ 3 & 2 & 1 & 0 & 0 \\ 4 & 3 & 2 & 1 & 0 \\ 5 & 4 & 3 & 2 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 5 \\ 4 \\ 3 \\ 2 \\ 1 \end{pmatrix}.$$

C.3 Approximation au sens des moindres carrés

Exercice C.11 Droite des moindres carrés (durée 25 minutes).

Calculez la droite des moindres carrés pour les observations $(1, 3)$, $(3, 3)$, $(5, 2)$ et $(7, 0)$. Faites un dessin. Calculez les résidus.

Exercice C.12 Droite des moindres carrés (durée 20 minutes).

Calculer la droite des moindres carrés puis les résidus pour les observations suivantes :

$i =$	1	2	3	4	5
$x_i =$	-2	0	3	4	5
$y_i =$	0	2	4	4	3

Exercice C.13 Moindres carrés : algorithme général (durée 35 minutes).

1. Écrire un algorithme `resout` de résolution de système linéaire de taille 2×2 :
 - paramètres : A et b (où A est supposée inversible) ;
 - résultat : le vecteur x qui vérifie $Ax = b$.

Indication : on pourra utiliser la formule $A^{-1} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}^{-1} = \frac{1}{\det(A)} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}$.
2. Écrire un algorithme `regression` qui calcule la droite des moindres carrés :
 - paramètres : deux vecteurs x et y (les observations), un entier n (la taille des vecteurs) ;
 - résultat : le vecteur $\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$ des coefficients de la droite des moindres carrés $y = \alpha x + \beta$.
3. Quelle est la complexité de `regression` ?

Exercice C.14 Chauffage au gaz (durée 40 minutes).

Un distributeur de gaz observe la quantité de gaz que ses clients consomment pendant plusieurs périodes (TABLE C.1). Il constate que plus il fait froid, plus les clients consomment de gaz. Pour bien s'approvisionner, il voudrait prévoir la quantité de gaz g en fonction de la température t .

1. On essaie une prévision par une droite des moindres carrés $g = at + b$. Calculez cette droite. Faites un dessin. Quelle quantité de gaz prévoyez vous pour $t = 5$?
2. On veut une prévision quadratique plus précise $g = at^2 + bt + c$. Proposez une méthode pour calculer cette prévision (le calcul sur l'exemple numérique n'est pas demandé).

Période $i =$	1	2	3	4
Température moyenne $t_i =$	-5	0	10	15
Quantité de gaz consommé $g_i =$	30	12	8	7

TABLE C.1 – Mesures de la température moyenne et de la consommation de gaz.

$i =$	1	2	3	4	5	6
$x_i =$	-3	-2	0	1	2	3
$y_i =$	5	4	2	0	-1	-2
$w_i =$	1	1/2	1/2	1	1	1

TABLE C.2 – On travaille sur l'échantillon (x_i, y_i) , avec des pondérations w_i .**Exercice C.15 Moindres carrés pondérés (durée 40 minutes).**

On veut réaliser une approximation par une droite $y = \alpha x + \beta$, en utilisant des moindres carrés pondérés. Notons $M \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = B$ le système linéaire des équations normales.

1. Pour cet échantillon, calculer M et B (il n'est pas demandé de résoudre le système).
2. On s'intéresse maintenant à un échantillon quelconque de taille n . Exprimer chaque coefficient de M et de B directement en fonction des x_i , y_i , et w_i .
3. Écrire un algorithme de calcul de M et B .

Exercice C.16 Approximation polynômiale (durée 15 minutes).

On veut approximer les données

$i =$	1	2	3	4	5	6
$x_i =$	-3	-2	0	1	2	3
$y_i =$	5	4	2	0	-1	-2

par un polynôme $y = ax^2 + bx + c$, en utilisant les moindres carrés. Calculer, sur cet échantillon, les équations normales (il n'est pas demandé de les résoudre).

Exercice C.17 Moindres carrés (40 minutes)

$i =$	1	2	3	4	5
$x_i =$	-2	-1	0	2	3
$y_i =$	-4	-2	-2	0	2
$w_i =$	1/2	1	1	1/2	1

On cherche des approximations au sens des moindres carrés pour l'échantillon (x_i, y_i) ci-dessus, avec des pondérations w_i . Vous allez écrire plusieurs systèmes d'équations normales, il n'est pas demandé de résoudre les systèmes.

1. Calculer les équations normales pour une droite $y = \alpha x + \beta$, *sans pondération*.
2. Calculer les équations normales pour une droite $y = \alpha x + \beta$, *avec pondération*.
3. Calculer les équations normales pour une parabole $y = ax^2 + bx + c$, *sans pondération*.

Exercice C.18 Moindres carrés, approximation linéaire et polynômiale (durée 45 minutes).

On considère l'échantillon (x_i, y_i) suivant et on cherche à trouver la loi liant x à y .

$i =$	1	2	3	4	5
$x_i =$	1	$\exp(1)$	$\exp(2)$	$\exp(3)$	$\exp(4)$
$y_i =$	1	1	$\exp(0.5)$	$\exp(-2)$	$\exp(3)$

TABLE C.3 – Échantillon (x_i, y_i) étudié. La fonction \exp est la fonction exponentielle.

1. **(Changement de variables)** On définit de nouvelles variables $x' = \log(x)$ et $y' = \log(y)$. Exprimer les valeurs du nouvel échantillon (x'_i, y'_i) . On pourra représenter les résultats sous forme d'un tableau.
2. **(Équations normales)** On veut réaliser une approximation linéaire par une droite $y' = \alpha + \beta x'$, en utilisant les moindres carrés. Les équations normales associées s'écrivent $\mathcal{A} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \mathcal{B}$. Que valent \mathcal{A} et \mathcal{B} ?
3. **(Valeurs de α et β)** Résoudre ce système pour trouver α et β .
4. **(Conclusion)** En revenant aux variables initiales, exprimer la relation liant x et y .
5. **(Complexité algorithmique)** Si on avait n échantillons (x'_i, y'_i) , quelle serait la complexité du calcul de la matrice \mathcal{A} ?
6. **(Approximation polynômiale)** On veut maintenant réaliser une approximation polynômiale par une parabole $y' = \alpha + \beta x' + \gamma (x')^2$. Comment sont modifiées les équations normales ? (on ne demande pas de les résoudre).

Exercice C.19 Moindres carrés et diagonalisation (durée 60 minutes).

On considère un échantillon de 5 couples de valeurs (x_i, y_i) , avec, pour chaque couple (x_i, y_i) , le poids $w_i \in [0, 1]$ correspondant à l'indice de confiance.

