

---

# Raport științific

Proiect PN-III-P1-1.1-PD-2016-0127

Prelucrarea eficientă a semnalelor biomedicale folosind  
codarea rară (SCEBIOS)

Etapă 1, 2018

dr.ing. Nicolae Cleju

## Cuprins

<b>Semnale și reprezentări rare</b>	<b>3</b>
Norma $\ell_p$ . . . . .	3
Semnale rare . . . . .	3
Reprezentări rare . . . . .	4
<b>Codarea rară a semnalelor</b>	<b>5</b>
Convexitate . . . . .	5
Reprezentări aproximative . . . . .	6
Achiziția comprimată a semnalelor . . . . .	7
<b>Algoritmi pentru codarea rară a semnalelor</b>	<b>8</b>
Programare liniară (Basis Pursuit)} . . . . .	8
Orthogonal Matching Pursuit . . . . .	9
Algoritmul <i>Smoothed</i> L0 . . . . .	11
<b>Algoritmi de învățare a dicționarelor</b>	<b>13</b>
Algoritmul K-SVD . . . . .	13
Algoritmul de antrenare continuă (Mairal <i>et al.</i> ) . . . . .	16
<b>Utilizări ale codării rare în procesarea semnalelor biomedicale</b>	<b>17</b>
Clasificare pe baza codării rare . . . . .	17
Clasificare prin codare rară cu dicționar unic . . . . .	18
Detecția fibrilației atriale folosind codarea rară a semnalelor ECG . . . . .	18
Algoritm de clasificare liniară optimizată de tip K-SVD . . . . .	19
<b>Bibliografie</b>	<b>21</b>

## Semnale și reprezentări rare

### Norma $\ell_p$

Una dintre direcțiile fundamentale de cercetare în domeniul prelucrărilor de semnale o reprezintă eficientizarea aplicațiilor, sub toate aspectele. În domeniul reprezentării semnalelor, eficiența se referă la posibilitatea reprezentării datelor și semnalelor într-un mod cât mai compact posibil, fără a sacrifica acuratețea reprezentării.

Din punct de vedere matematic, instrumentul matematic pentru măsurarea unui vector îl reprezintă funcția *normă*. În spațiile vectoriale de tip Lebesgue, care reprezintă cadrul matematic fundamental pentru modelarea datelor în prelucrările de semnal, norma  $\ell_p$  se definește astfel:

$$\|\mathbf{x}\|_p = \left( \sum_i |x_i|^p \right)^{\frac{1}{p}} \quad (1)$$

Relația de mai sus definește o funcție de tip normă pentru  $\forall p \geq 1$ . În particular, pentru  $p = 2$  se obține norma Euclidiană obișnuită. Pentru  $p < 1$ , relația nu mai definește în mod riguros o normă, întrucât nu se mai respectă proprietățile necesare de subaditivitate și/sau omogenitate. Cu toate acestea, ea rămâne extrem de utilă și folosită în literatura științifică din domeniul prelucrărilor de semnal.

În cazul limită al valorii  $p = 0$  se obține pseudo-norma  $\ell_0$ , definită ca fiind numărul de elemente diferite de zero ale vectorului:

$$\|\mathbf{x}\|_0 = \sum_i c_i, \text{ unde } c_i = \begin{cases} 1, x_i \neq 0 \\ 0, x_i = 0 \end{cases} \quad (2)$$

Deși, pentru  $p < 1$ , relațiile de mai sus nu definesc norme în sens matematic matematic, întrucât nu satisfac toate proprietățile necesare, ele sunt totuși utile în problemele de optimizare întâlnite de obicei în prelucrări de semnal. Din acest motiv, admitând un abuz de limbaj, ne vom referi totuși la ele cu termenul de normă, situație care se întâlnește relativ frecvent în literatura de specialitate.

### Semnale rare

În general, în domeniul prelucrărilor de semnal un semnal  $\mathbf{x}$  se consideră semnal rar dacă are un număr redus de coeficienți dominanți. Acest lucru poate fi în mod convenabil măsurat folosind norma  $\ell_p$  a semnalului,  $\|\mathbf{x}\|_p$ , pentru o oarecare valoare  $p < 2$ : un semnal este rar dacă norma sa

$\ell_p$  are valoare mică. Necesitatea de a alege un  $p < 2$  derivă din penalizarea impusă de ridicarea la puterea  $p$  din (1) asupra coeficienților de valoare mare sau mică, în contextul unei valori totale reduse a normei  $\ell_p$ . Efectul ridicării la puterea  $p$  este explicat în cele ce urmează:

- pentru  $p = 2$ , în norma  $\ell_2$  se ridică la pătrat fiecare element. În condițiile menținerii unei valori finale reduse, ridicarea la pătrat penalizează mai puternic coeficienții de valoare ridicată decât pe cei de valoare redusă, întrucât ridicarea la pătrat crește mai mult contribuția termenilor de valoare mare. Din acest motiv, vectorii cu norma  $\ell_2$  de valoare mică tind să aibă coeficienți de valoare mică, fără coeficienți dominanți, astfel încât aceștia nu sunt, în general, rari;
- valori  $p < 2$  reduc penalizarea impusă coeficienților de valoare mare, ceea ce permite ca vectorii să conțină și coeficienți de valoare mai ridicată, dominanți;
- pentru  $p = 1$ , toți coeficienții unui semnal sunt tratați în mod similar, norma  $\ell_1$  reprezentând suma valorilor absolute ale coeficienților, fără alte ridicări la putere;
- pentru  $p < 1$ , expresia normei  $\ell_p$  penalizează mai mult coeficienții de valoare redusă decât pe cei de valoare ridicată, lucru care încurajează ca energia semnalelor să fie concentrată în câțiva coeficienți de valoare mare, dominanți, restul coeficienților fiind de valori neglijabil. Aceste norme încurajează în acest fel raritatea semnalelor;
- în cazul limită când  $p = 0$ , norma  $\ell_0$  din (2) reprezintă numărul coeficienților nenuli. Prin urmare, o valoare redusă a normei  $\ell_0$  înseamnă în mod riguros că semnalul conține doar un număr redus de componente, fiind așadar un semnal rar. Aceasta coincide cu definiția uzuală a termenului de “vector rar” sau “matrice rară” așa sunt ele întâlnite, de exemplu, în informatică și știința calculatoarelor (conceptul opus fiind de vectori sau matrici *dense*).

## Reprezentări rare

În majoritatea cazurilor, semnalele de interes nu sunt ele însele rare, ci doar admit o descompunere rară într-o bază aleasă în mod convenabil. De altfel, într-un spațiu vectorial, orice reprezentare a unui semnal se face în raport cu o anumită bază vectorială, fie exprimată explicit, fie subînțeleasă a fi baza canonică. Un semnal  $\mathbf{x}$  se consideră semnal cu reprezentare rară dacă admite o reprezentare rară  $\gamma$  în raport cu o bază  $\mathbf{D}$ . Considerând  $\mathbf{x} \in \mathbb{R}^n$  (vector coloană) și  $\mathbf{D} \in \mathbb{R}^{n \times n}$  (cu vectorii bazei aranjați în coloane), acest lucru se exprimă sub forma:

$$\mathbf{x} = \mathbf{D}\gamma, \text{ cu } \|\gamma\|_p = \text{mic}. \quad (3)$$

Condiția de semnal rar se impune așadar asupra descompunerii  $\gamma$ , care trebuie să aibă energia concentrată într-un număr redus de coeficienți dominanți. Semnalul  $\mathbf{x}$  reprezintă așadar o

combinație liniară a vectorilor bazei  $\mathbf{D}$  (uzual denumiți atomi), în care doar câțiva vectori au o contribuție dominantă.

