Raport științific final

privind implementarea proiectului PCE 6/2022 “AsyDil”

0. Introducere

Proiectul “Învățarea asimetrică a dicționarelor” (Asymmetric Dictionary Learning—AsyDiL) are ca scop principal deducerea unor algoritmi pentru reprezentări rare și învățarea dicționarelor atunci când, în reprezentările rare aferente, atomii nu mai sunt simpli vectori ficși, ci sunt aleși dintr-o mulțime infinită. În acest fel cresc flexibilitatea și precizia reprezentării. Proiectul s-a derulat în perioada 30.05.2022 – 31.12.2024.

Conținutul raportului:

1. Obiective prevăzute/realizate
2. Prezentarea rezultatelor obţinute
3. Impactul estimat al rezultatelor obţinute, cu sublinierea celui mai semnificativ rezultat obţinut
4. Concluzii
5. Bibliografie

1. Obiective prevăzute/realizate

Vom prezenta succint obiectivele și livrabilele, precum și o listă a realizărilor.

1.1 Obiective prevăzute

Obiectivele din propunerea de proiect (care se regăsesc ca activități și în planul de realizare) sunt următoarele.

O1. Calculul reprezentărilor asimetrice
- O1.1. Reprezentări rare cu atomi-mulțime uniformi
- O1.2. Reprezentări rare cu atomi-mulțime probabiliști
- O1.3. Versiuni kernel ale reprezentărilor de la O1.1 și O1.2

O2. Algoritmi pentru învățarea asimetrică a dicționarelor
- O2.1. Învățarea dicționarelor cu atomi-mulțime bazată pe euristici
- O2.2. Învățarea dicționarelor cu atomi-mulțime bazată pe optimizare
- O2.3. Învățarea dicționarelor de tip kernel cu atomi-mulțime
- O2.4. Optimizarea parametrilor dicționarelor cu atomi-mulțime
- O2.5. Biblioteci MATLAB și Python pentru metodele propuse

O3. Aplicații la detecție de anomalii
- O3.1. Proiectare de algoritmi și acordare hiper-parametri

O3.2. Detecție de anomalii pe date de test publice
O3.3. Detecție de anomalii pe date din tranzacții bancare și alte date de tip graf

Țintele de publicare din propunerea de proiect au fost:

- 4 articole în reviste
- 6 articole la conferințe

De asemenea, a fost prevăzută publicarea softului realizat, în MATLAB și Python.

1.2 Obiective realizate

Pentru referire mai simplă, începem cu publicațiile. Ca rezultat al cercetării desfășurate în cadrul proiectului, au fost obținute următoarele publicații. Versiuni pdf sau legături la versiunile apărute efectiv se pot găsi pe site-ul proiectului, asydil.upb.ro. Aproape toate aceste articole sunt asociate cu câte un set de programe (în MATLAB, Python sau ambele), care pot fi găsite pe același site.

Articole de revistă:

[R1] D.C. Ilie-Ablachim, B.Dumitrescu, "Incoherent frames design and dictionary learning using a distance barrier", *Signal Processing*, vol.209, art. 109019, August 2023.
[R2] A. Băltoiu, D.C. Ilie-Ablachim, B.Dumitrescu, "Dictionary learning with cone atoms and application to anomaly detection", *Signal Processing*, vol.219, art.109398, June 2024.
[R3] D.C. Ilie-Ablachim, A. Băltoiu, B.Dumitrescu, "Sparse Representation With Gaussian Atoms and Its Use in Anomaly Detection", *IEEE Open Journal of Signal Processing*, vol.5, pp.168-176, 2024.
[R4] D.Ilie-Ablachim, B.Dumitrescu, "SimCDL: A Simple Framework for Contrastive Dictionary Learning", Applied Sciences, vol.22, no.14, art.10082, 2024.
[R5] A.R. Mănescu, B.Dumitrescu, "HyperDE: An Adaptive Hyper-Heuristic for Global Optimization", *Algorithms*, vol.16, no.9, 2023.
[R6] X.Zheng, B.Dumitrescu, J.Liu, C.D.Giurcaneanu – Multivariate time series imputation: An approach based on dictionary learning, *Entropy,* vol.24, no.8, pp.1057, July 2022.

Articole la conferințe:

[C1] D.C. Ilie-Ablachim, A.Băltoiu, B.Dumitrescu, "Sparse representations with cone atoms", IEEE Int. Conf. Acoustics Speech Signal Processing (ICASSP), Rhodes, Greece, June 2023.
[C2] D.C. Ilie-Ablachim, B.Dumitrescu, "Classification with dictionary learning and a distance barrier promoting incoherence", IEEE Machine Learning in Signal Processing (MLSP), Rome, Italy, Sept. 2023.
[C3] T.A. Badea, B.Dumitrescu, "Community-Augmented Local-Link Intensity: a score for anomaly detection in graphs", Int. Conf. Control Decision Information Technologies (CoDIT), Rome, Italy, July 2023.
[C4] C.E. Zica, B.Dumitrescu, "Online Computation of Reduced Egonet Features for Anomaly Detection in Bank Transactions Graphs", IEEE Machine Learning in Signal Processing (MLSP), Rome, Italy, Sept. 2023.