On souhaite réaliser une approximation polynômiale par une parabole d'équation : $y = \alpha x^2 + \beta x + \gamma$, en utilisant les moindres carrés pondérés.

1. **(Équations normales)** Les équations normales associées s'écrivent $\mathcal{M} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \mathcal{C}$.
 - (a) Quelles sont les tailles de la matrice \mathcal{M} et du vecteur \mathcal{C} ?
 - (b) Que valent \mathcal{M} et \mathcal{C} ? Écrire leurs coefficients en fonction de x_i, y_i et w_i .
2. **(Implémentation numérique des matrices)**
 - (a) Écrire un algorithme pour définir les matrices \mathcal{M} et \mathcal{C} (sans utiliser le produit matriciel).
 - (b) Quelle est sa complexité algorithmique (on exprimera le nombre de multiplications et le nombre d'additions) ?
3. **(Inversion de la matrice \mathcal{M})** Soit $\mathcal{M} = PDP^{-1}$ la diagonalisation de \mathcal{M} , avec $D = \begin{pmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_5 \end{pmatrix}$.
 - (a) Comment s'appellent les λ_i ?
 - (b) Exprimer \mathcal{M}^{-1} en fonction de P, P^{-1} et des λ_i .
4. **(Résolution théorique)** Exprimer le vecteur $\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$, solution des équations normales, en fonction de P, P^{-1}, λ_i et \mathcal{C} .

Exercice C.20 Moindres carrés (durée 25 minutes).

On considère un échantillon de quatre valeurs $(x_i, y_i) = \{(3, 8), (-1, 3), (2, -5), (1, 1)\}$. Le but de cet exercice est de calculer la droite des moindres carrés pour ces quatre observations : $y = \alpha x + \beta$.

1. (Mise sous forme matricielle)

- (a) Mettre ce problème sous forme matricielle $AX = B$, avec une matrice rectangulaire A .
- (b) Quelle est la taille de A et de B ?
- (c) Que vaut le vecteur X ?

2. (Équations normales) Ecrire les équations normales associées à ce problème.**3. (Résolution)**

- (a) Résoudre les équations normales en utilisant le pivot de Gauss.
- (b) Quelle est alors l'équation de la droite des moindres carrés ?

Exercice C.21 Moindres carrés avec régularisation (durée 25 minutes).

Soit A une matrice à m lignes et n colonnes et b un vecteur de \mathbb{R}^m . Nous cherchons une solution \hat{x} au sens des moindres carrés à $Ax = b$ en imposant en plus une hypothèse de régularisation sur \hat{x} . Au lieu de minimiser $\|Ax - b\|$, cela se traduit par la minimisation de

$$\|Ax - b\| + \lambda \sum_{i=1}^n x_i^2,$$

avec $\lambda \in \mathbb{R}$ un scalaire fixé.

- 1. Écrire les équations normales associées au problème lorsque $\lambda = 0$. On mettra ces équations sous la forme d'un système linéaire $C\hat{x} = d$. Que valent la matrice C et le vecteur d ? Quelles sont les dimensions de C et de d ?
- 2. L'hypothèse de régularisation sur \hat{x} revient à rajouter λI à la matrice C où I représente la matrice identité. Les équations normales se réécrivent alors sous la forme $E\hat{x} = d$, que valent E et d ? Quelles sont leurs dimensions ?
- 3. Écrire un algorithme qui assemble la matrice E et le vecteur d (nous ne demandons pas de l'écrire en langage scilab).
- 4. En détaillant votre réponse, calculer la complexité de cet algorithme. Lorsque n et m deviennent très grands, quelle est la complexité asymptotique de cet algorithme ?

Exercice C.22 Approximation polynomiale à partir de l'écriture matricielle (35 minutes).

- 1. On s'intéresse à la résolution au sens des moindres carrés du système suivant :

$$\begin{pmatrix} 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad (\text{C.2})$$

- (a) Ecrire les équations normales associées à ce problème.
- (b) Appliquer la méthode du pivot de Gauss sur ces équations normales.
- (c) Appliquer ensuite l'algorithme de remontée.
- (d) Quel(s) couple(s) d'observation (x_i, y_i) et quelle droite de régression linéaire conduisent à la résolution du système (C.2) ?
- 2. On s'intéresse maintenant à la résolution au sens des moindres carrés du second système :

$$\begin{pmatrix} 1 & -1 \\ 4 & 2 \\ 4 & -2 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 0 \\ 7 \end{pmatrix} \quad (\text{C.3})$$

- Ecrire les équations normales associées à ce second système.
- Les résoudre.
- Quel(s) couple(s) d'observation (x_i, y_i) et quel polynôme d'approximation conduisent à la résolution du système (C.3) ?

Exercice C.23 Moindres carrés en échelle logarithmique (durée 45 minutes).

On cherche une approximation au sens des moindres carrés après changement de variables pour l'échantillon (x_i, y_i) ci-dessous (où la fonction exp correspond à l'exponentielle).

$i =$	1	2	3	4	5
$x_i =$	1	exp(1)	exp(2)	exp(3)	exp(4)
$y_i =$	1	exp(1)	1	exp(2)	1

On suppose que les variables sont liées selon une loi de puissance : $y = \alpha x^\beta$, avec α, β deux réels à déterminer.

- Effectuer un changement de variables logarithmique en x et en y , en notant x' et y' les nouvelles variables. Exprimer les valeurs du nouvel échantillon (x'_i, y'_i) sous forme d'un tableau.
- Que devient la loi de puissance $y = \alpha x^\beta$ dans les nouvelles variables (x', y') ?
- En utilisant les moindres carrés, écrire les équations normales associées aux variables (x', y') .
- Quelle est la taille de la matrice et des vecteurs des équations normales précédentes ?
- Résoudre les équations normales au moyen de la méthode de Gauss et de l'algorithme de remontée.
- Que valent α et β dans la loi de puissance initiale $y = \alpha x^\beta$?

C.4 Applications linéaires

Exercice C.24 Bases et applications linéaires (durée 45 minutes).

On se place dans l'espace E des polynômes de degré inférieur ou égal à 3. On définit les deux bases :

$$e = \{e_1(X) = 1, e_2(X) = X, e_3(X) = X^2, e_4(X) = X^3\}$$

$$e' = \{e'_1(X) = X, e'_2(X) = 1 - X, e'_3(X) = X^3, e'_4(X) = (1 - X)^3\}$$

On appelle d l'application qui dérive un polynôme :

$$d: E \longrightarrow E$$

$$p(X) \longmapsto p'(X)$$

- Montrer que d est une application linéaire de E dans E .
- Calculer la matrice D qui représente d dans la base e .
- Calculer $\text{Ker}(d)$, le noyau de d .
- Calculer $P_{e' \rightarrow e}$ la matrice de passage de e' dans e .
- Calculer la matrice D' qui représente d dans la base e' .