În ultimele două decenii, a suscitat un real interes subiectul reprezentărilor în dicționare supra-complete de semnale. Un dicționar supracomplet  $\mathbf{D} \in \mathbb{R}^{n \times N}$  reprezintă o colecție de vectori (atomi) în număr  $N$  mai mare decât dimensiunea spațiului vectorial. Din acest motiv, atomii dicționarului nu pot fi toți liniar independenți, ceea ce înseamnă că un semnal  $\mathbf{x}$  poate avea, în general, o infinitate de descompuneri  $\gamma$  în raport cu dicționarul.

Prin extensie a relației (3), un semnal  $\mathbf{x}$  se consideră semnal cu reprezentare rară în raport cu un dicționar  $D$  dacă admite o descompunere rară  $\gamma$  în raport cu dicționarul, cu valoare redusă a normei  $\ell_p$ . Deosebirea esențială față de cazul unei baze este că  $\gamma$  este acum doar una dintre multiplele descompuneri posibile ale lui  $x$ . Dintre toate aceste descompuneri, se pune așadar problema găsirii descompunerii  $\gamma$  de normă  $\ell_p$  minimă, care va determina în ce măsură este semnalul  $\mathbf{x}$  cu reprezentare rară sau nu. Aceasta se poate formula ca o problemă de optimizare cu constrângeri, a cărei rezolvare constituie procesul de codare rară, prezentat în cele ce urmează.

## Codarea rară a semnalelor

Prin codare rară se înțelege procesul de găsim, pentru un semnal  $\mathbf{x} \in \mathbb{R}^n$ , a celei mai rare descompuneri  $\gamma$  în raport cu un dicționar supracomplet de semnale  $\mathbf{D} \in \mathbb{R}^{n \times N}$ . Acest proces poate fi exprimat sub forma unei probleme de optimizare cu constrângeri:

$$\gamma^* = \arg \min_{\gamma} \|\gamma\|_p \quad \text{s.t.} \quad \mathbf{x} = \mathbf{D}\gamma \quad (4)$$

Această problemă poate fi reformulată în diverse variante, în funcție de aspectele discutate în cele ce urmează.

## Convexitate

Alegerea valorii  $p$  în definiția normei  $\ell_p$  folosite în (4) are un impact major asupra modalității de soluționare a problemei, din următorul motiv:

- pentru  $p \geq 1$ , relația (4) reprezintă o problemă de optimizare convexă. Acest lucru permite utilizarea algoritmilor și a întregii literaturi științifice dezvoltate pentru rezolvarea optimizărilor convexe.
- valori  $p < 1$  conduc la optimizarea unei funcții non-convexe, ceea ce face mult mai dificil procesul de codare rară.

Aceste considerente fac ca în literatura de specialitate sa fie folosite practic doar două valori ale lui  $p$ ,  $p = 1$  sau  $p = 0$ :

- $p = 1$  reprezintă valoarea minimă a lui  $p$  care încă permite ca optimizarea să fie convexă. Acest lucru garantează existența unei singure soluții optime local/global, pentru găsirea acestuia beneficiind de întreg aparatul matematic și de algoritmi dezvoltati pentru optimizări convexe. În schimb, soluția de normă  $\ell_1$  minimă nu este neapărat și cea mai bună (cea mai rară), întrucât norma  $\ell_1$  nu încurajează raritatea descompunerii pe cât o fac normele cu  $p < 1$ . Problema de optimizare se formulează astfel:

$$\gamma^* = \arg \min_{\gamma} \|\gamma\|_1 \quad \text{a.î. } \mathbf{x} = \mathbf{D}\gamma \quad (5)$$

- $p = 0$  reprezintă valoarea utilizată de obicei atunci când se acceptă o optimizare non-convexă, întrucât asigură constrângerea cea mai riguroasă a rarității. Minimizarea normei  $\ell_0$  este însă o problemă NP-hard [1] în general, astfel încât nu există algoritmi eficienți pentru găsirea optimului global. În schimb, există în literatura de specialitate o varietate de algoritmi de rezolvare conduc la o soluție aproximativă (o descompunere rară, dar nu neapărat *cea mai rară*), soluție care coincide cu soluția optimă doar dacă dicționarul  $\mathbf{D}$  respectă anumite condiții suplimentare, specifice fiecărui algoritm. Problema de minimizare  $\ell_0$  se formulează astfel:

$$\gamma^* = \arg \min_{\gamma} \|\gamma\|_0 \quad \text{a.î. } \mathbf{x} = \mathbf{D}\gamma \quad (6)$$

## Reprezentări aproximative

Un aspect care trebuie luat în calcul este că semnalele reale pot fi avea reprezentări doar *aproximativ rare*. Este posibil ca mare parte dintr-un semnal să poată fi reprezentată cu un număr redus de termeni, dar o mică parte din energia semnalului să necesite în descompunere, în mod legitim, un număr mare de coeficienți foarte mici, dar nenuli. Acest lucru se întâmplă în cazul semnalelor afectate de zgomot, zgomotul fiind imposibil de reprezentat în general în mod compact.

Pentru a trata și această situație, constrângerea exactă  $\mathbf{x} = \mathbf{D}\gamma$  din relațiile (5) și (6) se înlocuiește cu o constrângere pătratică de forma  $\|\mathbf{x} - \mathbf{D}\gamma\|_2 \leq \epsilon$ . ceea ce conduce la expresiile următoare:

$$\gamma^* = \arg \min_{\gamma} \|\gamma\|_1 \quad \text{a.î. } \|\mathbf{x} - \mathbf{D}\gamma\|_2 \leq \epsilon \quad (7)$$

$$\gamma^* = \arg \min_{\gamma} \|\gamma\|_0 \quad \text{a.î. } \|\mathbf{x} - \mathbf{D}\gamma\|_2 \leq \epsilon \quad (8)$$

În acest fel, se caută descompunerea  $\gamma$  de normă minimă care aproximează semnalul în limita

unei toleranțe  $\epsilon$  permise. Acest fapt asigură robustețe la zgomot și alte eventuale inexactități ale semnalelor, și este importantă în special în cazul folosirii normei  $\ell_0$ , care este sensibilă la orice coeficienți nenuli (minimizarea normei  $\ell_0$  cu constrângeri exacte este instabilă în prezența zgomotului [2]).

Există și alte variante de formulare a codării rare, de exemplu în formă neconstrânsă, integrând ambii termeni în funcția optimizată:

$$\gamma^* = \arg \min_{\gamma} \|\mathbf{x} - \mathbf{D}\gamma\|_2 + \lambda \|\gamma\|_1 \quad (9)$$

Subliniem că toate aceste variante exprimă, în fond, același deziderat: obținerea unei descompuneri cât mai rare, care să reprezinte cât mai exact semnalul original. Forma particulară care se alege depinde de caracteristicile aplicației și de algoritmul de soluționare folosit, și nu este de obicei un factor critic.

### Achiziția comprimată a semnalelor

Una din aplicațiile fundamentale ale codării rare, care a alimentat dezvoltarea teoriei asociată acestui domeniu și constituie exemplul său de referință, o reprezintă achiziția comprimată a semnalelor (*“compressed sensing”*)[3].