[C5] D.C. Ilie-Ablachim, B.Dumitrescu, “Angle-Based Dictionary Learning for Outlier Detection”, IEEE Int. Conf. Signal Control and Communication (SCC), Hammamet, Tunisia, Dec. 2023
[C6] T.A. Badea, B.Dumitrescu, “DualGCN: A convolutional network for anomaly detection in directed graphs based on symmetrized and skew-symmetrized Laplacians”, IEEE Machine Learning in Signal Processing (MLSP), London, UK, Sept. 2024.
[C7] A.R. Manescu, B.Dumitrescu, “Evolutionary hyper-heuristics for improving global optimization algorithms”, Int. Conf. Control Systems and Computer Science (CSCS), Bucharest, Romania, 2023, May 2023.

Alte articole:

[A1] A. Băltoiu, D.C. Ilie-Ablachim, B.Dumitrescu, “Atom dimension adaptation for infinite set dictionary learning”, arXiv, Sept. 2024. https://arxiv.org/abs/2409.06831
[A2] T.A. Badea, B.Dumitrescu, “Haar-Laplacian for directed graphs“, arXiv, Nov. 2024. https://arxiv.org/abs/2411.15527
[A3] D.C.Ilie-Ablachim, B.Dumitrescu, C.Rusu, “Kernel t-distributed stochastic neighbor embedding “, arXiv, July 2023. https://arxiv.org/abs/2307.07081
[A4] T.A. Badea, B.Dumitrescu, “i-DGCN: A Spectral Convolutional Network For Directed Graphs Using An Intensity Laplacian“, 2024, manuscris pregătit pentru trimitere la o conferință.

Am publicat deci 6 articole în reviste, dintre care 3 în reviste cu calitate foarte bună [R1,R2,R3]; menționăm că la revistele MDPI nu am plătit taxă de publicare. Dintre articolele publicate pe arXiv, [A1] și [A2] au fost sau vor fi trimise spre publicare în reviste, cu șanse bune de reușită, în special pentru [A2]. În mod normal, vom încheia deci proiectul cu 7-8 articole publicate în reviste.

La conferințe am publicat 7 articole, dintre care 4 la conferințe foarte bune [C1,C2,C4,C6].

Trecem acum în revistă realizarea obiectivelor. Anticipând concluziile, putem spune că am îndeplinit complet planul de realizare, micile nereușite (într-o singură subdirecție) fiind compensate de reușite pe direcții conexe neprevăzute în planul initial, dar rezultând natural din acesta. Pentru fiecare obiectiv vom arăta articolele rezultate; unele articole au rezultate corespunzând mai multor obiective. Software-ul asociat va fi prezentat doar atunci când nu este în legătură unu-la-unu cu articolele.

O1. Calculul reprezentărilor rare pentru atomi-mulțime a fost prezentat în [C1] pentru atomi con (obiectivul O1.1) și [R3] pentru atomi gaussieni (obiectivul O1.2). De asemenea, în [R1] am folosit o idee apărută la studiul atomilor con, referitoare la necesitatea nesuprapunerii lor; optimizarea asociată poate fi utilizată pentru proiectarea frame-urilor incoerente. Originalitatea ideilor a fost bine apreciată, atât revistele cât și conferința având calitate de vârf. Variantele kernel ale reprezentărilor cu atomi-mulțime (O1.3) s-au dovedit ineficiente, de aceea nu am trimis spre publicare nici un articol pe această temă. Totuși, investigațiile pe tema kernel au produs rezultate conexe [A3] în problema vizualizării datelor multidimensionale.

O2. Rezultatele în privința antrenării dicționarelor (dictionary learning – DL) au fost mult mai diverse, de aceea am reușit și mai multe publicații. Metodele de antrenare a dicționarelor cu atomi mulțime au fost publicate în [R2] (pentru atomi con) și [R3] (pentru atomi gaussieni). Este vorba de metode bazate în special pe optimizare (pentru atomii gaussieni), dar și care combină optimizarea cu euristici simple (pentru atomii con), acoperind deci obiectivele O2.1 și O2.2. Optimizarea parametrilor atomilor-mulțime (O2.4) a fost efectuată în [A1].

Metode DL conexe au fost utilizate în mai multe articole:

- utilizând o funcție barieră care apare în optimizarea atomilor con [R1,C2];
- utilizând funcții de optimizare contrastive, cu rol similar de îndepărtare a atomilor [R4];
- pentru imputarea seriilor de timp, folosind o funcție obiectiv bazată pe norma 1 [R6];
- în cadrul optimizării pentru proiectarea unui detector de anomalii [C5].

Bibliotecile software aferente (O2.5) sunt disponibile la https://gitlab.cs.pub.ro/asydil. Majoritatea sunt organizate ca module asociate articolelor publicate, asigurând replicabilitatea rezultatelor. În plus, algoritmii de reprezentare rară și DL cu atomi mulțime sunt grupați în implementarea https://pypi.org/project/dictlearn/, inserate în contexul unei biblioteci cu metode pentru problema DL standard.

Alte articole legate de optimizare, în direcția metodelor de optimizare globală au fost [R5,C7].

O3. Multe din articolele dedicate atomilor-mulțime conțin rezultate în domeniul detecției de anomalii, de cele mai multe ori fiind vorba de rezultate mai bune decât cele prezente în literatura de specialitate.

- În [C1] am aplicat reprezentarea rară cu atomi con la detecția de anomalii în semnale cardiace.
- În [R2,R3] am efectuat detecție de anomalii pe date din biblioteca ADBench [HHH22], comparând cu metodele implementate acolo cu cele bazate pe DL cu atomi con și gaussieni. Am obținut rezultate mai bune decât toate metodele ADBench pe clasa de date „dependency”.