Exercice C.25 Changement de base (durée 20 minutes).

On se donne deux bases de \mathbb{R}^2 : e la base canonique, et $e' = \left\{ e'_1 = \begin{bmatrix} 1 \\ 3 \end{bmatrix}, e'_2 = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\}$.

Soit $l: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ l'application linéaire représentée de la base e' dans la base e' par la matrice

$$\begin{bmatrix} 2 & 0 \\ 0 & -1 \end{bmatrix}.$$

Calculez la matrice qui représente l de la base e dans la base e (détailler les calculs).

Exercice C.26 Changement de base (durée 40 minutes).

On se donne deux bases de \mathbb{R}^2 : e la base canonique, et $f = \left\{ \mathbf{f}_1 = \begin{bmatrix} -3 \\ 2 \end{bmatrix}, \mathbf{f}_2 = \begin{bmatrix} 2 \\ -1 \end{bmatrix} \right\}$.

Soit $l : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ l'application linéaire représentée par la matrice $L = \begin{bmatrix} 3 & 0 \\ 1 & 3 \end{bmatrix}$ de la base f dans la base f .

1. Calculez la matrice M qui représente l de la base e dans la base e (détaillez les calculs).
2. Soit $\mathbf{u} = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$. Calculez $\mathbf{u}^{(f)}$, les coordonnées de \mathbf{u} dans la base f .
3. Soit $\mathbf{v} = l(\mathbf{u})$. Calculez $\mathbf{v}^{(e)}$ et $\mathbf{v}^{(f)}$.

Exercice C.27 Changement de base (40 minutes).

On se donne deux bases de \mathbb{R}^2 : e la base canonique, et $f = \left\{ \mathbf{f}_1 = \begin{bmatrix} 3 \\ -1 \end{bmatrix}, \mathbf{f}_2 = \begin{bmatrix} 5 \\ -2 \end{bmatrix} \right\}$.

Soit $l : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ l'application linéaire représentée par la matrice $L = \begin{bmatrix} 4 & 2 \\ -1 & 1 \end{bmatrix}$ de la base f dans la base f .

1. Calculez la matrice M qui représente l de la base e dans la base e (détaillez les calculs).
2. Soit $\mathbf{u} = \begin{bmatrix} -2 \\ 1 \end{bmatrix}$. Calculez $\mathbf{u}^{(f)}$, les coordonnées de \mathbf{u} dans la base f .
3. Soit $\mathbf{v} = l(\mathbf{u})$. Calculez $\mathbf{v}^{(e)}$ et $\mathbf{v}^{(f)}$.

Exercice C.28 Matrice d'application linéaire (durée 40 minutes).

On se place dans $\mathbb{R}_2[X]$ l'espace des polynômes réels de degrés inférieur ou égal à 2. On note e sa base canonique et f la base $f = \{f_1 = X^2 + X, f_2 = X, f_3 = 2\}$. On considère $p : \mathbb{R}_2[X] \mapsto \mathbb{R}_2[X]$ l'application telle que

$$p(e_1) = f_2, \quad p(e_2) = f_3, \quad p(e_3) = f_1.$$

1. **(Matrice de p dans e et f)** Que vaut la matrice $L^{(e,f)}$, matrice de l'application p dans les bases e et f ?
2. **(Décomposition de f)** Exprimez les vecteurs de la base f par rapport à la base e .
3. **(Matrice de p dans e et e)** Que vaut la matrice $L^{(e,e)}$, matrice de l'application p dans les bases e et e ?

Exercice C.29 Noyau et image d'une matrice (durée 10 minutes).

Soit B la matrice $B = \begin{pmatrix} 1 & 3 \\ 4 & 12 \end{pmatrix}$.

1. **(Noyau de B)** Que vaut le noyau de B ? En donner une base.
2. **(Image de B)** Quelle est la dimension de l'image de B ?

Exercice C.30 Inversibilité et noyau d'une matrice (durée 20 minutes).

On se place dans $\mathbb{R}_2[X]$, l'espace des polynômes de degrés inférieur ou égal à 2. On note e sa base canonique $e = \{e_1 = 1, e_2 = X, e_3 = X^2\}$. On considère $p : \mathbb{R}_2[X] \rightarrow \mathbb{R}_2[X]$ l'application linéaire telle que

$$p(e_1) = X^2 + 3, \quad p(e_2) = \frac{1}{3}X + 2, \quad p(e_3) = 6.$$

1. **(Matrice de p dans e)** Que vaut la matrice \mathcal{P} de l'application p dans la base canonique e ?
2. **(Noyau de p)** En déduire le noyau de p .

3. **(Inversibilité)** Pouvait-on prévoir ce résultat ? (on calculera le déterminant de la matrice \mathcal{P}).

Exercice C.31 Etude d'une application linéaire (durée 45 minutes).

On se place dans $\mathbb{R}_2[X]$, l'espace des polynômes réels de degré inférieur ou égal à 2. On notera \mathcal{E} et \mathcal{F} les deux bases suivantes $\mathcal{E} = \{e_1(X), e_2(X), e_3(X)\} = \{(X+1)^2, 3X, 4\}$ et $\mathcal{F} = \{f_1(X), f_2(X), f_3(X)\} = \{X^2, X, 1\}$. Le but de cet exercice est d'étudier l'application :

$$d : \mathbb{R}^2 \rightarrow \mathbb{R}^2 \\ P(X) \mapsto P(X+1) - P(X).$$

1. **(Etude préliminaire)** Vérifier que d est une application linéaire.
2. **(Représentation matricielle)**
 - (a) Exprimer les images par d des éléments de la base \mathcal{E} (les écrire en fonction des éléments de la base \mathcal{F}).
 - (b) Reprendre la question précédente mais en écrivant ces images en fonction des éléments de la base \mathcal{E} .
 - (c) Ecrire la matrice $L^{(\mathcal{E}, \mathcal{F})}$, matrice de l'application d dans les bases \mathcal{E} et \mathcal{F} .
3. **(Noyau et image)** Que vaut le noyau de cette application ? En déduire la dimension de son image.
4. **(Changement de base)**
 - (a) Ecrire les matrices de passage de \mathcal{E} dans \mathcal{F} et réciproquement, qu'on notera $\mathcal{P}_{\mathcal{E} \rightarrow \mathcal{F}}$ et $\mathcal{P}_{\mathcal{F} \rightarrow \mathcal{E}}$.
 - (b) Exprimer la matrice $L^{(\mathcal{E}, \mathcal{E})}$, matrice de l'application d dans la base \mathcal{E} .
 - (c) Vérifier l'expression de cette matrice en utilisant la question 2b

Exercice C.32 Application linéaire et moindres carrés (durée 45 minutes).