Dat fiind un semnal  $\mathbf{x} \in \mathbb{R}^n$  care are o reprezentare rară  $\gamma$  într-o bază sau un dicționar  $\mathbf{D}$ , acesta poate fi achiziționat printr-o serie de măsurători liniare, exprimate sub formă de produs scalar cu un număr redus  $m < n$  de vectori aranjați ca linii ale unei matrici de proiecție  $\mathbf{P} \in \mathbb{R}^{m \times n}$ . Măsurătorile, posibil afectate de un zgomot necunoscut  $\mathbf{z}$ , constituie un vector de măsurători  $\mathbf{y}$ . Acest proces de achiziție este descris de ecuația:

$$\mathbf{y} = \mathbf{P}\mathbf{x} + \mathbf{z} = \mathbf{P}\mathbf{D}\gamma + \mathbf{z} \quad (10)$$

Achiziția comprimată exploatează faptul că semnalul  $\mathbf{x}$  poate fi recuperat din vectorul de măsurători  $\mathbf{y}$ , mai mic ca dimensiune decât semnalul original, prin găsirea descompunerii rare  $\gamma$  a lui  $\mathbf{y}$  în dicționarul efectiv  $\mathbf{D}_{ef} = \mathbf{P}\mathbf{D}$ , care este aceeași cu descompunerea rară a lui  $\mathbf{x}$  în  $\mathbf{D}$ . Deși  $\mathbf{y}$  are dimensiunea mai mică decât  $\mathbf{x}$ , și deci ecuația (10) reprezintă un sistem supracomplet (nedeterminat) cu o infinitate de soluții posibile în general, vectorul  $\gamma$  rămâne cea mai rară soluție a sistemului, și deci poate fi identificat prin codarea rară a lui  $\mathbf{y}$ . Pentru ca acest lucru să fie posibil este necesar ca matricea de achiziție  $\mathbf{P}$  și dicționarul  $\mathbf{D}$  să respecte o serie de condiții de incoerență [2],[4], [5], care au fost studiate în amănunt în teoria achiziției comprimate.

Importanța achiziției comprimate rezidă în faptul că numărul de măsurători necesare pentru achiziționarea informației din semnal,  $m$ , este redus, prin exploatarea informației *a priori* că

semnalul are o reprezentare suficient de rară într-un dicționar  $\mathbf{D}$ . Subliniem ca reprezentarea rară nu este cunoscută, singura informație necesară este că ea există. Mai mult, este posibil ca nici dicționarul  $\mathbf{D}$  să nu fie cunoscut în etapa de achiziție, el fiind necesar doar la reconstrucția ulterioară a semnalului  $x$ .

Achiziția comprimată a semnalelor este utilă în aplicații unde semnalele sunt de mari dimensiuni, dar informația utilă este concentrată într-un număr redus de componente. Exemplul cel mai relevant îl reprezintă imagistica medicală prin rezonanță magnetică [], unde achiziția comprimată poate reduce timpul de achiziție al imaginilor, aspect de o importanță deosebită pentru minimizarea riscului de expunere la radiații al pacienților.

## Algoritmi pentru codarea rară a semnalelor

Prezentăm în această secțiune o serie de algoritmi des utilizați în literatură pentru obținerea codărilor rare ale semnalelor. Altfel spus, acești algoritmi rezolvă diversele formulări ale problemei de optimizare prezentate anterior.

Subliniem faptul că dacă dicționarul  $\mathbf{D}$  folosit pentru reprezentare este de fapt o bază, adică conține atomi liniar independenți în număr egal cu dimensiunea spațiului vectorial, atunci problema nu pune în general dificultăți deosebite. În acest caz orice semnal din acest spațiu vectorial are doar o singură reprezentare, rară sau nu, care se poate obține prin tehnicile uzuale (de exemplu folosind baza bi-ortogonală atașată lui  $\mathbf{D}$ ).

Dificultatea apare în condițiile unui dicționar supracomplet  $\mathbf{D}$ , care permite o infinitate de reprezentări posibile. În acest caz devine critic ca algoritmul folosit să poată selecta cea mai rară reprezentare. Uzual, fiecare algoritm garantează obținerea soluției optime în anumite condiții specifice impuse asupra dicționarului.

## Programare liniară (Basis Pursuit)}

Problemele de optimizare de tip *programare liniară* reprezintă o clasă foarte bine studiată de probleme de optimizare convexă. Programarea liniară se definește ca minimizarea unei funcții liniare cu constrângeri liniare de egalitate și inegalitate de asemenea liniare. Problemele de programare liniară pot fi formulate sub diverse forme, forma standard fiind următoarea [6]:

$$\mathbf{d}^* = \arg \min_{\mathbf{d}} \sum_{p=0}^{N-1} w_p \cdot d_p \text{ cu } \mathbf{c} = \mathbf{P} \cdot \mathbf{d} \text{ and } d_i \geq 0, \quad (11)$$

Prin ecuația de mai sus se caută așadar un vector  $\mathbf{d}$ , cu elemente ne-negative, care minimizează o funcție liniară exprimată ca un produs scalar cu un vector de ponderi  $\mathbf{w}$ , între toți vectorii



care satisfac constrângerea liniară impusă  $\mathbf{c} = \mathbf{P} \cdot \mathbf{d}$ .

Programarea liniară reprezintă unul dintre cele mai simple tipuri de optimizare convexă, și ca atare este foarte bine studiată în matematică, existând o serie de algoritmi clasici, bine cunoscuți, pentru rezolvarea lor (de exemplu algoritmi de tip *simplex*, *primal-dual* sau *cu punct interior*). Din acest motiv nu vom prezenta mai multe detalii despre modul lor de operare.

În ceea ce privește codarea rară, programarea liniară se folosește pentru a rezolva problema de optimizare (5) ce utilizează norma  $\ell_1$ , cunoscută și sub numele de *Basis Pursuit*. Ecuația (5) poate fi rescrisă sub forma unei probleme de programare liniară de dimensiune dublă tip(11) [7], prin introducerea unor variabile alternative  $u_p \geq 0$  și  $v_p \geq 0$ , astfel încât  $\mathbf{y} = \mathbf{u} - \mathbf{v}$ , cu definițiile de mai jos (notația  $\square$  semnifică concatenare):

$$\begin{aligned}\mathbf{P} &= [\mathbf{D}, -\mathbf{D}] \\ w_p &= 1, \forall p \\ \mathbf{d} &= \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} \\ \mathbf{c} &= \mathbf{x}\end{aligned}$$

Cu aceste definiții ale parametrilor, optimizarea din (11) se poate rezolva cu unul dintre algoritmii existenți în literatură. Odată obținută soluția  $\mathbf{d}$ , descompunerea rară a semnalului,  $\gamma$ , se obține ca diferența dintre  $\mathbf{u} - \mathbf{v}$ . Aceasta satisface condiția inițială de reprezentare  $\mathbf{x} = \mathbf{D} \cdot \gamma$ , întrucât  $\mathbf{x} = \mathbf{c} = \mathbf{P} \cdot \mathbf{d} = \mathbf{D} \cdot \mathbf{u} - \mathbf{D} \cdot \mathbf{v} = \mathbf{D}(\mathbf{u} - \mathbf{v}) = \mathbf{D} \cdot \gamma$ .

## Orthogonal Matching Pursuit

Familie de algoritmi *Matching Pursuit* reprezintă o categorie importantă de algoritmi de codare rară, dintre care se distinge algoritmul *Orthogonal Matching Pursuit*, cel mai des utilizat.

Algoritmii de acest tip sunt utilizați pentru a rezolva probleme de optimizare cu constrângeri tip  $\ell_0$ , (6) și (8), care permit utilizarea doar a unui număr limitat de atomi în reprezentarea unui semnal. În aceste situații, identificarea reprezentării rare  $\gamma$  se reduce, în fapt, la identificarea suportului său, uzual notat cu  $T$ , adică indicii atomilor care sunt efectiv utilizați în reprezentare, având un coeficient nenul. Știind suportul  $T$  al unui semnal, valoarea coeficienților descompunerii rare se poate afla fără dificultăți prin descompunere  $\ell_2$  obișnuită.