Alte rezultate pe date generale:

- O îmbunătățire folosind DL a unei metode clasice de detecție de anomalii a fost propusă în [C5].

Și pe date de tip graf am obținut rezultate foarte bune.

- În [C3,C6] am utilizat datele Libra [DBB22] de tranzacții bancare, obținând rezultate mai bune, cel puțin pentru anumiți indicatori, decât în articolul original.
- De asemenea, în [C4] am propus o variantă on-line a unui algoritm utilizând egonet, cu aceleași performanțe ca în [DBB22], dar cu execuție semnificativ mai rapidă.
- Trecând la probleme mai generale pentru date de tip graf, în [A2] am propus un nou tip de Laplacian pentru grafuri orientate, care, pe probleme cu date continue, cum este predicția de ponderi ale arcelor, obtine rezultate mai bune decât cele mai recente metode.

Putem astfel spune că obiectivele proiectului au fost atinse, în special în privința temei principale, învățarea dicționarelor cu atomi-mulțime. Singurul punct în care nu au fost atinse așteptările inițiale este cel al algoritmilor kernel. S-au obținut în schimb rezultate peste așteptări în privința detecției de anomalii pe date de tip graf.

2. Prezentarea rezultatelor obținute

2.1 Scurtă descriere a ideilor științifice principale

Începem prin a prezenta ideile aflate la baza proiectului, pe care se bazează majoritatea contribuțiilor originale obtinute.

Învățarea dicționarelor (dictionary learning—DL) [DuIr18] este caracterizată, în varianta standard, de optimizarea unui dicționar $D \in \mathbb{R}^{m \times n}$ astfel încât reprezentările rare ale unor semnale aflate pe coloanele unei matrice $Y \in \mathbb{R}^{m \times N}$ să fie cât mai aproape de optim, în sensul minimizării erorii de reprezentare $||Y - DX||$, unde $X \in R^{n \times N}$ este o matrice rară, de obicei având cel mult s elemente nenule pe fiecare coloană.

În acest proiect am propus extinderea problemei la cazul în care atomii (coloanele dicționarului) nu mai sunt simpli vectori, ci mulțimi infinite. Am propus două tipuri de astfel de dicționare pentru reprezentări rare.

Atomi con. Primul tip de reprezentare este cel în care fiecare atom al dicționarului pentru reprezentare rară este un con $C(d, \rho)$, în care d, cu $||d|| = 1$, este atomul central și ρ este raza conului, care conține toți vectorii a, cu $||a|| = 1$, pentru care $||a - d|| \leq \rho$. Pentru reprezentarea rară, atunci când un atom-com d este selectat, din el se utilizează atomul efectiv a care este cel mai util în minimizarea erorii de reprezentare. Fig.1 ilustrează reprezentarea unui semnal y ca o combinație liniară a doi atomi-com, $x_1 a_1 + x_2 a_2$. Atomii efectivi sunt aleși astfel încât planul produs de ei să fie cel mai apropiat de y, aceasta fiind reprezentarea optimă.

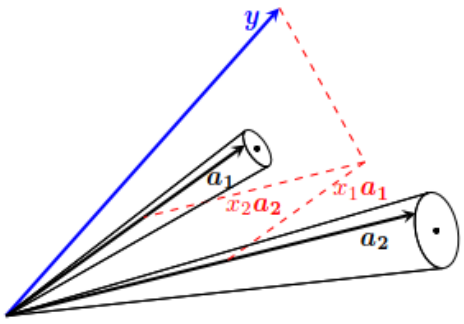

Fig.1 Aproximarea unui semnal prin combinația liniară a doi atomi con

Asociind un con fiecărui atom dintr-un dicționar $D \in R^{m \times n}$, astfel încât un atom-con să fie caracterizat de atomul central d_i și raza ρ_i, pentru reprezentarea unui semnal *y* trebuie minimizată funcția obiectiv

$$||y - \sum_{i=1}^{n} x_i a_i ||^2 \quad (1)$$

cu restricțiile $||x||_0 \leq s$ (numărul de coeficienți nenuli este cel mult *s*) și $a_i \in C(d_i, \rho_i)$ (atomul efectiv aparține conului asociat atomului central respectiv).

Atomi gaussieni. În al doilea tip de reprezentare asociem o probabilitate fiecărui potențial atom efectiv. Dacă d este un atom central, atunci un atom efectiv a aparține mulțimii $G(d, \sigma)$ cu probabilitatea

$$p(a, d) \sim exp\left(-\frac{||a - d||^2}{2\sigma^2}\right)$$

Pentru un dicționar $D \in R^{m \times n}$, asociem fiecărui atom d_j o distribuție $G(d_j, \sigma_j)$. Pentru a calcula o reprezentare în acest context, maximizăm probabilitatea atomilor din reprezentare, asigurând în același timp și o eroare mică de reprezentare. Funcția obiectiv este

$$\sum_{i=1}^{n} \frac{1}{\sigma_i^2} ||a_i - d_i||^2 + \lambda ||y - \sum_{i=1}^{n} x_i a_i ||^2 \quad (2)$$

Primul termen reprezintă logaritmul negativ al probabilității (log-likelihood) atomilor efectivi $a_i \in G(d_i, \sigma_i)$, deci minimizarea lui maximizează probabilitatea. Al doilea termen, ponderat cu o constantă λ, este eroarea de reprezentare a semnalului y.