On note $\mathcal{E} = \{e_1, e_2, e_3\}$ la base canonique de \mathbb{R}^3 . Soit $l : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ l'application linéaire définie par

$$l(u) = \langle u, \begin{bmatrix} 1 \\ 2 \end{bmatrix} \rangle e_1 + \langle u, \begin{bmatrix} 2 \\ 3 \end{bmatrix} \rangle e_2 + \langle u, \begin{bmatrix} 0 \\ 1 \end{bmatrix} \rangle e_3,$$

où $\langle \cdot, \cdot \rangle$ représente le produit scalaire dans \mathbb{R}^2 .

1. Montrer que l est une application linéaire de \mathbb{R}^2 dans \mathbb{R}^3 .
2. Calculer la matrice M qui représente l de la base canonique de \mathbb{R}^2 dans la base \mathcal{E} .
3. Que vaut l'image de l , notée $Im(l)$? Le vecteur $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ appartient-il à $Im(l)$? Si oui, donner un vecteur $u \in \mathbb{R}^2$ tel que $l(u) = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$.
4. Retrouver le résultat précédent en résolvant le système $Mu = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ au sens des moindres carrés. On détaillera les équations normales et leur résolution.
5. En utilisant le théorème du rang, que vaut le noyau de l , noté $Ker(l)$?

Exercice C.33 Application linéaire, image et noyau (durée 50 minutes).

Soit $a = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$, $b = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix}$ deux vecteurs de \mathbb{R}^3 et $c = \begin{bmatrix} 1 \\ 1 \\ 2 \\ 2 \end{bmatrix}$ un vecteur de \mathbb{R}^4 . On définit l'application

l suivante :

$$l: \mathbb{R}^3 \longrightarrow \mathbb{R}^4 \\ u \mapsto (\langle u, a \rangle - \langle u, b \rangle) c$$

où $\langle \cdot, \cdot \rangle$ correspond au produit scalaire dans \mathbb{R}^3 .

1. Montrer que l est une application linéaire.
2. Calculer les images $l(a)$ et $l(b)$.
3. Exprimer la matrice de l de la base canonique de \mathbb{R}^3 dans la base canonique de \mathbb{R}^4 . Combien de colonnes et de lignes possède cette matrice ?
4. Que vaut le noyau de l , noté $\text{Ker}(l)$? Donner une base de ce noyau.
5. Que vaut l'image de l , notée $\text{Im}(l)$? Donner une base de cette image.
6. Est-ce cohérent avec le théorème du rang ?

Exercice C.34 Application linéaire à partir du numéro étudiant (durée 35 minutes).

Nous allons définir trois vecteurs (deux vecteurs de \mathbb{R}^3 et un vecteur de \mathbb{R}^2) à partir de vos numéros étudiants. Pour cela, nous supposons que votre numéro d'étudiant est décomposé en un vecteur de \mathbb{R}^3 noté v_1 , un vecteur de \mathbb{R}^2 noté u_1 et un autre vecteur de \mathbb{R}^3 noté v_2 . Par exemple, si mon numéro étudiant est le 12345678, alors il se décompose de la manière suivante :

$$\underbrace{1 \ 2 \ 3}_{=v_1} \underbrace{4 \ 5}_{=u_1} \underbrace{6 \ 7 \ 8}_{=v_2}.$$

Dans ce cas, j'ai donc $v_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$, $u_1 = \begin{bmatrix} 4 \\ 5 \end{bmatrix}$, $v_2 = \begin{bmatrix} 6 \\ 7 \\ 8 \end{bmatrix}$.

Nous considérons également $u_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ un vecteur de \mathbb{R}^2 commun à tous les étudiants.

1. Écrire sur votre copie votre numéro d'étudiant ainsi que vos vecteurs v_1 , u_1 et v_2 correspondants.
2. Soit $l: \mathbb{R}^3 \rightarrow \mathbb{R}^2$ l'application linéaire définie par

$$l(x) = \langle v_1, x \rangle u_1 + \langle v_2, x \rangle u_2,$$

où $\langle \cdot, \cdot \rangle$ représente le produit scalaire dans \mathbb{R}^3 .

Nous rappelons que les vecteurs v_1 , u_1 et v_2 ont été définis à la question précédente et que

$u_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ est un vecteur commun à tout le monde.

Calculer l'image par l de $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$.

3. Calculer la matrice de l de la base canonique de \mathbb{R}^3 dans la base $\mathcal{U} = \{u_1, u_2\}$.
4. Calculer la matrice de l de la base canonique de \mathbb{R}^3 dans la base canonique de \mathbb{R}^2 .

Exercice C.35 Noyau et image (25 minutes).

Soit l une application linéaire de \mathbb{R}^3 dans \mathbb{R}^4 représentée dans les bases canoniques par

$$L = \begin{bmatrix} 1 & 0 & 2 \\ 1 & 2 & 0 \\ 0 & 1 & -1 \\ 1 & 0 & 2 \end{bmatrix}$$

1. Calculer le noyau $\text{Ker}(l)$: donner sa dimension, et calculer une base de $\text{Ker}(l)$.
2. Quelle est la dimension de l'image $\text{Im}(l)$?
3. Calculer une base de $\text{Im}(l)$.

C.5 Éléments propres**Exercice C.36 Éléments propres, puissance d'une matrice (durée 45 minutes).**

Soit la matrice A suivante :

$$A = \begin{bmatrix} 0 & -1 \\ 3 & 4 \end{bmatrix}$$

1. Calculer les valeurs propres de la matrice A .
2. Calculer les vecteurs propres correspondants.
3. En déduire la diagonalisation de la matrice A .
4. Calculer A^k .

Exercice C.37 Diagonalisation (durée 35 minutes).

Soit la matrice $A = \begin{bmatrix} 0 & -2 \\ -1 & -1 \end{bmatrix}$

1. Calculer les valeurs propres de la matrice A .
2. Calculer les vecteurs propres de la matrice A .
3. Écrire la diagonalisation de la matrice A .

Exercice C.38 Éléments propres (durée 15 à 20 minutes).

