Tehnici de optimizare
Laborator 1: Introducere in toolbox-ul de optimizare Matlab

1 Introducere

Formularea standard a unei probleme de optimizare:

$$
\begin{aligned}
f^* = \min_{x \in \mathbb{R}^n} \quad & \text{sau} \ \max_{x \in \mathbb{R}^n} f(x) \\
\text{s.l.:} \quad & Cx \le d, \quad g(x) \le 0 \\
& Ax = b, \quad h(x) = 0 \\
& x \in S
\end{aligned}
\tag{1}
$$

unde:

- x - variabila de decizie
- $f(x)$ - functia obiectiv
- S - multimea fezabila de baza (de exemplu $S = [a, b]$ -se mai numeste si constrangere de tip box)
- $Cx \le d$, $g(x) \le 0$ - constrangeri de inegalitate
- $Ax = b$, $h(x) = 0$ - constrangeri de egalitate
- $P = \{x \in S | \, Ax = b, \, h(x) = 0, \, Cx \le d, g(x) \le 0\}$ - multimea fezabila
- x^* - punctul de optim
- $f^* = f(x^*)$ -valoarea de optim

O clasificare naturala a problemelor de minimizare este:

1. **Probleme fara constrangeri**: $\min_{x \in \mathbb{R}^n} f(x)$.

   In acest caz, multimea fezabila este spatiul Euclidian $\mathbb{R}^n$

2. **Probleme cu constrangeri**: Formularea problemei 1

   Dupa natura functiei obiectiv, mai putem identifica urmatoarele tipuri de probleme:

- **Problema LP** (*eng. Linear Programing*): functia obiectiv si constrangerile sunt lineare.
- **Problema QP** (*eng. Quadratic Programing*): functia obiectiv este patratica si constrangerile sunt liniare

1.1 Probleme fara constrangeri

x=**fminunc**(fun,x_0) - *returneaza un punct de minim local al functiei fun, cu ajutorul informatiei de ordinul I (gradient) si a celei de ordin II (Hessiana).*

Variabila *fun* reprezinta functia obiectiv diferentiabila ce trebuie furnizata ca o variabila de tip *function handle.* O variabila de tip *function handle* poate fi, e.g. un fisier objfun.m care primeste la intrare variabila de decizie x si returneaza valoarea functiei in x. Vectorul x_0 reprezinta punctul initial de unde porneste procesul de cautare al minimului.

Pentru mai multe detalii, apelati la una din cele doua comenzi ajutatoare din Matlab: **help** si **doc**.

Exemplu: Consideram problema de optimizare:

$$\min_{x\in\mathbb{R}^2} f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

Sa se rezolve problema (sa se gaseasca un punct de minim local) utilizand rutina fminunc, cu punctul de initializar $x_0 = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$

Solutie: Se creaza fisierul objfun.m ce va contine:

```
function f=objfun(x)
f = 100*(x(2)-x(1)^2)^2 + (1-x(1))^2;
```

Se apeleaza rutina fminunc in urmatorul mod:

```
x0=[-1;2];
x=fminunc(@objfun,x0)
```

Gasiti in anexa versiunea in care functia este declarata in acelasi fisier cu fminunc.

1.2 Probleme cu constrangeri

x=**fmincon**(fun,x_0,C,d,A,b,l,u,nonlcon,optiuni) - *returneaza un punct de minim local pentru problema de optimizare 1*

Variabila **fun** reprezinta functia obiectiv ce trebuie furnizata ca o variabila de tip *function handle.*

Matricele si vectorii **C, d,A, b, l, u** definesc *constrangerile liniare* cu structura prezentata in modelul 1 -cu mentiunea ca S este o constrangere de tip box ($l \le x \le u$)

Variabila de tip handle **nonlcon** returneaza *constrangerile neliniare* de egalitate si inegalitate reprezentate in 1(adica $g(x)$ si $h(x)$);

Vectorul x_0 reprezinta *punctul initial* de unde porneste procesul de cautare al minimului

Variabila **optiuni** reprezinta setul de optiuni specific fiecarei rutine MATLAB. Comanda *optiuni=optimset('fmincon')* afiseaza setul de optiuni implicit. Fiecare dintre acestea se poate modifica in functie de problema de minimizare.

Pentru mai multe detalii, apelati: **help fmincon**.