Figura 2 sugerează reprezentarea unui semnal (cu negru), folosind doi atomi gaussieni; atomii centrali sunt roșii, iar distribuția asociată este gaussiană; atomii efectivi sunt trasați cu verde. Combinația liniară a atomilor efectivi (verde punctat) este o aproximare mai bună a semnalului dat decât combinația liniară a atomilor centrali (roșu punctat); probabilitatea lor este însă mai mică, dar optimă din punctul de vedere al compromisului dat de criteriul (2).

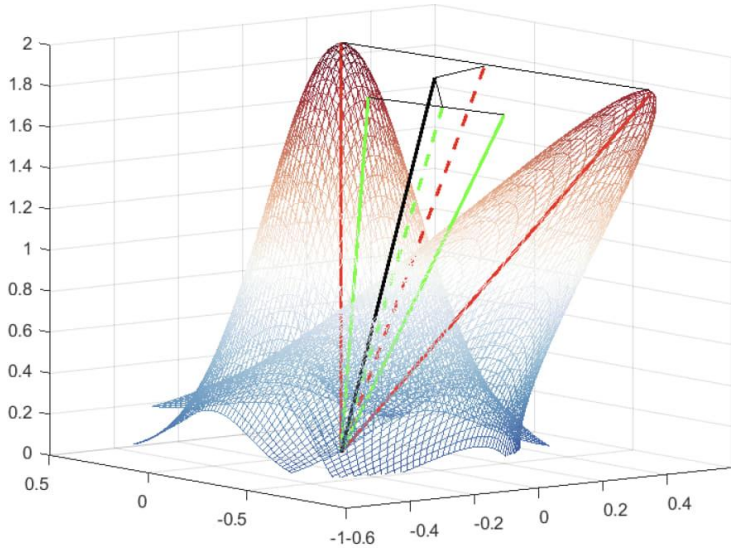

Fig.2 Aproximarea unui semnal prin combinația liniară a doi atomi gaussieni

Faptul că pentru fiecare semnal y se utilizează alți atomi efectivi corespunzători unui atom central face ca memorarea întregii informații pentru reprezentare să fie nepractică. Din acest motiv am numit reprezentarea *asimetrică*: calculul reprezentării este mereu posibil, dar refacerea semnalului din reprezentare nu se poate face. Cu toate acestea, dicționarele cu atomi-mulțime sunt utile în mai multe aplicații în care reprezentarea nu trebuie memorată. De exemplu, în detecția de anomalii se utilizează de regulă norma erorii de reprezentare, care are semnificația de scor de anomalie (cu cât mai mare, cu atât mai probabil ca semnalul să fie anomalie), adică pentru fiecare semnal este memorată o singură valoare.

2.2. Publicații

Publicațiile realizate până în prezent au fost prezentate în Sec.1. Cele 6+7 articole în reviste+conferințe depășesc așteptările inițiale din propunerea de proiect (4+6). De asemenea, avem pregătite alte 4 articole, din care cel puțin 3, aflate acum pe arXiv, își vor găsi drumul către o revistă sau conferință.

Am reușit publicare în reviste bune. Signal Processing (2 articole) este una din revistele cu tradiție în prelucrarea semnalelor. IEEE Open Journal of Signal Processing este o revistă în plină ascensiune, inaugurată în 2020, în care publică mulți autori cunoscuți; revista e destul de selectivă, în 2023 a publicat 46 de articole; articolul nostru publicat acolo a fost prezentat și la ICASSP 2024. Am publicat și la MDPI, dar în reviste Q1/Q2 (Applied Sciences, Entropy, Algorithms).

Între conferințele la care am avut articole acceptate, ICASSP (un articol în 2023) este cea mai mare și prestigioasă din domeniul prelucrării semnalelor. Tot cu tradiție, dar mai mică, este IEEE MLSP (Machine Learning in Signal Processing), la care am avut două lucrări în 2023 și una în 2024. Celelalte trei conferințe la care am avut lucrări sunt mai mici; două dintre ele sunt organizate sub tutela IEEE.

2.3. Rezultate științifice obținute

Vom descrie în continuare cele mai semnificative rezultate obținute, în modul în care apar ele în articolele realizate.

2.3.1. Reprezentări rare cu atomi con

Un algoritm pentru calculul reprezentărilor rare cu atomi con a fost prezentat în [C1]. El a pornit de la ideea din Orthogonal Matching Pursuit (OMP) [PRK93], unul din algoritmii de bază pentru calculul reprezentărilor rare, care este preferat pentru complexitatea scăzută și calitatea foarte bună. Dându-se un dicționar $D \in \mathbb{R}^{m \times n}$, cu $m \leq n$, un semnal y și un nivel de sparsitate s, se dorește reprezentarea x conținând s elemente nenule astfel încât $||y - Ax||$ să fie minimă. Problema este NP-completă, dar OMP calculează de obicei soluția optimă sau o bună aproximare a acesteia.