Soit l l'application linéaire de \mathbb{R}^2 dans \mathbb{R}^2 ayant pour éléments propres :

$$\lambda_1 = -1, \mathbf{v}_1 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \lambda_2 = 2, \mathbf{v}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Calculez la matrice A représentant l dans la base canonique.

Exercice C.39 Éléments propres (durée 20 minutes).

On se donne la matrice $A = \begin{bmatrix} -4 & 0 & 4 \\ 3 & 1 & -3 \\ -2 & 0 & 2 \end{bmatrix}$ et le vecteur $\mathbf{v} = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$.

1. Calculer $\det(A)$.
2. Qu'en déduisez-vous (plusieurs réponses) ?
3. Calculer $A\mathbf{v}$.
4. Qu'en déduisez vous ?

Exercice C.40 Valeurs propres (durée 25 minutes).

Soit A une matrice $n \times n$ inversible, avec des valeurs propres λ_i triées $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| > 0$.

Soit $A = PDP^{-1}$ la diagonalisation de A , avec $D = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$.

1. Exprimer A^{-1} en fonction de P , P^{-1} , et des λ_i .
2. Proposez un algorithme d'approximation de λ_n et du vecteur propre associé v_n .

Remarque : on pourra s'appuyer sur les algorithmes classiques sans les ré-écrire (produit matriciel, produit scalaire et normes, inversion de matrice et de système...).

Exercice C.41 Éléments propres et changement de base (durée 30 minutes).

On considère $l : \mathbb{R}^3 \mapsto \mathbb{R}^3$ une application linéaire, représentée par la matrice A dans la base canonique, notée e , de \mathbb{R}^3 . On suppose que A a les éléments propres suivants :

$$\lambda_1 = 2 \text{ associée à } u_1 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

$$\lambda_2 = 3 \text{ associée à } v_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \text{ et } v_2 = \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix}.$$

1. **(Matrice de l)** Que vaut la matrice représentant l dans la base des vecteurs propres ?
2. **(Matrice de passage)** Quelle est la matrice de passage de la base des vecteurs propres dans la base e ?
3. **(Matrice A)** Que vaut A ? (on détaillera les calculs)

Exercice C.42 Pivot de Gauss et changement de base (durée 45 minutes).

On se place dans \mathbb{R}^2 , on notera e sa base canonique. Soit $\ell : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ l'application linéaire représentée par la matrice $L = \begin{pmatrix} 4 & 3 \\ -1 & 0 \end{pmatrix}$ dans la base canonique. On cherche à déterminer l'antécédent par ℓ du vecteur $\begin{pmatrix} 2 \\ -0.25 \end{pmatrix}$ par deux méthodes distinctes. C'est-à-dire qu'on cherche le vecteur X tel que $LX = \begin{pmatrix} 2 \\ -0.25 \end{pmatrix}$.

1. **(Méthode 1 : pivot de Gauss)** On applique tout d'abord la méthode du pivot de Gauss (on détaillera les calculs dans les questions suivantes).
 - (a) Appliquez l'algorithme du pivot de Gauss pour obtenir un système triangulaire supérieur.
 - (b) Trouvez le vecteur X grâce à l'algorithme de remontée.
2. **(Éléments propres de L)**
 - (a) Que vaut le polynôme caractéristique de L ? Que valent ses valeurs propres ?
 - (b) Vérifiez que $u = \begin{pmatrix} -3 \\ 1 \end{pmatrix}$ et que $v = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$ sont des vecteurs propres associés aux valeurs propres trouvées précédemment. On note \mathcal{B} la base formée de ces vecteurs $\mathcal{B} = \{u, v\}$.
3. **(Matrice de ℓ dans \mathcal{B})**
 - (a) Que vaut la matrice de ℓ dans \mathcal{B} , notée par la suite $L^{(\mathcal{B})}$?
 - (b) Quel est le lien entre $L^{(\mathcal{B})}$ et L ? (on détaillera les matrices de passage)
4. **(Méthode 2 : résolution par la diagonalisation)**

(a) Résoudre le système $L^{(B)} \tilde{X} = \begin{pmatrix} -\frac{7}{5} \\ \frac{8}{8} \end{pmatrix}$.

(b) En déduire le vecteur X recherché initialement.

Exercice C.43 Numéro étudiant et changement de bases (durée 45 minutes).

Nous allons créer une application linéaire propre à chaque étudiant, basée sur votre numéro d'étudiant (ce numéro est noté sur votre carte pass campus).

Pour cela, nous supposons que votre numéro étudiant est composé d'un chiffre noté λ_1 , d'un vecteur de \mathbb{R}^3 noté v_1 , d'un autre chiffre noté λ_2 et à nouveau d'un vecteur de \mathbb{R}^3 noté v_2 . Par exemple, si mon numéro d'étudiant est le 12345678, alors il se décompose de la manière suivante :

$$\underbrace{1}_{=\lambda_1} \underbrace{2 \ 3 \ 4}_{=v_1} \underbrace{5}_{=\lambda_2} \underbrace{6 \ 7 \ 8}_{=v_2}.$$

Dans ce cas, j'ai donc $\lambda_1 = 1$, $v_1 = \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix}$, $\lambda_2 = 5$ et $v_2 = \begin{bmatrix} 6 \\ 7 \\ 8 \end{bmatrix}$.

Nous considérons également $\lambda_3 = 0$ et $v_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$. Il s'agira d'un chiffre et d'un vecteur de \mathbb{R}^3

communs à tous les étudiants.

1. Écrire sur votre copie votre numéro d'étudiant ainsi que vos λ_1 , v_1 , λ_2 et v_2 correspondants.
2. On se donne deux bases de \mathbb{R}^3 : \mathcal{E} la base canonique et $\mathcal{V} = \{v_1, v_2, v_3\}$. Nous rappelons

que v_1 et v_2 ont été déterminés à la question précédente et que $v_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$ est commun à tous les étudiants.

Soit $l : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ l'application linéaire représentée par une matrice, notée A , de la base \mathcal{E} dans la base \mathcal{E} . On suppose que A a pour éléments propres λ_1 associé à v_1 , λ_2 associé à v_2 et λ_3 associé à v_3 . Nous rappelons que λ_1 et λ_2 ont été déterminés à la question précédente et que $\lambda_3 = 0$ est commun à tous les étudiants.

Calculer la matrice B qui représente l de la base \mathcal{V} dans la base \mathcal{V} .