Exemplu: Consideram problema de optimizare:

$$\begin{aligned} &\min_{x \in \mathbb{R}^2} f(x) \quad \left(= e^{x_1}\left(4x_1^2 + 2x_2^2 + 4x_1x_2 + 2x_2 + 1\right)\right) \\ &\text{s.l.: } g(x) \leq 0 \quad \left(\begin{cases} x_1x_2 - x_1 - x_2 + 1.5 \leq 0 \\ -x_1x_2 - 10 \leq 0 \end{cases}\right) \end{aligned}$$

Sa se rezolve local problema utilizand rutina fmincon, cu punctul de initializare $x_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

Solutie: Se creeaza fisierul objfun.m ce defineste functia obiectiv si confun.m in care se definesc constrangerile:

```
function f=objfun(x)
f=exp(x(1))*(4*x(1)^2+2*x(2)^2+4*x(1)*x(2)+2*x(2)+1);

function[g,h]=confun(x)
g=[x(1)*x(2)-x(1)-x(2)+1.5 -x(1)*x(2)-10];
h=[];
```

Apoi se apeleaza rutina pentru minimizare cu restrictii:

```
x0=[-1,1];
optiuni=optimset(fmincon);
optiuni.LargeScale=off;
x=fmincon(@objfun,x0,[],[],[],[],[],[],@confun,optiuni)
```

1.3 Problema LP

x = **linprog**(c,C,d,A,b,lb,ub,x_0,optiuni) - *rezolva probleme de programare liniara, ce vizeaza minimizarea unei functii obiectiv liniare, in prezenta unui set de constrangeri liniare.*

O problema de programare liniara are urmatoarea formulare standard:

$$\begin{aligned} &\min_{x \in \mathbb{R}^n} c^T x \\ &\text{s.l.: } Cx \leq d \\ &\qquad Ax = b \\ &\qquad lb \leq x \leq ub \end{aligned} \tag{2}$$

unde $c, lb, ub \in \mathbb{R}^n, d \in \mathbb{R}^m, b \in \mathbb{R}^p, C \in \mathbb{R}^{m \times n}, A \in \mathbb{R}^{p \times n}$

Exemplu: Formulati problema LP pentru o clasificare binara si rezolvati problema utilizand functia linprog.
Solutie: Clasificarea binara constituie procesul de separare a elementelor unui set in doua grupuri pe baza unei reguli de clasificare. Astfel ca pentru clasificarea unei multimi de dimensiune $N(\{(z_1, y_1), ..., (z_N, y_N)\})$ operam cu urmatoarele notiuni:

- $z_i \in \mathbb{R}^n$ - vectorul de caracteristici al instantei i ($z \in \mathbb{R}^{N\times n}$);
- $y_i \in \{-1, 1\}$ - eticheta unei instantei i;

Presupunem pentru simplitate ca setul de date este impartit in:

- primele n_1 date pentru $y = 1$,
- restul $n_2 = N - n_1$ pentru $y = -1$

Aceasta ipoteza este usor realizabila in etapa de prelucrare a datelor.

Pentru regula de clasificare, consideram ca setul de date este separabil linear (vezi figura 1) si introducem *teorema de separare a hiperplanelor*:

Fie doua multimi A si B, subseturi convexe [1] din $\mathbb{R}^n$. Daca $\exists\, w \in \mathbb{R}^n$ nenul si un numar real b astfel incat

$$\begin{aligned} w^T x \geq b, &\quad \forall x \in A \\ w^T y \leq b, &\quad \forall y \in B \end{aligned}$$

atunci hyperplanul $w^T z = b$, $\forall z \in A \cup B$ separa cele doua multimi A si B

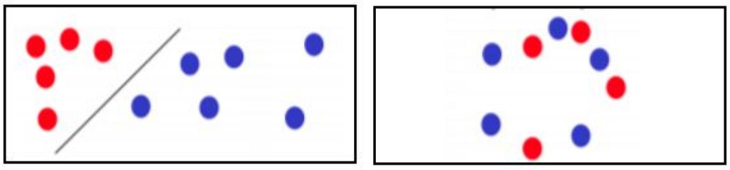

Figure 1: Stanga: 2 multimi separabile linear; Dreapta: multime neseparabila linear

Asadar avem regula de clasificare:

$$\begin{cases} w^T z_i - b \geq 0, \, \forall y_i = 1 \\ w^T z_j - b \leq 0, \, \forall y_j = -1 \end{cases} \tag{3}$$

Prin prisma acestei alegeri, gasirea clasificatorului binar se echivaleaza cu gasirea ponderilor $w = [w_1, \dots, w_n]^T$ si a deplasarii b ce definesc hiperplanul de separare $w^T z - b = 0$.

