



# CALCUL DIFFÉRENTIEL ET OPTIMISATION CONVEXE

## APPLICATION AUX RÉSEAUX DE NEURONES

1ÈRE ANNÉE DU CYCLE D'INGÉNIEUR À L'ECOLE SUPÉRIEURE  
DE LA STATISTIQUE ET DE L'ANALYSE DE L'INFORMATION

INES ABDELJAOUED TEJ  
ANNÉE UNIVERSITAIRE 2016/2017

# Table des matières

<b>1</b>	<b>Calcul différentiel et algèbre bilinéaire</b>	<b>4</b>
1.1	Algèbre linéaire et bilinéaire . . . . .	5
1.2	Calcul différentiel . . . . .	5
1.3	Fonctions convexes . . . . .	6
1.4	Formulation mathématique d'un problème d'optimisation . . . . .	8
1.5	Définition d'un algorithme . . . . .	10
1.6	Travaux dirigés sur le chapitre 1 . . . . .	10
<b>2</b>	<b>Minimisation sans contraintes</b>	<b>13</b>
2.1	Résultats fondamentaux . . . . .	13
2.2	Algorithmes déterministes . . . . .	16
2.2.1	Méthode de descente du gradient . . . . .	16
2.2.2	Méthode de relaxation successive . . . . .	18
2.2.3	Méthode de Newton . . . . .	18
2.2.4	Méthode du gradient conjugué . . . . .	19
2.3	Exercices sur la minimisation sans contraintes . . . . .	22
<b>3</b>	<b>Minimisation avec contraintes</b>	<b>23</b>
3.1	Résultats fondamentaux . . . . .	23
3.1.1	Résultats d'existence et d'unicité . . . . .	23
3.1.2	Conditions d'optimalité . . . . .	24
3.2	Algorithmes classiques . . . . .	26
3.2.1	Méthode du gradient projeté . . . . .	26
3.2.2	Méthode de pénalisation . . . . .	27
3.2.3	Méthode de Lagrange-Newton . . . . .	29
3.2.4	Méthode de programmation quadratique successive . . . . .	30
3.3	Exercices sur la minimisation avec contraintes . . . . .	31
3.4	Dualisation de problème de minimisation convexe . . . . .	32
3.4.1	Problème dual . . . . .	32
3.4.2	Point-selle du lagrangien . . . . .	33
3.4.3	Méthode d'Uzawa . . . . .	33
<b>4</b>	<b>Application aux réseaux de neurones</b>	<b>35</b>
4.1	Historique . . . . .	35
4.2	Définitions . . . . .	36
4.2.1	Neurone formel . . . . .	36
4.2.2	Fonctions d'activation . . . . .	36
4.2.3	Perceptron multi-couches . . . . .	37
4.3	Fonctionnement du réseau de neurones . . . . .	37

---

4.4	Statistique et PMC . . . . .	41
4.5	Logiciels utilisés . . . . .	41
4.5.1	nnet de R pour le perceptron multi-couches . . . . .	41
4.5.2	Proc Neural pour le perceptron de SAS . . . . .	41
4.5.3	Sage, Python . . . . .	41
4.6	conclusion . . . . .	41
<b>5</b>	<b>Exercices de révision</b>	<b>42</b>
5.1	Correction de la série d'exercices numéro 1 . . . . .	42
5.2	Correction de la série d'exercices numéro 2 . . . . .	44
5.3	Correction de la série d'exercices numéro 3 . . . . .	46
5.4	Correction de la série d'exercices numéro 4 . . . . .	48
5.5	Correction de la série d'exercices numéro 5 . . . . .	53
5.6	Exercices de révision . . . . .	55
<b>A</b>	<b>Applications sur Python</b>	<b>62</b>
A.1	Optimisation unidimensionnelle . . . . .	62
A.2	Minimisation sans contraintes . . . . .	62
A.3	Application aux réseaux de neurones . . . . .	62
A.3.1	Nnets . . . . .	62

# Introduction

Le cours que nous proposons ici porte sur la Programmation Mathématique et plus particulièrement sur la Programmation non Linéaire. Nous nous sommes intéressés à l'Optimisation et à l'Analyse convexe avec comme principale application les réseaux de neurones formels.

L'Optimisation est une approche généraliste qui relève des sciences de la décision, et à ce titre, elle doit faire partie du bagage de tout ingénieur. A chaque fois que les ressources sont limitées, les décideurs se doivent derésoudre les problèmes rencontrés de la meilleure façon possible. Ce cours comprend aussi bien un volet théorique que pratique.

L'objectif de ce cours est de présenter la théorie de l'optimisation en dimension finie et de répondre aux 2 questions suivantes :

- comment caractériser des solutions optimales d'un problème ?
- comment résoudre un problème d'optimisation à l'aide d'algorithmes numériques ou formels ?

L'Analyse convexe est traitée par l'utilisation qu'on peut en avoir en Optimisation : d'abord avec l'étude des ensembles convexes puis de celle des fonctions convexes. Il s'agit ensuite de savoir reconnaître les problèmes d'optimisation convexe puis d'étudier les principales méthodes en optimisation numérique (algorithmes du gradient, du gradient conjugué, de Newton, etc...). Nous nous intéressons enfin aux principes qui sous-tendent ces méthodes ainsi qu'à leur mise en oeuvre pratique en langage PYTHON sous la plateforme SAGEMATH. Il s'agira enfin d'implémenter et de tester l'algorithme du Gradient conjugué sur une des principales application qu'est les réseaux de neurones.

Nous donnons à la fin de chaque chapitre une liste d'exercices de difficultés variables. Ces exercices ne sont profitables que si l'élève-ingénieur les travaille.

# Chapitre 1

## Calcul différentiel et algèbre bilinéaire

Le cadre de travail choisi est celui des espaces de dimension finie. Nous insisterons sur les idées et mécanismes de base davantage que sur les généralisations possibles ou les techniques de contextes particuliers.

Les problèmes d'optimisation se posent dans des espaces de matrices : le produit scalaire standard sur l'ensemble des matrices à  $m$  lignes et  $n$  colonnes et à coefficient dans  $\mathbf{R}$  (noté  $\mathcal{M}_{m,n}(\mathbf{R})$ ) est défini par  $\langle\langle M, N \rangle\rangle = \text{tr}({}^tMN)$ . Soit  $A$  une matrice carrée réelle, la norme de Frobenius est définie par  $\|A\|_F = (\langle\langle A, A \rangle\rangle)^{\frac{1}{2}}$ .

Le produit scalaire usuel dans  $\mathbf{R}^n$  est noté par  $\langle \cdot, \cdot \rangle$  et est défini par  $\langle u, v \rangle = {}^tuv$  pour tout  $u, v \in \mathbf{R}^n$ . Soient  $A \in \mathcal{M}_n(\mathbf{R})$ ,  $u$  et  $v$  deux vecteurs dans  $\mathbf{R}^n$  :  ${}^tAv = u^tAv \in \mathbf{R}$ . Cette propriété risque d'engendrer des erreurs (par ce que nous utilisons la transposée de matrices pour la prouver). Elle est équivalente à l'égalité  $\langle u, Av \rangle = \langle {}^tAu, v \rangle$ .

Notons  $S_n(\mathbf{R})$  l'ensemble des matrices symétrique d'ordre  $n$  à coefficient dans  $\mathbf{R}$ . Une matrice  $A \in S_n(\mathbf{R})$  est semi-définie positive lorsque  $\langle Ax, x \rangle \geq 0$ . Nous notons  $\mathcal{P}_n(\mathbf{R})$  l'ensemble des matrices semi-définie positive de taille  $n$ .

Par exemple, la matrice  $A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{pmatrix}$  est définie positive et  $B = \begin{pmatrix} -3 & 0 & 0 \\ -8 & 1 & 0 \\ -12 & 0 & 3 \end{pmatrix}$  n'est pas une matrice semi-définie positive.

Si  $S \in \mathcal{P}_n(\mathbf{R})$  alors les valeurs propres de  $S$  sont positives ou nulles. Si  $S$  est une matrice définie positive alors les valeurs propres de  $S$  sont strictement positives.

Rappelons que  $\|\cdot\|$  désigne la norme usuelle de  $\mathbf{R}^n$ . Les formulations variationnelle suivantes de la plus grande et la plus petite valeur propre de  $S \in S_n(\mathbf{R})$  sont essentielles :

$$\lambda_{\max}(S) = \max_{x \neq 0} \frac{\langle Ax, x \rangle}{\|x\|^2}, \quad \lambda_{\min}(S) = \min_{x \neq 0} \frac{\langle Ax, x \rangle}{\|x\|^2}$$

Les normes matricielles subordonnées aux normes vectorielles usuelles sont :

$$\begin{aligned} \|A\|_{\infty} &= \max_{1 \leq i \leq n} \sum_{j=1}^{j=n} |a_{ij}| && \text{maximum sur les lignes} \\ \|A\|_1 &= \max_{1 \leq j \leq n} \sum_{i=1}^{i=n} |a_{ij}| && \text{maximum sur les colonnes} \\ \|A\|_2 &= \sqrt{\max_{1 \leq i \leq n} \{|\lambda_i| \mid \lambda_i \text{ v.p. de } {}^tAA\}} && \text{racine carré du rayon spectral de } {}^tAA \end{aligned}$$

La norme de  $A$  est égale à  $\|A\|_{\infty} = \|A\|_1 = \|A\|_2 = 4$ .

## 1.1 Algèbre linéaire et bilinéaire

Si  $l : \mathbf{R}^n \rightarrow \mathbf{R}$  est une application linéaire alors il existe un unique vecteur  $v \in \mathbf{R}^n$  tel que  $l(x) = \langle v, x \rangle$  pour tout  $x \in \mathbf{R}^n$ . Les fonctions affines à valeurs réelles sont de la forme  $x \mapsto \langle v, x \rangle + c$  où  $v \in \mathbf{R}^n$  et  $c \in \mathbf{R}$ .

Si  $q : \mathbf{R}^n \rightarrow \mathbf{R}$  est une forme quadratique, il existe un unique  $Q \in S_n(\mathbf{R})$  tel que  $q(x) = \frac{1}{2} \langle Qx, x \rangle$  pour tout  $x \in \mathbf{R}^n$ . Le coefficient  $\frac{1}{2}$  est là pour simplifier les calculs. Si  $S \in S_n(\mathbf{R})$  alors toutes les valeurs propres  $\lambda_i$  de  $S$  sont réelles et il existe une matrice orthogonale  $U$  telle que  ${}^tUSU = \text{diag}(\lambda_1, \dots, \lambda_n)$  : ainsi il existe des vecteurs propres unitaires  $x_i$  (associés aux  $\lambda_i$ ) tels que  $S = \sum_{i=1}^n \lambda_i x_i x_i^t$ . C'est la décomposition spectrale de  $S$ .

## 1.2 Calcul différentiel

$f : O \subset \mathbf{R}^n \rightarrow \mathbf{R}$  définie sur un ouvert  $O$  de  $\mathbf{R}^n$  est dite différentiable en  $\bar{x} \in O$  s'il existe  $l : \mathbf{R}^n \rightarrow \mathbf{R}$  linéaire telle que :

$$f(\bar{x} + h) = f(\bar{x}) + l(h) + \|h\| \epsilon(h)$$

avec  $\epsilon(h) \rightarrow 0$  quand  $h \rightarrow 0$ . Dans  $\mathbf{R}^n$  (muni du produit scalaire usuel), l'application  $l$  est représentée par un unique vecteur, appelé gradient de  $f$  en  $\bar{x}$ , et noté  $\nabla f(\bar{x})$  (ça se lit nabla f de  $\bar{x}$ ) :  $\nabla f(\bar{x})$  est le vecteur de composantes  $\partial_1 f(\bar{x}), \dots, \partial_n f(\bar{x})$ .

Plus généralement,  $F : O \subset \mathbf{R}^n \rightarrow \mathbf{R}^m$  qui à  $x \in O$  associe  $F(x) = \begin{pmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{pmatrix}$  est dite différentiable en  $\bar{x}$  si chacune des fonctions-composantes  $f_1, \dots, f_m$  l'est. Nous appellerons alors matrice jacobienne de  $F$  en  $\bar{x}$  et on note  $JF(\bar{x})$  la matrice de  $\mathcal{M}_{m,n}(\mathbf{R})$  dont les lignes sont  ${}^t\nabla f_1(\bar{x}), \dots, {}^t\nabla f_m(\bar{x})$  c'est-à-dire que le terme  $(i, j)$  de  $JF(\bar{x})$  est  $\partial_j f_i(\bar{x})$ .

Nous disons qu'une fonction numérique  $f$  est deux fois différentiable en  $\bar{x} \in O$  lorsque  $\nabla f : O \subset \mathbf{R}^n \rightarrow \mathbf{R}^n$  est différentiable en  $\bar{x}$ . La matrice jacobienne de  $\nabla f$  en  $\bar{x}$  est appelée matrice hessienne de  $f$  en  $\bar{x}$  et notée  $\nabla^2 f(\bar{x})$ . Il s'agit d'une matrice symétrique de taille  $n$  dont le terme  $(i, j)$  est  $\partial_{ij}^2 f(\bar{x})$ .

Enfin deux ensembles de résultats de Calcul différentiel sont essentiels en Optimisation : le théorème de la fonction implicite et le théorème d'inversion locale ainsi que les développements de Taylor sous leurs formes diverses. Par exemple, le développement en série de Taylor avec reste intégral est donné par :

$$f(y) = f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 \langle \nabla f(x + t(y - x)) - \nabla f(x), y - x \rangle dt$$

Dans le cas où  $f : O \subset \mathbf{R}^n \rightarrow \mathbf{R}$  est deux fois différentiable en  $\bar{x} \in O$ , nous avons le développement de Taylor-Young d'ordre deux suivant :

$$f(\bar{x} + h) = f(\bar{x}) + \langle \nabla f(\bar{x}), h \rangle + \frac{1}{2} \langle \nabla^2 f(\bar{x})h, h \rangle + \|h\|^2 \epsilon(h); \quad \lim_{h \rightarrow 0} \epsilon(h) = 0.$$

*Exemple 1.2.1.* Nous considérons la fonction  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  définie par  $J(x) = \frac{1}{2} \langle Ax, x \rangle - \langle b, x \rangle + c$  où  $A \in \mathcal{M}_n(\mathbf{R})$ ,  $b \in \mathbf{R}^n$  et  $c \in \mathbf{R}$ . Alors  $\nabla J(x) = \frac{1}{2}({}^tA + A)x - b$  pour tout  $x \in \mathbf{R}^n$ . La hessienne de  $J$  en  $x$  est la matrice symétrique  $\nabla^2 J(x) = \frac{1}{2}({}^tA + A)$ .

## 1.3 Fonctions convexes

L'ensemble  $C \subset \mathbf{R}^n$  est *convexe* si

$$\forall (x, y) \in C \times C, \forall t \in [0, 1] \quad tx + (1 - t)y \in C.$$

Autrement dit,  $C$  est convexe s'il contient tout segment reliant deux quelconques de ses points.

*Exemple 1.3.1* (Ensembles convexes). Une droite, un plan, l'intérieur d'un triangle, un intervalle  $[a, b]$  est convexe dans  $\mathbf{R}$ . Une réunion disjointe d'intervalles de  $\mathbf{R}$  n'est pas convexe ( $\mathbf{R}^*$  par exemple).

**Définition 1.3.1** (Fonction convexe). La fonction  $J : C \subset \mathbf{R}^n \longrightarrow \mathbf{R} \cup \{\infty\}$  est convexe si  $C$  est convexe et si

$$\forall (x, y) \in C \times C, \forall t \in [0, 1] \quad J(tx + (1 - t)y) \leq tJ(x) + (1 - t)J(y).$$

La fonction  $J$  est dite strictement convexe lorsque l'inégalité est stricte dès que  $x \neq y$ .

*Exemple 1.3.2.* — Exemple de fonction convexe : la corde AB est au-dessus de l'arc AB.

- $J(x) = \|x\|$  est convexe.
- Toute application affine, c'est-à-dire de la forme  $J(x) = \langle b, x \rangle + \beta$  où  $b$  et  $x$  sont dans  $\mathbf{R}^n$  et  $\beta \in \mathbf{R}$  est convexe mais pas strictement.

*Exemple 1.3.3.* Exemple de fonctions non convexes (la corde AB n'est pas au-dessus de l'arc) ; exemples de fonctions convexes non strictement convexes.

Toute fonction convexe sur un espace de dimension finie ( $\mathbf{R}^n$ ) est continue sur l'intérieur de son domaine (voir nb[?]).

**Théorème 1.3.1.** Soit  $J : C \subset \mathbf{R}^n \longrightarrow \mathbf{R}$ , différentiable sur  $C$ , avec  $C$  convexe.  $J$  est convexe si et seulement si  $\nabla J$  est un opérateur croissant de  $\mathbf{R}^n$  dans  $\mathbf{R}^n$ , c'est-à-dire

$$\forall (u, v) \in C \times C \quad \langle \nabla J(u) - \nabla J(v), u - v \rangle \geq 0.$$

**Preuve :** Si  $J$  est convexe alors

$$\forall (u, v) \in C \times C \quad \forall t \in ]0, 1[ \quad J(u + t(v - u)) - J(u) \leq t(J(v) - J(u)).$$

En divisant par  $t$  et en faisant tendre  $t$  vers  $0^+$ , on obtient :

$$\forall (u, v) \in C \times C \quad J(v) \geq J(u) + \langle \nabla J(u), v - u \rangle. \quad (1.3.1)$$

De même :

$$\forall (u, v) \in C \times C \quad J(u) \geq J(v) + \langle \nabla J(v), u - v \rangle.$$

En sommant les deux inégalités précédentes, on obtient le résultat. Réciproquement, supposons que

$$\forall (u, v) \in C \times C \quad \langle \nabla J(u) - \nabla J(v), u - v \rangle \geq 0.$$

Nous définissons  $\phi : [0, 1] \rightarrow \mathbf{R}$  tel que  $\phi(t) = tJ(u) + (1-t)J(v) - J(v + t(u-v))$ . Il est facile de vérifier que grâce à l'hypothèse, la fonction  $\phi$  est dérivable et que  $\forall t_1, t_2 \in [0, 1]$ ,  $(\phi'(t_1) - \phi'(t_2))(t_1 - t_2) \leq 0$ . La fonction  $\phi'$  est décroissante sur  $[0, 1]$  et elle vérifie  $\phi(0) = \phi(1) = 0$ . D'après le théorème de Rolle, il existe  $a \in ]0, 1[$  tel que  $\phi'(a) = 0$ . Sur  $[0, a]$ ,  $\phi'$  est décroissante, positive et sur  $[a, 1]$ ,  $\phi'$  est décroissante négative. Donc  $\phi$  croît sur  $[0, a]$  et décroît sur  $[a, 1]$  avec  $\phi(0) = \phi(1) = 0$ . Donc  $\phi$  est positive sur  $[0, 1]$ .

Supposons que  $\nabla J$  soit un opérateur strictement monotone :

$$\forall (u, v) \in C \times C, u \neq v, \quad \langle \nabla J(u) - \nabla J(v), u - v \rangle > 0$$

alors  $J$  est strictement convexe.

**Théorème 1.3.2.** *Soit  $J$  deux fois différentiable sur un ouvert convexe  $O$ . Alors :*

1.  *$J$  est convexe sur  $O$  si et seulement si  $\nabla^2 J(x)$  est semi-définie positive pour tout  $x \in O$  ;*
2. *Si  $\nabla^2 J(x)$  est définie positive pour tout  $x \in O$ , alors  $J$  est strictement convexe sur  $O$  ;*

**Preuve :** Soient  $s \neq 0$  et  $x_1 = x_0 + \alpha s$ . Le développement en série de Taylor de  $\nabla J(x_1)$  donne :

$$\nabla J(x_1) = \nabla J(x_0) + \alpha \nabla^2 J(x_0)s + o(\alpha) \quad (1.3.2)$$

Comme  $J$  est convexe, on a grâce à l'inéquation (1.3.1) :  $J(x_1) \geq J(x_0) + {}^t(x_1 - x_0)\nabla J(x_0)$  nous avons :

$${}^t(x_1 - x_0)\nabla J(x_1) \geq J(x_1) - J(x_0) \geq {}^t(x_1 - x_0)\nabla J(x_0). \quad (1.3.3)$$

Ainsi, en combinant (1.3.2) et (1.3.3), nous obtenons

$${}^t(x_1 - x_0)(\nabla J(x_0) + \alpha \nabla^2 J(x_0)s + o(\alpha)) \geq J(x_1) - J(x_0) \geq {}^t(x_1 - x_0)\nabla J(x_0).$$

Ce qui est équivalent à

$$\alpha {}^t s(\nabla J(x_0) + \alpha \nabla^2 J(x_0)s + o(\alpha)) \geq J(x_0 + \alpha s) - J(x_0) \geq \alpha {}^t s \nabla J(x_0)$$

Donc :

$$\alpha^2 {}^t s \nabla^2 J(x_0)s + {}^t s \alpha^2 o(\alpha) \geq J(x_0 + \alpha s) - J(x_0) - \alpha {}^t s \nabla J(x_0) \geq 0$$

Ainsi, nous obtenons

$${}^t s \nabla^2 J(x_0)s + {}^t s o(\alpha) \geq 0$$

Lorsque  $\alpha$  tend vers 0, nous obtenons le résultat sur la courbure positive de  $J$ .

Réciproquement, le développement en série de Taylor de  $J(x_1)$  donne  $J(x_1) = J(x_0) + {}^t(x_1 - x_0)\nabla J(x_0) + \frac{1}{2}\alpha^2 {}^t s \nabla^2 J(x_\theta)s$  avec  $x_\theta = x_0 + \theta(x_1 - x_0)$  et  $\theta \in [0, 1]$ . D'après l'hypothèse, nous obtenons

$$J(x_1) \geq J(x_0) + {}^t(x_1 - x_0)\nabla J(x_0)$$

donc  $J$  est convexe d'après (1.3.1).



*Exemple 1.3.4.* Soit  $A = \begin{pmatrix} -3 & 2 & 4 \\ -8 & 5 & 8 \\ -12 & 2 & 13 \end{pmatrix}$  une matrice carrée symétrique d'ordre 3 semi-définie positive et  $b = \begin{pmatrix} -3 \\ 1 \\ 2 \end{pmatrix}$  un vecteur de  $\mathbf{R}^3$ . Alors  $J$  définie par  $J(x) = \frac{1}{2}(Ax, x) - (b, x)$  est convexe. Si nous considérons la matrice  $A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 3 \end{pmatrix}$  définie positive alors  $J$  est strictement convexe.

## 1.4 Formulation mathématique d'un problème d'optimisation

Rappelons qu'une boule ouverte de  $\mathbf{R}^n$ , centrée en  $x^*$  de rayon  $r \geq 0$  est l'ensemble  $B(x^*, r) = \{x \in \mathbf{R}^n \mid \|x - x^*\| < r\}$ , et la boule est dite fermée si l'égalité est large.

**Définition 1.4.1** (Minimum (maximum) local). Soient  $C$  un ensemble non vide de  $\mathbf{R}^n$  et  $J$  une fonction de  $C$  dans  $\mathbf{R}$ .

On dit que  $x^* \in C$  réalise un *minimum local* de  $J$  sur  $C$  si on peut trouver une boule ouverte  $B(x^*)$  de  $\mathbf{R}^n$  centrée en  $x^*$  telle que

$$\forall x \in B(x^*) \cap C \quad J(x^*) \leq J(x)$$

On dit que  $x^* \in C$  réalise un *maximum local* de  $J$  sur  $C$  si on peut trouver une boule ouverte  $B(x^*)$  de  $\mathbf{R}^n$  centrée en  $x^*$  telle que

$$\forall x \in B(x^*) \cap C \quad J(x^*) \geq J(x)$$

**Définition 1.4.2** (Minimum (maximum) global). On dit que  $x^* \in C$  réalise un *minimum global* de  $J$  sur  $C$  si  $\forall x \in C, J(x^*) \leq J(x)$ . On dit que  $x^* \in C$  réalise un *maximum global* de  $J$  sur  $C$  si  $\forall x \in C, J(x^*) \geq J(x)$ .

Les minima et maxima sont dits *stricts* si les inégalités dans les définitions précédentes sont strictes.

*Exemple 1.4.1.* Minima et maxima locaux et globaux ; infinité de minima et maxima globaux ; pas de maximum global et infinité de maxima et minima locaux.

Nous nous intéresserons essentiellement à la recherche des points réalisant des minima car la recherche de maxima peut se ramener à celle des minima comme le montre la proposition suivante :

**Proposition 1.4.1.** Si  $x^*$  réalise un maximum (local ou global) de  $J$  sur  $C$ ,  $x^*$  réalise un minimum (local ou global) de  $-J$  sur  $C$ .

**Preuve :** voir [?].

*Exemple 1.4.2.* En considérant la fonction  $J : \mathbf{R}^2 \rightarrow \mathbf{R}$  vérifiant  $J(x_1, x_2) = 3x_1^2 - 4x_1^2x_2 + x_2^2$ , nous remarquons que  $(0, 0)$  est un point critique, mais qu'il n'est pas un minimum global.