Ideea fundamentală a algoritmilor din familia *Matching Pursuit* este de a efectua, la fiecare iterație, o căutare de tip *greedy* după cel mai reprezentativ atom din dicționar, care se adaugă la suportul curent. Pornind de la o listă de atomi inițial nulă, aceștia se adaugă unul câte unul, adăugând mereu pe cel mai promițător atom de la fiecare iterație, până când sunt în număr

suficient pentru a reprezenta întregul semnalul. Optimizările tip  $\ell_0$  fiind NP-hard, cu acești algoritmi se obțin în general doar soluții aproximative, suficient de rare, dar nu neapărat cele mai rare. În anumite condiții specifice impuse asupra dicționarului, se poate garanta de obicei că soluțiile obținute sunt identice cu cele optime.

Algoritmul cel mai reprezentativ din familia aceasta este *Orthogonal Matching Pursuit* (OMP) [8],[9], prezentat mai jos.

---

**Algoritmul 1** Orthogonal Matching Pursuit (OMP)

---

- 1:  $\mathbf{T}^0 \leftarrow \emptyset$
  - 2:  $\gamma^0 \leftarrow \mathbf{0}$
  - 3: **repetă**
  - 4:   Se calculează reziduul curent  $\mathbf{r}^k \leftarrow \mathbf{x} - \mathbf{D}\gamma^{k-1}$
  - 5:   Se găsește atomul  $\mathbf{d}_m \in \mathbf{D}$  cu corelație maximă  $\langle \mathbf{r}^k, \mathbf{d}_m \rangle$
  - 6:   Se actualizează suportul prin adăugarea indicelui  $m$ :  $\mathbf{T}^k \leftarrow \mathbf{T}^{k-1} \cup \{m\}$
  - 7:   Se proiectează  $\mathbf{x}$  pe semnalele din suportul  $\mathbf{T}^k$ , obținându-se soluția rară la iterația  $k$ :  
 $\gamma_{\mathbf{T}^k} = \mathbf{D}_{\mathbf{T}^k}^\dagger \mathbf{x}$
  - 8: **până când** criteriu de oprire, de exemplu  $\|\mathbf{r}^k\|_2 \leq \epsilon$  sau număr impus de iterații
  - 9: **returnează**  $\gamma^k$
- 

La inițializare, suportul  $\mathbf{T}^0$  este vid, iar soluția curentă este 0, indicând faptul că nici un atom nu este încă folosit în reprezentare. Algoritmul OMP adaugă, la fiecare iterație, un nou atom la setul suport. Fiecare iterație debutează prin calcularea reziduului, care reprezintă eroarea de aproximare a semnalului  $\mathbf{x}$  obținută cu soluția  $\gamma^{(k-1)}$  de la iterația precedentă:

$$\mathbf{r}^k = \mathbf{x} - \mathbf{D}\gamma^{k-1}.$$

La pasul următor se caută atomul  $\mathbf{d}_m$  care ar reduce, de unul singur, cât mai mult posibil acest reziduu. Această condiție înseamnă căutarea atomului din dicționar cel mai corelat cu  $\mathbf{r}^k$ , care se adaugă în continuare la setul suport  $\mathbf{T}^k$ . Coeficienții atomilor din setul suport se obțin prin proiectarea semnalului  $\mathbf{x}$  pe acesta, obținându-se astfel reprezentarea de eroare minimă folosind acești atomi. Atomii care nu fac parte din suport rămân cu coeficientul egal cu 0. Eroarea de aproximare în urma proiecției va fi utilizată apoi la pasul următor, pentru alegerea atomului următor.

Se observă că, la fiecare iterație, suportul  $\mathbf{T}$  crește cu un atom, iar eroarea de aproximare scade întrucât se face proiecție pe un număr din ce în ce mai mare de atomi. Acest lucru garantează convergența algoritmului, eroarea de aproximare ajungând la 0 într-un număr de iterații cel mult egal cu dimensiunea semnalului. Pentru o robustețe crescută la zgomot, algoritmul OMP se oprește atunci când eroarea de aproximare scade sub o toleranță acceptabilă  $\epsilon$ . Algoritmul OMP se poate rula și pentru un număr fixat de iterații,  $k$ , ceea ce asigură faptul că reprezentarea rară

va avea exact  $k$  coeficienți nenuli.

Familia *Matching Pursuit* cuprinde și alte variante de algoritm. Algoritmul *Matching Pursuit* (MP) [8] este o versiune mai simplă, dar și mai puțin eficientă. La fiecare iterație, se selectează un atom după aceeași regulă, diferența constând în modul de actualizare a coeficienților pentru atomii existenți deja în setul suport. La *Matching Pursuit*, când se selectează un atom pe baza corelației maxime, acest coeficient de corelație devine coeficientul corespunzător din vectorul de codare rară  $\gamma$ . Specific acestui algoritm este faptul că coeficienții atomilor selectați anterior nu se mai modifică. Întrucât atomii nu sunt în general ortogonali, acest lucru face ca un atom să poate fi selectat de mai multe ori pe parcursul rulării algoritmului, ceea ce conduce la o rată de convergență slabă.

Spre deosebire de MP, la OMP are loc recalcularea coeficienților tuturor atomilor anteriori la fiecare adăugare a unui nou, prin proiecția ortogonală a semnalului pe suport. Acest lucru asigură ortogonalitatea erorii de aproximare pe setul atomilor din suport, ceea ce înseamnă că la fiecare iterație se va selecta un atom nou. Se garantează astfel convergența după un număr redus de iterații, în număr cel mult egal cu dimensiunea semnalului, dar fiecare iterație are complexitate de calcul crescută în raport cu dimensiunea semnalului ( $\mathcal{O}(n^2)$  față de  $\mathcal{O}(n)$  la MP), chiar și atunci când proiecția ortogonală se poate face prin algoritmi eficienți care refolosesc soluția precedentă pentru a reduce efortul de calcul.

Familia *Matching Pursuit* cuprinde și alți algoritmi înrudiți, precum *Compressive Sampling Matching Pursuit* (CoSaMP)[10], sau *Batch OMP*, care selectează mai mult de un atom la fiecare iterație. Aceștia sunt însă mult mai rar întâlniți în literatură, comparativ cu OMP.

## Algoritmul *Smoothed L0*

Algoritmul *Smoothed  $\ell_0$*  (SL0) [11] urmărește rezolvarea problemei de optimizare (6) a codării rare bazată pe norma  $\ell_0$ . Ideea fundamentală a algoritmului este înlocuirea funcției normă  $\ell_0$ , extrem de ne-convexă și deci dificil de minimizat, cu o secvență de funcții mai regulate, mai ușor de minimizat, care tind treptat către norma  $\ell_0$ .

Minimizarea normei  $\ell_0$  a descompunerii  $\gamma$ ,  $\|\gamma\|_0$ , se înlocuiește cu maximizarea unei funcții surogat definită în felul următor:

$$f(\gamma) = \sum_{i=1}^n e^{-\frac{\gamma_i^2}{2\sigma^2}}. \quad (12)$$

Funcția definită astfel poate fi văzută ca o aproximare mai netedă a complementului normei  $\ell_0$ : dacă varianța  $\sigma^2 \rightarrow 0$ , fiecare termen exponențial din definiția funcției tinde spre 0 dacă coeficientul corespunzător  $\gamma_i$  este nenul, dar tinde spre 1 dacă coeficientul este zero, iar valoarea funcției indică astfel numărul coeficienților egali cu 0. În aceste condiții, minimizarea normei  $\ell_0$

**Algoritmul 2** SL0

- 
- 1: Inițializare:  $\gamma^0 = \arg \min \|\gamma\|_2$  s.t.  $\mathbf{x} = \mathbf{D}\gamma$
  - 2: Inițializare: se alege o secvență descrescătoare  $\{\sigma_i\}$ ,  $1 \leq i \leq N$
  - 3: **pentru**  $i = 1 \rightarrow N$  **execută**
  - 4:   Inițializare cu soluția precedentă:  $\gamma^i \leftarrow \gamma^{i-1}$
  - 5:    $L$  pași de actualizare după gradient:
  - 6:   **pentru**  $k = 1, \dots, L$  **execută**
  - 7:     Se calculează gradientul  $\delta = [-s_1 e^{\frac{-s_1^2}{2\sigma^2}}, \dots, -s_n e^{\frac{-s_n^2}{2\sigma^2}}]^T$
  - 8:     Actualizarea soluției în direcția gradientului  $\gamma^i \rightarrow \gamma^i + \mu\delta$
  - 9:     Se impune constrângerea  $\mathbf{x} = \mathbf{A}\gamma$  prin proiecție ortogonală:

$$\gamma^i \leftarrow \gamma^i - \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}(\mathbf{D}\gamma^i - \mathbf{x})$$

10:   **sfârșit iterații**

11: **sfârșit iterații**

12: **returnează**  $\gamma^N$

---

a reprezentării rare echivalează cu maximizarea funcției, lucru pe care îl exploatează algoritmul.