Intrucat inegalitatile din 3 sunt omogene, le putem rescrie:

$$\begin{cases} w^T z_i - b \geq t, \, \forall y_i = 1 \\ w^T z_j - b \leq -t, \, \forall y_j = -1 \end{cases} \tag{4}$$

[1] Detalii despre aceasta notiune la curs

Deoarece dorim un t cat mai mare, problema LP se formuleaza astfel:

$$\begin{aligned} &\max_{w,b,t} t \\ &\text{s.l.: } w^T z_i - b \geq t, \forall y_i = 1 \\ &\qquad w^T z_j - b \leq -t, \forall y_j = -1 \end{aligned} \tag{5}$$

Observatie! Putem remarca ca functiile Matlab rezolva in general probleme de minimizare, astfel ca trebuie sa aducem problema de maximizare la una de minimizare. Acest lucru este posibil, intrucat avem urmatoarea relatie: $\max f(x) = -min - f(x)$.

Formularea din 5 trateaza cazul cand cele doua multimi sunt complet separabile. Dar aceasta situatie este un caz ideal, rar intalnit in realitate. Astfel ca s-a introdus o relaxare a constrangerilor 4 ($t = 1$):

$$\begin{cases} w^T z_i - b \geq 1 - u_i, \forall y_i = 1 \\ w^T z_j - b \leq -(1 - v_j), \forall y_j = -1 \end{cases}$$

pentru $\forall u_i, v_j \geq 0$. Observam ca pentru $u = v = 0$ recuperam cazul separabil.
Prin urmare dorim vectorii u si v sa fie cat mai mici:

$$\begin{aligned} &\min_{w,b,u,v} 1^T u + 1^T v \\ &\text{s.l.: } w^T z_i - b \geq 1 - u_i, \forall y_i = 1 \\ &\qquad w^T z_j - b \leq -(1 - v_j), \forall y_j = -1 \\ &\qquad u \geq 0;\ v \geq 0 \end{aligned} \tag{6}$$

Pentru un grad de generalitate mare, vom apela functia lingprog pentru problema 6. Comparama forma acesteia cu forma standard LP 2 si obtinem:

Variabila de decizie $x = \begin{bmatrix} w \\ b \\ u \\ v \end{bmatrix} \in \mathbb{R}^{(n+1+N)}$, $c = \begin{bmatrix} zeros(n,1) \\ 0 \\ ones(N,1) \end{bmatrix}$.

Restrictii de egalitate nu avem deci $A = [\,]$, $b = [\,]$.
Constrangerile de inegalitate trebuiesc prelucrate:

$$\begin{cases} w^T z_i - b \geq 1 - u_i, \forall y_i = 1 \\ w^T z_j - b \leq -(1 - v_j), \forall y_j = -1 \end{cases} \Rightarrow \begin{cases} -w^T z_i + b - u_i \leq -1, \forall y_i = 1 \\ w^T z_j - b - v_j \leq -1, \forall y_j = -1 \end{cases}$$

Deci:
$C = \begin{bmatrix} -z(1:N/2,:) & ones(N/2,1) & -I_{N/2} & O_{N/2} \\ z(N/2+1:N,:) & -ones(N/2,1) & O_{N/2} & -I_{N/2} \end{bmatrix}$, $d = -ones(N,1)$

Pentru x nu avem un prag superior, astfel ca $u = [\,]$, dar pentru u si v avem conditiile $u \geq 0$ si $v \geq 0$. Prin urmare $l = \begin{bmatrix} -inf \\ \vdots \\ -inf \\ zeros(N,1) \end{bmatrix} \in \mathbb{R}^{(n+1+N)\times 1}$

In final apelam: $x =$ *linprog(c,C,d,A,b,l,u)*.