Soit  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  une fonction de plusieurs variables  $x = (x_1, \dots, x_n)$  à valeurs réelles. Elle est appelée *fonction coût* ou *fonction objectif*. Soit  $g : \mathbf{R}^n \rightarrow \mathbf{R}^p$  une fonction à  $p$  composantes et on peut écrire

$$g(x) = (g_1(x), \dots, g_p(x)), \quad x \in \mathbf{R}^n$$

où les  $g_i(x) : \mathbf{R}^n \rightarrow \mathbf{R}$  représentent les *contraintes en inégalités*. Soit  $h : \mathbf{R}^n \rightarrow \mathbf{R}^q$  une fonction à  $q$  composantes et on peut écrire

$$h(x) = (h_1(x), \dots, h_q(x)), \quad x \in \mathbf{R}^n$$

où les  $h_j(x) : \mathbf{R}^n \rightarrow \mathbf{R}$  représentent les *contraintes en égalités*. Un problème d'optimisation est donc défini par :

$$(\mathcal{P}) = \begin{cases} \min J(x) \\ g(x) \leq 0 \\ h(x) = 0 \end{cases}$$

On note  $C$  l'ensemble des contraintes, c'est-à-dire :  $C = \{x \in \mathbf{R}^n \mid g(x) \leq 0, h(x) = 0\}$ . On suppose que  $C$  est non vide. Un élément  $x \in C$  est dit *réalisable*.

**Définition 1.4.3** (Programmation quadratique). On dit que le problème  $(\mathcal{P})$  est un problème de *programmation quadratique* quand la fonction objectif est quadratique  $J(x) = \frac{1}{2} {}^t x A x - b x + c$  avec  $A$  symétrique définie positive et les fonctions contraintes sont affines  $a_i x \leq b_i$ ,  $i = 1..p + q$ .

**Définition 1.4.4** (Programmation convexe). On dit que le problème  $(\mathcal{P})$  est un problème de *programmation convexe* quand les fonctions  $J$ ,  $g_i$ ,  $i = 1..p$  sont convexes et les fonctions  $h_j$ ,  $j = 1..q$  sont affines.

**Définition 1.4.5** (Programmation linéaire). On dit que le problème  $(\mathcal{P})$  est un problème de *programmation linéaire* quand les fonctions  $J$ ,  $g_i$ ,  $i = 1..p$  et  $h_j$ ,  $j = 1..q$  sont affines.

Le cas de la programmation linéaire est un cas particulier de la programmation convexe mais il se présente comme un cas singulier pour lequel on n'est pas toujours sûr de trouver des solutions. De ce fait, les méthodes employées pour la résolution de ce problème sont des méthodes très spécifiques qui relèvent de la Recherche Opérationnelle (et non pas des cas particuliers des méthodes de programmation non linéaire).

*Exemple 1.4.3.* En statistique : la Régression linéaire. Considérons un nuage de  $n$  points de  $\mathbf{R}^2$  :  $M_i = (t_i, x_i)$ ,  $1 \leq i \leq n$ . Ces données sont souvent le résultat de mesures et on cherche à décrire le comportement global de ce nuage en approchant au mieux ses points par une droite  $x = at + b$ . On utilise pour cela la méthode des moindres carrés : on veut donc trouver un couple de réels  $(a, b)$  solution de

$$\min J(a, b) = \sum_{i=1}^{i=n} (x_i - at_i - b)^2 \quad .$$

1. Calculer le gradient de  $J$  en un point quelconque  $(a, b)$  de  $\mathbf{R}^2$ .

2. Ecrire que  $\nabla J(a, b) = 0$  en fonction de  $S_t = \sum_{i=1}^{i=n} t_i$ ,  $S_x = \sum_{i=1}^{i=n} x_i$ ,  $S_{xt} = \sum_{i=1}^{i=n} x_i t_i$  et  $S_{t^2} = \sum_{i=1}^{i=n} t_i^2$ .
3. On suppose que  $(S_t)^2 - nS_{t^2} \neq 0$ . Résoudre le système obtenu dans (b). Soit  $(a^*, b^*)$  la solution.
4. Vérifier que le couple  $(a^*, b^*)$  est bien un minimum.

Pour s'assurer de l'existence de solutions du problème  $(\mathcal{P})$ , nous précisons les hypothèses sur les fonctions  $J$ ,  $g$  et  $h$  dans les prochains chapitres. Nous proposerons également un certain nombre de méthodes et d'algorithmes permettant de résoudre les problèmes d'optimisation convexe.

## 1.5 Définition d'un algorithme

Un algorithme est défini par une application  $\mathcal{A}$  de  $\mathbf{R}^n$  dans  $\mathbf{R}^n$  permettant la génération d'une suite d'éléments de  $\mathbf{R}^n$  par la formule :

$$\begin{cases} x_0 \in \mathbf{R}^n \text{ donné,} & k = 0 & \text{Etape d'initialisation} \\ x_{k+1} = \mathcal{A}(x_k) & k = k + 1 & \text{Itération } k. \end{cases}$$

Ecrire un algorithme est donc se donner une suite  $(x_k)_{k \in \mathbf{N}}$  de  $\mathbf{R}^n$  ; étudier la convergence de l'algorithme, c'est étudier la convergence de la suite  $(x_k)_{k \in \mathbf{N}}$ .

**Définition 1.5.1.** Soit  $(x_k)_{k \in \mathbf{N}}$  une suite de limite  $x^*$  définie par la donnée d'un algorithme convergent  $\mathcal{A}$ . On dit que la convergence de  $\mathcal{A}$  est

- *linéaire* si l'erreur  $e_k = \|x_k - x^*\|$  décroît linéairement :

$$\exists C \in [0, 1[, \exists k_0, \forall k \geq k_0 \quad e_{k+1} \leq C e_k$$

- *Super-linéaire* si l'erreur  $e_k$  décroît de la manière suivante :

$$e_{k+1} \leq \alpha_k e_k$$

où  $\alpha_k$  est une suite positive convergente vers 0. Si  $\alpha_k$  est une suite géométrique, la convergence de l'algorithme est dite *géométrique*.

- *d'ordre p* si l'erreur  $e_k$  décroît de la manière suivante :

$$\exists C \geq 0, \exists k_0, \forall k \geq k_0 \quad e_{k+1} \leq C e_k^p$$

Si  $p = 2$ , la convergence est dite *quadratique*.

On a bien entendu intérêt à ce que la vitesse de convergence d'un algorithme soit la plus élevée possible afin d'obtenir la solution avec un minimum d'itérations pour une précision donnée.

## 1.6 Travaux dirigés sur le chapitre 1

*Exercice 1.* Soit  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  continûment différentiable. Soit  $x_0$  tel que  $\nabla f(x_0) \neq 0$ . Que représente  $\nabla f(x_0)$  pour la surface de niveau  $S := \{x \in \mathbf{R}^n | f(x) = f(x_0)\}$  ?

*Exercice 2.* Déterminer  $\nabla f(x)$  et  $\nabla^2 f(x)$  en tout point  $x$  pour :

1.  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  tel que  $f(x) = \langle c, x \rangle + \gamma$
2.  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  tel que  $f(x) = \sum_{i=1}^m (r_i(x))^2$  où les  $r_i : \mathbf{R}^n \rightarrow \mathbf{R}$  sont deux fois différentiables.

*Exercice 3.* Déterminer  $JF(x)$  avec  $F : \mathbf{R}^n \rightarrow \mathbf{R}^m$  tel que  $F(x) = Lx + b$  et  $L \in \mathcal{M}_{m,n}(\mathbf{R})$ .

*Exercice 4.* Trouver le ou les minima de la fonction  $f(x) = x^4 + 3x^2 + 1$  sur l'ensemble  $\mathbf{R}$ .

*Exercice 5.* Soit  $a$  une forme bilinéaire symétrique de  $\mathbf{R}^n \times \mathbf{R}^n$  dans  $\mathbf{R}$

1. Montrer que l'on peut trouver une matrice symétrique  $A$  d'ordre  $n$  telle que

$$\forall u, v \in \mathbf{R}^n \quad a(u, v) = \langle Au, v \rangle$$

2. Calculer le gradient et la dérivée seconde (Hessien) de la fonctionnelle  $J$  définie sur  $\mathbf{R}^n$  par :  $J(v) = \frac{1}{2} \langle Av, v \rangle - \langle b, v \rangle$  où  $b \in \mathbf{R}^n$  est fixé.
3. A quelle condition sur  $A$ , la fonction  $J$  est-elle convexe ? strictement convexe ?
4. Quels sont les points critiques de la fonction  $J$ .

*Exercice 6.* Montrer que la fonction indicatrice d'un ensemble  $K \subset \mathbf{R}^n$  définie par

$$1_K(x) = \begin{cases} 0 & \text{si } x \in K \\ +\infty & \text{sinon } (x \in \mathbf{R}^n / K) \end{cases}$$

est convexe si et seulement si  $K$  est convexe.

*Exercice 7.* Montrer l'inégalité de Young :  $\forall a, b > 0, \forall p, q \in \mathbf{N}$  tels que  $\frac{1}{p} + \frac{1}{q} = 1$ ,  $ab \leq \frac{a^p}{p} + \frac{b^q}{q}$ .

*Exercice 8.* On considère les formes quadratiques  $q$  suivantes. Dans chaque cas dites si  $q$  est définie positive, semi-définie positive ou ni l'un ni l'autre :

1.  $q(x) = 2x_1^2 + 8x_2x_3 + 3x_4^2 + 2x_1x_4$
2.  $q(x) = x_1^2 + x_2^2 + x_1x_3 + 4x_3^2$ .

*Exercice 9.* Etudier la convexité ou la concavité des fonctions définies par :

1.  $f(x) = 3x_1^2 + 2x_2^2 - 8x_1$
2.  $f(x) = x_1^2 - 10x_1x_2 + 9x_2^2$
3.  $f(x) = -x_1^4 + x_2x_1^2 + x_1^2 + x_2 - 3$ .

*Exercice 10.* Soit la fonction  $f(x, y) = 3xe^y - x^3 - e^{3y}$ .

1. Calculer le gradient de cette fonction ; en déduire le ou les points critiques ;
2. Calculer la matrice hessienne au(x) point(s) critique(s) ; Que peut-on conclure sur la matrice hessienne ?
3. Quelle est la plus grande valeur de la fonction  $f$  ? quelle est la plus petite valeur de la fonction  $f$  ?
4. Que peut-on conclure sur la nature de l'extremum trouvé ? Vérifier les résultats avec MUPAD ou MAPLE.

*Exercice 11.* Soit la fonction  $f(x, y) = -6x_1 - 8x_2 + x_1^2 + x_2^2$ . Calculer le gradient de  $f$  et la matrice hessienne de  $f$ . En déduire l'ensemble des points critiques de  $f$ .

*Exercice 12.* Chercher l'ensemble des points critiques de la fonction  $f(x_1, x_2) = x^4 + 3y^2x^2 + x^2 + y^4$  sur  $\mathbf{R}^2$ .

*Exercice 13.* Soit la fonction  $f(x, y) = -(x^2 - 1)^2 - (x^2y - x - 1)^2$ . Même question que l'exercice précédent.

*Exercice 14.* Calculer le gradient puis les points critiques pour chaque fonction  $f$  donnée ci-dessous :

1.  $f(x) = x_1^2 + x_2^2 - 12x_1 - 16x_2$
2.  $f(x) = 6x_1^2 - 8x_2^2 - 54$
3.  $f(x) = (-2x_1^2 - 11x_1 + 12)x_2 - x_1^2$
4.  $f(x) = x_1(x_1^2 + x_2^2) - (x_1^2 + x_2^2) - x_1 + 1$
5.  $f(x, y) = 9 - 2x + 4y - x^2 - 4y^2$
6.  $f(x, y) = x^3y + 12x^2 - 8y$
7.  $f(x, y) = e^x \cos(y)$
8.  $f(x, y) = x^2 + y^2 + \frac{1}{x^2y^2}$

Trouver, si possible, les extremas et donner leur nature.

# Chapitre 2

## Minimisation sans contraintes

Soit  $J$  une fonction de  $\mathbf{R}^n$  dans  $\mathbf{R}$ . Un problème d'optimisation sans contraintes est formulé de la façon suivante :

$$(\mathcal{P}) \quad \left\{ \min_{x \in \mathbf{R}^n} J(x) \right.$$

### 2.1 Résultats fondamentaux

**Définition 2.1.1** (Fonction coercive). On dit que  $J : \mathbf{R}^n \longrightarrow \mathbf{R}$  est coercive si

$$\lim_{\|x\| \rightarrow +\infty} J(x) = +\infty$$

*Exemple 2.1.1.*  $\|x\|$  est coercive. Pour  $n = 2$ ,  $J(x) = x_1^2 + x_2^2 - ax_1 - bx_2 - c$  est coercive  $\forall x \in \mathbf{R}^2$  et  $\forall a, b \in \mathbf{R}$ . Toute forme quadratique définie positive est coercive. Une fonction affine n'est jamais coercive.

*Exemple 2.1.2.* La fonction  $J$  définie par  $J(x_1, x_2) = x_1^2 - x_2^2$  n'est pas coercive. En effet, pour  $x = (0, n)$ ,  $\lim_{\|x\| \rightarrow +\infty} J(x) = \lim_{n \rightarrow +\infty} J(x) = \lim_{n \rightarrow +\infty} -n^2 = -\infty$ .

**Théorème 2.1.1** (Existence). Soit  $J : \mathbf{R}^n \longrightarrow \mathbf{R}$  continue et coercive. Alors le problème  $(\mathcal{P})$  admet au moins une solution.

**Preuve :** Soit  $(x_p)$  une suite minimisante telle que  $\lim_{p \rightarrow +\infty} J(x_p) = d = \inf(\mathcal{P})$ ;  $d < +\infty$ . Pour montrer que cette suite est bornée, raisonnons par l'absurde : si ce n'était pas le cas, on pourrait en extraire une sous-suite (encore notée  $(x_p)$ ) telle que  $\lim_{p \rightarrow +\infty} \|x_p\| = +\infty$ . Par coercivité de  $J$ , on aurait  $\lim_{p \rightarrow +\infty} J(x_p) = +\infty$  ce qui, d'après l'hypothèse, est absurde.

Comme  $(x_p)$  est bornée, on peut alors en extraire une sous-suite encore notée  $(x_p)$  qui converge vers  $x^* \in \mathbf{R}^n$ . Par continuité de  $J$ , on a alors :

$$d = \lim_{p \rightarrow +\infty} J(x_p) = J(x^*) > -\infty \quad .$$

Le point  $x^*$  est alors solution du problème  $(\mathcal{P})$ .

**Théorème 2.1.2** (Unicité). Soit  $J : \mathbf{R}^n \longrightarrow \mathbf{R}$  strictement convexe. Alors le problème  $(\mathcal{P})$  admet au plus une solution.

**Preuve :** Supposons que  $J$  admette au moins un minimum  $m$  réalisé en  $x_1 \neq x_2 : J(x_1) = J(x_2) = m$ . Comme  $J$  est strictement convexe, on a :  $J(\frac{x_1+x_2}{2}) < \frac{1}{2}J(x_1) + \frac{1}{2}J(x_2) = m$ . Ce qui contredit le fait que  $m$  est minimum. Donc nécessairement,  $x_1 = x_2$ .

**Théorème 2.1.3.** Soit  $J$  une fonction de classe  $C^1$  de  $\mathbf{R}^n$  dans  $\mathbf{R}$ . On suppose qu'il existe  $\alpha > 0$  tel que

$$\forall x, y \in \mathbf{R}^n, \quad \langle \nabla J(x) - \nabla J(y), x - y \rangle \geq \alpha \|x - y\|^2 \quad (2.1.1)$$

Alors  $J$  est strictement convexe et coercive ; en particulier le problème  $(\mathcal{P})$  admet une solution unique. On dit que  $J$  est elliptique et  $\alpha$  est appelée constante d'ellipticité.

**Preuve :** D'après le théorème 1.3.1 du chapitre 1, la condition (2.1.1) implique que  $J$  est strictement convexe. Pour montrer que  $J$  est coercive, il suffit d'appliquer la formule de Taylor avec reste intégral.

Les conditions que nous donnons dans la suite sont des conditions différentielles qui portent sur la dérivée de la fonction à minimiser.

**Théorème 2.1.4** (Condition nécessaire d'optimalité du premier ordre). Soit  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  une fonction différentiable sur  $\mathbf{R}^n$ . Si  $x^*$  réalise un minimum (global ou local) de  $J$  sur  $\mathbf{R}^n$  alors  $\nabla J(x^*) = 0$ .

**Preuve :** Si  $x^*$  réalise un minimum de  $J$  sur  $\mathbf{R}^n$  alors

$$\forall x \in B(x^*, r) \quad J(x^*) \leq J(x).$$

Soit  $h \in \mathbf{R}^n$ ,  $h \neq 0$ . On peut trouver  $t_h = \frac{r}{\|h\|}$  tel que  $\forall t \in ]0, t_h[, x^* + th \in B(x^*, r)$  et donc  $J(x^*) \leq J(x^* + th)$ . Comme  $J$  est différentiable, nous avons

$$J(x^* + th) = J(x^*) + t \langle \nabla J(x^*), h \rangle + t\epsilon(t)$$

comme  $t > 0$ , nous obtenons après simplification par  $t$  et passage à la limite :

$$\forall h \in \mathbf{R}^n \quad \langle \nabla J(x^*), h \rangle \geq 0 \quad \Leftrightarrow \quad \nabla J(x^*) = 0.$$

**Définition 2.1.2.** Un point  $x^*$  de  $\mathbf{R}^n$  vérifiant  $\nabla J(x^*) = 0$  est appelé *point critique* ou *point stationnaire*. La relation  $\nabla J(x^*) = 0$  est appelée *équation d'Euler*.

*Exemple 2.1.3.* Penser au cas  $J(x) = x^3$  qui admet un point critique  $x = 0$  non extremum.

**Théorème 2.1.5** (CNS du premier ordre dans le cas convexe). Soit  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  une fonction différentiable et convexe sur  $\mathbf{R}^n$ . Un point  $x^*$  réalise un minimum global de  $J$  sur  $\mathbf{R}^n$  si et seulement si  $\nabla J(x^*) = 0$ .

**Preuve :** Montrons la condition suffisante. Soit  $x^* \in \mathbf{R}^n$  tel que  $\nabla J(x^*) = 0$ . Comme  $J$  est convexe, on a d'après la preuve du théorème 1.3.1 :

$$\forall x \in \mathbf{R}^n \quad J(x) \geq J(x^*) + \langle \nabla J(x^*), x - x^* \rangle = J(x^*).$$

**Théorème 2.1.6** (Condition nécessaire d'optimalité du second ordre). *On suppose que  $x^*$  est un minimum (local) de  $J$  et que  $J$  est deux fois différentiable sur  $\mathbf{R}^n$ . Alors*

1.  $\nabla J(x^*) = 0$  et
2.  $\forall x \in \mathbf{R}^n, \langle \nabla^2 J(x^*)x, x \rangle \geq 0$ . C'est-à-dire que la matrice Hessienne  $\nabla^2 J(x^*)$  de  $J$  en  $x^*$  est semi-définie positive (toutes les valeurs propre de  $\nabla^2 J(x^*)$  sont positives ou nulles).

**Preuve :** Soit  $x \in \mathbf{R}^n$ . Appliquons la formule de Taylor sur la fonction  $t \rightarrow J(x^* + tx)$ . Comme  $\nabla J(x^*) = 0$ , on obtient :

$$0 \leq J(x^* + tx) - J(x^*) = \frac{t^2}{2} \langle \nabla^2 J(x^*)x, x \rangle + t^2 \epsilon(t)$$

Après division par  $t^2$ , on fait tendre  $t$  vers 0 et nous avons le résultat voulu.

**Théorème 2.1.7** (Condition suffisante du second ordre). *Soit  $J$  deux fois dérivable sur  $\mathbf{R}^n$  vérifiant  $\nabla J(x^*) = 0$  et*

$$\exists \alpha > 0 \quad | \quad \forall x \in \mathbf{R}^n : \quad {}^t x \nabla^2 J(x^*) x \geq \alpha \|x\|^2 \quad (2.1.2)$$

*Alors la fonction  $J$  admet un minimum local strict en  $x^*$ .*

**Preuve :** On suppose que  $x^*$  n'est pas un minimum local strict, alors il existe une suite  $(x_n)$  telle que  $x_n \neq x^*$  et  $J(x_n) \leq J(x^*)$ . On peut donc écrire  $x_n = x^* + t_n y_n$  avec  $\|y_n\| = 1$  et  $t_n > 0$ . La suite  $(y_n)$  étant bornée, elle contient une sous-suite  $y_n$  convergente  $(y_m)$  convergente vers une certaine direction  $y$ ;  $\|y\| = 1$ . Nous avons donc

$$\begin{aligned} J(x^*) &\geq J(x_m) = J(x^* + t_m y_m) \\ &\geq J(x^*) + {}^t t_m \nabla J(x^*) y_m + \frac{1}{2} t_m^2 {}^t y_m \nabla^2 J(x^*) y_m + o(t_m^2). \end{aligned}$$

Ayant  $\nabla J(x^*) = 0$ , le passage à la limite donne  ${}^t y \nabla^2 J(x^*) y \leq 0$ ,  $y \neq 0$  ce qui contredit l'hypothèse.

**Remarque 2.1.1.** La condition suffisante exige que le hessien soit une matrice définie positive. La condition (2.1.2) est une condition d'ellipticité locale.

**Exemple 2.1.4.** Soit  $J(x_1, x_2) = 3x_1^2 - 4x_1^2 x_2 - x_2^2$ . Il est facile de voir que l'unique point critique de  $J$  est  $(0, 0)$ . En ce point la Hessienne est semi-définie positive : on ne peut pas conclure immédiatement sur la nature de ce point. Mais une étude locale, montre que ce point n'est pas un extréma de  $J$ . En effet,  $J(0, 0) = 0$ ,  $J(1, 0) = 3$  et  $J(0, 1) = -1$ .

**Exemple 2.1.5.** Soit  $J(x_1, x_2) = 3x_1^2 - 4x_1^2 x_2 + x_2^2$ . Nous remarquons que  $(0, 0)$  est un point critique de  $J$ . En ce point la Hessienne est définie positive :  $(0, 0)$  est donc un minimum de  $J$ .



## 2.2 Algorithmes déterministes

Dans tout ce qui suit,  $\epsilon$  est un réel positif (petit) donné qui représente la précision désirée.

### 2.2.1 Méthode de descente du gradient

L'algorithme du gradient fait partie d'une classe plus grande de méthodes numériques appelées *méthodes de descente* : on veut minimiser une fonction  $J$ , pour cela on se donne un point de départ arbitraire  $x_0$ . Pour construire l'itéré  $x_1$  suivant, il faut se rapprocher du minimum de  $J$ , c'est-à-dire  $J(x_1) < J(x_0)$ . On cherche alors  $x_1 = x_0 + r_0 d_0$  où  $r_0 > 0$  et  $d_0 \in \mathbf{R}^n$  non nul. En pratique, on cherche donc  $d_0$  et  $r_0$  pour que  $J(x_0 + r_0 d_0) < J(x_0)$ . Quand  $d_0$  existe, on dit que c'est une *direction de descente* et  $r_0$  est le *pas de descente*. Ils peuvent être fixe ou changer à chaque itération.

Soit  $J$  une fonction de classe  $C^1$  de  $\mathbf{R}^n$  dans  $\mathbf{R}$ . Une idée naturelle pour trouver une direction de descente est de considérer le développement de Taylor formel de la fonction  $J$  entre les deux itérés  $x_0$  et  $x_1 = x_0 + r_0 d_0$  avec  $r_0 > 0$  :

$$J(x_0 + r_0 d_0) = J(x_0) + r_0 \langle \nabla J(x_0), d_0 \rangle + o(r_0 d_0)$$

Comme nous voulons que  $J(x_0 + r_0 d_0) < J(x_0)$ , nous pouvons choisir en première approximation  $d_0 = -\nabla J(x_0)$ . Plus généralement, le développement de Taylor formel de la fonction  $J$  entre deux itérés  $x_k$  et  $x_{k+1} = x_k + r_k d_k$  avec  $r_k > 0$  :

$$J(x_k + r_k d_k) = J(x_k) + r_k \langle \nabla J(x_k), d_k \rangle + o(r_k d_k)$$

Comme nous voulons que  $J(x_k + r_k d_k) < J(x_k)$ , nous pouvons choisir en première approximation  $d_k = -\nabla J(x_k)$ . La méthode ainsi obtenue s'appelle *l'algorithme du gradient*.

*Algorithme 1* (Gradient).

- 
1.  $k := 0$  ; choix de  $x_0 \in \mathbf{R}^n$
  2. **Tant que**  $\|\nabla J(x_k)\| > \epsilon$  **Faire**

$$d_k := -\nabla J(x_k) ;$$

$$\text{Déterminer } r_k > 0 ;$$

$$x_{k+1} := x_k + r_k d_k ;$$

$$k := k + 1 ;$$
  - Fin Faire** ;
  3. Retourner  $x_k$
- 

*Remarque 2.2.1.* Lorsque le pas  $r_k$  est fixé pour tout  $k$ , nous parlons de méthode du gradient à pas fixe. Lorsque  $r_k = \min_{r>0} J(x_k - r \nabla J(x_k))$ , nous parlons de méthode du gradient à pas optimal.

Cette méthode a pour avantage d'être très facile à mettre en oeuvre. Malheureusement les conditions de convergences sont assez lourde et la méthode est en général assez lente.