OMP este un algoritm lacom, care construiește suportul x adăugând pe rand câte un atom. OMP are două operații importante: i) alegerea următorului atom ca fiind cel care are cea mai mare proiecție pe reziduul curent al reprezentării (sau, altfel zis, atomul cel mai apropiat de reziduul curent); ii) calculul soluției în sensul celor mai mici pătrate (CMMP) pentru suportul curent (și deci a reziduului asociat). Pentru extinderea la atomi con trebuie găsite soluții eficiente pentru aceste două operații.

Atomul dintr-un con cel mai apropiat de un vector dat se poate calcula eficient așa cum se arată în [C1], proiecția asociată fiind și ea ușor de calculat. Problema se poate rezolva în planul format de vector și de atomul central al conului, adică este o simplă problemă de geometrie plană. Operațiile sunt vectoriale, deci complexitatea este similară cu cea a produsului scalar necesar în OMP.

Calculul soluției CMMP nu mai este așa ușor ca la OMP standard, pentru că atomii efectivi determinați la pasul anterior nu mai sunt optimi la pasul curent, după adăugarea unui nou atom (optim doar în contextul celorlalți). Soluția CMMP poate fi găsită iterativ, prin proiecții succesive pe fiecare con; se țin ficși toți atomii efectivi, mai puțin unul, și se determină noua valoare a acestui atom prin proiecția reziduului pe conul atomului (ceea ce schimbă atomul efectiv). Operația este identică cu proiecția de la alegerea următorului atom. După mai multe runde de proiecții succesive (algoritmic vorbind, aceasta este o operație de coborâre pe coordonate), soluția CMMP este aproximată suficient de bine.

Algoritmul 2 din [C1] prezintă în detaliu algoritmul schițat mai sus și numit Cone-OMP. Complexitatea lui este doar de câteva ori mai mare decât cea a algoritmului OMP, ceea ce este un rezultat remarcabil, având în vedere dificultatea aparentă a optimizării cu atomi-con.

2.3.2. Reprezentări rare cu atomi gaussieni

Pentru calculul atomilor gaussieni am propus doi algoritmi, numiți Gauss-OMP și Gauss-L1 [R3]; primul este în stil OMP (algoritm lacom); în continuare îl prezentăm pe scurt pe al doilea. Algoritmul se bazează pe optimizarea unei funcții obiectiv de tip relaxare convexă, de forma

$$\sum_{i=1}^{n} \frac{1}{\sigma_i^2} ||a_i - d_i||^2 + \lambda ||y - \sum_{i=1}^{n} x_i a_i||^2 + \gamma ||x||_1 \qquad (3)$$

în care ultimul termen este adăugat la (2) pentru a încuraja apariția zerourilor în vectorul de coeficienți x, pentru a obține o reprezentare rară. Se renunță astfel la un număr de coeficienți s impus, lăsând flexibil nivelul de raritate, care este reglat prin ponderea γ. Detaliile modului de rezolvare, bazat pe coborâre pe coordonate, se găsesc în [R3]. Prezentăm aici doar formulele rezultate. Presupunând toate variabilele fixate, mai puțin un atom efectiv a_j, optimizarea acestuia se efectuează cu

$$a_j = \frac{\sigma^2{}_j}{\lambda x^2{}_j \sigma^2{}_j + 1} \left(x_j \hat{y} + \frac{1}{\sigma_j^2} d_j\right)$$

unde $\hat{y} = y - \hat{A}\hat{x}$ este reziduul datorat celorlalți atom efectivi $\hat{A}$, din care lipsește atomul curent, cu reprezentările $\hat{x}$ aferente. Ținând acum toate variabilele fixate, mai puțin coeficientul x_j, expresia optimă a acestuia este

$$x_j = soft(a^T{}_j \hat{y}, \frac{\gamma}{2\lambda})$$

unde funcția *soft* reprezintă operația "soft thresholding".

După cum se vede, formulele de mai sus sunt simple, comparabile din punctul de vedere al complexității cu cele din AK-SVD, deși acolo atomii sunt vectori, nu mulțimi.

2.3.3. Antrenarea dicționarelor cu atomi mulțime

Vom prezenta în această secțiune modalitățile de actualizare a atomilor centrali în cadrul procedurilor DL pentru atomi-mulțime. Algoritmii DL au structura iterativă obișnuită; în fiecare iterație se reprezintă semnalele de antrenare cu ajutorul unui algoritm de reprezentare din cele ilustrate în secțiunile precedente, utilizând dicționarul curent; apoi, pe baza reprezentărilor, atomii centrali sunt actualizați.

Dicționare cu atomi-con. Am utilizat ca punct de plecare AK-SVD [RZE08], unul dintre cei mai simpli algoritmi DL eficienți; el optimizează atomii succesiv. Notăm d_j atomul central care se optimizează la un moment dat, ceilalți atomi fiind ficși, și F matricea de eroare fără contribuția acestui atom; coloana l a acestei matrice, corespunzând semnalului l, este

$$f_l = y_l - \sum_{i \neq j} a_{il} x_{il}$$

În această relație, a_{il} este atomul efectiv i utilizat pentru reprezentarea semnalului l, iar x_{il} este coeficientul asociat atomului în reprezentarea liniară; coeficientul este zero în cazul în care atomul nu este utilizat în reprezentarea rară.