1.4 Problema QP

x = **quadprog**(Q,q,C,d,A,b,l,u,x_0,optiuni) - *rezolva probleme de programare patratica, ce vizeaza minimizarea unei functii obiectiv patratice, in prezenta unui set de constrangeri liniare.*

O problema de programare patratica are urmatoarea formulare standard:

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} \frac{1}{2}x^TQx + q^Tx \\ &\text{s.l.: } Cx \leq d \\ &\qquad\; Ax = b \\ &\qquad\; l \leq x \leq u \end{aligned} \tag{7}$$

unde $Q = Q^T \in \mathbb{R}^{n\times n}, q, l, u \in \mathbb{R}^n, d \in \mathbb{R}^m, b \in \mathbb{R}^p, C \in \mathbb{R}^{m\times n}, A \in \mathbb{R}^{p\times n}$

Exemplu: Formulati problema QP pentru un clasificator binar si rezolvati-o utilizatand functia quadprog.
Solutie: Definim ca si in exemplul din sectiunea LP

- N - numarul de date
- $z_i \in \mathbb{R}^n$ - vectorul caracteristic instantei i ($z \in \mathbb{R}^{N\times n}$)
- $y_i \in \{-1,\, 1\}$ eticheta instantei i

De asemenea presupunem ca datele sunt separabile linear si ca sunt aranjate astfel: primele $N/2$ date sunt etichtate cu 1 si restul cu -1

Introducem pentru regula de separare tehnica *Support Vector Machine* (SVM). Aceasta calculeaza parametri hiperplanului de separare cu ajutorul vectorilor suport, de unde si numele algoritmului de invatare automata. Vectorii de suport sunt acele puncte de date cele mai apropiate de hiperplanul de separare (a se vedea figura 2). Atentia asupra acestora este motivata de faptul ca ele sunt cele mai greu de clasificat.

Asadar facem urmatoarea notatie:

- $H_1 : w^Tz - b = 1$ -dreapta pe care se gasesc punctele etichetate cu 1 si cele mai apropiate de hiperplanul de separare;
- $H_2 : w^Tz - b = -1$ -dreapta pe care se gasesc punctele etichetate cu -1 si cele mai apropiate de hiperplanul de separare;

Scopul SVM-ului este de a gasi hiperplanul care ofera cea mai buna margine, adica cea mai mare distanta dintre H_1 si H_2. Cum gasim aceasta margine? Pentru aceasta amintim distanta de la un punct z_i la o drapta $w^Tz - b$:

$$d = \frac{|w^Tz_i - b|}{\|w\|}$$

Astfel distanta de la un punct z_+ de pe H_1 la hiperplanul de separare este

$$d_+ = \frac{|w^Tz_+ - b|}{\|w\|} = \frac{|1|}{\|w\|} = \frac{1}{\|w\|}$$

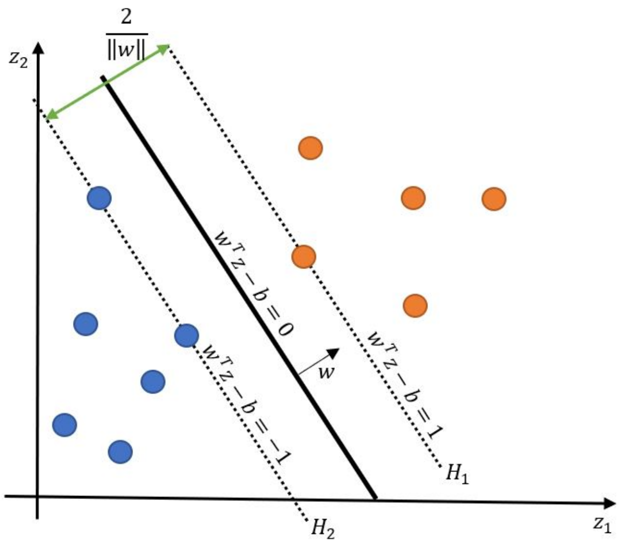

Figure 2: Rezentarea SVM pentru $z \in \mathbb{R}^2$ (punctele albastre au eticheta -1, iar cele portocalii 1)

Procedand in aceeasi maniera pentru punctele de pe H_2 obtinem:

$$d_- = \frac{|w^T z_- - b|}{\|w\|} = \frac{|-1|}{\|w\|} = \frac{1}{\|w\|}$$

Prin urmare gasim ca marginea este: $d_+ + d_- = \frac{2}{\|w\|}$
Cu conditia ca nu avem puncte intre H_1 si H_2 rescriem relatia 3:

$$\begin{cases} w^T z_i - b \geq 1, \, \forall y_i = 1 \\ w^T z_j - b \leq -1, \, \forall y_j = -1 \end{cases} \Rightarrow \begin{cases} y_i(w^T z_i - b) \geq 1 \\ y_j(w^T z_j - b) \geq 1 \end{cases}$$