**Théorème 2.2.1** (Critères de convergences). *Soit  $J$  une fonction de classe  $C^1$  de  $\mathbf{R}^n$  dans  $\mathbf{R}$ , coercive et strictement convexe. On suppose qu'il existe une constante  $M > 0$  telle que*

$$\forall (x, y) \in \mathbf{R}^n \times \mathbf{R}^n \quad \|\nabla J(x) - \nabla J(y)\| \leq M\|x - y\|. \quad (2.2.1)$$

**Preuve :** Voir [?]. La fonction  $J$  admet un minimum unique sur  $\mathbf{R}^n$  et ce minimum  $x^*$  est caractérisée par  $\nabla J(x^*) = 0$ , puisque  $J$  est strictement convexe. Sachant que le développement en série de Taylor avec reste intégral est donné par :

$$J(x_{k+1}) = J(x_k) + \langle \nabla J(x_k), x_{k+1} - x_k \rangle + \int_0^1 \langle \nabla J(x_k + t(x_{k+1} - x_k)) - \nabla J(x_k), x_{k+1} - x_k \rangle dt,$$

et comme  $x_{k+1} = x_k - r_k \nabla J(x_k)$ , on obtient avec (2.2.1) :

$$J(x_{k+1}) - J(x_k) \leq -\frac{1}{r_k} \|x_{k+1} - x_k\|^2 + \int_0^1 \|\nabla J(x_k + t(x_{k+1} - x_k)) - \nabla J(x_k)\| \|x_{k+1} - x_k\| dt.$$

Donc  $J(x_{k+1}) - J(x_k) \leq (-\frac{1}{r_k} + \frac{M}{2}) \|x_{k+1} - x_k\|^2$ . Si on choisit  $r_k$  dans un intervalle  $[\beta_1, \beta_2] \subset ]0, \frac{2}{M}[$ , nous obtenons alors

$$J(x_{k+1}) - J(x_k) \leq (-\frac{1}{\beta_2} + \frac{M}{2}) \|x_{k+1} - x_k\|^2.$$

La suite  $(J(x_k))$  est alors strictement décroissante et comme elle est minorée, elle est convergente. Cela entraîne d'une part que  $J(x_{k+1}) - J(x_k)$  tend vers 0 et d'autre part que la suite  $(x_k)$  est bornée (par coercivité). On peut donc extraire de la suite  $(x_k)$  une sous-suite convergente vers  $\bar{x}$ . De plus comme

$$\|x_{k+1} - x_k\|^2 \leq (-\frac{1}{\beta_2} + \frac{M}{2})^{-1} (J(x_{k+1}) - J(x_k)),$$

la suite  $(x_{k+1} - x_k)$  tend également vers 0. Par conséquent  $\nabla J(x_k) = \frac{x_{k+1} - x_k}{r_k}$  tend vers 0 ; Par continuité de  $\nabla J$ , on obtient  $\nabla J(\bar{x}) = 0$ . Donc  $\bar{x}$  est l'unique minimum  $x^*$  de  $J$  et toute la suite converge vers  $x^*$ .

*Remarque 2.2.2.* Si la fonction  $J$  est coercive strictement convexe (par exemple elliptique de constante d'ellipticité  $\alpha$ ) et de dérivée lipschitzienne de rapport  $M > 0$  avec un pas  $0 < r_k < \frac{2}{M}$ . Alors la méthode du gradient converge. Dans la pratique, on ne calcule pas  $\alpha$  et  $M$ , mais on fait plusieurs tests pour différentes valeurs.

*Exemple 2.2.1.* Appliquer la méthode du gradient pour résoudre le problème d'optimisation suivant, en prenant comme point de départ  $x^0 = (1, 1)$  :

$$\min_{(x_1, x_2) \in \mathbf{R}^2} \quad J(x_1, x_2) = x_1^2 + 4x_2^2 - 2x_1 - 12x_2 + 10 \quad .$$

## 2.2.2 Méthode de relaxation successive

L'algorithme précédent ne calcule pas en réalité les extréma d'une fonction, mais les points critiques qui vérifient la condition d'optimalité du premier ordre (équation d'Euler). La méthode de relaxation permet de ramener un problème de minimisation dans  $\mathbf{R}^n$  à un problème de minimisation dans  $\mathbf{R}$ .

*Algorithme 2* (relaxation).

- 
1.  $k := 0$  ;  
choix de  $X_0 = (x_1^0, x_2^0, \dots, x_n^0) \in \mathbf{R}^n$  et  $X_1 := X_0 + 2\epsilon$  ;
  2. **Tant que**  $\|X_{k+1} - X_k\| \geq \epsilon$  **Faire**  
  
 $k := k + 1$  ;  
 Calcul de  $X_{k+1} = (x_1^{k+1}, x_2^{k+1}, \dots, x_n^{k+1})$  :  
  
 pour  $i$  variant de 1 à  $n$ , on calcule la solution  $x_i^{k+1}$  de  
 $\min J(x_1^{k+1}, x_2^{k+1}, \dots, x_{i-1}^{k+1}, x, x_{i+1}^k, \dots, x_n^k) ; x \in \mathbf{R}$   
  
**Fin Tant que** ;
  3. Retourner  $X_k$  ;
- 

Cet algorithme est très simple mais on ne peut pas garantir sa convergence vers un point stationnaire. Si même il y a convergence, celle-ci est très lente. Les conditions de convergence de cette méthode sont analogues aux conditions de convergence de la méthode du gradient. Pour plus de détails et pour des exemples d'utilisation, on peut se référer à [?].

*Exemple 2.2.2.* Soit  $X_0 = (1, 5, 2)$  alors le calcul de  $X_1 = (x_1^1, x_2^1, x_3^1)$  se fait comme suit :

$$\begin{aligned} x_1^1 &= \min_{x \in \mathbf{R}} J(x, 5, 2) \\ x_2^1 &= \min_{x \in \mathbf{R}} J(x_1^1, x, 2) \\ x_3^1 &= \min_{x \in \mathbf{R}} J(x_1^1, x_2^1, x). \end{aligned}$$

Donc pour  $J(x_1, x_2, x_3) = x_1^2 + 4x_2^2 + 2x_3^2 - 2x_1 - 12x_2 - 7$ , le calcul se fait avec une méthode d'optimisation unidimensionnelle comme la méthode de Newton ou bien la méthode de dichotomie.

## 2.2.3 Méthode de Newton

La méthode de Newton est en réalité une méthode utilisée pour résoudre des équations non linéaires de la forme  $F(x) = 0$  où  $x \in \mathbf{R}^n$ . Elle est adaptée au calcul d'un point critique c'est-à-dire à la résolution de  $\nabla f(x) = 0$  où  $x \in \mathbf{R}^n$ . Soit  $J$  une fonction de classe  $C^2$  de  $\mathbf{R}^n$  dans  $\mathbf{R}$ .

*Algorithme 3* (Newton).

- 
1.  $k := 0$  ; choix de  $x_0$  dans un voisinage de  $x^*$
  2. **Tant que**  $\|\nabla J(x_k)\| \geq \epsilon$  **Faire**  

$$x_{k+1} := x_k - (\nabla^2 J(x_k))^{(-1)} \nabla J(x_k) ;$$

$$k := k + 1$$
**Fin Tant que**
  3. Retourner  $x_k$  ;
- 

*Remarque 2.2.3.* L'étape 2. revient à résoudre le système linéaire  $\nabla^2 J(x_k) \delta_k = \nabla J(x_k)$  puis à poser  $x_{k+1} = x_k - \delta_k$ .

L'inconvénient majeur de cette méthode est sa sensibilité au choix du point de départ : la convergence est locale. Ce point doit être choisi assez près de  $x^*$  : en pratique on essaie de s'approcher de la solution par une méthode de type gradient puis on applique la méthode de Newton.

*Remarque 2.2.4.* Lorsque  $x_0$  est suffisamment proche de  $x^*$ , la convergence de la suite  $(x_n)$  est d'ordre 2, c'est-à-dire que la suite  $\|\nabla J(x^*)\|$  converge vers 0 de façon quadratique : l'erreur est élevée au carré à chaque itération : concrètement, si  $e_k$  vaut  $10^{-2}$ , alors à l'étape  $k + 1$ , elle vaut  $10^{-4}$  et  $10^{-8}$  à l'étape suivante.

Les méthodes quasi-newtonienne remplacent la matrice hessienne par une approximation  $H_k$  qui vérifie la relation de la sécante :  $H_k(x_k - x_{k-1}) = \nabla J(x_k) - \nabla J(x_{k-1})$ .

## 2.2.4 Méthode du gradient conjugué

Cette méthode est très stable. Elle demande  $2n^3$  opérations dans le cas d'une matrice pleine et  $n$  itérations. Le nombre d'opérations diminue beaucoup dans le cas d'une matrice creuse.

*Algorithme 4* (Cas général).

1.  $k := 0$  ; choix de  $x_0 \in \mathbf{R}^n$  de  $\epsilon > 0$  et calcul de  $g_0 := \nabla J(x_0)$
2. **Tant que**  $\|g_k\| \geq \epsilon$  **Faire**
  - ◇  $w_k = \begin{cases} g_0 & \text{si } k = 0 \\ g_k + \alpha_k w_{k-1} & \text{si } k \geq 1 \end{cases} ; \alpha_k = -\frac{\|g_k\|^2}{\|g_{k-1}\|^2}$
  - ◇ **Si**  $\langle w_k, g_k \rangle \geq 0$  **Alors**  $w_k = g_k$  ; **Fin Si**
  - ◇ recherche d'un pas  $r_k$  approchant le minimum de  $J(x_k - r w_k)$  par  $\langle \nabla J(x_k - r w_k), w_k \rangle = 0$
  - ◇  $x_{k+1} := x_k - r_k w_k$  ;
  - ◇  $k := k + 1$
- Fin Tant que ;**
3. Retourner  $x_k$  ;

Nous adaptons la méthode du gradient conjugué à la résolution de système linéaire appliquée au calcul des points critiques d'une fonction quadratique. Cette méthode s'applique donc au cas particulier d'une fonction quadratique associée à une matrice  $A$  et à un vecteur  $b$  ; la recherche d'un point critique revient à résoudre le système  $Ax = b$ .

C'est une méthode de descente inspirée de la méthode du gradient. En effet, la direction de descente de la méthode du gradient  $g_k = Ax_k - b$  est corrigée de façon à ce que toutes les directions de descentes  $w_k$  obtenues soient  $A$ -conjuguées :  $w_k = g_k + \alpha_k w_{k-1}$  où  $\alpha_k$  et  $w_k$  sont tels que  $\langle w_k, Aw_{k-1} \rangle = 0$ .

*Algorithme 5* (gradient conjugué linéaire).

1.  $k := 0$  ; choix de  $x_0 \in \mathbf{R}^n$  et calcul de  $g_0 := Ax_0 - b$
2. **Tant que**  $\|g_k\| \geq \epsilon$  **Faire**
  - ◇  $w_k := \begin{cases} g_0 & \text{si } k = 0 \\ g_k + \alpha_k w_{k-1} & \text{si } k \geq 1 \end{cases} ; \alpha_k := -\frac{\langle g_k, Aw_{k-1} \rangle}{\langle Aw_{k-1}, w_{k-1} \rangle} ;$
  - ◇  $r_k := \frac{\langle g_k, w_k \rangle}{\langle Aw_k, w_k \rangle} ;$
  - ◇  $x_{k+1} := x_k - r_k w_k ;$
  - ◇  $g_{k+1} := Ax_{k+1} - b ;$
  - ◇  $k := k + 1 ;$
- Fin Tant que ;**
3. Retourner  $x_k$  ;

Dans le cas d'une forme quadratique avec une matrice associée  $A \in S_n$  définie positive, on atteint le minimum en au plus  $n$  itérations. On montre ceci par récurrence sur les égalités

suivantes :  $\langle g_k, g_j \rangle = \langle g_k, w_j \rangle = \langle w_k, Aw_j \rangle = 0$  pour  $0 \leq j < k$ . C'est-à-dire que les directions  $g_0, \dots, g_k$  sont 2 à 2 conjugués. Il ne peut y avoir  $n + 1$  directions conjugués de  $\mathbf{R}^n$ . Donc nécessairement si  $g_i \neq 0$  pour  $i = 0..n - 1$  on a  $g_n = 0$ .

**Théorème 2.2.2.** *La méthode du gradient conjugué trouve le minimum d'une fonction quadratique  $J$  associée à une matrice  $A$  de  $S_n(\mathbf{R})$  définie positive, en au plus  $n$  itérations.*

**Preuve :** Voir [?]. Si  $g_k = 0$  alors  $x_k = x^*$  est la solution de  $Ax = b$ . Pour  $k = 1$ , nous avons  $w_0 = g_0$  donc  $\langle g_1, w_0 \rangle = \langle Ax_1 - b, w_0 \rangle = \langle Ax_0 - b, w_0 \rangle - r_0 \langle Aw_0, w_0 \rangle$  et alors :  $\langle g_1, w_0 \rangle = \langle g_0, w_0 \rangle - r_0 \langle Aw_0, w_0 \rangle = 0$ . D'après la définition de  $r_0$ , ceci entraîne aussi

$$\langle g_1, g_0 \rangle = \langle g_1, w_0 \rangle = 0,$$

et

$$\langle w_1, Aw_0 \rangle = \langle g_1, Aw_0 \rangle + \alpha_0 \langle w_0, Aw_0 \rangle = 0.$$

d'après la définition de  $\alpha_0$ .

Nous faisons maintenant l'hypothèse de récurrence suivante :

$$\begin{cases} \langle g_k, g_j \rangle = 0 & \text{pour } 0 \leq j < k \\ \langle g_k, w_j \rangle = 0 & \text{pour } 0 \leq j < k \\ \langle w_k, Aw_j \rangle = 0 & \text{pour } 0 \leq j < k \end{cases}$$

Si  $g_k \neq 0$ , on peut construire l'algorithme au rang  $k + 1$  et obtenir  $(x_{k+1}, g_{k+1}, w_{k+1})$ . Par construction, on a vu que  $\langle g_{k+1}, w_k \rangle = 0$  et pour  $j < k$  :  $\langle g_{k+1}, w_j \rangle = \langle g_{k+1}, w_j \rangle - \langle g_k, w_j \rangle = -r_k \langle Aw_k, w_j \rangle = 0$ .

Pour  $j \leq k$ ,  $\langle g_{k+1}, g_j \rangle = \langle g_{k+1}, w_j \rangle - \alpha_j \langle g_{k+1}, w_{j-1} \rangle = 0$  car  $g_j = w_j - \alpha_j w_{j-1}$ .

Maintenant :  $w_{k+1} = g_{k+1} + \alpha_{k+1} w_k$ . Pour  $j < k$  :

$$\langle w_{k+1}, Aw_j \rangle = \langle g_{k+1}, Aw_j \rangle + \alpha_{k+1} \langle w_k, Aw_j \rangle = \langle g_{k+1}, Aw_j \rangle.$$

Comme  $g_{j+1} = g_j - r_j Aw_j$ , on obtient si  $r_j \neq 0$  :

$$\langle g_{k+1}, Aw_j \rangle = \frac{1}{r_j} \langle g_{k+1}, g_j - g_{j+1} \rangle = 0$$

Donc si  $r_j \neq 0$ ,  $\langle w_{k+1}, Aw_j \rangle = 0$  pour  $j < k$ . D'autre part,  $\langle w_{k+1}, Aw_k \rangle = 0$  par construction. Donc  $\langle w_{k+1}, Aw_j \rangle = 0$  pour  $j < k + 1$ . La récurrence est démontrée tant qu'on a  $r_j \neq 0$  et  $g_j \neq 0$ . Mais on a

$$\langle g_k, w_k \rangle = \langle g_k, g_k \rangle + \alpha_k \langle g_k, w_{k-1} \rangle = \|g_k\|^2,$$

et  $r_k = \frac{\langle g_k, w_k \rangle}{\langle Aw_k, w_k \rangle}$ . Donc  $r_k$  ne peut s'annuler que si  $g_k = 0$ , mais alors  $x_k = x^*$ . D'autre part,

$$\|w_k\|^2 = \|g_k\|^2 + \alpha_k^2 \|w_{k-1}\|^2.$$

Donc si  $g_k \neq 0$  alors  $w_k \neq 0$ . Par conséquent, si les vecteurs  $g_0, \dots, g_k$  sont non nuls il en est de même pour  $w_0, \dots, w_k$  et les  $k + 1$  directions  $g_0, \dots, g_k$  forment une famille orthogonale pour le produit scalaire usuel. Ces directions sont donc indépendantes. Par suite si  $g_0, \dots, g_{n-1}$  ne sont pas nuls on a nécessairement  $g_n = 0$  ce qui entraîne la convergence de l'algorithme à la  $n^{\text{e}}$  itération.

*Exercice 15.* Gradient conjugué : Programmer l'algorithme du gradient conjugué pour résoudre  $Ax = b$ ; comparer avec les procédures de MAPLE ou de MUPAD en terme de mémoire utilisée et de temps CPU.

## 2.3 Exercices sur la minimisation sans contraintes

*Exercice 16.* Montrer par un exemple que la condition  $\nabla J(x^*) = 0$  est une condition nécessaire mais pas suffisante.

*Exercice 17.* Les fonctions  $J$  suivantes sont-elles coercives :  $J(x) = x^3 + x + 1$ ,  $J(x) = (a, x) + b$ ,  $J(x) = 2x_1^2 + x_2 - 1$ ,  $J$  est une forme quadratique associée à une matrice symétrique définie positive.

*Exercice 18.* Chercher les dimensions d'un wagon rectangulaire non couvert telles que pour un volume donné  $V$ , la somme des aires des côtés et du plancher soit minimale.

*Exercice 19.* Soit la fonction  $f(x_1, x_2) = 3x_1^2 + 2x_1x_2 - x_2^2$ .

1. trouver, au point  $x_0 = (1, 1)$ , un vecteur unitaire  $v$  qui indique la direction du gradient
2. trouver si elles existent les valeurs  $\alpha_{max}$  et  $\alpha_{min}$  :  $\alpha_{max} = \max_{\alpha} f(x_0 + \alpha v)$  et  $\alpha_{min} = \min_{\alpha} f(x_0 + \alpha v)$ .

*Exercice 20.* Appliquer la méthode du gradient pour résoudre le problème suivant, en prenant comme point de départ  $x^0 = (1, 1)$  :

$$\min \quad f(x_1, x_2) = x_1^2 + x_2^2 - 3x_1 + 4x_2 + 2 \quad .$$

*Exercice 21.* Chercher les extrémums de la fonction  $f(x_1, x_2) = x_1^2 - x_1x_2 + \frac{1}{6}x_2^3$  sur  $\mathbf{R}^2$ .

*Exercice 22.* Caractériser les optima des fonctions suivantes définies sur  $\mathbf{R}^2$  :

1.  $f(x) = x_1^2 + x_2^2 - 12x_1 - 16x_2$
2.  $f(x) = 6x_1^2 - 8x_2^2 - 54$
3.  $f(x) = (-2x_1^2 - 11x_1 + 12)x_2 - x_1^2$
4.  $f(x) = x_1(x_1^2 + x_2^2) - (x_1^2 + x_2^2) - x_1 + 1$

*Exercice 23. Problème de gestion de stocks.* Soient  $p_1 = 52$  et  $p_2 = 44$  les prix respectifs de deux produits. Soient  $q_1$  et  $q_2$  les quantités respectives de ces produits. Le revenu issu de la vente est donc :  $R = p_1q_1 + p_2q_2$ . La fonction coût est :  $C = q_1^2 + q_1q_2 + q_2^2$  et le bénéfice réalisé est  $\Pi = R - C$ . Trouver les quantités  $q_1$  et  $q_2$  maximisant le bénéfice.

*Exercice 24.* Exercice 2.10 de [?] : On veut résoudre le système suivant par une méthode de gradient à paramètre optimal :

$$\begin{cases} \frac{1}{2}x = 0 \\ \frac{c}{2}y = 0 \end{cases} \quad \text{où } c \geq 1$$

1. Ecrire le système sous la forme  $Ax = b$  et calculer les valeurs propres de  $A$ .
2. Soit  $r$  le résidu :  $b - Ax$ . Calculer  $r$  et le paramètre  $\alpha$  correspondant à la minimisation sur  $\mathbf{R}$  de la fonction qui à  $\alpha$  associe  $J(x_k + \alpha r_k)$ .
3. Soit  $P_k$  le point de coordonnées  $x_k$  et  $y_k$ . Exprimer  $x_{k+1}$  et  $y_{k+1}$  en fonction de  $x_k$  et  $y_k$ .
4. Soit  $t_k = \frac{y_k}{x_k}$  la pente de la droite  $(OP_k)$ . Exprimer  $t_{k+2}$  en fonction de  $t_k$ . Interprétation géométrique. Conclusion ?

# Chapitre 3

## Minimisation avec contraintes

Dans ce chapitre nous nous intéressons au cas où le problème de minimisation comporte des contraintes : soit un sous-ensemble  $C$  non vide, fermé de  $\mathbf{R}^n$  et soit le problème de la minimisation d'une fonction objectif sous des contraintes :

$$(\mathcal{P}) = \begin{cases} \min J(x) \\ x \in C \end{cases}$$

où  $J : \mathbf{R}^n \longrightarrow \mathbf{R}$  une fonction de plusieurs variables  $x = (x_1, \dots, x_n)$  à valeurs réelles.

### 3.1 Résultats fondamentaux

#### 3.1.1 Résultats d'existence et d'unicité

**Théorème 3.1.1** (Existence). *Supposons que  $J$  est continue et que l'une des conditions suivantes est réalisée :*

1. *Soit  $C$  est borné*
2. *Soit  $J$  est coercive*

*Alors le problème  $(\mathcal{P})$  admet au moins une solution.*

**Preuve :** La démonstration est la même que dans le cas sans contraintes : nous montrons qu'une suite minimisante est bornée soit parce qu'elle est dans un borné  $C$  soit parce que  $J$  est coercive. La limite de la sous-suite extraite est dans  $C$  fermé. C'est donc une solution de  $(\mathcal{P})$ .

**Théorème 3.1.2** (Unicité). *Sous les hypothèse du théorème 3.1.1, si  $J$  est strictement convexe et si  $C$  est convexe, alors le problème  $(\mathcal{P})$  admet une solution unique\*.*

**Preuve :** Supposons que  $J$  admet au moins un minimum  $m$  réalisé en  $x_1 \neq x_2 : J(x_1) = J(x_2) = m$ . Comme  $J$  est strictement convexe, on a :  $J(\frac{x_1+x_2}{2}) < \frac{1}{2}J(x_1) + \frac{1}{2}J(x_2) = m$ . Ce qui contredit le fait que  $m$  est minimum. Donc nécessairement,  $x_1 = x_2$ .

*Exemple 3.1.1.* Les conditions du théorème 3.1.2 sont satisfaites dans le cas où  $J$  est une fonction quadratique associée à une matrice définie positive et  $C$  est convexe.



### 3.1.2 Conditions d'optimalité

**Théorème 3.1.3** (CN du premier ordre). *Soient  $J$  une fonction convexe et  $C$  un ensemble convexe fermé. Supposons que  $J$  est différentiable sur  $C$ . Un point  $x^* \in C$  est solution de  $(\mathcal{P})$  si*

$$\forall x \in C \quad \langle \nabla J(x^*), x - x^* \rangle \geq 0 \quad .$$

**Preuve :** Soient  $x^* \in C$  une solution de  $(\mathcal{P})$  et  $x \in C$ . Comme  $C$  est convexe nous avons  $x^* + t(x - x^*) \in C$  pour tout  $t \in [0, 1]$ . Donc  $J(x^* + t(x - x^*)) - J(x^*) \geq 0$ . En divisant par  $t$  et en faisant tendre  $t$  vers  $0^+$ , nous obtenons le résultat.

**Théorème 3.1.4** (CNS du premier ordre dans le cas convexe). *Soient  $J$  une fonction convexe et différentiable,  $C$  un ensemble convexe fermé. Un point  $x^* \in C$  est solution de  $(\mathcal{P})$  si et seulement si*

$$\forall x \in C \quad \langle \nabla J(x^*), x - x^* \rangle \geq 0 \quad . \quad (3.1.1)$$

**Preuve :** Montrons la condition suffisante d'optimalité : soit  $x^* \in C$  et  $J$  une fonction convexe. Alors pour tout  $x \in C$  :  $J(x) \geq J(x^*) + \langle \nabla J(x^*), x - x^* \rangle$ . Donc  $J(x) \geq J(x^*)$  pour tout  $x \in C$ .

*Remarque 3.1.1.* Si  $C = \mathbf{R}^n$ , la condition (3.1.1) est équivalente à l'équation d'Euler. Dans le cas général, cette relation reste abstraite, nous allons donc la détailler pour le cas où  $C$  est défini par (3.1.2), c'est-à-dire le cas de contraintes en égalités et en inégalités.

Nous traitons particulièrement le cas où  $C$  est défini par des égalités et des inégalités

$$C = \{x \in \mathbf{R}^n \mid g(x) \leq 0, h(x) = 0\} \quad , \quad (3.1.2)$$

avec  $g : \mathbf{R}^n \rightarrow \mathbf{R}^p$  une fonction à  $p$  composantes et on peut écrire

$$g(x) = (g_1(x), \dots, g_p(x)), \quad x \in \mathbf{R}^n$$

où les  $g_i(x) : \mathbf{R}^n \rightarrow \mathbf{R}$  représentent les contraintes en inégalités ;  $g(x) \leq 0$  si et seulement si  $g_i(x) \leq 0$  pour tout  $i = 1..p$ .

et  $h : \mathbf{R}^n \rightarrow \mathbf{R}^q$  une fonction à  $q$  composantes et on peut écrire

$$h(x) = (h_1(x), \dots, h_q(x)), \quad x \in \mathbf{R}^n$$

où les  $h_j(x) : \mathbf{R}^n \rightarrow \mathbf{R}$  représentent les contraintes en égalités ;  $h(x) = 0$  si et seulement si  $h_j(x) = 0$  pour tout  $j = 1..q$ .