Eroarea pătratică asociată este

$$\sum_l || f_l - a_{jl} x_{jl} ||^2$$

Egalând cu zero derivata acestei erori se obține, pentru definiția atomilor-con, regula de actualizare (justificarea este dată în [R2])

$$d_j \leftarrow \sum_l x_{jl} \left(f_l - x_{jl} (a_{jl} - d_j) \right)$$

urmată de normalizare. Observăm că actualizarea din această formulă se poate calcula pe măsură ce atomii efectivi și coeficienții lor sunt produși de Cone-OMP, deci nu este necesară memorarea lor. Așadar operațiile de reprezentare se pot întrețese cu cele de actualizare.

În algoritmul AK-SVD standard, actualizarea atomilor se face secvențial. În contextul atomilor con, modificarea unui singur atom d_j ar implica recalcularea tuturor atomilor efectivi care apar în semnalele unde d_j apare în reprezentare, ceea ce ar însemna un efort de calcul imens. De aceea,

am adaptat algoritmul Parallel AK-SVD (PAK-SVD) [DuIr18], în care actualizarea atomilor se face în paralel, atomii noi fiind calculați independent din cei curenți. Se observă că acest lucru este posibil în regula de actualizare de mai sus.

Ca ilustrare a algoritmului bazat pe actualizarea de mai sus, numit Cone-DL, prezentăm în Fig.3 evoluția erorii medii a Cone-DL pentru unul din seturile de date din ADBench [HHH22], comportament destul de tipic; pentru ceilalți algoritmi este arătat doar nivelul erorii finale, printr-o linie orizontală. Cone-DL reușește atingerea unei erori mai mici decât AK-SVD urmat de Cone-OMP, confirmând astfel că antrenarea are efect și dicționarul obținut este mai bun. Prin 'swap' este denumit algoritmul care, după terminarea fazei de antrenare, permută razele conurilor după numărul de semnale în care este utilizat atomul respectiv; atomii cei mai folosiți primesc raze mai mari; în acest exemplu, razele sunt aleator distribuite în intervalul [0.01,0.1]. Se observă că pentru Cone-DL, această realocare nu are efect benefic, ceea ce confirmă încă o dată că antrenarea este bine efectuată pe distribuția inițială a razelor. În schimb, pentru Cone-OMP și dicționarul preantrenat cu AK-SVD, eroarea scade, ceea ce arată suboptimalitatea acestei metode.

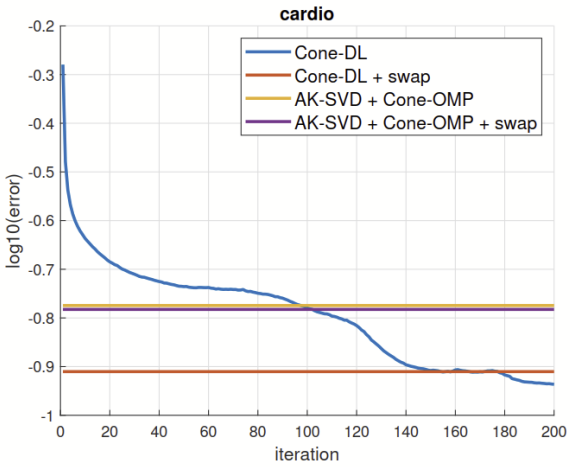

Fig.3 Evoluția erorii medii a Cone-DL pe setul de date *cardio*.

Tab.1 conține timpii de execuție pentru Cone-DL pe 30 de seturi de date din ADBench, alături de cei ai metodei clasice AK-SVD (care utilizează OMP pentru reprezentare rară, într-o implementare foarte eficientă și pre-compilată). Se observă că Cone-DL este de doar câteva ori mai lent, deși rezolvă o problemă mult mai complexă. În tabel, parametrul $c = n/m$ este raportul dintre dimensiunea dicționarului și cea a semnalelor; grade de supradimensionare cu valori 2-4 sunt tipice în DL.

	$s = 2$			$s = 3$	
Algorithm	$c = 2$	$c = 3$	$c = 4$	$c = 2$	$c = 3$
AK-SVD + OMP	59.45	76.84	89.70	70.16	92.84
Cone-DL	156.66	166.18	167.16	310.78	314.34

Tab.1. Timpi de execuție pentru 30 de seturi de date din ADBench.

Dicționare cu atomi gaussieni. Algoritmul Gauss-L1, prezentat în secțiunea 2.3.2, poate calcula reprezentarea rară pentru dicționare date. Prezentăm acum modul de actualizare a atomilor centrali, care este esențial pentru un algoritm DL. Algoritmul DL-Gauss-L1 poate fi găsit în detaliu în [R3]; dăm aici doar operația de bază. În etapa de reprezentare dintr-o iterație a algoritmului DL-Gauss-L1, pentru semnalul y_l sunt folosiți în reprezentare atomii efectivi a_{il}. Pe baza acestor atomi, noul atom central d_i are expresia optimă

$$d_i = \sum_{l, x_{il} \neq 0} a_{il}$$

adică o simplă mediere. Subliniem că acest rezultat nu este euristic, ci rezultă din criterii de optimalitate. Este esențial că această expresie poate fi calculată pe măsură ce se calculează reprezentările, fără a fi necesară memorarea atomilor efectivi: pe măsură ce atomii efectivi a_{il} sunt calculați, ei sunt adunați la valoarea curentă a atomului central d_i; după calculul tuturor reprezentărilor mai este necesară doar normalizarea atomului central.