Avantajul utilizării funcției surogat (12) este că, fiind continuă, se pot utiliza metode de optimizare de tip gradient. Însă, pentru valori mici ale lui  $\sigma^2$  funcția nu este convexă, așadar optimizarea bazată pe gradient se poate bloca în puncte de optim local. Pentru a evita acest lucru, autorii propun rezolvarea repetată a problemei folosind o secvență de valori ale  $\sigma^2$ : se pornește de la o valoare mare, care garantează convexitatea funcției; treptat, valoarea lui  $\sigma^2$  scade, dar se folosește soluția de la pasul precedent cu punct de pornire pentru noua problemă de optimizare, în ideea că pentru o scădere mică a  $\sigma^2$  soluția optimă rămâne în apropierea soluției precedente, încât poate fi găsită prin metode de tip gradient pornind de la aceasta, în ciuda faptului că funcția este în ansamblul ei ne-convexă.

Algoritmul complet, prezentat în figura alăturată, constă în maximizarea repetată a funcției  $f(\gamma)$  prin metoda *urcării pe gradient* (“steepest ascent”), urmată la fiecare iterație de o proiecție ortogonală pe spațiul soluțiilor  $\mathbf{x} = \mathbf{D}\gamma$ . Numărul de pași ai optimizării după gradient poate fi mic, întrucât soluția la fiecare iterație este folosită doar ca punct de plecare pentru iterația următoare, și deci poate fi doar aproximativă. Ca urmare, algoritmul este în general foarte rapid.

Un dezavantaj al algoritmului SL0 în forma prezentată aici este faptul că este sensibil la eventuale zgomote în reprezentare (reprezentări aproximative). O variantă robustă la zgomot este prezentată în [12].

## Algoritmi de învățare a dicționarului

În general, succesul oricărei aplicații bazate pe codarea rară depinde de calitatea dicționarului folosit pentru descompunere. Un dicționar adecvat trebuie să poată reprezenta în mod compact toate semnalele de interes, iar pentru aceasta este necesar ca toată trăsăturile particulare ale semnalelor să fie capturate în mod eficient de atomii dicționarului, în număr redus. Așadar, trebuie să existe o corespondență între semnalele de interes și dicționarul utilizat.

În unele aplicații se folosesc dicționare pre-determinate, construite pe baza unor familii de funcții standard, de exemplu dicționare de tip *wavelet* nedecimat (*undecimated wavelet*) [13]. Aceste dicționare se aleg *a priori* pe baza similitudinii cu semnalele de interes, și au avantajul simplității, și uneori și pe cel al eficienței, atunci când există algoritmi rapizi de calcul.

În multe aplicații, însă, este necesar a se estima dicționarul pornind de la un set de date de antrenare. Acest lucru permite adaptarea atomilor la semnalele particulare de antrenare, ceea ce de regulă conduce la performanțe îmbunătățite. În acest scop, în literatura de specialitate există mai mulți algoritmi care, pornind de la un set de semnale de intrare, urmăresc să obțină un dicționar pentru care descompunerile tuturor acestor semnale să fie rare.

În cele ce urmează, prezentăm o serie de algoritmi de învățare a dicționarului des întâlniți în literatura de specialitate.

### Algoritmul K-SVD

Algoritmul K-SVD [14] reprezintă unul dintre primii și cei mai utilizați algoritmi de învățare a dicționarului pentru codarea rară.

K-SVD este un algoritm iterativ, fiecare iterație fiind compusă din două etape:

1. Codarea rară a semnalelor de antrenare, considerând dicționarul curent la iterația  $i$ .
2. Actualizarea atomilor din dicționar, pe baza descompunerilor rare găsite la pasul anterior.

Algoritmul este sumarizat mai jos, și este explicat în detaliu în continuare.

Ca date de intrare, algoritmul necesită un set de semnale de antrenare  $\mathbf{X}$ . Dicționarul  $\mathbf{D}$  se inițializează cu valori aleatoare, sau cu combinații aleatoare ale semnalelor de antrenare.

La fiecare iterație, prima etapă o reprezintă codarea rară. Scopul acesteia este de a estima, pentru fiecare atom, care dintre semnalele de antrenare îl folosesc în descompunere. Numai aceste semnale vor fi apoi folosite pentru actualizarea atomului, fapt care permite atomilor să se specializeze pe anumite trăsături caracteristice unor subseturi de semnale. În urma codării se obține matricea descompunerilor rare la pasul  $i$ ,  $\mathbf{\Gamma}^{(i)}$ , în care fiecare coloană reprezintă descompunerea semnalului corespunzător din  $\mathbf{X}$ .

---

**Algoritmul 3** K-SVD

---

**Intrare** semnalele de antrenare  $\mathbf{X} \in \mathbb{R}^{n \times M}$ 

- 1: Inițializare aleatoare a dicționarului:  $\mathbf{D}^{(0)} \in \mathbb{R}^{n \times N}$
  - 2: **repetă**
  - 3:   Codare rară cu dicționarul curent la pasul  $i$ , cu raritatea impusă  $k$ :  $\mathbf{X} \approx \mathbf{D}^i \mathbf{\Gamma}^i$
  - 4:   Actualizarea dicționarului:
  - 5:   **pentru** fiecare atom  $\mathbf{d}_n = \mathbf{d}_1, \dots, \mathbf{d}_N$  **execută**
  - 6:     Se calculează setul  $\mathbf{X}_n$  al semnalelor relevante
  - 7:     Se scade contribuția celorlalți  $k - 1$  atomi, se obține matricea  $\mathbf{R}_n$
  - 8:     Se calculează descompunerea SVD a  $\mathbf{R}_n$ :  $\mathbf{R}_n = \mathbf{U}\mathbf{S}\mathbf{V}^T$
  - 9:     Se actualizează atomul cu cel mai semnificativ vector singular stânga  $\mathbf{d}_n \leftarrow \mathbf{u}_1$
  - 10:    Se actualizează coeficienții cu vectorul singular dreapta:
- $$\mathbf{\Gamma}_n^i \leftarrow \mathbf{v}_1^T \cdot S(1, 1)$$
- 11:   **sfârșit iterații**
  - 12: **până când** criteriu de oprire, de exemplu număr fix de iterații
  - 13: **returnează**  $\mathbf{\Gamma}^i$
- 

Subliniem că codarea rară se oprește după găsirea primilor  $k$  coeficienți ai descompunerii, în algoritmul original din [14] acest lucru fiind asigurat de algoritmul de codare folosit, respectiv Orthogonal Matching Pursuit, rulat doar pentru  $k$  iterații. Se obține astfel, în mod intenționat, doar o reprezentare aproximativă a semnalelor. Scopul este acela de a identifica principalele componente ale descompunerilor, care sunt reprezentate în mod compact prin câțiva atomi, și de a le separa de ceea ce rămâne în semnale, respectiv componentele ce nu au o reprezentare eficientă, necesitând un număr mai mare de atomi cu coeficienți de valori reduse. Actualizarea atomilor va avea în vedere aceste componente din urmă, urmărindu-se îmbunătățirea atomilor pentru a le reprezenta și pe acestea în mod mai compact.