Rappelons que l'ensemble  $C$  défini par (3.1.2) est convexe si les  $g_i$  sont convexes pour  $i = 1..p$  et les  $h_j$  sont affines pour  $j = 1..q$ .

**Définition 3.1.1** (Contrainte active ou saturé). Une contrainte en inégalité  $g_i$  est *active* ou *saturé* au point  $x^* \in \mathbf{R}^n$  si  $g_i(x^*) = 0$ . On note  $I(x^*) = \{i \in 1..p \mid g_i(x^*) = 0\}$  l'ensemble des indices  $i$  correspondant aux contraintes actives en  $x^*$ .

**Définition 3.1.2** (Qualification des contraintes). Un point  $x^* \in \mathbf{R}^n$  vérifie les conditions de qualification des contraintes pour les contraintes  $h$  et  $g$ ,

1. s'il est réalisable :  $h(x^*) = 0$  et  $g(x^*) \leq 0$ , ( $x^* \in C$ )
2. si les vecteurs  $\nabla h_j(x^*)$  et  $\nabla g_i(x^*)$  sont linéairement indépendants pour  $j = 1..q$  et  $i \in I(x^*)$ .

On dit aussi que  $x^*$  est un point régulier.

**Théorème 3.1.5** (Conditions de Karush-Kuhn-Tucker). *On suppose que  $J$ ,  $g$  et  $h$  de classe  $C^1$ . Soit  $x^*$  une solution du problème  $(\mathcal{P})$ . On suppose que  $x^*$  est régulier pour les contraintes  $h$  et  $g$ . Alors il existe  $\lambda^* \in \mathbf{R}^p$  et  $\mu^* \in \mathbf{R}^q$  tels que  $x^* \in C$  et*

$$\nabla J(x^*) + \sum_{i=1}^{i=p} \lambda_i^* \nabla h_i(x^*) + \sum_{j=1}^{j=q} \mu_j^* \nabla g_j(x^*) = 0 \quad (3.1.3)$$

$$\forall j = 1..q \quad \mu_j^* g_j(x^*) = 0 \quad \mu_j^* \geq 0 \quad (3.1.4)$$

**Définition 3.1.3** (lagrangien). On appelle lagrangien du problème  $(\mathcal{P})$  la fonction définie sur  $\mathbf{R}^n \times \mathbf{R}^p \times \mathbf{R}^q$  par

$$\mathcal{L}(x, \lambda, \mu) = J(x) + \sum_{i=1}^{i=p} \lambda_i h_i(x) + \sum_{j=1}^{j=q} \mu_j g_j(x) \quad .$$

La relation (3.1.3) s'écrit alors

$$\nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = 0$$

où  $\nabla_x$  désigne le gradient par rapport à la première variable.

*Exemple 3.1.2* (Régression linéaire avec contraintes). Considérons un nuage de  $n$  points de  $\mathbf{R}^2$  :  $M_i = (t_i, x_i)$ ,  $1 \leq i \leq n$ . On cherche la droite de régression  $x = at + b$  mais on impose par exemple  $b \geq 0$ . On veut donc trouver un couple de réels  $(a, b)$  solution de

$$\min \quad J(a, b) = \sum_{i=1}^{i=n} (x_i - at_i - b)^2 \quad , \quad b \geq 0$$

La fonction  $J$  est strictement convexe, coercive et  $C = \{(a, b) \in \mathbf{R}^2 \mid -b \leq 0\}$  est convexe ; ce problème a donc une solution unique.

On pose  $g(a, b) = -b$ , comme  $\nabla g(a, b) = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$  n'est pas nul, tout point est régulier.

Reste à écrire les équations de KKT : il existe  $\lambda \geq 0$  tel que

$$\lambda \cdot b = 0 \quad , \quad b \geq 0 \quad , \quad \nabla J(a, b) + \lambda \begin{pmatrix} 0 \\ -1 \end{pmatrix} = 0$$

*Exercice 25.* Minimiser la fonction  $J(x) = 2x_1^2 + 2x_1x_2 + x_2^2 - 10x_1 - 10x_2$  sous les contraintes  $g_1$  et  $g_2$  avec  $g_1(x) := x_1^2 + x_2^2 \leq 5$  et  $g_2(x) := 3x_1 + x_2 \leq 6$ .

*Indication :* on peut remarquer que  $J$ ,  $g_1$  et  $g_2$  sont convexes.

1. Ecrire les conditions de Kuhn et Tucker
2. Vérifier qu'il y a quatre cas possible : montrer que la seule solution possible de ce problème d'optimisation avec contraintes est  $x^* = (1, 2)$ .

**Théorème 3.1.6** (CNS dans le cas convexe). *Si  $J$ ,  $h$  et  $g$  sont de classe  $C^1$  et on suppose que  $J$  et  $g$  sont convexes,  $h$  affine et  $x^*$  régulier pour les contraintes  $h$  et  $g$  (vérifie les conditions de qualification des contraintes). Le point  $x^*$  est solution du problème  $(\mathcal{P})$  si et seulement si les conditions KKT sont satisfaites.*

Notons  $I^+(x^*) = \{i, 1 \leq i \leq p \mid g_i(x^*) = 0 \text{ et } \mu_i^* > 0\}$  l'ensemble des contraintes fortement actives. Notons

$$T(x^*) = \{d \in \mathbf{R}^n \mid \langle \nabla h_j(x^*), d \rangle = 0, j = 1..q, \langle \nabla g_i(x^*), d \rangle = 0, i \in I(x^*)\}$$

le sous-espace tangent de  $x^*$  à  $C$  et

$$C(x^*) = \{d \in \mathbf{R}^n \mid \langle \nabla h_j(x^*), d \rangle = 0, j = 1..q, \langle \nabla g_i(x^*), d \rangle = 0, i \in I^+(x^*)\}$$

le sous-espace critique de  $C$  en  $x^*$ .

**Théorème 3.1.7** (CS du second ordre). *On suppose que  $J$ ,  $h$  et  $g$  sont de classe  $C^2$ ; soit  $x^*$  un point de  $\mathbf{R}^n$  vérifiant les conditions nécessaires de KKT avec  $(\lambda^*, \mu^*)$ . Si la matrice hessienne  $\nabla_{xx}^2 \mathcal{L}$  du lagrangien au point  $(x^*, \lambda^*, \mu^*)$  est définie positive sur le sous-espace critique  $C(x^*)$  alors  $x^*$  est un minimum (local) strict de  $J$  sur  $C$ .*

Rappelons qu'un minimum local  $x^*$  est strict si pour toute autre solution  $x$  admissible dans un voisinage de  $x^*$ , nous avons  $x \in B(x^*) : J(x^*) \leq J(x)$ .

*Exemple 3.1.3.* Etudier, d'un point de vue graphique puis numérique, le problème d'optimisation avec contraintes suivant :

$$(\mathcal{P}) = \begin{cases} \min (x_1 - 3)^2 + (x_2 - 4)^2 \\ x \in C \end{cases}$$

où  $C = \{x \in \mathbf{R}^2 \mid x_1 \geq 0, x_2 \geq 0, 5 - x_1 - x_2 \geq 0, -2.5 + x_1 - x_2 \leq 0\}$ .

## 3.2 Algorithmes classiques

Nous présentons dans cette section les principaux algorithmes rencontrés dans la littérature sur la programmation non linéaire avec contraintes. La méthode du gradient projeté et la méthode de pénalisation n'utilisent pas le lagrangien contrairement à la méthode de Newton-Lagrange qui s'applique au cas de contraintes d'égalités. D'autres méthodes, comme la méthode SQP, utilisent le lagrangien qui joue alors le rôle de fonction objectif.

### 3.2.1 Méthode du gradient projeté

Soit  $C$  un sous-ensemble de  $\mathbf{R}^n$  convexe et fermé et soit  $x$  un point de  $\mathbf{R}^n$  n'appartenant pas à  $C$ . Alors le problème

$$\min ||x - y||^2 = \sum (x_i - y_i)^2, y \in C$$

admet une solution unique  $x^* = \pi_C(x) \in C$ . De plus,  $\pi_C(x)$  est caractérisé par

$$\forall y \in C \quad \langle x - \pi_C(x), y - \pi_C(x) \rangle \leq 0$$

En effet, la projection sur un convexe fermé de  $\mathbf{R}^n$  est un problème de moindres carrés. La fonction coût est continue, coercive et strictement convexe, on applique alors le théorème 3.1.2 et le théorème 3.1.4. On définit la fonction distance d'un point  $x$  à l'ensemble  $C$  par

$$d(x, C) = \|x - \pi_C(x)\|.$$

*Algorithme 6* (Gradient projeté).

- 
1.  $k := 0$ ;  
choix de  $x_0$  et de  $r_0 > 0$ ;
  2. **Tant que**  $\|x_{k+1} - x_k\| > \epsilon$  **Faire**  
 $x_{k+1} := \pi_C(x_k - r_k \nabla J(x_k))$ ;  
 $k := k + 1$   
**Fin Tant que**
  3. Retourner  $x_k$ ;
- 

**Théorème 3.2.1.** Soit  $J$  une fonction de classe  $C^1$  de  $\mathbf{R}^n$  dans  $\mathbf{R}$ . On suppose que  $J$  est elliptique de constante  $\alpha$  et de dérivée lipchitzienne de rapport  $M$ . Alors, si on choisit le pas  $0 < r_k < \frac{2\alpha}{M}$ , la suite  $x_n$  définie par la méthode du gradient projeté converge vers la solution du problème  $(\mathcal{P})$ .

*Exemple 3.2.1.* Lorsque  $C = \{x \in \mathbf{R}^n \mid x_i \geq 0, i = 1..n\}$  est l'orthant positif de  $\mathbf{R}^n$ , la projection sur  $C$  est définie par  $\pi_C(x) = (x_1^+, \dots, x_n^+)$  où  $x_i^+ = \max(0, x_i)$  est la partie positive du réel  $x_i$ .

*Exemple 3.2.2.* Si  $C = \{x \in \mathbf{R}^n \mid a_i \leq x_i \leq b_i, i = 1..n\}$  où  $a_i \leq b_i$  pour  $i = 1..n$ . Alors  $\pi_C(x) = (\hat{x}_1, \dots, \hat{x}_n)$  où

$$\hat{x}_i = \begin{cases} x_i & \text{si } a_i \leq x_i \leq b_i \\ a_i & \text{si } x_i < a_i \\ b_i & \text{si } x_i > b_i \end{cases}$$

Malgré son apparente simplicité, cette méthode est difficile à mettre en oeuvre. En effet, à chaque étape, il faut calculer le projeté d'un vecteur de  $\mathbf{R}^n$  sur  $C$ . Lorsque  $C$  est simple (voir exemple 3.2.1), c'est faisable. Dès que les contraintes ne sont pas des contraintes de borne, le calcul de la projection devient délicat.

### 3.2.2 Méthode de pénalisation

On remplace le problème  $(\mathcal{P})$  par un problème sans contraintes

$$(\mathcal{P}_r) = \min J(x) + r\alpha(x), \quad x \in \mathbf{R}^n$$

où  $\alpha : \mathbf{R}^n \rightarrow \mathbf{R}$  est une fonction de *pénalisation des contraintes* et  $r > 0$ . Le but est de trouver des fonctions  $\alpha$  telles que les problèmes  $(\mathcal{P})$  et  $(\mathcal{P}_r)$  soient équivalents. On parle alors de pénalisation *exacte*.

En général, on effectue une pénalisation *inexacte* où l'ensemble des solutions de  $(\mathcal{P}_r)$  ne couvre pas toutes les solutions de  $(\mathcal{P})$ . On prend par exemple  $\alpha : \mathbf{R}^n \rightarrow \mathbf{R}$  continue sur  $\mathbf{R}^n$ , positive ou nulle et vérifiant  $\alpha(x) = 0 \Leftrightarrow x \in C$ . Par exemple, pour la contrainte d'égalité  $h(x) = 0$ , on prend  $\alpha(x) = \|h(x)\|^2$ ; pour la contrainte d'inégalité  $g(x) \leq 0$ , on prend  $\alpha(x) = \|g(x)^+\|^2$  où  $g(x)^+ = (g_1(x)^+, \dots, g_n(x)^+)$  avec  $g_i(x)^+ = \max(0, g_i(x))$ .

*Algorithme 7* (Pénalisation extérieure).

---

1. **Initialisation**

$k = 1$ ; choix de  $x_0$  et de  $r_1 > 0$

2. **Itération  $k$**

Résoudre le sous-problème

$$(\mathcal{P}_{r_k}) \min J(x) + r_k \alpha(x), x \in \mathbf{R}^n$$

en prenant  $x_{k-1}$  comme point de départ.

3. **Critère d'arrêt**

Si  $x_k$  est satisfaisant, STOP

Sinon, on pose  $k := k + 1$ , on choisit  $r_{k+1} > r_k$  et on retourne à 2.

---

Le théorème suivant nous assure la convergence de la méthode de pénalisation :

**Théorème 3.2.2.** *Nous supposons que la fonction  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  est continue, coercive. Soit  $C$  un ensemble fermé non vide de  $\mathbf{R}^n$ . Supposons que  $\alpha$  est continue sur  $\mathbf{R}^n$ , positive ou nulle et vérifiant  $\alpha(x) = 0 \Leftrightarrow x \in C$ . Alors :*

- $\forall r > 0$ ,  $(\mathcal{P}_r)$  a au moins une solution  $x_r$ ,
- La famille  $(x_r)_{r>0}$  est bornée,
- Toute sous-suite convergente extraite de  $(x_r)_{r>0}$  converge vers une solution de  $(\mathcal{P})$  lorsque  $r$  tend vers  $+\infty$ .

**Preuve :** Soit  $x^*$  une solution du problème  $(\mathcal{P})$ . Nous montrons, grâce à la coercivité et à la continuité de  $J_r = J + r\alpha$ , que le problème de minimisation sans contraintes  $(\mathcal{P}_r)$  admet au moins une solution  $x_r$ . Cette solution vérifie :

$$J(x_r) \leq J_r(x_r) = J(x_r) + r\alpha(x_r) \leq J_r(x^*) = J(x^*)$$

Donc  $J(x_r)$  est uniformément bornée par rapport à  $r > 0$  et par coercivité de  $J$ ,  $x_r$  est bornée. On peut donc en extraire une sous-suite convergente vers  $\bar{x}$ . On en déduit que

$$0 \leq \alpha(x_{r_k}) \leq \frac{1}{r_k} (J(x^*) - J(x_{r_k})).$$

En faisant tendre  $k$  vers  $+\infty$ , on obtient que  $\alpha(\bar{x}) = 0$  et donc que  $\bar{x} \in C$ . Nous obtenons donc  $J(\bar{x}) \leq J(x^*)$ . Donc  $\bar{x}$  est solution de  $(\mathcal{P})$ .

### 3.2.3 Méthode de Lagrange-Newton

Cet algorithme s'applique au cas de contraintes en égalité. Si nous nous plaçons dans le cadre de la programmation quadratique avec contraintes en égalité affines

$$(\mathcal{P}) = \begin{cases} \min J(x) = \frac{1}{2} \langle Ax, x \rangle - \langle b, x \rangle + c \\ \text{s/c } Qx = d \end{cases}$$

avec  $A$  une matrice carré d'ordre  $n$ ,  $b \in \mathbf{R}^n$ ,  $c \in \mathbf{R}$ ,  $Q$  une matrice  $q \times n$  et  $d \in \mathbf{R}^q$ . Les conditions de KKT sont

$$\begin{cases} \nabla_x(J(x) + {}^t\lambda(Qx - d)) = 0 \\ Qx = d \end{cases}$$

où  $\lambda \in \mathbf{R}^q$  est le multiplicateur de Lagrange associé à la contrainte  $Qx = d$ . Le couple optimal  $(x^*, \lambda^*)$  est solution du système linéaire :

$$\begin{pmatrix} A & {}^tQ \\ Q & 0 \end{pmatrix} \begin{pmatrix} x^* \\ \lambda^* \end{pmatrix} = \begin{pmatrix} b \\ d \end{pmatrix}$$

Si la matrice du système est inversible alors ce système admet une solution unique que l'on peut calculer par une méthode d'analyse numérique matricielle.

Dans le cas général, on considère le problème d'optimisation suivant  $\min J(x) \text{ s/c } h(x) = 0$  où  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  et  $h : \mathbf{R}^n \rightarrow \mathbf{R}^q$  sont suffisamment régulière. Les conditions du premier ordre s'écrivent

$$\begin{cases} \nabla_x \mathcal{L}(x, \lambda) = 0 \\ h(x) = 0 \end{cases}$$

où  $\lambda \in \mathbf{R}^q$  et  $\mathcal{L}(x, \lambda) = J(x) + \lambda h(x)$  est le lagrangien du problème. On peut résoudre ce système par la méthode de Newton ce qui donne :

*Algorithme 8* (Lagrange-Newton).

1. **Initialisation**

$k = 0$ ; choix de  $(x_0, \lambda_0) \in \mathbf{R}^n \times \mathbf{R}^q$

2. **Itération**  $k$  : on connaît  $(x_k, \lambda_k)$

Résoudre

$$\nabla_{xx}^2 \mathcal{L}(x_k, \lambda_k) d_k + \nabla h(x_k) y_k = -\nabla_x \mathcal{L}(x_k, \lambda_k)$$

et  $\nabla h(x_k) d_k = -h(x_k)$

$$x_{k+1} := x_k + d_k;$$

$$\lambda_{k+1} = \lambda_k + y_k;$$

3. **Critère d'arrêt**

Si  $x_k$  est satisfaisant, STOP

Sinon, on pose  $k := k + 1$  et on retourne à 2.

### 3.2.4 Méthode de programmation quadratique successive

Si on linéarise les contraintes et on fait une approximation quadratique de  $\mathcal{L}$ , nous transformons un problème d'optimisation en un problème de programmation quadratique de  $\mathcal{L}$ . En effet, une solution de  $\mathcal{P}$  est un point critique du lagrangien du problème. nous pouvons mettre au point une méthode de descente qui consiste à résoudre à chaque étape un problème quadratique avec contraintes linéaires (ces problèmes sont des approximation du problème  $\mathcal{P}$ ). On appelle cette méthode la programmation quadratique successive ou SQP.

Etant donné  $x_k$ , on cherche  $x_{k+1} = x_k + r_k d_k$  où  $d_k$  est une direction de descente et  $r_k > 0$ . Une approximation des contraintes par la formule de Taylor donne la direction de descente :  $h(x_{k+1}) = h(x_k) + Jh(x_k).d_k = 0$  et  $g(x_{k+1}) = g(x_k) + Jg(x_k).d_k \leq 0$  (en négligeant les termes d'ordre supérieur ou égal à 2, nous effectuons une linéarisation des contraintes au voisinage de  $x_k$ ). D'autre part, il faudrait que  $x_{k+1}$  diminue la valeur du lagrangien : nous cherchons donc un point critique du lagrangien en effectuant une approximation d'ordre 2 :

$$\mathcal{L}(x_k + d, \lambda, \mu) = \nabla_x \mathcal{L}(x_k + d, \lambda, \mu) + \frac{1}{2} \langle \nabla_{xx}^2 \mathcal{L}(x_k + d, \lambda, \mu) d, d \rangle + O(\|d\|^3)$$

En négligeant les termes d'ordre supérieur ou égal à 3, on voit qu'il faut minimiser

$$\nabla_x \mathcal{L}(x_k + d, \lambda, \mu) + \frac{1}{2} \langle \nabla_{xx}^2 \mathcal{L}(x_k + d, \lambda, \mu) d, d \rangle .$$

Il reste à déterminer le pas  $r_k$ , le multiplicateur de Lagrange  $\lambda$  et le multiplicateur positif  $\mu$  à chaque itération :

*Algorithme 9 (SQP).*

---

1. **Initialisation**

$k = 0$  ; choix de  $(x_0, \lambda_0, \mu_0) \in \mathbf{R}^n \times \mathbf{R}^q \times \mathbf{R}_+^p$

2. **Itération**  $k$  : on connaît  $(x_k, \lambda_k, \mu_k)$

Résoudre le sous-problème quadratique en  $d^*$  solution de :

$$\begin{aligned} \min \quad & \frac{1}{2} \langle \nabla_{xx}^2 \mathcal{L}(x_k, \lambda_k, \mu_k) d, d \rangle + \langle \nabla_x \mathcal{L}(x_k, \lambda_k, \mu_k), d \rangle \\ \text{avec} \quad & Jh(x_k)d + h(x_k) = 0, \\ \text{et} \quad & Jg(x_k)d + g(x_k) \leq 0. \end{aligned}$$

$d_k := d^*$  ;

$\lambda_{k+1}$  est un multiplicateur de Lagrange associé à la contrainte d'égalité ;

$\mu_{k+1}$  est un multiplicateur positif associé à la contrainte d'inégalité ;

$x_{k+1} := x_k + r_k d_k$  ; (on peut choisir  $r_k := 1$ )

3. **Critère d'arrêt**

Si  $x_k$  est satisfaisant, STOP

Sinon, on pose  $k := k + 1$  et on retourne à 2.

---

### 3.3 Exercices sur la minimisation avec contraintes

*Exercice 26.* Trouver les extremums relatifs de la fonction  $J(v_1, v_2) = v_2^2$  par rapport à l'ensemble

$$U = \{v = (v_1, v_2) \in \mathbf{R}^2; v_1^2 + v_2^2 - 1 = 0\}.$$

*Exercice 27.* Une agence publicitaire doit concevoir une affiche ayant une surface totale de  $0.8m^2$ . Le client désire que la surface totale imprimée soit maximale, tout en conservant des marges de  $2.5cm$  en haut et en bas et de  $1cm$  de chaque côté. Quelles doivent être les dimensions de l'affiche ?

*Exercice 28.* Une usine de contenants a reçu une commande de boîtes rectangulaires ouvertes de  $1 m^3$ . Le coût de fabrication estimé est de  $0.600D$  le mètre carré pour la base,  $0.450D$  le mètre carré pour le devant et l'arrière et  $0.41D$  le  $m^2$  pour les deux autres côtés. Quelle doivent être les dimensions de la boîte afin de minimiser le coût de production ?

*Exercice 29.* Trouver les extremums relatifs de la fonction  $J(v_1, v_2) = v_1 + (v_2 - 1)^2$  par rapport à l'ensemble

$$U = \{v = (v_1, v_2) \in \mathbf{R}^2; v_1^2 = 0\}.$$

*Exercice 30.* Minimiser  $J(u) = u_1 u_2 u_3$  sous la contrainte  $u_1 + u_2 + u_3 - 3 = 0$ .

*Exercice 31.* On désire construire la plus grande boîte possible de surface donnée  $S$ . Déterminer  $L$ ,  $l$  et  $h$  : la longueur, la largeur et la hauteur de cette boîte.

*Exercice 32.* Maximiser la fonction  $J(x, y) = 14x - x^2 + 6y - y^2 + 7$  sous les contraintes  $x + y \leq 2$  et  $x + 2y \leq 3$ . On précisera l'existence et l'unicité et on dessinera l'ensemble réalisable  $S$ .

*Exercice 33.* Résoudre

$$\min \frac{1}{2}(Ax, x) - (b, x)$$

sous les contraintes  $x_1 \geq 1$  et  $x_2 - 2x_3 = 1$  avec  $A = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 3 \end{pmatrix}$  et  $b = \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix}$ .

Comparer avec la solution sans contraintes.

*Exercice 34.* On désire minimiser le coût de production de deux unités de la même compagnie. Ces unités produisent respectivement des camions et des voitures, à l'aide de machines-outils, ressource rare dans la compagnie. Le problème est modélisé de la façon suivante : Minimiser  $J_1(u_1) + J_2(u_2)$  sous la contrainte de la consommation  $g_1(u_1) + g_2(u_2) = c$  où  $c$  le nombre total de machines-outils disponibles dans la compagnie.

1. Enoncer les conditions d'optimalité
2. Poser  $c = 9$ ,  $g_1(u_1) = u_1$ ,  $g_2(u_2) = u_2$ ,  $J_1(u_1) = (u_1 - 10)^2$  et  $J_2(u_2) = 2(u_2 - 5)^2$ .
3. Supposons que l'on ait la possibilité de louer des machines-outils au prix unitaire  $q = 4$ , on veut savoir à quel prix la location est intéressante et combien de machines-outils il faudrait louer.

*Exercice 35.* Minimiser  $f(x_1, x_2) = x_1^2 + x_2^2 - 6x_1 - 8x_2$  sous les contraintes  $g_1(x) = 4x_1^2 + x_2^2 - 16$  et  $g_2(x) = 3x_1 + 5x_2 - 15$ .

*Exercice 36.* Minimiser  $\frac{1}{2}(u_1 - 1)^2 + \frac{1}{2}(u_2 - 2)^2$  sous les contraintes  $u_1 - u_2 = 1$ ,  $u_1 + u_2 \leq 2$ ,  $-u_1 \leq 0$  et  $-u_2 \leq 0$ .