3. Calculer la matrice C qui représente l de la base \mathcal{V} dans la base \mathcal{E} .

4. Soit $w = l(u)$ avec u le vecteur de coordonnées dans la base \mathcal{V} , $u^{(\mathcal{V})} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$. Calculer de deux manières différentes $w^{(\mathcal{E})}$ qui correspond aux coordonnées de w dans la base \mathcal{E} .

5. En notant P la matrice de passage de la base \mathcal{V} à la base \mathcal{E} , quel est le lien entre A , B et P ? Quel est le lien entre A , C et P ? (On ne demande pas de faire les calculs, ni d'explicitier la matrice A).

Exercice C.44 Diagonalisation (durée 25 minutes).

Soit la matrice $A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$.

1. Vérifier que 0, 2 et -1 sont les valeurs propres de A .
2. Vérifier que $\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$, $\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$ et $\begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}$ sont les vecteurs propres de A .
3. Réécrire la matrice A sous forme $A = PDP^{-1}$ en explicitant les matrices P et D .

Exercice C.45 Matrice d'une application linéaire (durée 55 minutes).

Soit $l : \mathbb{R}^d \rightarrow \mathbb{R}^n$ l'application linéaire représentée par la matrice A suivante de la base canonique de \mathbb{R}^d dans la base canonique de \mathbb{R}^n

$$A = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 3 & 2 \\ 0 & 0 & 4 \end{bmatrix}.$$

1. D'après la taille de la matrice A , que valent les dimensions d et n ?
2. Les vecteurs propres de A sont $v_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$, $v_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$ et $v_3 = \begin{bmatrix} \frac{5}{3} \\ 2 \\ 1 \end{bmatrix}$. Que valent les valeurs propres correspondantes ?
3. On note $\mathcal{V} = \{v_1, v_2, v_3\}$, la base des vecteurs propres. Que vaut B la matrice de l'application l de la base \mathcal{V} dans la base \mathcal{V} ?
4. Que vaut C la matrice de l'application linéaire l de la base \mathcal{V} dans la base canonique de \mathbb{R}^n ? On calculera la matrice C de deux manières différentes.
5. Existe-il des bases dans lesquelles la matrice de l'application l devient $\begin{bmatrix} 1 & -1 & \frac{1}{3} \\ 0 & 3 & -6 \\ 0 & 0 & 4 \end{bmatrix}$? Justifier votre réponse.
6. Soit u le vecteur de coordonnées dans la base canonique de \mathbb{R}^d : $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$. Que valent les coordonnées de l'image de u par l dans la base \mathcal{V} ?

Annexe D

Exercices de travaux dirigés

Séance 1/3 : inversion de matrices, résolution de systèmes

Exercice D.1 – Inversion de matrice

1. La méthode de Gauss peut être utilisée pour obtenir l'inverse d'une matrice carrée inversible. Soit le système suivant :

$$(A \mid I_n) = \left(\begin{array}{ccc|ccc} a_{1,1} & \cdots & a_{1,n} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,n} & 0 & \cdots & 1 \end{array} \right)$$

En appliquant les règles du pivot de Gauss en descente puis en remontée, on peut obtenir la matrice identité à gauche, et le système suivant :

$$(I_n \mid A^{-1})$$

Appliquez cela à la matrice suivante et vérifiez le résultat obtenu.

$$\left(\begin{array}{ccc} 1 & -1 & 2 \\ 4 & -6 & 12 \\ -1 & -5 & 12 \end{array} \right)$$

2. Donnez le nombre d'additions/soustractions et le nombre de multiplications/divisions nécessaires à l'inversion d'une matrice $n \times n$ par cette méthode. En déduire l'ordre de grandeur de l'algorithme sous-jacent.

Exercice D.2 – Décomposition LU

La méthode du pivot de Gauss permet de transformer une matrice A en une matrice triangulaire supérieure U . Cette transformation passe par une suite de matrices que l'on note ici $A = A_0, A_1, A_2, \dots, A_k = U$. Le passage de A_i à A_{i+1} correspond à une étape élémentaire de la méthode et consiste à combiner linéairement deux lignes. Il y a $k = n(n-1)/2$ étapes.

Dans la suite, on exprimera les combinaisons sous la forme $L_j \leftarrow L_j + m_{ij}L_i$ avec $i < j$ (L_i sera la ligne contenant le pivot, L_j la ligne dont un élément est mis à zéro).

1. Prenons pour A_0 la matrice A de l'exercice précédent. La matrice A_1 s'obtient par combinaison des lignes 1 et 2. Donnez la matrice $E_{1,2}$ (appelée matrice élémentaire) telle que $E_{1,2}.A_0 = A_1$.
2. Trouvez ensuite la matrice $E_{1,3}$ puis $E_{2,3}$. On obtient ainsi $E_{2,3}.E_{1,3}.E_{1,2}.A = U$.

3. Dans une matrice correspondant à la matrice identité avec une seule valeur $a_{ij} = m$ hors diagonale, la matrice inverse est la même matrice avec $a_{ij} = -m$. Ainsi par exemple,

$$\begin{pmatrix} 1 & 0 & 0 \\ m & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -m & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Déterminez la matrice $E_{1,2}^{-1}$. Donnez ensuite $E_{1,3}^{-1}$ et $E_{2,3}^{-1}$.

4. On cherche la matrice L telle que $A = L.U$. À partir de la relation $E_{2,3}.E_{1,3}.E_{1,2}.A = U$, déduisez-en la forme de la matrice L en fonction des $E_{i,j}$.
Donnez finalement L pour la matrice A .
5. Donnez la complexité de l'algorithme suggéré dans les questions précédentes et qui produit une décomposition LU . À l'aide de la relation $A = L.U$, résolvez $A.X = B$ avec $B = \begin{pmatrix} 1 \\ 4 \\ 1 \end{pmatrix}$.

Idem avec $B = \begin{pmatrix} 2 \\ 2 \\ -24 \end{pmatrix}$.

6. Dans la cas général d'une matrice A de taille $n \times n$ dont on connaît une décomposition LU , donnez le nombre d'additions/soustractions et de multiplications/divisions requises pour résoudre un système $AX = B$. Déduisez-en la complexité de l'algorithme sous-jacent.
7. Écrivez en pseudo-code l'algorithme de résolution d'un système $AX = B$ connaissant une décomposition LU de A .
8. Comment la décomposition LU d'une matrice A peut permettre de calculer A^{-1} . Comparez, en terme de complexité, les deux méthodes de calcul de l'inverse d'une matrice (méthode de Gauss comme dans l'exercice 1, décomposition LU).