Deși convergența algoritmilor propuși nu e garantată, comportarea practică este foarte bună. Fig.4 ilustrează convergența funcției obiectiv a algoritmului Gauss-L1 pentru 100 de semnale din setul *landsat* din ADBench [HHH22]. Se observă scăderea cvasi-permanentă a obiectivului. Comportamentul DL-Gauss-L1 este similar. Cu roșu este reprezentată evoluția pentru anomalii, în timp ce albastru este pentru semnalele normale. Vom rediscuta chestiunea mai pe larg ulterior, dar remarcăm de acum că, în multe cazuri, scăderea este mai pronunțată pentru semnalele normale, putându-se astfel face distincția între ele și anomalii.

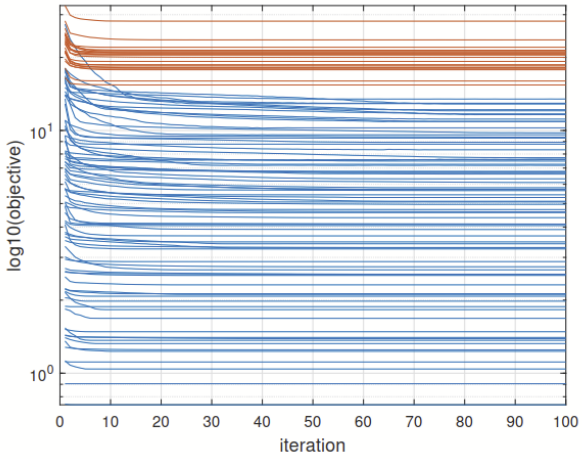

Fig.4 Evoluția funcției obiectiv a algoritmului Gauss-L1 pe 100 de semnale.

2.3.4. Adaptarea parametrilor dicționarelor cu atomi mulțime

În algoritmii prezentați în secțiunile anterioare, parametrii atomilor-mulțime, adică razele ρ_i pentru atomii-con sau deviațiile standard σ_i pentru atomii gaussieni, erau aleși la început. Doar în [R2] a fost propus ca, la sfârșitul antrenării, razele să fie alocate atomilor în ordine descrescătoare a utilizării lor (cele mai mari raze celor mai utilizați atomi). În [A1] am propus algoritmi în care

permutarea setului de raze (sau deviații standard) să se obțină pe parcursul algoritmului, astfel încât antrenarea să adapteze mai bine atomii centrali, ținând seama de razele curente din fiecare iterație a antrenării. Vom prezenta doar algoritmul pentru atomi gaussieni, cel pentru atomi-con fiind similar ca idee.

Formulăm problema DL aducând o mica modificare problemei de reprezentare (3) în felul următor. Se dă un set de deviații standard $\sigma_i, i = 1{:}n$. Scopul este de a găsi asocierea optimă dintre σ_i și atomii central d_i, adică o permutare π a mulțimii de indici $1{:}n$ astfel încât funcția obiectiv

$$\sum_{\ell=1}^{N}\left[\sum_{i=1}^{n}\frac{1}{\sigma_{\pi_i}^2}||a_{i\ell}-d_{i\ell}||^2+\lambda||y_\ell-\sum_{i=1}^{n}x_{i\ell}a_{i\ell}||^2+\gamma||x_\ell||_1\right] \quad (4)$$

să fie minimă. Desigur, având în vedere natura combinatorială a problemei, nu putem spera decât la o rezolvare aproximativă.

Algoritmul propus este foarte simplu: pornim iterațiile DL cu deviații standard egale pentru toți atomii. La fiecare ν iterații modificăm deviațiile standard către distribuția lor dată, cu un pas mic; de asemenea, permutăm valorile lor astfel încât cei mai utilizați atomi să primească deviațiile standard cele mai mari. Aceste modificări treptate au scopul de a permite adaptarea dicționarului astfel încât semnalele să fie reprezentate mai bine.

Mai precis, operațiile de adaptare a deviațiilor standard sunt următoarele. Valoarea unică de pornire este media valorilor din distribuția dată, notată $\bar{\boldsymbol{\sigma}} \in \mathbb{R}^n$, adică

$$\mu = \frac{1}{n}\sum_{\mathrm{i}=1}^{n}\bar{\sigma}_i \quad (5)$$

Trecerea către distribuția finală se face în pași de mărime

$$\boldsymbol{\delta} = (\bar{\boldsymbol{\sigma}} - \mu\mathbf{1})/\lfloor n_{it}/\nu \rfloor$$

unde $\mathbf{1}$ este vectorul cu toate elementele egale cu 1 și n_{it} este numărul de iterații DL care se execută. La iterația k (multiplu de ν), vectorul curent de deviații standard are forma

$$\boldsymbol{\sigma} = \mu\mathbf{1} + (k/\nu)\boldsymbol{\delta}$$

De fiecare dată când se modifică acest vector, se face și realocarea razelor după utilizarea atomilor. Măsurăm gradul de utilizare în două moduri, anume după numărul de semnale în care apare atomul, adică norma 0: $u_0(a_i) = \sum_{\ell, x_{i\ell}\neq 0} 1$, sau după norma 1 a coeficienților atomilor: $u_1(a_i) = \sum_\ell |x_{i\ell}|$. Actualizarea atomilor se face cu algoritmul prezentat în secțiunea anterioară. Denumim DL-G-L1-adapt algoritmul rezultat.

Vom prezenta în secțiunea 2.3.6 rezultatele obținute cu acest algoritm în detecția de anomalii.