*Exercice 37.* Soit le problème d'optimisation suivant : Minimiser  $f(x) = x_1^2 + x_2^2$  sous la contrainte  $h(x) = (x_1 - 1)^3 - x_2^2 = 0$ .



1. Résoudre ce problème graphiquement.
2. Montrer qu'il n'existe pas de multiplicateurs de Lagrange pour cet optimum.
3. Expliquez ce résultat

*Exercice 38.* Trouver les optima locaux sur  $\mathbf{R}^2$  par la règle des multiplicateurs de Lagrange :

1.  $y = x_1^2 + x_2^2$  sous la contrainte  $x_1 x_2 = 2$
2.  $y = x_1 x_2$  sous la contrainte  $x_1^2 + x_2^2 = 2$ .

*Exercice 39.* Vérifier, pour chacun des problèmes suivants, qu'au point  $x^*$  donné, les conditions suffisantes d'optimum sont satisfaites :

1. Min  $f(x) = x_1^2 + 4x_2^2$  s/c  $x_1 + 2x_2 = 6$  ;  $x^* = (3, \frac{3}{2})$
2. Min  $f(x) = x_1 x_2$  s/c  $x_1^2 + x_2^2 = 1$  ;  $x^* = (-\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}})$ .

### 3.4 Dualisation de problème de minimisation convexe

Soit le problème de la minimisation d'une fonction objectif sous des contraintes exprimées par des égalités et des inégalités :

$$(\mathcal{P}) = \begin{cases} \min & J(x) \\ \text{s/c} & h(x) = 0 \\ & g(x) \leq 0 \end{cases}$$

où  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  une fonction de plusieurs variables  $x = (x_1, \dots, x_n)$  à valeurs réelles et avec  $g : \mathbf{R}^n \rightarrow \mathbf{R}^p$  une fonction à  $p$  composantes :  $g(x) = (g_1(x), \dots, g_p(x))$ ,  $x \in \mathbf{R}^n$  où les  $g_i(x) : \mathbf{R}^n \rightarrow \mathbf{R}$  représentent les contraintes en inégalités ;  $g(x) \leq 0$  si et seulement si  $g_i(x) \leq 0$  pour tout  $i = 1..p$ . Les contraintes en égalités sont données par  $h : \mathbf{R}^n \rightarrow \mathbf{R}^q$  une fonction à  $q$  composantes et on peut écrire  $h(x) = (h_1(x), \dots, h_q(x))$ ,  $x \in \mathbf{R}^n$  où les  $h_j(x) : \mathbf{R}^n \rightarrow \mathbf{R}$  représentent les contraintes en égalités ;  $h(x) = 0$  si et seulement si  $h_j(x) = 0$  pour tout  $j = 1..q$ . Notons  $C = \{x \in \mathbf{R}^n \mid h(x) = 0, g(x) \leq 0\}$ . Le problème  $(\mathcal{P})$  est appelé problème primal par opposition au problème dual que nous allons maintenant introduire.

#### 3.4.1 Problème dual

Soit  $\Phi$  la fonction de  $\mathbf{R}^n$  dans  $\mathbf{R}$  définie par  $\Phi(x) = \max_{\lambda, \mu \geq 0} \langle h(x), \lambda \rangle + \langle g(x), \mu \rangle$ .

Alors  $\Phi(x) = \begin{cases} 0 & \text{si } x \in C \\ +\infty & \text{sinon} \end{cases}$  En effet, si  $x \in C$  alors  $g(x) \leq 0$  et le max est atteint pour  $\mu = 0$  donc  $\Phi(x) = 0$ . D'autre part, si  $x$  n'est pas dans  $C$  alors il existe  $i$  tel que  $g_i(x) > 0$  et  $\langle g_i, \lambda_i \rangle$  peut devenir très grand en faisant tendre  $\lambda_i$  vers  $+\infty$ .

La fonction  $\Phi$  est appelée indicatrice de  $C$  et le problème  $(\mathcal{P})$  est équivalent à

$$\begin{cases} \min & f(x) + \max_{\lambda, \mu \geq 0} \langle h(x), \lambda \rangle + \langle g(x), \mu \rangle \\ & x \in \mathbf{R}^n \end{cases}$$

Ce qui, en utilisant le lagrangien, donne  $\min_{x \in \mathbf{R}^n} \max_{\lambda, \mu \geq 0} \mathcal{L}(x, \lambda, \mu)$ .

**Définition 3.4.1.** On appelle problème dual de  $(\mathcal{P})$  la problème

$$\left\{ \max_{\lambda, \mu \geq 0} \min_{x \in \mathbf{R}^n} \mathcal{L}(x, \lambda, \mu) \right.$$

On appelle  $\omega(\lambda, \mu) = \min_{x \in \mathbf{R}^n} \mathcal{L}(x, \lambda, \mu)$  la fonction duale.

La fonction duale  $\omega(\lambda, \mu)$  est concave. De plus,  $\omega(\lambda, \mu) \leq \min_{x \in \mathbf{R}^n} \max_{\lambda, \mu \geq 0} \mathcal{L}(x, \lambda, \mu)$ . On a donc  $\max_{\lambda, \mu \geq 0} \omega(\lambda, \mu) \leq \min_{x \in \mathbf{R}^n} \max_{\lambda, \mu \geq 0} \mathcal{L}(x, \lambda, \mu)$ . Si l'on note que, par construction,  $\min_{x \in \mathbf{R}^n} \max_{\lambda, \mu \geq 0} \mathcal{L}(x, \lambda, \mu) = f(x^*)$  où  $x^*$  est la solution du problème primal, on a donc

$$\max_{\lambda, \mu \geq 0} \omega(\lambda, \mu) \leq f(x^*)$$

### 3.4.2 Point-selle du lagrangien

Un couple  $(x^*, (\lambda^*, \mu^*))$  est appelé point-selle (ou point-col) du lagrangien sur  $\mathbf{R}^n \times (\mathbf{R}^q \times \mathbf{R}_+^p)$  lorsque

$$\mathcal{L}(x^*, (\lambda, \mu)) \leq \mathcal{L}(x^*, (\lambda^*, \mu^*)) \leq \mathcal{L}(x, (\lambda^*, \mu^*)) \quad .$$

La valeur  $\mathcal{L}(x^*, (\lambda^*, \mu^*))$  est appelée valeur-selle.

**Théorème 3.4.1.** Les points-selles de  $\mathcal{L}$  sur  $\mathbf{R}^n \times (\mathbf{R}^q \times \mathbf{R}_+^p)$  sont exactement les points  $(x^*, (\lambda^*, \mu^*))$  tels que :

1.  $x^*$  minimise  $x \mapsto \mathcal{L}(x, (\lambda^*, \mu^*))$  ;
2.  $x^* \in C$  ;
3.  $\mu_j^* g_j(x^*) = 0$  pour tout  $j = 1..p$ .

En particulier, si  $(x^*, (\lambda^*, \mu^*))$  est un point-selle de  $\mathcal{L}$  sur  $\mathbf{R}^n \times (\mathbf{R}^q \times \mathbf{R}_+^p)$  alors  $x^*$  est une solution de  $(\mathcal{P})$  et  $(\lambda^*, \mu^*)$  sont les multiplicateurs de Kuhn et Tucker associé.

**Théorème 3.4.2.** Si  $\mathcal{P}$  est un problème d'optimisation convexe alors  $(x^*, (\lambda^*, \mu^*))$  est un point-selle de  $\mathcal{L}$  si et seulement si  $x^*$  est une solution de  $(\mathcal{P})$  et  $(\lambda^*, \mu^*)$  sont les multiplicateurs de Lagrange solutions du système de Karush Kuhn et Tucker associé à  $x^*$ .

Notons que si  $f$  est strictement convexe, si les  $g_i$  sont convexes et les  $h_j$  affines, et si  $C$  est non vide alors l'existence de  $x^*$  est équivalente à celle de  $(\lambda^*, \mu^*)$  et on a

$$\omega(\lambda^*, \mu^*) = \mathcal{L}(x^*, \lambda^*, \mu^*) = f(x^*)$$

### 3.4.3 Méthode d'Uzawa

Le théorème 3.4.2 indique que nous allons chercher  $(x^*, (\lambda^*, \mu^*)) \in \mathbf{R}^n \times \mathbf{R}^q \times \mathbf{R}_+^p$  vérifiant les conditions de KKT de la manière suivante :

1. Pour  $(\lambda^*, \mu^*)$  fixés dans  $\mathbf{R}^q \times \mathbf{R}_+^p$ , nous cherchons le minimum sans contraintes de la fonction  $x \mapsto \mathcal{L}(x, \lambda^*, \mu^*)$  ;
2. Pour  $x^* \in \mathbf{R}^n$  fixé, on cherche le maximum sur  $\mathbf{R}^q \times \mathbf{R}_+^p$  de la fonction  $(\lambda, \mu) \mapsto \mathcal{L}(x^*, \lambda, \mu)$ . C'est un problème de maximisation avec des contraintes de bornes simples.

Le principe de la méthode d'Uzawa est d'utiliser la méthode du gradient pour maximiser la fonction duale, tout en tenant compte de la contrainte  $\mu \geq 0$ . Cela donne  $\mu_{k+1} = (\mu_k + r \nabla \omega(\mu_k))^+$ . Si le minimum en  $x$  du lagrangien est unique, on note  $x(\lambda, \mu)$  le vecteur tel que

$$\omega(\lambda, \mu) = \mathcal{L}(x(\lambda, \mu), \lambda, \mu)$$

On peut écrire que  $\nabla \omega(\lambda, \mu) = g(x(\lambda, \mu))$ .

*Algorithme 10* (Algorithme d'Uzawa).

---

1. **Initialisation**

$k := 0$ ; choix de  $(\lambda^0, \mu^0) \in \mathbf{R}^q \times \mathbf{R}_+^p$

2. **Itération**  $k$  : on connaît  $(\lambda^k, \mu^k)$  et on fixe  $r > 0$ ;

on calcule  $x_k$  solution de :

$$\min \mathcal{L}(x, \lambda^k, \mu^k) \text{ pour } x \in \mathbf{R}^n.$$

on calcule  $(\lambda^{k+1}, \mu^{k+1})$  avec

$$\lambda_i^{k+1} = \lambda_i^k + r h_i(x_k) \text{ pour } i = 1..q.$$

$$\mu_j^{k+1} = \max(0, \mu_j^k + r g_j(x_k)) \text{ pour } j = 1..p.$$

3. **Critère d'arrêt**

Si  $\max_i g_i(x_k) < \epsilon$  alors STOP

Sinon, on pose  $k := k + 1$  et on retourne à 2.

---

# Chapitre 4

## Application aux réseaux de neurones

Un réseau de neurones artificiel est une structure composée d'entités (les neurones) capables de calcul interagissant entre eux. Il permet de traiter, par le biais de l'informatique, des problèmes de différentes natures que les outils classiques ont du mal à résoudre. Les explications et autres définitions sont tirés du polycopé de Marie Cottrel de l'Université de la Sorbonne<sup>1</sup>.

### 4.1 Historique

Les recherches sur le cerveau ont fait d'énormes progrès, tant du point de vue de la compréhension du fonctionnement qu'à l'échelle cellulaire et membranaire. On sait désormais mesurer et enregistrer l'activité électrique d'un neurone ou d'un ensemble de neurones. Le cerveau humain dispose de l'ordre de  $10^{11}$  avec plus de  $10^{15}$  connexions. Il y a des propriétés collectives des neurones qui agissent en relations excitatrices ou inhibitrices. On distingue dans le cerveau des sous-ensembles, des connexions internes et externes, des entrées et des sorties, pour réaliser certaines tâches (parole, vision, émotion,...).

Les années 1940 sont connues pour l'apprentissage du codage et de la recherche opérationnelle : A. Turing avait mis en place la machine de Turing, premier ordinateur conçu pour déchiffrer le code Enigma. Depuis, le besoin de proposer des machines ou des modèles reproduisant des fonctions des cerveaux humains ont été développés. Il s'agit d'outils de calcul, de mémoire pour reproduire des comportements intelligents. Des chercheurs dans différents domaines ont travaillé sur l'intelligence artificielle.

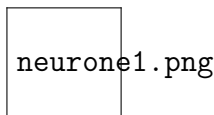
Le tableau suivant donne brièvement un historique des réseaux de neurones artificiels :

1940 - 1960 concepts	Mac Culloch & Pitts 43 Hebb Rosenblatt 58	modèle de neurone loi d'adaptation perceptron
1960 - 1980 transition et déclin	Widrow-Hoff 58 Minsky & Papert 69	adaline limites aux perceptron
1980 - ... renouveau	Hopfield 82 Kohonen 72, 82, 84 Rumelhart, Le Cun 86 Werbos 74	réseaux dynamiques auto-organisation rétro-propagation

1. Marie Cottrel, Les réseaux de neurones historique, méthodes et applications SAMOS-MATISSE, Université Paris 1, Sorbonne

## 4.2 Définitions

Un neurone biologique est une cellule nerveuse dont la fonction est de transmettre, dans certaines conditions, un signal électrique (0.01s). C'est comme un relai entre une couche de neurones et celle qui la suit. Le corps



Le corps d'un neurone est relié d'une part par un ensemble de *dendrites* (entrées du neurone) et d'autre part à un *axone*, qui est la partie étirée de la cellule. L'axone représente pour nous la sortie du neurone. Les charges électriques, reçues par les *connexions synaptiques* (en amont), s'accumulent dans le noyau du neurone jusqu'à atteindre un certain *seuil*. Au delà de ce seuil, la transmission du signal électrique se déclenche via son axone vers d'autres neurones (en aval). On remarque que les liaisons axone-dendrite entre deux neurones (connexions synaptiques) ne sont pas toutes de la même efficacité. Ainsi l'entrée associée à une certaine dendrite du neurone pourra avoir plus d'influence qu'une autre sur la valeur de sortie. On peut représenter la qualité de la liaison par un *poids*, sorte de coefficient s'appliquant au signal d'entrée. Le poids sera d'autant plus grand que la liaison est bonne. Un poids négatif aura tendance à inhiber une entrée, tandis qu'un poids positif viendra l'accentuer<sup>2</sup>.

### 4.2.1 Neurone formel

Le neurone formel reproduit le neurone biologique. Il doit être capable de recevoir, en entrée, différentes informations provenant des neurones environnants. Il doit analyser et ajuster ces informations, de manière à envoyer en sortie une réponse. On assimile donc à un neurone un quadruplet (poids  $w_1, w_n$ , biais  $w_0$ , fonction d'activation  $f$ , seuil  $\theta$ ) :

$$f(x_1, \dots, x_n) = \begin{cases} 1 & \text{si } w_0 + w_1x_1 + \dots + w_nx_n > \theta \\ -1 & \text{sinon.} \end{cases}$$

On utilise aussi une notation plus simple :



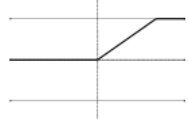
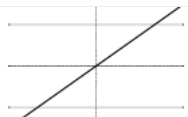
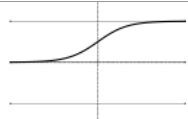
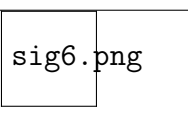
$$f(\vec{x}) = \begin{cases} 1 & \text{si } \vec{w} \cdot \vec{x} > \theta \\ -1 & \text{sinon.} \end{cases}$$

### 4.2.2 Fonctions d'activation

Il y a plusieurs exemples de fonctions d'activation utilisées pour les réseaux de neurones. Elle ressemblent le plus souvent à des fonctions *Tout ou rien* :

---

2. Texte et dessin d'Adrien Vergé, Réseaux de neurones artificiels, Lycée Michelet, TIPE 2009.

Nom	Valeur	Représentation
Seuil	$f(x) = 0$ si $x \leq 0$ et $f(x) = 1$ sinon	 sig1.png
Seuil symétrique	$f(x) = -1$ si $x < 0$ $f(0) = 0$ $f(x) = 1$ si $x > 0$	
Linéaire saturée	$f(x) = 0$ si $x < 0$ $f(x) = x$ pour $0 \leq x \leq 1$ $f(x) = 1$ si $x > 1$	
Linéaire	$f(x) = x$	
Sigmoïde	$f(x) = \frac{1}{1 + e^{-x}}$	
Tangente hyperbolique	$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	 sig6.png

### 4.2.3 Perceptron multi-couches

On souhaite classer des images de 4x6 pixels dans deux catégories, selon des critères dépendant de la couleur et de la disposition de ces pixels. La première couche de notre réseau devra comporter 24 entrées, une entrée pour la valeur de chaque pixel. La couche de sortie fournira deux valeurs selon la catégorie. Le classement des images se fera selon les «affinités» : on placera l'image dans la catégorie avec laquelle l'affinité est grande.

Chaque neurone comportant plusieurs entrées et une seule sortie, on peut d'une façon générale, superposer dans une même couche, la couche 1 qui a  $s_1$  neurones formels à  $e_1$  entrées. On obtient alors un système admettant  $e_1$  entrées et  $s_1$  sorties. Puis, lorsqu'on concatène  $N$  couches connectées les unes à la suite des autres (le nombre d'entrées d'une couche étant égal au nombre de sorties de la couche précédente :  $s_{n-1} = e_n$ ), on obtient un réseau de  $N$  couches de neurones.

## 4.3 Fonctionnement du réseau de neurones

Les applications des réseaux neuronaux vont de la reconnaissance vocale, en passant par la classification d'image ou de la prédiction financière. Les réseaux de neurones sont bien indiqués lorsqu'il s'agit de modéliser des structures complexes et des données irrégulières. En particulier quand les relations entre les variables sont non linéaires. Ils requièrent un pré-traitement sur les données aberrantes auxquelles ils sont sensibles. Pour les PMC, les données doivent être normalisées. La particularité du réseau de neurones comme outil ma-

thématique repose sur sa capacité d'apprentissage. Grâce à un processus d'entraînement, on peut améliorer les performances du réseau, c'est-à-dire qu'en réponse à une certaine entrée, il fournit la bonne sortie. Il s'agira de modifier les poids des liaisons en réaction aux entrées qu'on lui soumet, de manière à ce que la sortie corresponde au résultat désiré. Le réseau s'améliore en réduisant l'erreur qu'il commet à chaque simulation. On utilise pour cela un algorithme d'apprentissage. Il y a deux types d'algorithme d'apprentissage :

- Les apprentissages supervisés : on fournit un grand nombre de couples  $(x, c)$  où  $x$  est l'entrée et  $c$  est la sortie désirée. La correction s'effectue selon l'erreur obtenue pour chaque couple. Dans le cas d'un perceptron, la correction de l'erreur se fait comme suit :

$$w_i \leftarrow w_i + \Delta w_i$$

avec

$$\Delta w_i = \eta(t - c)x_i$$

où

$t = f(\vec{x})$  est la valeur cible

$c$  est la sortie désirée du perceptron

$\eta$  une constante de petite taille (e.g., .1) appelée *taux d'apprentissage*

On prouve la convergence lorsque les données d'apprentissage sont linéairement séparables et  $\eta$  suffisamment petit. On parle en général d'apprentissage par l'exemple.

- Les apprentissages non supervisés : ne connaissant pas la sortie désirée pour une entrée donnée, on n'a pas de signal d'erreur le réseau apprendra par lui-même à classer des entrées similaires en trouvant des points communs aux stimuli appliqués.

La correction de l'erreur en réaction à des stimuli en entrée constitue le processus d'apprentissage

To understand, consider simpler *linear unit*, where

$$o = w_0 + w_1x_1 + \dots + w_nx_n$$

Let's learn  $w_i$ 's that minimize the squared error

$$E[\vec{w}] \equiv \frac{1}{2} \sum_{d \in D} (t_d - o_d)^2$$

Where  $D$  is set of training examples.

$$\nabla E[\vec{w}] \equiv \left[ \frac{\partial E}{\partial w_0}, \frac{\partial E}{\partial w_1}, \dots, \frac{\partial E}{\partial w_n} \right]$$

Training rule :

$$\Delta \vec{w} = -\eta \nabla E[\vec{w}]$$

i.e.,

$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i}$$

$$\begin{aligned}
\frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2} \sum_d (t_d - o_d)^2 \\
&= \frac{1}{2} \sum_d \frac{\partial}{\partial w_i} (t_d - o_d)^2 \\
&= \frac{1}{2} \sum_d 2(t_d - o_d) \frac{\partial}{\partial w_i} (t_d - o_d) \\
&= \sum_d (t_d - o_d) \frac{\partial}{\partial w_i} (t_d - \vec{w} \cdot \vec{x}_d) \\
\frac{\partial E}{\partial w_i} &= \sum_d (t_d - o_d) (-x_{i,d})
\end{aligned}$$

### Gradient-Descent(*training\_examples*, $\eta$ )

Each training example is a pair of the form  $\langle \vec{x}, t \rangle$ , where  $\vec{x}$  is the vector of input values, and  $t$  is the target output value.  $\eta$  is the learning rate (e.g., .05).

- Initialize each  $w_i$  to some small random value
- Until the termination condition is met, Do
  - Initialize each  $\Delta w_i$  to zero.
  - For each  $\langle \vec{x}, t \rangle$  in *training\_examples*, Do
    - Input the instance  $\vec{x}$  to the unit and compute the output  $o$
    - For each linear unit weight  $w_i$ , Do

$$\Delta w_i \leftarrow \Delta w_i + \eta(t - o)x_i$$

- For each linear unit weight  $w_i$ , Do

$$w_i \leftarrow w_i + \Delta w_i$$

$\sigma(x)$  is the sigmoid function

$$\frac{1}{1 + e^{-x}}$$

Nice property :  $\frac{d\sigma(x)}{dx} = \sigma(x)(1 - \sigma(x))$

L'erreur :

$$\begin{aligned}
\frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2} \sum_{d \in D} (t_d - o_d)^2 \\
&= \frac{1}{2} \sum_d \frac{\partial}{\partial w_i} (t_d - o_d)^2 \\
&= \frac{1}{2} \sum_d 2(t_d - o_d) \frac{\partial}{\partial w_i} (t_d - o_d) \\
&= \sum_d (t_d - o_d) \left( -\frac{\partial o_d}{\partial w_i} \right) \\
&= - \sum_d (t_d - o_d) \frac{\partial o_d}{\partial net_d} \frac{\partial net_d}{\partial w_i}
\end{aligned}$$



But we know :

$$\frac{\partial o_d}{\partial net_d} = \frac{\partial \sigma(net_d)}{\partial net_d} = o_d(1 - o_d)$$

$$\frac{\partial net_d}{\partial w_i} = \frac{\partial (\vec{w} \cdot \vec{x}_d)}{\partial w_i} = x_{i,d}$$

So :

$$\frac{\partial E}{\partial w_i} = - \sum_{d \in D} (t_d - o_d) o_d (1 - o_d) x_{i,d}$$

Backpropagation Algorithm :

Initialize all weights to small random numbers.

Until satisfied, Do

— For each training example, Do

1. Input the training example to the network and compute the network outputs
2. For each output unit  $k$

$$\delta_k \leftarrow o_k(1 - o_k)(t_k - o_k)$$

3. For each hidden unit  $h$

$$\delta_h \leftarrow o_h(1 - o_h) \sum_{k \in \text{outputs}} w_{h,k} \delta_k$$

4. Update each network weight  $w_{i,j}$

$$w_{i,j} \leftarrow w_{i,j} + \Delta w_{i,j}$$

where

$$\Delta w_{i,j} = \eta \delta_j x_{i,j}$$

La faiblesse des réseaux de neurones réside dans le caractère non explicite des résultats. La convergence n'est pas forcément vers une solution optimale globale. Le choix peu guidé de la structure du PMC (nombre de couches, nombre de neurones par couches). Sensible à un très grand nombre de variables discriminantes.

Input	Output
10000000	→ 10000000
01000000	→ 01000000
00100000	→ 00100000
00010000	→ 00010000
00001000	→ 00001000
00000100	→ 00000100
00000010	→ 00000010
00000001	→ 00000001

Learned hidden layer representation :

Input		Hidden Values				Output
10000000	→	.89	.04	.08	→	10000000
01000000	→	.01	.11	.88	→	01000000
00100000	→	.01	.97	.27	→	00100000
00010000	→	.99	.97	.71	→	00010000
00001000	→	.03	.05	.02	→	00001000
00000100	→	.22	.99	.99	→	00000100
00000010	→	.80	.01	.98	→	00000010
00000001	→	.60	.94	.01	→	00000001

## 4.4 Statistique et PMC

Supposons que  $X_1, \dots, X_n$  soient des valeurs observées et que l'on cherche  $\phi(X_1, \dots, X_n)$  telle que

$$E(X_{n+1}|X_1, \dots, X_n) = \phi(X_1, \dots, X_n) .$$

Toute fonction  $\phi(x_1, \dots, x_n)$  suffisamment régulière peut être approchée par une fonction de la forme :

$$g\left(\sum_{i=1}^n a_i x_i + b_i\right) .$$

Ce qu'il faut chercher est l'estimation statistique  $\hat{g}(X_1, \dots, X_n)$  de  $g$  s'exprimant sous la forme :

$$\hat{g} = g\left(\sum_{i=1}^n \hat{a}_i x_i + \hat{b}_i\right)$$

où, pour  $i \in [1, n]$ , les valeurs  $\hat{a}_i$  et  $\hat{b}_i$  sont respectivement des valeurs approchées des  $a_i$  et  $b_i$ . (Voir [2] et [3] pour une vitesse d'approximation).