Séance 2/3 : approximation, comparaison de croissances

Exercice D.3 – Droite des moindres carrés

On considère la suite de points $(1, 2), (2, 2), (3, 3), (4, 2), (5, 4)$.

1. Calculez la droite des moindres carrés.
2. Calculez les résidus.
3. Extrapolez la valeur du point d'abscisse 6.
4. Faites un graphique qui permet de tout observer.
5. Donnez la complexité du calcul des coefficients de la droite.

Exercice D.4 – Comparaison de croissances

Rappel. Soient $f(x)$ et $g(x)$ deux fonctions définies sur \mathbb{R} , positives pour x assez grand. On dit que f est négligeable devant g en $+\infty$ si $\frac{f(x)}{g(x)} \xrightarrow{x \rightarrow +\infty} 0$. Avec les notations de Landau, on note $f(x) = o(g(x))$.

Question. Classer les fonction suivantes en fonction de leur croissance en $+\infty$, au sens des o (chacune doit être négligeable par rapport à la suivante) :

$x; \log_2 x; \sqrt{x}; 1; x^k; x^x; k^x; \ln x; x^{1/2}; k^{-x}$

Exercice D.5 – Manipulation de fonctions

Simplifier les expressions suivantes :

$$f_1(x) = \log x + \log(x^2)$$

$$f_2(x) = \log x - \log 1/x$$

$$f_3(x) = \log_2 x + \log_{10} x$$

$$f_4(x) = (e^x)^3$$

$$f_5(x) = e^{\log(x)+3}$$

$$f_6(x) = e^{5 \log(x)}$$

Séance 3/3 : changements de base, éléments propres

Exercice D.6 Polynômes

On se place dans l'espace E des polynômes de degré inférieur ou égal à 3. On définit les deux bases :

$$e = \{e_1(X) = 1, e_2(X) = X, e_3(X) = X^2, e_4(X) = X^3\}$$

$$e' = \{e'_1(X) = X, e'_2(X) = 1 - X, e'_3(X) = X^3, e'_4(X) = (1 - X)^3\}$$

On appelle d l'application qui dérive un polynôme :

$$\begin{aligned} d : E &\longrightarrow E \\ p(X) &\longmapsto p'(X) \end{aligned}$$

1. Montrer que d est une application linéaire de E dans E .
2. Calculer la matrice D qui représente d dans la base e .
3. Calculer $\text{Ker}(d)$, le noyau de d .
4. Calculer $P_{e' \rightarrow e}$ la matrice de passage de e' dans e .
5. Calculer la matrice D' qui représente d dans la base e' . (*Question optionnelle*)

Exercice D.7 – Éléments propres

On s'intéresse à la matrice $A = \begin{pmatrix} 11 & -18 \\ 6 & -10 \end{pmatrix}$.

1. Calculer les valeurs propres de A .
2. Calculer les vecteurs propres de A .
3. Écrivez la diagonalisation de A .
4. Calculez A^9 .
5. Supposons maintenant que l'on cherche à calculer A^p pour $p > 0$ et M de taille $n \times n$. On suppose A diagonalisable. Donnez la complexité du calcul de A^p de deux manières : classique et sous forme diagonalisée (sans compter le coût de la mise sous forme diagonale).

Annexe E

Exercices de travaux pratiques

Voici un tableau détaillé des compétences visées. Complétez le à mesure que vous progressez, en inscrivant des preuves que vous montrez à votre enseignant.

Je suis capable de ...	Preuve
tracer une fonction simple	<i>exemple : exo 1 question 2, le 4 octobre</i>
tracer plusieurs fonctions sur le même graphique	
exporter un graphique	
sélectionner des sous matrices	
utiliser des opérations matricielles et des opérations terme à terme	
importer des données dans scilab depuis un fichier texte	
calculer une régression linéaire de deux manières différentes	
calculer les équations normales	
tracer le résultat d'une régression	
calculer l'erreur d'approximation	
modifier les propriétés d'un graphique (titre, axes, couleur, etc.)	
calculer et stocker les étapes d'un système dynamique discret	
utiliser la régression linéaire pour estimer un taux de croissance	
afficher plusieurs graphes dans une même figure	
interpréter un résultat sous forme graphique	
effectuer un changement de variable sur les données d'entrée	
effectuer une régression polynômiale	
calculer les éléments propres d'une matrice	
changer de repère (de base)	

Séance 1/3

Exercice E.1 Découverte de Scilab

1. Ouvrir `Scilab` et créer un fichier de code `decouverte.sce`
2. Tracer la fonction racine entre 0 et 10 avec un pas de 1.
3. Regarder l'utilisation des fonctions `clf()` et `scf()`.
4. Tracer les fonctions racine et x^2 sur la même figure, puis sur des figures différentes.
5. Exporter le graphe réalisé au format `.png`. Il y a deux façons de procéder.
6. Quelle est la syntaxe de création de fonction ? Créer une fonction qui somme deux éléments.
7. Écrire dans `Scilab` la matrice de l'exercice 1 du TD 1.
8. Additionner et multiplier cette matrice à elle-même. Comment effectuer une multiplication terme à terme ?
9. Calculer le déterminant de la matrice et son inverse.
10. Extraire la première ligne et sommer ses éléments. Calculer le vecteur des sommes ligne par ligne.
11. Appliquer la fonction Cosinus sur une de vos matrices, qu'observez-vous ?

Exercice E.2 Manipulation de données

1. Importer les données sur fichier "Kangourous.txt" dans une matrice. Ce fichier contient la taille du nez de différents kangourous gris. La longueur est en première colonne, la largeur en deuxième colonne.
2. Tracer la longueur en fonction de la largeur. Comment pré-traiter les données pour que le graphique soit lisible ?
3. Calculer une régression linéaire sur les données en utilisant la fonction `reglin()`.
4. Tracer sur la même figure et le même graphe les données brutes ainsi que la droite des moindres carrés. Ajouter au graphe un titre et des noms d'axes.
5. Calculer une régression linéaire sur les données en utilisant un calcul de variance et de covariance. $ax + b$ avec $a = \frac{cov(X,Y)}{var(X)}$ et $b = \bar{y} - a\bar{x}$
6. Calculer une régression linéaire sur les données en utilisant en résolvant les équations normales.
7. Calculer les résidus entre la fonction obtenue par régression linéaire et les données brutes.
8. Est-ce qu'il y a une forte corrélation entre la longueur et la largeur d'un nez de kangourou ?