A doua etapă a fiecărei iterații o reprezintă actualizarea dicționarului, care se face secvențial, pentru fiecare atom  $\mathbf{d}_n$  în parte. Pentru fiecare atom se folosesc doar semnalele care îl utilizează în reprezentare, grupate în matricea notată  $\mathbf{X}_n$ . Contribuția celorlalți  $k - 1$  atomi dominanți, rezultând din etapa de codare rară, este scăzută din fiecare semnal. Matricea reziduală obținută, notată  $\mathbf{R}_n$ , mai cuprinde doar:

- i. contribuția atomului  $\mathbf{d}_n$  însuși la semnalele din  $\mathbf{X}_n$ , și
- ii. componentele ne-dominante din semnale, pentru care nu există o reprezentare eficientă.

Se urmărește actualizarea atomului  $\mathbf{d}_n$  pentru a minimiza, într-un anume sens, matricea  $\mathbf{R}_n$ . Matematic, aceasta se formulează ca o problemă de aproximare matricială cu o matrice de rang

egal cu 1:

$$\mathbf{d}_n^* = \arg \min_{\mathbf{d}_n} \|\mathbf{R}_n - \mathbf{d}_n \alpha_n^T\|$$

Produsul  $\mathbf{d}_n \alpha_n^T$  formează o matrice de rang 1 (vector coloană înmulțit cu un vector linie) care minimizează distanța față de  $\mathbf{R}_n$ . Dacă se folosește norma Frobenius ca măsură a distanțelor, soluția este dată de descompunerea valorilor singulare (SVD), prin teorema clasică Eckart-Young. Dat fiind o descompunere SVD a matricii  $\mathbf{R}_n$

$$\mathbf{R}_n = \mathbf{U} \mathbf{S} \mathbf{V}^T,$$

cea mai bună aproximare de rang  $r$ ,  $\hat{\mathbf{R}}_n$ , în sensul minimizării normei Frobenius, este dată de restrângerea descompunerii SVD la cei mai semnificativi  $r$  vectori și valori singulare:

$$\hat{\mathbf{R}}_n = \mathbf{U}_r \mathbf{S}_r \mathbf{V}_r^T,$$

unde  $\mathbf{S}_r$  conține doar cele mai mari  $r$  valori ale lui  $\mathbf{S}$ , iar  $\mathbf{U}_r$  și  $\mathbf{V}_r$  conțin doar vectorii singulari corespunzători.

Ca urmare, atomul  $\mathbf{d}_n$  se va înlocui cu cel mai semnificativ vector singular stânga rezultat din descompunerea SVD,  $\mathbf{u}_1$ . Termenul din partea dreaptă,  $S_{11} \cdot \mathbf{v}_1^T$ , formează coeficienții corespunzători acestui nou atom în descompunerea semnalelor din  $\mathbf{X}_n$ , și ca atare va înlocui linia corespunzătoare din matricea  $\mathbf{\Gamma}^{(i)}$ .

Algoritmul rulează în mod obișnuit pentru un număr fixat de iterații, dar se pot utiliza și alte criterii de oprire, de exemplu atunci când descompunerile rare devin suficient de bune, sau când modificările atomilor devin suficient de mici.

Se observa că algoritmul K-SVD necesită o operație de descompunere SVD pentru fiecare atom, la fiecare iterație (de unde și numele “K-SVD”, prin analogie cu numele algoritmului de clusterizare *K-Means*). Acest lucru este poate fi prohibitiv din punct de vedere computațional, în special atunci când semnalele au dimensiuni mari. Totuși, întrucât se folosește numai cel mai semnificativ vector singular, se pot folosi algoritmi mult mai eficienți pentru o descompunere SVD parțială, care furnizează numai vectorul necesar fără a calcula întreaga descompunere (de exemplu, algoritmul Lanczos).

Este posibil ca, în urma iterațiilor, unii atomi să devină supra-specializați pe un număr prea mic de semnale particulare, ceea ce dăunează abilității generale de reprezentare a dicționarului. Pentru a evita acest lucru, dacă un atom este folosit de prea puține semnale în etapa de codare, atunci el este reinițializat în mod aleator.

### Algoritmul de antrenare continuă (Mairal *et al.*)

Un alt algoritm des întâlnit în literatură este algoritmul de antrenare continuă (“*online dictionary training*”) [15], [16], pe care îl vom denumi mai departe “algoritmul Mairal”, după numele autorului principal.

Ca și K-SVD, algoritmul Mairal este iterativ, fiecare iterație cuprinzând etapa de codare rară urmată de etapa de actualizare a dicționarului. Diferența principală este aceea că cele două etape se execută după fiecare semnal din setul de antrenare, și nu după parcurgerea întregului set ca în K-SVD (de unde și numele de antrenare “continuă”), actualizarea tuturor atomilor făcându-se așadar după fiecare semnal de intrare.

Pentru fiecare semnal  $\mathbf{x}_t$ , are loc mai întâi procesul de codare rară a acestuia cu dicționarul curent  $\mathbf{D}^{(t-1)}$ . Spre deosebire de K-SVD, în care se minimizează norma  $\ell_0$  a descompunerilor, în algoritmul Mairal codarea rară de face prin minimizarea normei  $\ell_1$ , sub formă neconstrânsă:

$$\gamma_t = \arg \min_{\gamma} \frac{1}{2} \|\mathbf{x}_t - \mathbf{D}_{t-1}\gamma\|_2^2 + \lambda \|\gamma\|_1. \quad (13)$$

Se observă că și aici există o eroare de reprezentare nenulă,  $\|\mathbf{x} - \mathbf{D}\gamma\|_2$

Urmează apoi actualizarea întregului dicționar  $\mathbf{D}^t$ , care se face prin minimizarea unei funcții de cost pătratică:

$$\mathbf{D}^t = \arg \min_{\mathbf{D}} \frac{1}{t} \sum_{i=1}^t \frac{1}{2} \|\mathbf{x}_i - \mathbf{D}\gamma_i\|_2^2 + \lambda \|\gamma_i\|_1. \quad (14)$$

Păstrând ficși coeficienții  $\gamma$  găsiți în urma codării rare, (14) urmărește obținerea unui nou dicționar care îmbunătățește eroarea de reprezentare a tuturor semnalelor  $\mathbf{x}_i$  observate până la momentul curent.

Rezolvarea ecuației (14) pentru obținerea întregului dicționar  $\mathbf{D}_t$  se face prin optimizarea repetată a fiecărui atom, secvențial, de fiecare dată actualizând un atom în timp ce ceilalți sunt considerați fixați (*block-coordinate descent* în raport cu atomii dicționarului). Fiecare actualizare a unui atom este o operație liniară de tip coborâre după gradient (*gradient descent*).