Ce type de problème classique de la statistique est résolu dans des cas très généraux. L'intérêt des réseaux neuronaux intervient dans les cas de grandes dimensions.

Nous avons  $x_1, \dots, x_n$  des variables observées et nous cherchons  $x_{n+1}$ . Pour  $i = 1, \dots, n+1$ , notons  $X_i$  la variable aléatoire de la  $i$ -ième valeur. Posons  $X = (X_1, \dots, X_n)$  et  $Y = (X_{n+1})$ . Pour approcher  $Y$ , nous cherchons à construire un réseau qui approxime au mieux l'espérance  $E[Y|X]$  par une fonction  $f_W(X)$ , où  $W$  est une matrice de poids synaptiques.

La couche d'entrée comportera  $n$  neurones (pour  $X$ ) et la couche de sortie 1 neurone (pour  $f_W(X)$ ).

Qui dit estimer, sous-entend choisir une fonction d'erreur  $\pi(Y, f_W(X))$  entre la sortie obtenue  $f_W(x)$  avec  $x = (x_1, \dots, x_n)$  et celle désirée  $Y = (x_{n+1})$ .

--

Une fois cette fonction d'erreur choisie, nous désirons qu'elle soit aussi petite que possible. Donc nous cherchons  $f_W$  qui minimise au mieux l'espérance  $E[\pi(Y, f_W(X))]$  de la fonction d'erreur  $\pi$ .

La question est de savoir si approcher  $Y$  permet de trouver  $E[Y|X]$ . Cela dépend de la fonction d'erreur choisie.

**Exemple 2.1.** Choisissons  $\pi(Y, f_W(X)) = (Y - f_W(X))^2$ . Avec cette fonction d'erreur, nous calculons de manière classique

$$E[(Y - f_W(X))^2|X] = E[(Y - E[Y|X])^2|X] + E[(E[Y|X] - f_W(X))^2|X].$$

(l'erreur totale est la somme de l'erreur de prédiction et de l'erreur statistique de prédiction). Donc minimiser l'erreur entre la sortie du réseau  $f_W(X)$  et la valeur désirée  $Y$  revient à minimiser celle entre  $f_W(X)$  et l'espérance  $E[Y|X]$ , ce qu'on cherche à estimer.

D'autres fonctions d'erreur entre la sortie du réseau  $f_W(X)$  et la valeur désirée  $Y$  partagent cette propriété d'*estimation optimale*. Par exemple, la *Cross Entropy* souvent employée dans les PMC :

$$Y \log(f_W(X)) + (1 - Y) \log(1 - f_W(X)).$$

Comme nous l'avons étudié dans le PMC, nous savons que les poids optimaux sont obtenus en minimisant l'espérance d'erreur entre les calculs du réseau et les valeurs désirées obtenues à partir d'un corpus.

## 4.5 Logiciels utilisés

### 4.5.1 nnet de R pour le perceptron multi-couches

### 4.5.2 Proc Neural pour le perceptron de SAS

### 4.5.3 Sage, Python

## 4.6 conclusion

Un réseau de neurones est avant tout un outil de traitement de l'information. À force d'apprentissage, on peut le former à des tâches de reconnaissance, classification, approximation, prévision...

On utilise par exemple des réseaux de neurones pour réduire l'écho dans les communications téléphoniques outre-mer (réseau Adaline). Ils peuvent aussi servir pour l'élimination du bruit, les prévisions météorologiques, la compression de données, la modélisation du marché monétaire, le développement d'Intelligence Artificielle...

Grâce à leur capacité d'apprentissage, les réseaux de neurones fournissent de très bons résultats pour des tâches de reconnaissance. Il existe de nombreuses applications dans le traitement du son et de l'image, par exemple dans des logiciels de reconnaissance optique de caractères. C'est parce qu'un tel programme est assez facile à mettre en oeuvre que nous avons choisi de le développer et de le tester dans la troisième partie.

Une application qui a fait parler d'elle est le projet *NetTalk*, réseau de 309 neurones relié en entrée à un scanner et branché en sortie à un haut-parleur, qui a appris à lire un texte à haute voix ! Alors que les outils informatiques traditionnels sont inadaptés à ce genre de problème, les réseaux constituent une nouvelle approche prometteuse. Car une fois entraîné, un réseau est capable de lire des nouveaux mots qu'ils n'a jamais vu auparavant (grâce à sa capacité d'adaptation). Ainsi, l'entraînement de *NetTalk* qui a consisté à lui rabâcher un dictionnaire de 50 000 mots pendant toute une nuit, a donné de bons résultats : au matin il lisait sans faute 95% des mots du dictionnaire d'apprentissage et 75% des nouveaux mots.

Enfin, on peut citer des applications de réseaux faits de vrais neurones "animaux" cette fois-ci. Des chercheurs de l'Université de Reading en Angleterre ont réussi à développer une intelligence artificielle pour robot à l'aide de neurones de rats : mis en culture et connectés par des électrodes au système moteur de la machine, celle-ci apprend à éviter les murs à force de se cogner !

# Chapitre 5

## Exercices de révision

### 5.1 Correction de la série d'exercices numéro 1

*Exercice 40.* Problème de transport. Le but est d'optimiser la livraison de fournitures scolaires d'un entrepôt  $M$  vers deux clients  $N_1$  et  $N_2$ . L'entrepôt  $M$  dispose d'un stock de 3 tonnes de marchandises. Il faut livrer les 2 clients qui ont commandé chacun 1 tonne de fournitures scolaires. Le coût de transport unitaire entre l'entrepôt  $M$  et le client  $N_1$  (respectivement  $N_2$ ) est de 20 TND/tonne (respectivement 30 TND/tonne). On veut minimiser le coût du transport tout en satisfaisant les commandes des clients.

1. Donnez les contraintes de ce problème mathématique.
2. Donnez la fonction coût de ce problème mathématique.

La solution de ce problème de transport est donnée dans ce qui suit : on note  $v_{i,j}$  les quantités de marchandise partant de l'entrepôt  $i$  vers le client  $j$ .

1. La contrainte de limite de stocks et de satisfaction clients est égale à  $v_{1,1} + v_{1,2} \leq 3$
2. La fonction coût de ce problème mathématique est  $20v_{1,1} + 30v_{1,2}$  qu'on cherche à minimiser.

*Exercice 41.* (Exercice de Paul Dorbec). Une entreprise souhaite acheter des machines outils. Deux machines aux caractéristiques différentes peuvent lui servir. Les caractéristiques des machines sont résumées ci-dessous :

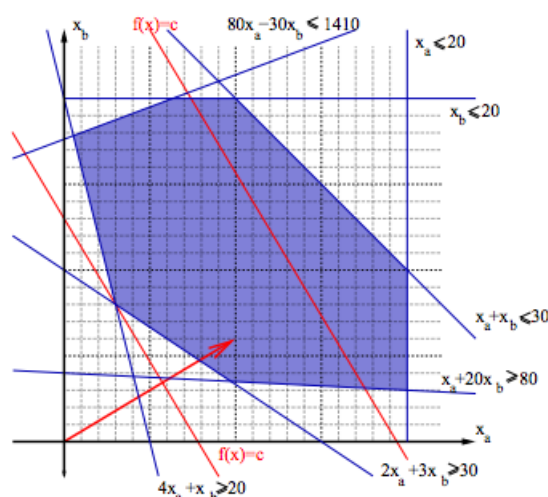
Type de machine	A	B
Coût à l'achat	14570	8742
Production pièce $\alpha$	2/h	3/h
Production pièce $\beta$	4/h	1/h
Production pièce $\gamma$	1/h	20/h
Consommation électrique	80kW/h	-30 kW/h

Voici les contraintes qui guident le choix de l'entreprise. L'entreprise n'a pas la surface pour entreposer plus de 30 machines. De plus, les fournisseurs n'ont que 20 machines de chaque type à vendre. L'entreprise commercialise 3 types de pièces, désignées par  $\alpha, \beta, \gamma$ . Selon les commandes, elle peut avoir à fabriquer 720 pièces  $\alpha$  par jour, 480 pièces  $\beta$  par

jour ou 1920 pièces  $\gamma$  par jour. Enfin, l'entreprise ne peut fournir une puissance électrique supérieure à 1410 kW/h. On remarquera que la machine B produit sa propre électricité et permet même d'alimenter d'autres machines. Ceci est représentée par une consommation négative. En revanche, le critère du coût de fonctionnement des machines est comparable, et on n'en tiendra donc pas compte.

1. Modélisez le problème sous forme d'un programme mathématique. Pour le simplifier, on supposera dans la suite que les nombres de machines peuvent prendre des valeurs non entières. (On parle de relaxation du problème).
2. Représentez les contraintes sur un graphique, et mettez en évidence l'ensemble des solutions. Est-il convexe ? Est-il compact ? Qu'en déduire ?
3. On désigne par  $f(x)$  la fonction économique. Dessinez sur le graphique les droites  $f(x) = c$ , où  $c$  est une constante. Vous choisirez les valeurs de  $c$  qui vous semblent pertinentes.
4. Calculez le gradient de la fonction économique. Vérifiez qu'il est orthogonal aux droites dessinées précédemment
5. Devinez la solution optimale du problème. Soit  $\text{copt}$  son coût. Vous pouvez vérifier graphiquement en dessinant la droite  $f(x) = \text{copt}$  qu'aucune autre solution n'est meilleure. Pouvez-vous aussi deviner la meilleure solution entière ?

On modélise le problème comme suit : Minimiser  $14570x_1 + 8742x_2$  sous les contraintes  $x_1 \leq 20$ ,  $x_2 \leq 20$ ,  $x_1 + x_2 \leq 30$ ,  $2x_1 + 3x_2 \geq \frac{720}{24}$ ,  $4x_1 + x_2 \geq \frac{480}{24}$ ,  $x_1 + 20x_2 \geq \frac{1920}{24}$ ,  $80x_1 - 30x_2 \leq 1410$ .



La solution optimale est  $(3, 8)$ . Elle est entière.

*Exercice 42.* Consommation des ménages. On considère un ménage qui peut consommer  $n$  types de marchandise dont les prix forment un vecteur  $p \in \mathbb{R}_+^n$ . Son revenu à dépenser est un réel  $b > 0$ , et ses choix de consommation sont supposés être modélisés par une fonction d'utilité  $u(x)$  de  $\mathbb{R}_+^n$  dans  $\mathbb{R}$  (croissante et concave), qui mesure le bénéfice que le ménage tire de la consommation de la quantité  $x$  des  $n$  marchandises.

1. Donnez la consommation du ménage sous contrainte de budget.
2. Est-ce un problème de minimisation ou de maximisation de l'utilité ?

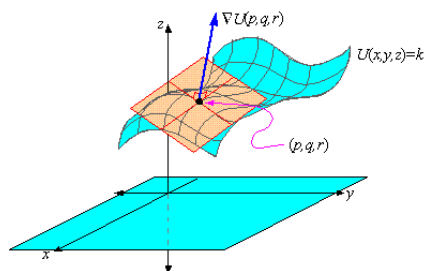
La consommation des ménages est un vecteur  $x^*$  qui réalisera le maximum de  $u(x)$  pour  $x \in \mathbf{R}_+^n$  sous la contrainte  $p \leq b$ . Il s'agit de maximiser l'utilité sous une contrainte de budget maximal.

*Exercice 43.* Problème du sac à dos. Un problème classique est celui du sac-à-dos. Soit  $n$  objets de poids respectifs  $p_1, \dots, p_n$  dans  $R$ , et d'utilités respectives  $u_1, \dots, u_n \in R$ , et  $P \in R$  un poids maximal que l'on est disposé à porter. On pose  $x_i = 1$  si on met l'objet  $i$  dans le sac-à-dos, et  $x_i = 0$  sinon. On veut maximiser l'utilité du sac à dos sous contrainte de poids.

1. Donnez la fonction objectif de ce problème.
2. Quelle est la contrainte de ce problème ?

La fonction objectif est égale à  $\sum_{i=1}^n x_i u_i$ , qu'on cherche à maximiser pour  $x \in \{0, 1\}^n$ . La contrainte de ce problème est  $\sum_{i=1}^n x_i p_i \leq P$ .

*Exercice 44.* Soit  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  continûment différentiable. Soit  $x_0$  tel que  $\nabla f(x_0) \neq 0$ . Que représente  $\nabla f(x_0)$  pour la surface de niveau  $S := \{x \in \mathbf{R}^n | f(x) = f(x_0)\}$  ?  
Le gradient est orthogonal à la surface de niveau  $S$  :



## 5.2 Correction de la série d'exercices numéro 2

*Exercice 45.* Soit  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  continûment différentiable. Soit  $x_0$  tel que  $\nabla f(x_0) \neq 0$ . Que représente  $\nabla f(x_0)$  pour la surface de niveau  $S := \{x \in \mathbf{R}^n | f(x) = f(x_0)\}$  ?

Le gradient est un vecteur qui indique la direction et le sens de croissance de la fonction  $f$  dans l'espace. Sur une surface de niveau  $S$ , la fonction  $f$  est constante. Ainsi, pour tout déplacement élémentaire sur cette surface, la variation de  $f$  est nulle. On a  $df = 0$  donc que  $\nabla f(x)$  est normal aux surfaces de niveau.

*Exercice 46.* Déterminer  $\nabla f(x)$  et  $\nabla^2 f(x)$  en tout point  $x$  pour :

1.  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  tel que  $f(x) = \langle c, x \rangle + \gamma$
2.  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  tel que  $f(x) = \frac{1}{2} \langle Ax, x \rangle + \langle d, x \rangle + \delta$
3.  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  tel que  $f(x) = \sum_{i=1}^m (r_i(x))^2$  où les  $r_i : \mathbf{R}^n \rightarrow \mathbf{R}$  sont deux fois différentiables.

*Exercice 47.* Comme  $r(x)$  est différentiable, on a :

$$r(x+h) = r(x) + \langle \nabla r(x), h \rangle + \|h\| \epsilon(h)$$

ce qui donne

$$r(x+h)^2 = r(x)^2 + 2r(x) \langle \nabla r(x), h \rangle + \langle \nabla r(x), h \rangle^2 + \|h\| \epsilon'(h)$$

La quantité  $\langle \nabla r(x), h \rangle^2$  est de la forme  $\|h\| \epsilon'(h)$  avec  $\epsilon'(h)$  tend vers 0 quand  $\|h\|$  tend vers 0. En effet :

$$\langle \nabla r(x), h \rangle^2 = \langle \nabla r(x), h \rangle \langle \nabla r(x), h \rangle = {}^t h \nabla r(x) {}^t \nabla r(x) h$$

et en terme de normes on a :  $\| {}^t h \nabla r(x) {}^t \nabla r(x) h \| \leq \|h\|^2 \| \nabla r(x) {}^t \nabla r(x) \|$ . Donc  $\langle \nabla r^2(x), h \rangle = 2r(x) \langle \nabla r(x), h \rangle$  et

$$\nabla r^2(x) = 2r(x) \nabla r(x)$$

Nous pouvons aussi considérer que  $r^2(x)$  est le produit  $r(x)r(x)$  et que le gradient de ce produit est la somme  $r(x)\nabla r(x) + \nabla r(x)r(x)$ .

D'autre part, en écrivant le développement de Taylor à l'ordre 2 de  $r(x+h)$ , on peut obtenir l'expression de la matrice hessienne  $\nabla^2 r^2(x)$  :

$$r(x+h) = r(x) + \langle \nabla r(x), h \rangle + \frac{1}{2} \langle \nabla^2 r(x) h, h \rangle + \|h\|^2 \epsilon(h)$$

ce qui donne  $r(x+h)^2 = r(x)^2 + 2r(x) \langle \nabla r(x), h \rangle + \langle \nabla r(x), h \rangle^2 + r(x) \langle \nabla^2 r(x) h, h \rangle + \|h\|^2 \epsilon'(h)$  Donc  $\nabla^2 r^2(x) = 2\nabla r(x) {}^t \nabla r(x) + 2r(x) \nabla^2 r(x)$ .

*Exercice 48.* Montrer qu'une norme est convexe.

Soit  $N$  une norme sur  $K$ . Pour tout  $x, y$  dans  $K$  et  $\theta$  dans  $[0, 1]$ , on a d'après la définition de la norme :  $N(\theta x + (1-\theta)y) \leq N(\theta x) + N((1-\theta)y) \leq \theta N(x) + (1-\theta)N(y)$ . La norme est donc une fonction convexe.

*Exercice 49.* Montrer l'inégalité de Young :  $\forall a, b > 0, \forall p, q \in \mathbf{N}$  tels que  $\frac{1}{p} + \frac{1}{q} = 1$ ,  $ab \leq \frac{a^p}{p} + \frac{b^q}{q}$ .

On utilise la convexité de la fonction exponentielle avec  $\frac{1}{p} + \frac{1}{q} = 1$ . On obtient pour  $a > 0$  et  $b > 0$  :

$$\exp\left(\frac{1}{p} \log a^p + \frac{1}{q} \log b^q\right) \leq \frac{1}{p} \exp(\log a^p) + \frac{1}{q} \exp(\log b^q)$$

D'où l'inégalité à prouver.

*Exercice 50.* Montrer que la fonction indicatrice d'un ensemble  $K \subset \mathbf{R}^n$  définie par

$$1_K(x) = \begin{cases} 0 & \text{si } x \in K \\ +\infty & \text{sinon } (x \in \mathbf{R}^n/K) \end{cases}$$

est convexe si et seulement si  $K$  est convexe.

Supposons que  $K$  est convexe et soient  $x, y \in K$  et  $t \in [0, 1]$ . Si  $x, y \in K$  alors  $tx + (1-t)y \in K$  et  $1_K(tx + (1-t)y) = 0 = t1_K(x) + (1-t)1_K(y)$ . Si  $x$  (ou  $y$ )  $\notin K$  alors  $1_K(x)$  (ou  $1_K(y)$ ) est égale à  $+\infty$  et l'inégalité de convexité est trivialement vérifiée.

Réciproquement, soient  $x, y \in K$  et  $t \in [0, 1]$ . Par convexité de  $1_K$ , on a :  $1_K(tx + (1-t)y) \leq t1_K(x) + (1-t)1_K(y) = 0$ . Comme  $1_K$  ne peut prendre comme valeur que 0 ou  $+\infty$ ,  $1_K(tx + (1-t)y) = 0$  et  $tx + (1-t)y \in K$ .

*Exercice 51.* Représenter graphiquement chacun de ces ensembles en précisant sa convexité :

- $A = \{(x, y) \in \mathbb{R}^2 \mid x \geq 0, y \geq 0, x + y \leq 1\}$
- $B = \{(x, y) \in \mathbb{R}^2 \mid xy \geq 0\}$
- $C = \{(x, y) \in \mathbb{R}^2 \mid x \geq 0 \text{ ou } y \geq 0\}$
- $D = \{(x, y) \in \mathbb{R}^2 \mid x \geq 0, y \leq 0, y \leq x^3\}$
- $E = \{(x, y) \in \mathbb{R}^2 \mid x^2 + 4y^2 \leq 4, 3x + 3y + 1 \geq 0\}$ .

*Exercice 52.*  $A$  est convexe ;  $B$  n'est pas convexe ;  $C$  n'est pas convexe ;  $D$  est réduit à  $\{(0, 0)\}$  il est par convention convexe ;  $E$  est convexe.

Soit  $E$  un espace vectoriel réel. On dit qu'un sous ensemble  $K$  de  $E$  est un cône si et seulement si

$$\forall x \in K, \forall \lambda > 0, \lambda x \in K.$$

Montrer qu'un cône  $K$  est convexe si et seulement si  $\forall x, y \in K : x + y \in K$ .

Supposons que  $K$  est un cône convexe. Alors  $\forall x, y \in K$  et  $\forall t \in [0, 1] : tx + (1-t)y \in K$ . On prend  $t = \frac{1}{2}$  et  $\lambda = 2$ , on obtient :  $\frac{1}{2}(x + y) \in K$  et  $2\frac{1}{2}(x + y) \in K$ .

Réciproquement, supposons que  $\forall x, y \in K : x + y \in K$ . Comme  $K$  est un cône alors pour  $x \in K$ ,  $\lambda x \in K$  pour  $\lambda > 0$ . Pour  $t \in [0, 1] : (1-t)y \in K$  et  $tx \in K$ . Ce qui implique que  $tx + (1-t)y \in K$  donc  $K$  est un cône convexe.

*Exercice 53.* Les fonctions  $J$  suivantes sont-elles coercives ? -  $J(x) = x^3 + x + 1$  -  $J(x) = \langle a, x \rangle + b$  -  $J(x) = 2x_1^2 + x_2 - 1$  -  $J$  est une forme quadratique associée à une matrice symétrique définie positive.

- $J(x) = x^3 + x + 1$  n'est pas coercive : on prend pour cela  $x = -n$  où  $n \rightarrow +\infty$ .
- $J(x) = \langle a, x \rangle + b$  n'est pas coercive : on prend pour cela  $x = (-an, 0, \dots, 0)$

$$J(-an, 0, \dots, 0) = -na^2 + b \rightarrow -\infty \text{ lorsque } n \rightarrow +\infty.$$

- $J(x) = 2x_1^2 + x_2 - 1$  n'est pas coercive : tester pour  $x = (0, -n)$ . - Si  $A$  est symétrique définie positive alors  $J$  est coercive.

*Exercice 54.* Etudier la convexité ou la concavité des fonctions définie par  $f(x) = 3x_1^2 + 2x_2^2 - 8x_1$  :

Pour  $f(x) = 3x_1^2 + 2x_2^2 - 8x_1$ ,  $\nabla f(x)^t = (6x_1 - 8, 4x_2)$  et  $\nabla^2 f(x) = \begin{pmatrix} 6 & 0 \\ 0 & 4 \end{pmatrix}$ .

## 5.3 Correction de la série d'exercices numéro 3

*Exercice 55.* Soient  $E, F$  deux espaces vectoriels normés sur  $R$ ,  $A \subset E$  et  $f : A \rightarrow F$ . On a les trois équivalences suivantes :  $f$  est continue sur  $A$



$\Leftrightarrow$  L'image réciproque de tout ouvert de  $F$  est un ouvert relatif de  $A$  (i.e.  $\forall V$  ouvert de  $F$ , il existe  $O$  ouvert de  $E$  tel que  $f^{-1}(V) = O \cap A$ )

$\Leftrightarrow$  L'image réciproque de tout fermé de  $F$  est un fermé relatif de  $A$  (i.e.  $\forall U$  fermé de  $F$ , il existe  $W$  fermé de  $E$  tel que  $f^{-1}(U) = W \cap A$ ).

1. Soit  $X = \{(x_1, x_2) \in \mathbb{R}^2, 3x_1^2 + 2x_1x_2 + x_2^3 < 1\}$ . Montrez que  $X$  est un ouvert de  $\mathbb{R}^2$ .
2. Soit  $Y = \{(x_1, x_2, x_3) \in \mathbb{R}^3, x_1^3 + 2x_1x_2x_3 \leq 4, x_2^4 + x_3^2 \leq 3\}$ . Montrez que  $Y$  est un fermé de  $\mathbb{R}^3$ .

On considère  $f$  de  $\mathbb{R}^2$  dans  $\mathbb{R}$  vérifiant  $f(x_1, x_2) = 3x_1^2 + 2x_1x_2 + x_2^3$ .  $f$  est polynomiale donc elle est continue sur  $\mathbb{R}^2$ . Grâce au théorème précédent,  $f^{-1}(]-\infty, 1]) = X$  est un ouvert de  $\mathbb{R}^2$  puisque  $]-\infty, 1]$  est un ouvert de  $\mathbb{R}$ .

On considère  $f(x_1, x_2, x_3) = x_1^3 + 2x_1x_2x_3$  et  $g(x_1, x_2, x_3) = x_2^4 + x_3^2$ . Les fonctions  $f$  et  $g$  sont deux fonctions polynomiales, donc continues sur  $\mathbb{R}^3$  et  $Y = f^{-1}(]-\infty, 0]) \cap g^{-1}(]-\infty, 0])$ . Comme  $]-\infty, 0]$  est un fermé de  $\mathbb{R}$  alors  $Y$  est un fermé de  $\mathbb{R}^3$ .

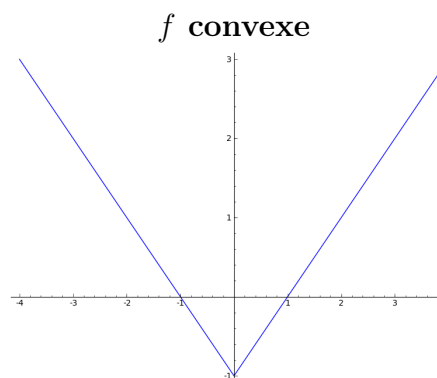
*Exercice 56.* Soit  $E$  un espace vectoriel muni de la norme  $\|\cdot\|$ . On dit que  $f : E \rightarrow \mathbb{R}$  est une application coercive si  $\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty$ . Les fonctions suivantes sont :

1.  $f(x) = x + 1$  n'est pas coercive, car  $\lim_{x \rightarrow -\infty} x + 1 = \lim_{|x| \rightarrow +\infty} x + 1 = -\infty$ .
2.  $f(x) = \exp(x)$  n'est pas coercive :  $\lim_{x \rightarrow -\infty} e^x = \lim_{|x| \rightarrow +\infty} e^x = 0$ .
3.  $f(x) = x^2 + 2$  est coercive car  $\lim_{|x| \rightarrow +\infty} |x|^2 + 2 = +\infty$ .
4. Pour  $f(x) = \sum_{i=1}^n a_i x^i$  avec  $a_n \neq 0$  et  $n \geq 1$  : si  $n$  est impair, alors  $f$  n'est pas coercive. En effet,  $\lim_{x \rightarrow -\text{signe}(a_n) \cdot \infty} f(x) = -\infty$ . Si  $n$  est pair, alors  $f$  est coercive si et seulement si  $a_n > 0$ .