Asadar am gasit problema de optimizare:

$$\begin{aligned} &\max_{w,b} \frac{2}{\|w\|} & & \min_{w,b} \frac{\|w\|^2}{2} \\ &\text{s.l.: } y(w^T z - b) \geq 1 & \Leftrightarrow \quad & \text{s.l.: } y(w^T z - b) \geq 1 \end{aligned} \tag{8}$$

Echivalenta problemelor este sustinuta de faptul ca maximizarea $\frac{2}{\|w\|}$ presupune gasirea celui mai mic w. Remarcam ca problema obtinuta este un QP.

Continuam prin a rescrie problema 8 pentru cazul neseparabil, relaxand restrictiile:

$$\begin{aligned} &\min_{w,b.v} \frac{1}{2}\|w\|^2 + \lambda \, 1^T v \\ &\text{s.l.: } y_i(w^T z_i - b) \geq 1 - v_i, i = 1 : N \\ &\qquad v \geq 0, \end{aligned} \tag{9}$$

unde $\lambda \in \mathbb{R}$ este o pondere care decide influenta componentei liniare din functia obiectiv.

Comparand 9 cu formularea standard 7 gasim:

Variabila de decizie $x = \begin{bmatrix} w \\ b \\ v \end{bmatrix} \in \mathbb{R}^{(n+1+N)}$

Functia obiectiv: $Q = \begin{bmatrix} I_n & O_n \\ O_{1+N} & O_{1+N} \end{bmatrix}, \quad q = \lambda \begin{bmatrix} zeros(n,1) \\ 0 \\ ones(N,1) \end{bmatrix}.$

Restrictii de egalitate nu avem deci $A = [\,]$, $b = [\,]$.
Constrangerile de inegalitate trebuiesc prelucrate:

$$y_i(w^T z_i - b) \geq 1 - v_i, \, \forall i = 1 : N \Rightarrow -y_i w^T z_i + y_i b - v_i \leq -1, \, \forall i = 1 : N$$

Deci: $C = \begin{bmatrix} -y. * z(1:N,:) & y & -I_N \end{bmatrix}$, $d = -ones(N,1)$
Pentru x nu avem un prag superior, astfel ca $u = [\,]$, dar pentru v avem conditia $v \geq 0$.

Prin urmare $l = \begin{bmatrix} -inf \\ \vdots \\ -inf \\ zeros(N,1) \end{bmatrix} \in \mathbb{R}^{(n+1+N)\times 1}$

In final apelam: $x =$*quadprog(Q,q,C,d,A,b,l,u)*

2 Aplicatii

1. Exersati comenzile *fmincon* si *fminunc* implementand exemplele din sectiunile 1.1 si 1.2 din laborator.

2. Fie setul de date de mai jos:

   unde

z^1	0	0	1	2	1	4	1	2	3	4	14	12	10	13	14	14	1	7	9	6
z^2	5	1	1	0	2	6	3	3	2	5	10	5	10	8	8	6	24	8	10	10
y	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	1	1	1	1	1	1	1	1	1	1

- z^1 reprezinta numarul de prezente al unui student la curs;
- z^2 este numarul de ore dedicate proiectului de catre student;
- y este eticheta studentului cu semnificatia -1 pentru picat, respectiv 1 pentru promovat;

Cerinte:

(a) Plotati setul de date. Cum este acesta?

(b) Calculati clasificatorul din formularea LP pentru cazul neseparabil si plotati hiperplanul de separare gasit. Adaugati punctele $[z^1, z^2, y] = \{[2; 5; -1], [15; 11; -1]\}$; si recalculati clasificatorul. Ce puteti spune despre u si v ?

(c) Calculati clasificatorul din formularea QP cu $c = 0.001$ pentru cazul neseparabil si plotati hiperplanul de separare gasit. Schimbati valoarea lui $\lambda = \{0.1; 1\}$ si recalculati clasificatorul. Ce observati?

3 Anexa

3.1 Probleme fara constrangeri

```
%% fminunc
% min: 100(x2-x1^2)^2 + (1-x1)^2
% x0 = [-1,2]
%%
f = @(x) 100*(x(2) -x(1)^2)^2 + (1 -x(1))^2;
x0 = [-1 2];
sol =fminunc(f,x0)
```