Séance 2/3

Exercice E.3 Un système proie/prédateur

Le système rats/chouettes est défini ainsi :

$$\begin{aligned} C_k &= 0,5C_{k-1} + 0,4R_{k-1} \\ R_k &= -0,104C_{k-1} + 1,1R_{k-1} \end{aligned}$$

1. Modéliser le système rats/chouettes : tracer le nombre de chouettes et de rats en fonction du temps en considérant qu'il y a $C_0 = 13$ chouettes et $R_0 = 10$ (milliers de) rats au départ.
2. Comment calculer le ratio rat/chouettes vers lequel le système converge en utilisant une régression linéaire ? Le calculer.
3. La population de rats semble croître à un taux fixe de $t\%$ par mois. Si cette intuition est juste, que vaut R_k en fonction de R_{k-1} ? En déduire une estimation de t par régression linéaire. Même question pour les chouettes.
4. Essayons de confirmer le taux de croissance d'une autre manière. Si l'intuition est juste, que vaut R_k en fonction de k ? En déduire une estimation de t par régression linéaire. Même question pour les chouettes.
5. Utiliser la fonction `subplot()` pour tracer dans la même figure plusieurs graphes. Afficher les courbes pour différentes configurations (C_0, R_0) de départ. Reprendre les questions précédentes pour calculer le ratio et les taux dans chaque configuration. Quel est l'effet sur la précision des estimations ? Comment améliorer cette précision ?

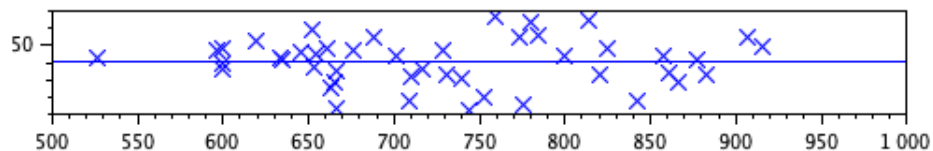
Séance 3/3

Exercice E.4 Régression polynomiale

1. Comment écrire un polynôme sous `scilab` ? Ecrire le polynôme $x^3 + 2x^2 + 3$. Tracer sa courbe en utilisant la fonction `horner()`.
2. Importer les données de "polynome_bruit.txt" dans une matrice. Tracer.
3. Effectuer un changement de variables en `log` pour obtenir une évaluation grossière du degré du polynôme qui permettrait de modéliser ces données.
4. Effectuer une régression polynomiale de degré 2, 3 et 4. Calculer les résidus pour chaque polynôme.

Exercice E.5 Changement de base

1. Reprendre les données des Kangourous, importer les données dans une matrice `Kc`. On prendra soin d'écrire `Kc` sous forme de matrice à deux lignes, n colonnes.
2. Comme lors du TP1, effectuer une régression linéaire sur les données. Évaluer la droite sur $x = 0:900$, noter cet échantillon `Rc`. Tracer la droite et les données sur la même figure.
3. Calculer la matrice `C` de covariance associée aux données.
4. Calculer les valeurs propres et vecteurs propres de la matrice `C` à l'aide la la fonction `spec`. Nommer `P` et `D` les matrices dans la décomposition $C = PDP^{-1}$.
5. Calculer `Kvp`, les données dans la base des vecteurs propres et les tracer.
6. Effectuer une régression linéaire sur les données dans cette base, et tracer la droite.
7. Gérer l'affichage de façon à obtenir la figure suivante :



Annexe F

Projet : la valeur des signes

Pendant les vacances, le quotidien l'Est Républicain amuse la galerie avec un jeu intitulé « valeur des signes ». Deux exemples sont reproduits dans la figure.

1. Formalisez le problème mathématiquement.
2. Résolvez le problème à l'aide de Scilab.

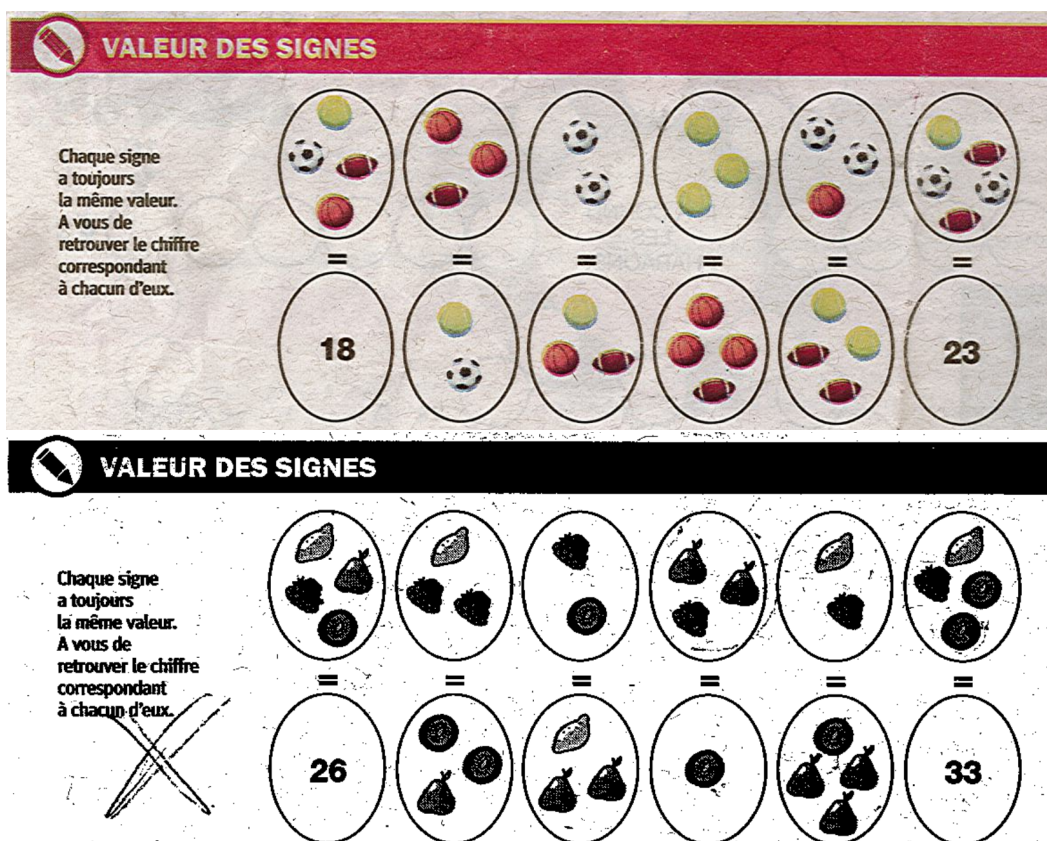


FIGURE F.1 – « valeur des signes » dans l'Est Républicain.