*Exercice 57.* Répondre par vrai ou faux en justifiant votre réponse :

1.  $f$  convexe  $\Rightarrow |f|$  convexe.
2. Si  $f$  est convexe alors  $\lambda f$  est convexe pour tout  $\lambda \in \mathbb{R}$ .

1. FAUX : si  $f$  est convexe,  $|f|$  convexe n'est pas forcément convexe.



2.  $f$  convexe  $\Rightarrow |f|$  convexe. FAUX. Par exemple, si  $f$  est convexe, alors  $-f$  n'est pas convexe mais concave.

*Exercice 58.* Soit  $A$  une matrice carrée dans  $M_n(\mathbb{R})$  et  $b \in \mathbb{R}^n$ . Montrez que l'application  $f(x) = Ax + b$  est deux fois dérivable en tout point de  $\mathbb{R}^n$ . Déterminez  $df(x)$  et  $d^2f(x)$ .

Il suffit de vérifier que  $f(x+h) = f(x) + df(x).h + \frac{1}{2}d^2f(x)(h, h)$ . Il en ressort que  $df(x) = A$  et que  $d^2f(x) = 0$ .

*Exercice 59.* Déterminez les valeurs des paramètres rendant les fonctions suivantes convexes :

1.  $f(x) = ax^2 + bx + c$
2.  $f(x) = ae^{2x} + e^x + c$
3.  $f(x, y) = ax^2 + 2bxy + cy^2$ .

Il suffit de calculer la hessienne pour chaque fonction et de vérifier qu'elle est définie-positive.

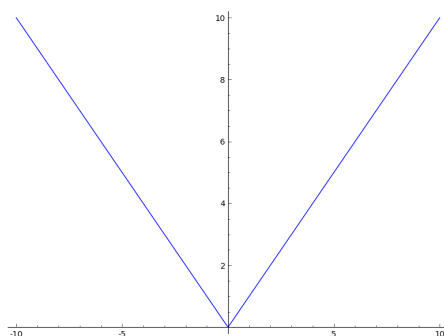
1.  $\nabla^2 f(x) = 2a$  est définie positive lorsque  $a \geq 0$ .
2. Une simple étude de fonction permet de caractériser  $a$  qui assure que  $f$  soit convexe :  $f''(x) = 4ae^{2x} + e^x$ , si  $a \geq 0$  alors  $f$  est convexe, sinon, si  $a < 0$ , alors

$$\lim_{x \rightarrow +\infty} f''(x) = -\infty$$

donc  $f$  est non convexe.

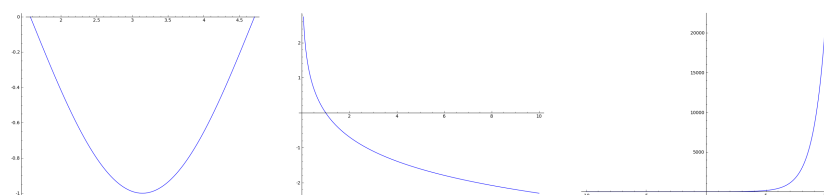
*Exercice 60.* Soit  $E$  un e.v.n. et  $A \subset E$ . Une fonction  $f : A \rightarrow R$  est convexe si son épigraphe est une partie convexe de  $E \times R$  i.e.  $\text{epi}(f) = \{(x, \lambda), x \in A, f(x) \leq \lambda\}$  est un ensemble convexe. Montrez que  $f(x) = |x|$  est une fonction convexe sur  $R$  et donnez l'épigraphe de  $f$ .

L'épigraphe est la partie au dessus de  $f$  délimitée par sa courbe :



*Exercice 61.* Les fonctions suivantes sont-elles convexes ?

1.  $f(x) = e^x$  pour  $x \in R$
2.  $f(x) = -\log(x)$  pour  $x > 0$
3.  $f(x) = \cos(x)$  pour  $x \in [\frac{\pi}{2}, \frac{3\pi}{2}]$
4.  $f(x) = \|x\|$  pour  $x \in R^n$ .



Ce sont des fonctions convexes. Pour la fonction  $\|x\|$ , il suffit de noter que  $\|\theta x + (1 - \theta)y\| \leq \|\theta x\| + \|(1 - \theta)y\|$  pour tout  $\theta$  et  $x, y \in R^n$ .

*Exercice 62.* Soit la fonction  $f(x_1, x_2) = 3x_1^2 + 2x_1x_2 - x_2^2$ . Trouver, au point  $x_0 = (1, 1)$ , un vecteur unitaire  $v$  qui indique la direction du gradient.

Le gradient de  $f$  est égal à  $\nabla f(x) = \begin{pmatrix} 6x_1 + 2x_2 \\ -2x_2 + 2x_1 \end{pmatrix}$ . Au point  $(1, 1)$ , la direction du gradient est égale à  $(1, 0)^t$ .

*Exercice 63.* Cherchez les dimensions d'un wagon rectangulaire non couvert telles que pour un volume donné  $V$ , la somme des aires des côtés et du plancher soit minimale.

On note  $x, y, z$  les dimensions du wagon. Ce sont des quantités strictement positives vérifiant  $xyz = V$  fixé. L'aire de ce wagon est égale à  $A = xy + 2(xz + yz)$ . On cherche donc à minimiser  $A$  sous la contrainte  $V = xyz$  et  $x > 0, y > 0, z > 0$ . Par exemple, on pose  $z = \frac{V}{xy}$ . On doit donc minimiser la fonction  $A = xy + 2V \frac{x+y}{xy} = xy + \frac{2V}{y} + \frac{2V}{x}$ . Le système d'optimalité est  $\begin{cases} y^3 = 2V \\ x^3 = 2V \end{cases}$  et on obtient  $x = y = \sqrt[3]{2V}$ .

*Exercice 64.* Soit la fonction  $f(x, y) = x_1^2 + x_2^2 - 4x_1 - 2x_2$ . Calculez le gradient de  $f$  et la matrice hessienne de  $f$ . En déduire l'ensemble des points critiques de  $f$ .

## 5.4 Correction de la série d'exercices numéro 4

*Exercice 65.* Trouver les minima et les maxima sur  $R^2$  de la fonction  $J$  définie sur  $R^2$  par :

1.  $J(x_1, x_2) = x_1^2 - x_1x_2 + \frac{1}{6}x_2^3$
  2.  $J(x_1, x_2) = x_1^2 - 2x_1x_2 + 1$
  3.  $J(x_1, x_2) = x_1^3 + x_2^3 - 9x_1x_2 + 27$ .
1.  $J(x_1, x_2) = x_1^2 - x_1x_2 + \frac{1}{6}x_2^3$  a deux points critiques  $(0, 0)$  et  $(\frac{1}{2}, 1)$ . La matrice Hessienne vaut  $\begin{pmatrix} 2 & -1 \\ -1 & x_2 \end{pmatrix}$ . Pour  $x_2 = 0$  la matrice a deux valeurs propres de signes différents, donc  $(0, 0)$  n'est ni un minimum ni un maximum. Pour  $x_2 = 1$ , la matrice est définie positive et le point  $(\frac{1}{2}, 1)$  est un minimum strict.
  2.  $J(x_1, x_2) = x_1^2 - 2x_1x_2 + 1$  : le point  $(0, 0)$  est un point critique mais ce n'est ni un minimum ni un maximum.
  3.  $J(x_1, x_2) = x_1^3 + x_2^3 - 9x_1x_2 + 27$  : les deux points critiques sont  $(0, 0)$  et  $(3, 3)$ . Le point  $(0, 0)$  n'est ni un minimum ni un maximum car la matrice Hessienne n'est ni semi-définie positive ni semi-définie négative. Le point  $(3, 3)$  est un minimum strict.

*Exercice 66.* Montrer par un exemple que  $\nabla J(x) = 0$  est une condition nécessaire d'optimalité et pas suffisante.

Il suffit de considérer la fonction  $J(x) = x^3$  dans  $R$ .

*Exercice 67.* Soit  $A$  une matrice symétrique, définie positive, à coefficients réels. Montrer qu'il existe une constante  $\alpha > 0$  telle que

$$\forall \nu \in R^n \quad \langle A\nu, \nu \rangle \geq \alpha \|\nu\|^2$$

où  $\langle, \rangle$  est le produit scalaire sur  $R^n$  et  $\|\cdot\|$  la norme euclidienne associée.

$A$  est symétrique, il existe donc une base de vecteurs propres orthogonaux  $u_i$  associés aux valeurs propres  $\lambda_i$  qui sont strictement positives (car  $A$  est définie positive). Soit  $x = \sum_{i=1}^n x_i u_i$  dans  $R^n$ . Nous avons

$$\langle Ax, x \rangle = \sum_{i,j=1}^n \lambda_i x_i x_j \langle u_i, u_j \rangle = \sum_{i=1}^n \lambda_i x_i^2 \geq \lambda_{\min} \|x\|^2$$

On peut prendre pour  $\alpha$  la plus petite valeur propre  $\lambda_{\min}$  de  $A$ .

*Exercice 68.* Appliquer la méthode de Newton à l'équation  $F(x) = \frac{1}{2}x^2 + e^{-x}$  pour calculer le minimum de  $F$  sur  $[\frac{1}{10}, 1]$ . Pour cela,

1. montrer que les conditions du théorème de convergence globale de la méthode de Newton sont satisfaites.
2. Vérifier la convergence de la suite de Newton puis calculer la suite et montrer que sa convergence est quadratique.

On pose  $f = F'$ .

1. Le théorème de convergence globale de la méthode de Newton est donné ci-dessous : Soit  $f$  une fonction de Classe  $C^2$  sur  $[a, b]$  et  $g(x) = x - \frac{f(x)}{f'(x)}$ . Si  $f$  vérifie :

(a)  $f(a).f(b) < 0$

(b)  $\forall x \in [a, b], f'(x) \neq 0$  (c'est la stricte monotonie),

(c)  $\forall x \in [a, b], f''(x) \neq 0$  (concavité dans le même sens).

Alors en choisissant  $x_0 \in [a, b]$  tel que  $f(x_0).f''(x_0) > 0$ , la suite  $(x_n)$  définie par  $x_0$  et  $x_{n+1} = g(x_n)$  converge vers l'unique solution de  $f(x) = 0$  dans  $[a, b]$ .

Ici,  $F'(x) = x - e^{-x}$  sur  $[0.1, 1]$ . On a pour  $f(x) = F'(x)$  :  $f(0.1) = -0.8$  et  $f(1) = 0.6$ ,  $f'(x) = 1 + e^{-x}$  et  $f''(x) = -e^{-x}$  sont deux fonctions de signes constants (tous deux positifs). En choisissant  $x_0 = 1$ , les conditions du théorème ci-dessus sont satisfaites et  $(x_n)$  converge vers le minimum de  $F$  sur  $[0.1, 1]$ . En effet, pour  $x_0 = 1$ ,  $x_1 = 0.537883$ ,  $x_2 = 0.566989$ ,  $x_3 = 0.567143 = x_4$ . Le minimum à  $10^{-6}$  près est égal à 0.567143.

2. Les deux premières conditions du théorème assurent l'existence et l'unicité d'une racine  $x^*$  unique de  $f$  dans  $[a, b]$ . D'autre part, comme  $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$  on obtient :

$$\begin{aligned} x_{n+1} - x^* &= x_n - x^* - \frac{f(x_n)}{f'(x_n)} \\ &= x_n - x^* + \frac{f(x^*) - f(x_n)}{f'(x_n)} \\ &= x_n - x^* + \frac{(x^* - x_n)f'(x_n) + \frac{(x^* - x_n)^2}{2}f''(\zeta_n)}{f'(x_n)} \end{aligned}$$

Ainsi :

$$\begin{aligned} x_{n+1} - x^* &= (x_n - x^*) \left( 1 - \frac{f'(x_n) + \frac{x^* - x_n}{2} f''(\zeta_n)}{f'(x_n)} \right) \\ &= \frac{(x_n - x^*)^2}{2} \frac{f''(\zeta_n)}{f'(x_n)} \end{aligned}$$

Si  $f''$  et  $f'$  sont de même signe, alors  $x_{n+1} - x^* > 0$  et la suite est alors minorée par  $x^*$  à partir du rang 1. Sinon, si  $f''$  et  $f'$  sont de signes contraires, alors  $x_{n+1} - x^* < 0$  et la suite est alors majorée. De plus, et selon les cas, la suite va être soit décroissante ( $f'' > 0$  et  $f' > 0$  ou bien  $f'' < 0$  et  $f' < 0$ ) donc convergente soit croissante ( $f'' > 0$  et  $f' < 0$  ou bien  $f'' < 0$  et  $f' > 0$ ) et donc convergente.

On remarque que pour  $g(x) = x - f(x)f'(x)$ ,  $g'(x) = \frac{f(x)f''(x)}{(f'(x))^2}$  donc  $g'(x^*) = 0$ . D'autre part :

$$\begin{aligned} x_{n+1} - x^* &= g(x_n) - g(x^*) \\ &= (x_n - x^*)^2 \frac{g''(x^*)}{2} + (x_n - x^*)^2 \epsilon(e_n) \\ &= e_n^2 \frac{g''(x^*)}{2} + e_n^2 \epsilon(e_n) \end{aligned}$$

Comme  $\epsilon(e_n)$  est négligeable au voisinage de  $+\infty$  on obtient le résultat. A chaque itération, on multiplie par 2 le nombre de chiffres exacts de l'approximation.

D'une façon générale, si  $f$  est de Classe  $C^p$  alors :  $e_{n+1} = e_n g'(x^*) + \frac{e_n^2}{2!} g''(x^*) + \dots + \frac{e_n^p}{p!} g^{(p)}(x^*) + e_n^p \epsilon(e_n)$ . La méthode est alors d'ordre  $p$  si  $g'(x^*) = \dots = g^{(p-1)}(x^*) = 0$  et  $g^{(p)}(x^*) \neq 0$ .

*Exercice 69.* Soit la fonction  $f(x) = \frac{2}{3}x^3 - \frac{1}{2}x^2 - 2x + 1$ , on se propose de trouver les extremums réels de  $f$  par la méthode de Fibonacci.

1. Montrer que  $f$  possède un seul point critique (ou point stationnaire)  $\alpha$  sur  $[1, 2]$ .
2. Donner les trois premières itérations de la méthode de Fibonacci pour calculer  $\alpha$ .

On remarque que  $f'(x) = 2x^2 - x - 2$  est strictement croissante sur  $[1, 2]$  car  $f''(x) = 4x - 1$  est positive strictement sur  $[1, 2]$ . Elle réalise donc une bijection sur cet ensemble. D'autre part,  $f'$  change de signe donc elle s'annule nécessairement en un point unique  $\alpha$  de  $[1, 2]$ . La fonction  $f$  admet alors un unique minimum sur  $[1, 2]$ .

La fonction  $f$  est unimodale car strictement convexe. On applique l'algorithme de Fibonacci pour  $N = 10$  et on obtient les itérations suivantes :  $a_1 = 1$ ,  $b_1 = 2$ ,  $x_1^G = 1.382022$  et  $x_1^D = 1.617978$ . On note que  $-0.959276 = f(x_1^G) < f(x_1^D) = -0.721131$ . On pose donc  $a_2 = a_1 = 1$  et  $b_2 = x_1^D$ .

Les différentes itérations donnent les intervalles suivants :

$$\begin{aligned} &\left[1, \frac{110}{89}, \frac{123}{89}, \frac{144}{89}\right] \\ &\left[1, \frac{102}{89}, \frac{110}{89}, \frac{123}{89}\right] \\ &\left[\frac{102}{89}, \frac{110}{89}, \frac{115}{89}, \frac{123}{89}\right] \\ &\left[\frac{110}{89}, \frac{115}{89}, \frac{118}{89}, \frac{123}{89}\right] \\ &\left[\frac{110}{89}, \frac{113}{89}, \frac{115}{89}, \frac{118}{89}\right] \end{aligned}$$

$$\left[ \frac{110}{89}, \frac{112}{89}, \frac{113}{89}, \frac{115}{89} \right]$$

$$\left[ \frac{112}{89}, \frac{113}{89}, \frac{114}{89}, \frac{115}{89} \right]$$

$$\left[ \frac{113}{89}, \frac{114}{89}, \frac{114}{89}, \frac{115}{89} \right]$$

$$\left[ \frac{114}{89}, \frac{114}{89}, \frac{114}{89}, \frac{114}{89} \right]$$

Le minimum est égal à  $\frac{114}{89} = 1.280898$ .

Le code ayant permis de calculer cete suite est le suivant :

```
def fib(n) :
    if n<2 :
        return(n)
    else :
        return(fib(n-1)+fib(n-2))

N = 10
a = 1
b = 2
k = 1
X=[a, a + fib(N-k)/fib(N-k+2)*(b-a),
a + fib(N-k+1)/fib(N-k+2)*(b-a), b]
while k <=N :
    if f(X[1])<f(X[2]) :
        b = X[2]
        k = k+1
    elif f(X[1])>f(X[2]) :
        a = X[1]
        k = k+1
    else :
        a = X[1]
        b = X[2]
        k = k+2
    X=[a, a + fib(N-k)/fib(N-k+2)*(b-a),
    a + fib(N-k+1)/fib(N-k+2)*(b-a), b]
print X
```

*Exercice 70.* Soit  $f$  une fonction de  $\mathbf{R}$  dans  $\mathbf{R}$  telle que

$$f(x) = \frac{1}{4}x^4 + x^3 - x + 1 \quad \text{pour} \quad x \in \mathbf{R}.$$

1. Montrer que l'équation  $f'(x) = 0$  admet une unique solution sur  $[0, 1]$ .
2. Nous désirons déterminer la solution de l'équation  $f'(x) = 0$  avec la méthode de Newton sur  $[0, 1]$ . En quoi consiste cette méthode ?
3. Ecrire l'algorithme général de la méthode de Newton qui prend en entrée les points  $x_0$ ,  $a < b$  et une fonction  $f'$  et rend la racine de  $f'(x) = 0$  sur  $[a, b]$  ou bien un message d'erreur.

4. Est-ce que la fonction  $f'$  vérifie les conditions du théorème de convergence globale de la méthode de Newton pour  $x_0 = 1$ . Donner le théorème.
5. Calculer la racine de  $f'$  sur l'intervalle  $[0, 1]$  avec une erreur de  $10^{-6}$  en proposant un bon choix de  $x_0$ .

Suivre les mêmes étapes que l'exercice 4.

*Exercice 71.* On considère le problème d'optimisation sans contrainte d'une fonction  $J$  définie sur  $R^n$ .

1. Rappeler la définition de l'algorithme du gradient à pas optimal.
2. On rappelle que le pas optimal  $r_k$  à l'itération  $k$  vérifie la relation

$$\langle \nabla J(u_k - r_k d_k), d_k \rangle = 0$$

avec  $d_k = \nabla J(u_k)$ . Donner une expression simple du pas optimal  $r_k$  dans le cas d'une fonction quadratique

$$J(v) = \frac{1}{2} \langle Av, v \rangle - \langle b, v \rangle.$$

Voir le cours.

*Exercice 72.* On considère la fonction  $J$  définie sur  $R^5$  par

$$J(v) = \frac{1}{2} \langle Av, v \rangle - \langle b, v \rangle$$

que l'on souhaite minimiser sans contraintes à l'aide de l'algorithme du gradient à pas fixe. Compléter les pointillés de la fonction Python suivante. `def GPF(A,b,u0,epsilon,max_iter, r) :`

```

    u = .....
    nb_iter = .....
    d = .....
    nr = norm(d)
    while (..... and .....):
        u = .....
        nb_iter = .....
        d = .....
        nr = .....
    return(u,nb_iter)

```

où `epsilon`, `max_iter` et `nb_iter` représentent respectivement la précision souhaitée de la solution numérique, le nombre maximal d'itérations à ne pas dépasser et le nombre d'itérations effectuées.

```

def GPF(A,b,u0,epsilon,max_iter, r) :
    u = u0
    nb_iter = 0
    d = Au-b
    nr = norm(d)
    while (..... and .....):

```

```

    u = .....
    nb_iter = .....
    d = .....
    nr = .....
    return(u,nb_iter)

```

## 5.5 Correction de la série d'exercices numéro 5

*Exercice 73.* Vérifier que les contraintes suivantes vérifient la condition de qualification à l'application du théorème de Lagrange donnée en cours. :

1.  $2x + y \leq 2, x \geq 0, y \geq 0$
2.  $x^2 + y^2 \leq 1, x \geq 0, y \geq 0$
3.  $2x - y = -5, 5x + 2y \geq 37, x \geq 0, y \geq 0$
4.  $x^2 + y^2 + z \leq 6, x \geq 0, y \geq 0, z \geq 0$
5.  $4x + 3y \leq 10, y - 4x^2 \geq -1, x \geq 0, y \geq 0$

*Exercice 74.* Déterminer les quantités suivantes :

1.  $\max x^2 + y^2$  s/c  $2x + y \leq 2, x \geq 0, y \geq 0$
2.  $\min x^2 - 2y$  s/c  $x^2 + y^2 \leq 1, x \geq 0, y \geq 0$
3.  $\max 3xy - x^3$  s/c  $2x - y = -5, 5x + 2y \geq 37, x \geq 0, y \geq 0$
4.  $\max x^2 + x + 4y^2$  s/c  $2x + 2y \leq 1, x \geq 0, y \geq 0$

*Exercice 75.* Résoudre les problèmes d'optimisation de l'exercice 2. en utilisant la condition de Kuhn et Tucker.

*Exercice 76.* Optimiser localement  $f(x, y, z) = (x-1)^2 + (y-1)^2 + z^2$  sous les contraintes  $x^2 + y^2 + z^2 = 1$  et  $x + y + z = 0$ .

*Exercice 77.* On considère la fonction  $f$  définie sur  $R^3$  par

$$f(x, y, z) = x^2 + 2y^2 - z^2 + 2xy + 2xz + x$$

1. Montrer que  $f$  n'est pas coercive sur  $R^3$
2. Déterminer les points critiques de  $f$  sur  $R^3$ .  $f$  admet-elle des extréma locaux ?
3. Soit  $A = \{(x, y, z) \in R^3, z = \frac{1}{2}(x + y)\}$ .  $A$  est-il fermé ? borné ?
4. Déterminer les points critiques de la restriction de  $f$  à  $A$  de deux manières : en utilisant d'une part les multiplicateurs de Lagrange, et d'autre part en remplaçant  $z$  par sa valeur dans l'expression de  $f$ .
5. Calculer les points où  $f$  est minimale sur  $A$ .

*Exercice 78.* Résoudre les problèmes d'optimisation avec contraintes égalité suivants :

1. Extremum de  $(x^2 + y^2)$  sous la contrainte  $3x^2 + 4xy + 6y^2 = 140$ ,
2.  $\max(x^2 + xy + y^2 + yz + z^2)$  sous la contrainte  $x^2 + y^2 + z^2 = 1$ .



*Exercice 79.* On considère le problème d'optimisation

$$\min_{x+y \leq 1} 7x^2 + 2y^2 + 7xy + 7x$$

1. Montrer que ce problème est un problème convexe
2. Montrer que ce problème admet une solution unique notée  $(a, b)$
3. Ecrire les conditions KKT et trouver  $(a, b)$ .

*Exercice 80.* On considère la fonction  $f$  définie sur  $R^2$  par

$$(x-1)^2 + (y-2)^2,$$

1. Montrer que  $f$  est strictement convexe sur  $R^2$ .
2. On considère les trois problèmes d'optimisation suivants :

$$(P_i) \quad \min_{(x,y) \in K_i} f(x, y), \quad \{i \in \{1, 2, 3\}\}$$

où  $K_1 = R^2$ ,  $K_2 = \{(x, y) \in R^2, x^2 + y^2 \leq 10\}$  et  $K_3 = \{(x, y) \in R^2, x^2 + \frac{1}{3}y^2 \leq 1\}$ .

- (a) Expliquer pourquoi chaque problème  $P_i$  admet une unique solution
- (b) Résoudre  $P_1$  puis  $P_2$ .
- (c) Résoudre  $P_3$ .

*Exercice 81.* On considère la fonction  $u$  définie sur  $R^2$  par

$$u(x, y) = x^2 + y^2 + xy$$

On note

$$D = \{(x, y) \in R^2, x^2 + y^2 \leq 1 \text{ et } x + y \geq 1\}$$

1. Montrer que  $u$  est strictement convexe sur  $R^2$  et qu'elle possède un unique minimum sur  $D$ .
2. En écrivant les relations de KKT, déterminer le point où  $u$  atteint son minimum sur  $D$ .

## 5.6 Exercices de révision

*Exercice 82.* Les fonctions  $J$  suivantes sont-elles coercives :  $J(x) = x_1^2 - x_2^2 - 1$  et  $J(x) = x_1^2 + 2x_2^2 - 2x_1 + 1$  pour  $x = (x_1, x_2) \in \mathbf{R}^2$ .