Avantajul principal al algoritmului Mairal față de K-SVD este că se înlocuiesc operațiile de descompunere SVD, costisitoare din punct de vedere computațional, cu operații liniare de tip coborâre după gradient, mult mai eficiente. Motivația care stă în spatele acestei alegeri este aceea că aproximările bazate pe descompunerile SVD, foarte precise dar cu efort mare de calcul, nu se justifică. Semnalele de antrenare  $\mathbf{x}_n$  reprezintă doar eșantioane din distribuția  $p(\mathbf{x})$  a clasei de semnale. În evaluarea dicționarului nu contează atât performanța de reprezentare a acestor semnale, cât performanța medie pentru alte semnale viitoare provenind din aceeași distribuție. Cu alte cuvinte, puterea de generalizare a dicționarului este mai importantă decât performanța



obținută pentru setul particular de antrenare folosit. Prin urmare, în locul optimizărilor costisitoare de tip SVD, efectuate după parcurgerea întregului set de date, se preferă optimizări inexacte, dar rapide, după fiecare semnal de antrenare. De asemenea, pentru eficiență sporită, optimizările atomilor se pot face și o dată la un grup mai mare de semnale (“*mini-batch*”).

Subliniem că aceleași motivații stau la baza utilizărilor algoritmilor de tip coborâre stohastică după gradient (“*stochastic gradient descent*”) utilizați în antrenarea sistemelor de tip rețele neurale artificiale: se preferă optimizări inexacte dar dese, după un număr redus de semnale, scopul fiind capacitatea de generalizare a modelului și nu supra-specializarea pe setul de antrenare.

În final, menționăm că algoritmul Mairal este implementat în biblioteca software *scikit-learn* [17], un pachet software *open-source* foarte popular care reunește diverși algoritmi din domeniul învățării automate, dezvoltat în limbajul de programare Python.

## Utilizări ale codării rare în procesarea semnalelor biomedicale

### Clasificare pe baza codării rare

Codarea rară poate fi folosită cu succes în aplicații de clasificare (a se vedea exemplele de mai jos). La baza acestor aplicații stă ideea că vectorul care conține descompunerea rară a unui semnal poate fi văzut ca un vector de trăsături ale semnalului, capabil a fi utilizat ca dată de intrare într-un algoritm clasificator.

Dat fiind un set de date sub forma coloanelor unei matrici  $\mathbf{X}$ , prin codare rară se obține matricea descompunerilor rare,  $\mathbf{\Gamma}$ , în care fiecare coloană  $\gamma$  reprezintă descompunerea rară corespunzătoare semnalului de pe aceeași poziție din  $\mathbf{X}$ . Descompunerea rară exprimă faptul că fiecare vector  $\mathbf{x}$  reprezintă o combinație liniară a unor componente reprezentate de atomii dicționarului folosit. Fiecare atom al dicționarului este considerat a fi o componentă particulară, specifică.

Prin urmare, pentru fiecare semnal  $\mathbf{x}$ , coeficienții descompunerii  $\gamma$  pot fi interpretați ca niște trăsături (“*features*”) care descriu absența sau prezența unor anumite componente din semnal: un coeficient de valoare mare indică faptul că semnalul conține componenta specifică reprezentată de atomul corespunzător, și în ce măsură, în timp ce un coeficient egal cu zero sau de valoare mică indică faptul că acea componentă lipsește din compoziția semnalului. Impunând un număr mic de coeficienți dominanți, codarea rară asigură faptul că acești coeficienți sunt relevanți, conținând mare parte din energia semnalului.

Vectorul descompunerii  $\gamma$  poate fi utilizat așadar ca vector de trăsături, folosit ca dată de intrare în algoritmi clasici (de exemplu SVM). O serie de exemple și aplicații specifice sunt prezentate mai jos.

## Clasificare prin codare rară cu dicționar unic

Unul din primii algoritmi folosind cu succes codarea rară pentru clasificarea datelor îl reprezintă cel din [18].

Dat fiind un set de date de antrenare  $\mathbf{X}$  cuprinzând date reprezentative din toate clasele  $c \in \mathcal{C}$ , pentru clasificarea unui semnal de test  $\mathbf{x}$  se procedează la codarea rară a lui  $\mathbf{x}$  în raport cu dicționarul  $\mathbf{X}$ :

$$\gamma^* = \arg \min_{\gamma} \|\gamma\|_1 \text{ a.î. } \mathbf{x} = \mathbf{X}\gamma.$$

Se observă faptul că setul de date de antrenare  $\mathbf{X}$  este utilizat pe post de dicționar de reprezentare, ceea ce are la bază ipoteza că semnalele care aparțin aceleiași clase aparțin unei suprafețe de dimensiuni reduse în spațiul vectorial, ceea ce permite ca un semnal să poate fi aproximat în mod rezonabil ca o combinație liniară a unui număr mic de semnale din aceeași clasă; această ipoteză se verifică pentru numeroase categorii de semnale.

Odată găsită descompunerea rară  $\gamma$ , clasificarea propriu-zisă se face pe baza atomilor care aproximează cel mai bine semnalul de test. Pentru fiecare clasă de semnal  $c \in \mathcal{C}$ , se restrânge reprezentarea rară doar la atomii din acea clasă,  $\mathbf{X}_c$ . Clasa a căror atomi reprezintă cu eroare minimă semnalul de test este cea selectată:

$$\hat{c} = \arg \min_{c \in \mathcal{C}} \|\mathbf{x} - \mathbf{D}_c \gamma_c\|_2$$

Deși autorii utilizează această metodă de clasificare pentru identificarea fețelor, abordarea este suficient de generală pentru a putea fi adaptată și la semnale de tip biomedical, de exemplu pentru semnale tip ECG [19].

## Detecția fibrilației atriale folosind codarea rară a semnalelor ECG

O aplicație de detecție a unei patologii medicale pe baza semnalelor de tip ECG este prezentată în [20]. Se urmărește detecția automată a fibrilației atriale a inimii pe baza semnalelor de tip ECG înregistrate cu un singur senzor, disponibile ca parte a competiției *PhysioNet/CinC Challenge 2017*. Clasificarea se face între 4 clase de semnal:

- ECG normal
- ECG cu fibrilație arterială prezentă
- semnal zgomotos, clasificare imposibilă
- semnal de tip necunoscut

Procesul de clasificarea urmează în mare o arhitectură clasică, compusă dintr-o etapă de pre-procesare, urmată de extragerea trăsăturilor și aplicarea unui clasificator tip SVM. În faza

de preprocesare are lor detecția undelor QRS, P și T, precum și extragerea intervalelor RR (intervalul între două unde R consecutive). Vectorul de trăsături conține numeroase statistici ale semnalului, în principal bazate pe intervalele RR (de exemplu media și dispersia intervalelor RR), dar principalul element de noutate îl constituie trăsăturile obținute din codarea rară.

Pentru codarea rară, se învață un dicționar specific semnalelor ECG analizate, folosind o variantă simplificată a algoritmului Mairal. Odată obținut dicționarul, codarea rară a semnalelor se face prin minimizarea normei  $\ell_1$  a descompunerilor. Vectorii descompunerilor obținuți prin codarea rară sunt utilizați apoi fără altă modificare pentru a antrena un clasificator de tip SVM liniar, pentru discriminarea celor 4 clase de semnale. Cele 4 scoruri obținute de acest clasificator pentru fiecare semnal, care exprimă probabilitatea ca semnalul să aparțină fiecărei clase, sunt apoi adăugate la vectorul de trăsături final, alături de trăsăturile clasice extrase din statisticile semnalului și ale unei R. Clasificatorul final este de tip SVM-RBF.

Se observă că, în soluția prezentată, codarea rară este utilizată ca modalitate de mapare neliniară din spațiul semnalelor în spațiul de dimensiune mai mare al descompunerilor. Utilizarea unui clasificator de tip SVM liniar pentru descompuneri indică ipoteza că, în acest spațiu de dimensiuni mai mari, clasele sunt liniar separabile într-o măsură mai mare decât în spațiul original. Abordarea este astfel similară cu un SVM bazat pe funcții kernel ("*kernel-based SVM*"), cu diferența că maparea neliniară se face în mod explicit, prin procesul de codare rară, fiind în mare măsură învățată din setul de date, prin învățarea dicționarului.

Reținem de asemenea ideea utilizării succesive a diferiți clasificatori: un clasificator inițial operează direct asupra descompunerilor rare, iar rezultatele obținute de acesta reprezintă o parte din trăsăturile de intrare pentru un al doilea clasificator, care produce rezultatele finale.

### Algoritm de clasificare liniară optimizată de tip K-SVD

În vederea unor aplicații de clasificare a semnalelor pe baza descompunerilor rare, algoritmi de antrenare a dicționarului pot fi extinși pentru a asigura și antrenarea parametrilor clasificatorului,

O astfel de soluție este propusă în [21]. Se urmărește antrenarea unui clasificator liniar  $\mathbf{W}$  pentru a aproxima matricea  $\mathbf{H}$  a indicatorilor claselor:

$$\mathbf{H} \approx \mathbf{W}\mathbf{T}.$$

În ecuația de mai sus, fiecare coloană din matricea țintă  $\mathbf{H}$  indică clasa semnalului respectiv, fiind o coloană de zerouri cu o singură valoare de 1 pe poziția  $c$  care indică clasa. Matricea  $\mathbf{W}$  conține coeficienții clasificatorului, iar  $\mathbf{T}$  reprezintă matricea descompunerilor rare ale fiecărui semnal, pe coloane. Matricea  $\mathbf{W}$  este astfel antrenată încât să producă o valoare apropiată de 1 pe poziția care indică clasa unui semnal, și 0 în rest.

Conform [21], optimizarea clasificatorului împreună cu dicționarul poate fi scrisă sub forma:

$$\arg \min_{\mathbf{D}, \mathbf{W}, \mathbf{\Gamma}} \|\mathbf{X} - \mathbf{D}\mathbf{\Gamma}\|_2 + \lambda_1 \|\mathbf{H} - \mathbf{W}\mathbf{\Gamma}\|_2 + \lambda_2 \|\mathbf{W}\|_2, \text{ cu } \|\gamma\| \leq k$$

Se urmărește optimizarea comună a tuturor celor trei matrici: optimizarea unui dicționar  $\mathbf{D}$  care să conducă la descompuneri rare  $\mathbf{\Gamma}$  (cu cel mult  $k$  termeni nenuli) suficient de discriminative astfel încât să permită clasificarea semnalelor cu clasificatorul liniar  $\mathbf{W}$  optimizat și el. O soluție naivă, prin extinderea algoritmului K-SVD, ar fi efectuarea tuturor celor trei optimizări în mod succesiv, în cadrul unei iterații:

- i. codare rară a semnalelor cu dicționarul curent
- ii. actualizarea dicționarului cu K-SVD, și
- iii. actualizarea clasificatorului  $\mathbf{W}$ . Această înșiruire de optimizări succesive are însă probleme de convergență și eficiență.

Soluția propusă de autori pentru eficientizarea algoritmului este de a cupla primii doi termeni într-o singură matrice, reformulând problema de optimizare sub forma:

$$\arg \min_{\mathbf{D}, \mathbf{W}, \mathbf{\Gamma}} \left\| \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_1} \mathbf{H} \end{pmatrix} - \begin{pmatrix} \mathbf{D} \\ \sqrt{\lambda_1} \mathbf{W} \end{pmatrix} \mathbf{\Gamma} \right\|_2, \text{ cu } \|\gamma\| \leq k.$$

Această formulare este similară cu cea a algoritmului K-SVD, și ca atare se poate rezolva în mod eficient folosind acest algoritm, dar considerând matricile extinse obținute prin concatenarea celor doi termeni. La o singură iterație a K-SVD se actualizează o coloană din întreaga matrice  $\begin{pmatrix} \mathbf{D} \\ \sqrt{\lambda_1} \mathbf{W} \end{pmatrix}$ , așadar atât  $\mathbf{D}$  cât și  $\mathbf{W}$  simultan. Termenul de regularizare  $\lambda_2 \|\mathbf{W}\|_2$ , care avea rolul de a controla mărimea coeficienților rezultați pentru clasificator, nu mai este necesar întrucât fiecare coloană rezultată din K-SVD este de normă egală cu 1, fiind un vector singular, ceea ce limitează în mod automat mărimea coeficienților din  $\mathbf{W}$ .

Autorii lucrării utilizează algoritmul pe semnale de tip imagine, în vederea identificării fețelor, dar algoritmul propus poate fi utilizat pentru orice tip de semnale, atât timp cât aplicația utilizează dicționare optimizate și clasificarea liniară pe baza descompunerilor rare. Reținem din această lucrare în principal ideea fundamentală de a grupa cei doi termeni de aproximare  $\ell_2$  (termenul de codare rară și cel de clasificare liniară) într-o singură matrice, care apoi este optimizată simultan prin K-SVD.

## Bibliografie

- [1] B. K. Natarajan, “Sparse approximate solutions to linear systems,” *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, Apr. 1995.
- [2] D. Donoho, M. Elad, and V. Temlyakov, “Stable recovery of sparse overcomplete representations in the presence of noise,” *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, Jan. 2006.
- [3] D. L. Donoho, “Compressed sensing,” *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [4] E. J. Candes and J. Romberg, “Sparsity and incoherence in compressive sampling,” *Inverse Problems*, vol. 23, no. 3, pp. 969–985, Jun. 2007.
- [5] A. Cohen, W. Dahmen, and R. DeVore, “Compressed sensing and best k-term approximation,” *Journal of the American Mathematical Society*, vol. 22, no. 1, pp. 211–231, Jul. 2008.
- [6] S. Boyd and L. Vandenberghe, *Convex optimization*. New York, NY, USA: Cambridge University Press, 2004.
- [7] S. Mallat, *A wavelet tour of signal processing, third edition: The sparse way*. Academic Press, 2008.
- [8] S. Mallat, “Matching pursuits with time-frequency dictionaries,” *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.
- [9] Y. C. Pati, R. Rezaeiifar, and P. S. Krishnaprasad, “Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition,” in *Proc. 27th annual asilomar conf. On signals, systems, and computers*, 1993, pp. 40–44.
- [10] D. Needell and J. A. Tropp, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, p. 30, 2008.
- [11] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, “A fast approach for overcomplete sparse decomposition based on smoothed l0 norm,” *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, 2009.
- [12] A. Eftekhari, M. Babaie-Zadeh, C. Jutten, and H. Moghaddam, “Robust-SL0 for stable sparse representation in noisy settings,” in *Proc. ICASSP 2009*, 2009, pp. 3433–3436.
- [13] R. Rubinstein, A. M. Bruckstein, and M. Elad, “Dictionaries for sparse representation modeling,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045–1057, 2010.
- [14] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on signal processing*, vol. 54, no. 11, p. 4311, 2006.

- [15] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online dictionary learning for sparse coding,” in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 689–696.
- [16] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online learning for matrix factorization and sparse coding,” *Journal of Machine Learning Research*, vol. 11, no. Jan, pp. 19–60, 2010.
- [17] “Scikit-learn - machine learning in python.”
- [18] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.
- [19] H. F. Huang, G. S. Hu, and L. Zhu, “Sparse representation-based heartbeat classification using independent component analysis,” *J. Med. Syst.*, vol. 36, no. 3, pp. 1235–1247, Jun. 2012.
- [20] B. M. Whitaker, M. Rizwan, V. B. Aydemir, J. M. Rehg, and D. V. Anderson, “AF classification from ecg recording using feature ensemble and sparse coding,” *Computing*, vol. 44, p. 1, 2017.
- [21] Q. Zhang and B. Li, “Discriminative K-SVD for dictionary learning in face recognition,” in *The twenty-third IEEE conference on computer vision and pattern recognition, CVPR 2010, san francisco, ca, usa, 13-18 june 2010*, 2010, pp. 2691–2698.