*Exercice 83.* Chercher les extrémums de la fonction  $f(x_1, x_2) = x_1^2 - x_1x_2 + \frac{1}{6}x_2^2$  sur  $\mathbf{R}^2$ .

*Exercice 84.* Soit  $x_0 = (1, 1)$ . Calculer les premiers itérés engendrés par la méthode du gradient conjugué sur les fonctions suivantes définies sur  $\mathbf{R}^2$  :

1.  $f(x) = 2x_1^2 + 4x_2^2 - 12x_1 - 16x_2$
2.  $f(x) = 6x_1^2 + 8x_2^2 - 54$ .

*Exercice 85.* Appliquer la méthode du gradient pour résoudre le problème suivant, en prenant comme point de départ  $x^0 = (1, 1)$  :

$$\min \quad f(x_1, x_2) = 2x_1^2 + 6x_2^2 - 2x_1 - 2x_2 + 1 \quad .$$

*Exercice 86.* Soit la fonction  $f(x_1, x_2) = 4x_1^2 + 2x_1x_2 - x_2^2$ .

1. trouver, au point  $x_0 = (1, 1)$ , le vecteur  $d$  qui indique le gradient.
2. trouver si elle existe la valeur  $r_0 = \min_r f(x_0 - rd)$ .

*Exercice 87.* Répondre par vrai ou faux.

1. La fonction  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  définie par  $J(x) = \|x\|$  est convexe? Justifier votre réponse.
2. Une fonction quadratique associée à une matrice symétrique semi-définie positive est strictement convexe?
3. La fonction  $J : \mathbf{R}^n \rightarrow \mathbf{R}$  définie par  $J(x_1, x_2) = 2x_1^2 - 3x_2^2$  est coercive? Justifier votre réponse.
4. Soit  $J$  une fonction de classe  $C^1$  coercive et strictement convexe sur  $\mathbf{R}^n$ . Alors le problème d'optimisation  $\min J(x)$ ,  $x \in \mathbf{R}^n$  admet une unique solution.
5. La méthode du gradient est une méthode de descente? Justifier votre réponse.
6. La méthode du gradient conjugué ne s'applique qu'au cas particulier d'une fonction quadratique? Justifier votre réponse.
7. La fonction `dotprod` de **Maple** permet de calculer le produit de deux matrices?

*Exercice 88.* Soit  $x = (x_1, x_2, x_3) \in \mathbf{R}^3$ . Posons  $f(x) = \frac{1}{2}(Ax, x) - (b, x)$  avec

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -1 \\ 1 & -1 & 3 \end{pmatrix} \quad \text{et} \quad b = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}.$$

1. Montrer que la matrice  $A$  est définie positive.
2. Vérifier que  $f$  admet un optimum unique. Donner une caractérisation de cet optimum (c'est-à-dire une condition nécessaire et suffisante d'optimalité).
3. Soit  $X^0 = (x_1^0, x_2^0, x_3^0) \in \mathbf{R}^3$ . Notons  $X^{k+1} = (x_1^{k+1}, x_2^{k+1}, x_3^{k+1})$  le point déterminé itérativement par la méthode de relaxation successive à partir de  $X^k = (x_1^k, x_2^k, x_3^k)$  avec  $k \geq 0$ . Donner la forme générale de  $x_1^{k+1}, x_2^{k+1}, x_3^{k+1}$  en fonction de  $X^k$ .

*Exercice 89.*

1. Une forme quadratique semi-définie positive est-elle convexe? (justifier la réponse).
2. Soit  $J$  une fonction convexe continûment différentiable. Donner une condition nécessaire et suffisante pour que  $x^*$  soit un optimum global de  $J$  sur  $\mathbf{R}^n$ .
3. Enoncer une condition nécessaire et une condition suffisante d'optimalité pour une fonction quelconque de classe  $C^2$ .

*Exercice 90.* Soit  $x \in \mathbf{R}^n$ . On pose  $f(x) = \frac{1}{2}A(x, x)$  où  $A$  est une matrice symétrique d'ordre  $n$ , définie-négative.

1. Montrer que  $f$  admet un optimum unique. Quelle est sa nature?
2. Soit  $x^0 \in \mathbf{R}^n$ . On note  $x^{k+1}$  le point déterminé itérativement par la méthode du gradient optimal à partir de  $x^k$  avec  $k \geq 0$ .
  - (a) Donner la forme générale de  $x^{k+1}$  en fonction de  $x^k$  et  $\nabla f(x^k)$ .

- (b) Montrer que les directions de déplacement successifs de la méthode du gradient optimal sont orthogonales (zigzag).
3. En déduire que le vecteur  $x^{k+1} - x^k$  est orthogonal à  $Ax^{k+1}$ .

*Exercice 91.* Soient  $A$  une matrice symétrique d'ordre  $n$ ,  $B$  une matrice symétrique définie-positive d'ordre  $n$  et  $c$  un vecteur de  $\mathbf{R}^n$ . On définit la fonctionnelle

$$J : v \in \mathbf{R}^n \longrightarrow J(v) = \frac{1}{2}A(v, v) + (Bv, v) - (c, v) \in \mathbf{R} \quad .$$

1. Calculer le gradient de la fonction  $J$  et déterminer l'ensemble des points stationnaires de  $J$ .
2. Calculer la dérivée seconde de  $J$ . Quelle est la nature de cette matrice ?
3. En prenant  $n = 2$ ,  $A = B = I_2$  (la matrice identité d'ordre 2) et  $c = {}^t(3, 3)$ , calculer l'extremum  $x^*$  de la fonction  $J$ .
4. En déduire la nature de  $x^*$ .

*Exercice 92.* Notons  $x = (x_1, x_2) \in \mathbf{R}^2$ . Soit le problème  $(\mathcal{Q})$  :  $\min 4x_1^2 + x_2^2$  sous les contraintes  $x_1 + 2x_2 - 8 \leq 0$  et  $x_1^2 + x_2 \geq 0$ .

1. Déterminer l'ensemble des points de  $\mathbf{R}^2$  pour lesquels les deux contraintes sont actives (au même temps).
2. Enoncer les conditions de Karush, Kuhn et Tucker associées à  $(\mathcal{Q})$ .
3. Résoudre le problème  $(\mathcal{Q})$ .

*Exercice 93.* Vérifier que les points  $\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$  et  $\begin{pmatrix} -1/3 \\ 2/3 \\ 2/3 \end{pmatrix}$  sont solutions des conditions de Karush, Kuhn et Tucker du problème d'optimisation avec contraintes :

$$(\mathcal{R}) = \begin{cases} \min x_2 + x_3 \\ \text{s/c } x_1 + x_2 + x_3 = 1 \\ \text{et } x_1^2 + x_2^2 + x_3^2 = 1 \end{cases}$$

*Indications :* déterminer les multiplicateurs de Lagrange  $\lambda_1$  et  $\lambda_2$  associés à  $(\mathcal{R})$ .

*Exercice 94.* Répondre par vrai ou faux.

1. La méthode du gradient à pas optimal est une méthode de descente. Justifier votre réponse.
2. La méthode du gradient conjugué est une méthode itérative qui permet de résoudre des systèmes d'équations linéaires. Justifier votre réponse.

*Exercice 95.* Soit le problème de programmation non linéaire de  $\mathbf{R}^2$  :

$$(\mathcal{P}) = \begin{cases} \min \frac{1}{2}x_1^2 + \frac{1}{2}x_2^2 - 2x_1 + x_2 + 3 \\ \text{s/c } x_1 \leq 1 \\ \text{et } x_2 \geq 0 \end{cases}$$

1. Déterminer le Lagrangien associé au problème  $(\mathcal{P})$ .

2. Le problème  $(\mathcal{P})$  admet-il une solution ? est-elle unique ? Justifier votre réponse.
3. Posons  $x_0 = \begin{pmatrix} 0.8 \\ -1 \end{pmatrix}$ . Calculer les points  $x_1$  et  $x_2$  avec la méthode du gradient projeté à pas fixe  $r = 0.1$ .
4. Citer deux autres algorithmes qui permettent de résoudre le problème  $(\mathcal{P})$ .

*Exercice 96.* Soit  $x = (x_1, x_2, x_3) \in \mathbf{R}^3$ . Posons  $J(x) = \frac{1}{2}(Ax, x) - (b, x)$  avec

$$A = \begin{pmatrix} 1 & -1 & -1 \\ -1 & 4 & -1 \\ -1 & -1 & 4 \end{pmatrix} \quad \text{et} \quad b = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}.$$

Nous considérons le problème d'optimisation avec contraintes :

$$(\mathcal{P}) = \begin{cases} \min J(x) \\ x \in C \end{cases}$$

avec  $C = \{x = (x_1, x_2, x_3) \in \mathbf{R}^3 \mid h(x) = x_1 + x_2 = 0, \quad g(x) = x_1 - 2x_2 \leq 0\}$ .

1. Vérifier que le polynôme caractéristique de  $A$  est égal à

$$P_A(x) = (x - 5)(x^2 - 4x + 1) \quad .$$

2. Le problème  $(\mathcal{P})$  admet-il une solution ? Est-elle unique ?
3. Qu'est-ce qu'une contrainte active ?
4. Donner la définition d'un point régulier (on parle aussi de condition de qualification des contraintes).
5. Ecrire les conditions de Karush, Kuhn et Tucker
6. Donner une caractérisation de l'optimum de  $(\mathcal{P})$  c'est-à-dire une condition nécessaire et suffisante d'optimalité.
7. Calculer l'optimum de  $(\mathcal{P})$ .
8. Citer trois algorithmes qui s'appliquent dans ce cas précis.

*Exercice 97.* On désire construire la plus grande boîte possible de surface donnée  $S$ . Déterminer  $x$ ,  $y$  et  $z$  : la longueur, la largeur et la hauteur de cette boîte. *Indications : il faut modéliser le problème, donner les conditions d'optimalité et résoudre.*

*Exercice 98.* Soient  $A$  et  $b$  définis par

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

et  $J(y) = \frac{1}{2}(Ay, y) + (b, y)$  où  $y \in \mathbf{R}^n$ . Soit le problème d'optimisation avec contraintes :

$$(P) = \begin{cases} \min & J(y) \\ s/c & y_1 \geq 0 \\ & y_2 + y_3 = 0 \end{cases}$$

1. Le problème  $(P)$  admet-il une solution ? est-elle unique ?
2. Ecrire les conditions d'optimalité KT que doit vérifier l'optimum.
3. Résoudre le problème  $(P)$ .

*Exercice 99.* Soient  $A = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 3 \end{pmatrix}$  et  $b = \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix}$ .

1. Résoudre

$$\min \frac{1}{2}(Ax, x) - (b, x)$$

sous les contraintes  $x_1 \geq 1$  et  $x_2 - 2x_3 = 1$

2. Comparer avec la solution du problème sans contraintes.

*Exercice 100.* On désire construire la plus grande boîte rectangulaire de surface donnée  $S = 6m^2$ .

1. Modéliser le problème et énoncer les conditions d'optimalité
2. Déterminer  $x$ ,  $y$  et  $z$  : la longueur, la largeur et la hauteur de cette boîte.

*Exercice 101.* Une usine de contenants a reçu une commande de boîtes rectangulaires ouvertes de  $1 m^3$ . Le coût de fabrication estimé est de 0.600D le mètre carré pour la base, 0.450D le mètre carré pour le devant et l'arrière et 0.410D le  $m^2$  pour les deux autres côtés. Quelle doivent être les dimensions de la boîte afin de minimiser le coût de production ?

*Exercice 102.* Une agence publicitaire doit concevoir une affiche ayant une surface totale de  $0.8m^2$ . Le client désire que la surface totale imprimée soit maximale, tout en conservant des marges de  $2.5cm$  en haut et en bas et de  $1cm$  de chaque côté. Quelles doivent être les dimensions de l'affiche ?

*Exercice 103.* Minimiser la fonction  $J(x) = 2x_1^2 + 2x_1x_2 + x_2^2 - 10x_1 - 10x_2$  sous les contraintes  $g_1$  et  $g_2$  avec  $g_1(x) := x_1^2 + x_2^2 \leq 5$  et  $g_2(x) := 3x_1 + x_2 \leq 6$ .

*Indication : on peut remarquer que  $J$ ,  $g_1$  et  $g_2$  sont convexes.*

1. Ecrire les conditions de Kuhn et Tucker
2. Vérifier qu'il y a quatre cas possible sur (a)
3. Montrer que la seule solution possible de ce problème d'optimisation avec contraintes est  $x^* = (1, 2)$ .

*Exercice 104.* Trouver les extremums relatifs de la fonction  $J(v_1, v_2) = v_2^2$  par rapport à l'ensemble

$$U = \{v = (v_1, v_2) \in \mathbf{R}^2; v_1^2 + v_2^2 - 2 = 0\}.$$

*Exercice 105.* Trouver les extremums relatifs de la fonction  $J(v_1, v_2) = v_1^2 + (v_2 - 1)^2$  par rapport à l'ensemble

$$U = \{v = (v_1, v_2) \in \mathbf{R}^2; v_2^2 = 0\}.$$

*Exercice 106.* On désire construire la plus grande boîte possible de surface donnée  $S = 2m^2$ . Déterminer  $L$ ,  $l$  et  $h$  : la longueur, la largeur et la hauteur de cette boîte.

*Exercice 107.* Maximiser la fonction  $J(x, y) = 4x - x^2 + y - y^2 + 1$  sous les contraintes  $x + y \leq 2$  et  $x + 2y \leq 1$ . On précisera l'existence et l'unicité et on dessinera l'ensemble réalisable  $S$ . Appliquer l'algorithme de pénalisation extérieure.

*Exercice 108.* Résoudre

$$\min \frac{1}{2}(Ax, x) - (b, x)$$

sous les contraintes  $x_1 \geq 0$  et  $x_1 - x_2 - x_3 = 1$  avec  $A = \begin{pmatrix} -3 & 4 & 0 \\ -2 & 3 & 0 \\ -2 & 2 & 1 \end{pmatrix}$  et  $b = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ .

Comparer avec la solution sans contraintes.

*Exercice 109.* Etudier la convexité ou la concavité des fonctions définies par :

1.  $f(x) = 3x_1^2 + 2x_2^2 - 8x_1$
2.  $f(x) = x_1^2 - 10x_1x_2 + 9x_2^2$
3.  $f(x) = -x_1^4 + x_2x_1^2 + x_1^2 + x_2 - 3$ .

*Exercice 110.* Dans  $\mathbf{R}^2$ , trouver le point qui minimise la distance entre  $x = y = 0$  et la courbe  $y^2 = (x - 1)^3$ . Vérifier qu'il n'existe pas de multiplicateurs de Lagrange.

*Exercice 111.* Trouver le rayon de la sphère de  $\mathbf{R}^3$  centrée en 0 tangente au plan  $u_1 + u_2 + u_3 = 1$ .

*Exercice 112.* Soit  $A$  une matrice symétrique d'ordre  $n$ ,  $B$  une matrice symétrique définie positive d'ordre  $n$  et  $b$  un vecteur de  $\mathbf{R}^n$ . Enoncer une condition nécessaire d'extremum relatif de la fonctionnelle

$$J : v \in \mathbf{R}^n \longrightarrow J(v) = \frac{1}{2}(Av, v) - (b, v)$$

par rapport à l'ensemble

$$U = \{v \in \mathbf{R}^n; (Bv, v) = 1\} \quad .$$

*Exercice 113.* Minimiser  $f(x_1, x_2) = x_1^2 + x_2^2 - x_1 + x_2$  sous les contraintes  $g_1(x) := x_1^2 + x_2^2 \leq 6$  et  $g_2(x) := 3x_1 + 5x_2 \leq 5$ .

*Exercice 114.* Minimiser  $\frac{1}{2}(u_1 - 1)^2 + \frac{1}{2}(u_2 - 2)^2$  sous les contraintes  $u_1 - u_2 \leq 1$ ,  $u_1 + u_2 \leq 2$ ,  $-u_1 \leq 0$  et  $-u_2 \leq 0$ .

*Exercice 115.* Trouver les optima locaux sur  $\mathbf{R}^2$  par la règle des multiplicateurs de Lagrange :

1.  $y = 2x_1^2 + x_2^2$  sous la contrainte  $x_1x_2 = 2$
2.  $y = x_1x_2$  sous la contrainte  $2x_1^2 + x_2^2 = 2$ .

*Exercice 116.* Donner une représentation graphique de l'ensemble des points admissibles pour les problèmes suivants :

1.  $\min f(x) = x_1^2 + 4x_2^2$  s/c  $x_1 + x_2 = 9, x \in \mathbf{R}^2$ ,
2.  $\min f(x) = x_1x_2$  s/c  $x_1 + x_2 + x_3 = 1, x_1 - x_2 = 0, x \in \mathbf{R}^3$ ,
3.  $\min f(x) = x_1x_2 + x_1^2 + 4x_2^2$  s/c  $x_1 + x_2 = 2, x_1 - x_2^2 = 0, x \in \mathbf{R}^2$ .

*Exercice 117.* 1. Tracer les courbes de niveau pour les fonctions définies sur  $\mathbf{R}^2$  par

$$f(x) = |x_1 - 1| + |x_2 - 2|.$$

2. Trouver par une méthode graphique le max de  $f$  en présence de la contrainte  $h(x) = x_1^2 + x_2^2 - 2 = 0$ .
3. Vérifier qu'en ce point, les conditions suffisantes d'optimum sont satisfaites.
4. Trouver par une méthode graphique le min de  $f$  en présence de la contrainte  $h(x) = 0$ .
5. Montrer qu'en ce point, la condition d'optimum du premier ordre ne peut être utilisée pour le caractériser.

Soit  $J(x) = -\frac{3}{2}x_1^2 + x_1x_2 + \frac{3}{2}x_2^2 - x_3x_1 + x_3x_2 + \frac{1}{2}x_3^2 + 2x_1 - 3x_2$  pour  $x = (x_1, x_2, x_3)$ .

1. Calculer le gradient de  $J$  en  $x_0 = (0, 1, 0)$ .
2. Déterminer l'ensemble des points critiques de  $J$ .
3. Calculer les matrices hessiennes sur les points stationnaires. Donner la nature de ces matrices.
4. Conclure sur les extremums de  $J$ .
5. Vérifier que la fonction est strictement convexe.
6. Montrer que la fonction  $J$  est coercive.
7. Montrer que  $\nabla J$  est une fonction lipshitzienne.
8. Donner l'énoncé du théorème de convergence de la méthode du gradient conjugué.
9. Pour  $x_0 = (0, 1, 0)$ , calculer les deux premiers itérés  $x_1$  et  $x_2$  de la méthode du gradient (le pas est à déterminer).

*Exercice 118.* Refaire le même exercice avec  $J(x) = -\frac{1}{2}x_1^2 - \frac{1}{2}x_1x_2 - \frac{1}{2}x_1x_3 + \frac{1}{2}x_2^2 - x_3x_2 - \frac{1}{2}x_3^2 + 2x_1 - 2x_3$  pour  $x = (x_1, x_2, x_3) \in \mathbf{R}^3$ .

# Annexe A

## Applications sur Python

### A.1 Optimisation unidimensionnelle

*Exercice 119.* Considérons la fonction  $f(x) = x^4 + 3x^2 + 1$  pour tout  $x \in \mathbf{R}$ . Déterminer les points stationnaires de  $f$  par la méthode de Newton et par la méthode de la sécante. En quoi consiste la méthode de dichotomie ?

*Exercice 120.* Soit  $J(x, y) = x^4 + 3x^2y^2 + x^2 + y^4$  pour tout  $x, y \in \mathbf{R}$ . Calculer le gradient de  $J$  en  $(x, y) \in \mathbf{R}^2$  ainsi que la matrice hessienne de  $J$ . Est-ce que le point  $(0, 0)$  est un extrémum de  $J$ . Refaire cet exercice avec  $J(x, y) = 3x \exp(y) - x^3 - \exp(3y)$ ,  $J(x, y) = -(x^2 - 1)^2 - (x^2y - x - 1)^2$ ,  $J(x, y) = 9 - 2x + 4y - x^2 - 4y^2$ ,  $J(x, y) = x^3y + 12x^2 - 8y$ ,  $J(x, y) = \exp(x) \cos(y)$  et  $J(x, y) = x^2 + y^2 + \frac{1}{x^2y^2}$ .

*Exercice 121.* Vérifier que les formes quadratiques suivantes sont des fonctions convexes :  $J(x_1, x_2, x_3) = x_1^2 - 6x_1x_2 + 10x_2^2 + 5x_3^2$ ,  $J(x_1, x_2, x_3, x_4) = 2x_1^2 + 2x_1x_4 + 8x_3x_2 + 3x_4^2$ ,  $J(x_1, x_2, x_3) = x_1^2 + x_1x_3 + x_2^2 + 4x_3^2$  et  $q(x_1, x_2, x_3) = x_1^2 + 10x_2^2 + 5x_3^2 + 2x_1x_3 - 2x_2x_3 - 6x_1x_2$ .

### A.2 Minimisation sans contraintes

Programmer les algorithmes suivants : Gradient à pas constant, Algorithme de Newton, Relaxation avec sous programme Newton (pour  $\mathbf{R}$ ). Les tester en particulier sur les exemples  $f$  suivants :  $f(x, y) = x^2 - 5xy + y^4 - 25x - 8y$ ,  $f(x, y) = (x^4 - 3)^2 + y^4$ ,  $f(x) = x_1^4 - 4x_1^3 + 6(x_1^2 + x_2^2) - 4(x_1 + x_2)$ . Gradient conjugué pour  $J(x_1, x_2) = x_1^2 + x_2^2 - 5$ .

### A.3 Application aux réseaux de neurones

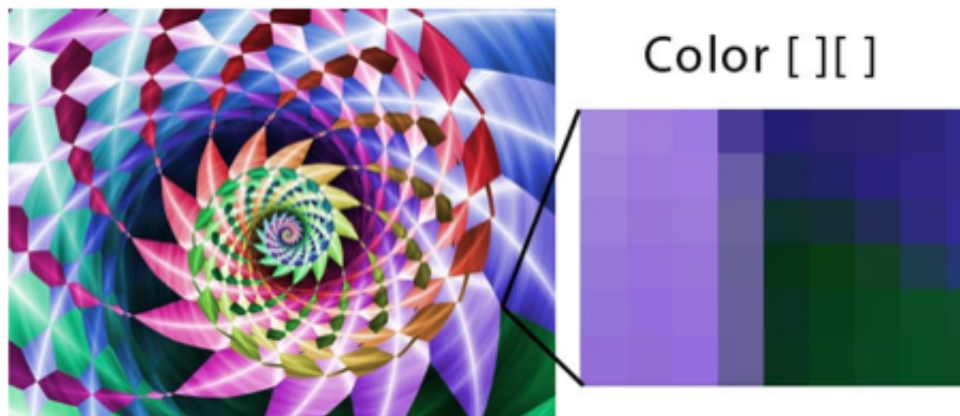
L'objectif de ce TP est d'effectuer des simulations de réseaux de neurones. D'abord, un réseau de neurone formel puis un réseau de neurones multicouches.

Programmer les algorithmes suivants : Gradient projeté, Algorithme de Newton-Lagrange, Pénalisation extérieure.

#### A.3.1 Nnets

Une image peut être représentée par un tableau à deux dimensions  $x_i$  et  $y_j$ , où chaque élément contient l'information sur la couleur  $z_{i,j}$  du pixel correspondant.

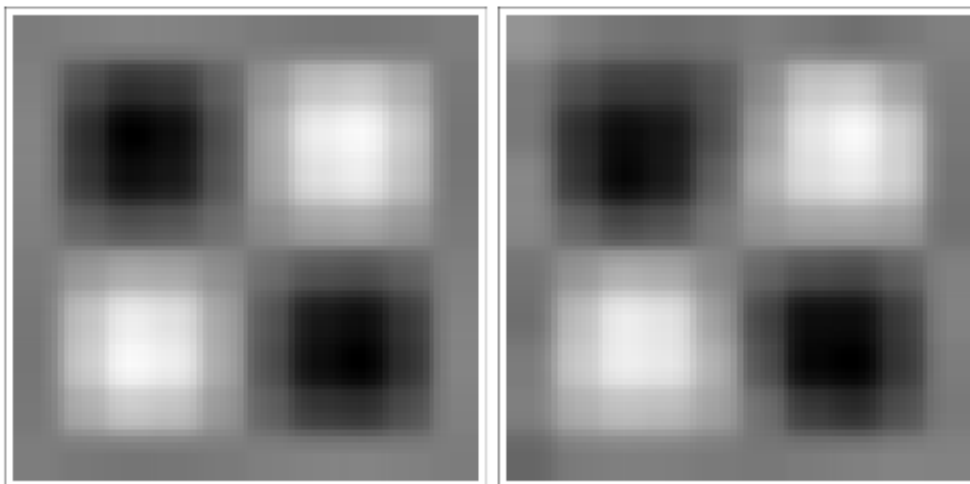




Ces informations correspondent à l'Entrée du réseau de neurones artificiels. On sélectionne une image pour laquelle on souhaite que le Nnets reconnaisse. Initialiser le réseau de neurones avec des poids aléatoires. Apprendre à ce réseau à reconnaître l'image et ceci en fonction de l'erreur  $E(w, \theta) = \sum \sum (o(x_i, y_j) - z_{i,j})^2$ .

Check the results of the trained neural network

Compare original (left) with NN output (right):



Une fois entraîné, on peut utiliser le RdN à reconnaître d'autres images. C'est une application directe de la reconnaissance de visages.