2.3.5. Proiectare de frame-uri incoerente

Frame-urile grassmanniene sunt matrice $D \in R^{m\times n}$, cu $m > n$, în care coloanele (atomii, în terminologia DL) se află la distanța maximă posibilă. Ele sunt utile în codare și comunicații, printre altele [StHe03,MeDa14]. Pentru proiectarea lor, ar trebui optimizată funcția

$$\min_{D}\ \max_{1\le i<j\le n}\left|d_i^T d_j\right|$$

cu condiția $||d_j|| = 1$ a normalizării atomilor. Aceasta este o funcție neconvexă greu de optimizat, în special atunci când dimensiunile matricei sunt mari. Multe metode [TDHS05,ZoBo15,RuGo16] dau rezultate bune pentru matrice mici, dar devin mult prea lente pentru matrice medii sau mari. Alte metode sunt mai rapide și vor fi menționate ulterior.

Pentru obținerea de frame-uri grassmaniene am minimizat funcția

$$f(d_j) = ||W\overline{D^T}d_j||^2 + \lambda b(d_j)$$

unde $\overline{D}$ este dicționarul fără coloana j, iar W o matrice diagonală ce depinde de produsele scalare $|d_i^T d_j|$ și de valoarea dorită a coerenței; primul termen al funcției obiectiv este preluat dintr-un articol anterior [Dumi17]; al doilea este reprezentat de funcția barieră

$$b(d_j) = \sum_{i \neq j} \left[\max(0, M - ||d_i - d_j||^2) + \max(0, M - ||d_i + d_j||^2)\right]$$

unde M este un parametru pentru nivelul acceptabil de coerență. Această funcție a fost utilizată și în contextul atomilor mulțime. Se observă că dacă atomii sunt apropiați, atunci fie diferența lor, fie suma lor (în funcție de orientarea relativă), este mică și atunci funcția barieră este pozitivă. În schimb, atunci când atomii sunt depărtați, funcția barieră este nulă. Deci, minimizarea funcției barieră asigură în principiu o distanță minimă între atomi.

Pentru optimizare, am propus o metodă de gradient cu pas adaptiv. Mai multe detalii pot fi găsite în [R1].

Câteva rezultate semnificative sunt prezentate în Tabelul 2. Am comparat metoda propusă, numită IDB (Incoherence via a Distance Barrier), cu două dintre metodele cu rezultate bune, anume ISPM [Dumi17] și FLIP [JBS21]. Se observă că IDB dă rezultate mai bune decât ambele metode pentru o gamă largă de dimensiuni. Coloana ISPM+FLIP conține rezultatele pentru metoda FLIP (care este mai lentă) inițializată cu rezultatul ISPM (care este mult mai rapid). Ca viteză, IDB este de câteva ori mai lentă decât ISPM, dar clar mai rapidă decât FLIP.

m	n	bound	ISPM	FLIP	ISPM +FLIP	IDB
5	10	0.3333	0.3350	0.3338	0.3338	0.3333
50	60	0.0582	0.0775	0.0666	0.0634	0.0625
90	100	0.0335	0.0537	0.0384	0.0384	0.0375
100	110	0.0303	0.0495	0.0349	0.0350	0.0340
200	210	0.0155	0.0286	0.0181	0.0181	0.0175
300	310	0.0104	0.0209	0.0122	0.0123	0.0120
500	510	0.0063	0.0137	0.0074	0.0075	0.00732
500	550	0.0135	0.0203	0.0156	0.0151	0.0151
700	710	0.0045	0.0104	0.0053	0.0054	0.00527
900	1100	0.0142	0.0187	0.0166	0.0156	0.0156
2000	2500	0.0100	0.0130	0.0126	0.0121	0.01094
25	800	0.2871	0.3696	0.3896	0.3694	0.3687
30	800	0.2417	0.3189	0.3408	0.3189	0.3172
100	1000	0.0949	0.1350	0.1470	0.1350	0.1342

Tabel 2. Coerențele frame-urilor calculate de câteva metode

Alte rezultate prezentate în [R1] arată că IDB este mai bună decât alte metode foarte recente, anume ICBP [TSR19] și TELET [JyBa22].

2.3.6. Detecție de anomalii pe date generale

Detecția de anomalii cu DL se face de obicei [AEH15, YMW19, PBT20, HZS21] în două etape, în mod nesupervizat. În prima se antrenează un dicționar cu toate datele disponibile pentru antrenare, indiferent de tipul lor (semnale normale sau anomalii). În a doua etapă se face ordonarea semnalelor după scorul de anomalie ales, care este de obicei eroarea de reprezentare. Deoarece se presupune că semnalele normale sunt multe și similare între ele, procesul de optimizare asociat DL tinde să reprezinte bine aceste semnale, în detrimentul anomaliilor, care sunt puține. DL produce mulți atomi care sunt bine specializați pentru semnalele normale, dar alocă puțini atomi anomaliilor, pentru care o eroare mai mare nu afectează prea mult funcția obiectiv a DL. Așadar, se consideră anomalii semnalele reprezentate cu eroare mare. De exemplu, în Fig.4, anomaliile sunt reprezentate cu roșu și semnalele normale cu albastru; este un caz fericit, în care, după DL cu atomi-mulțime, toate anomaliile reale au într-adevăr o eroare mai mare decât a semnalelor normale (ceea ce nu se întâmpla înainte, când dicționarul era antrenat cu DL standard).

În cazul dicționarelor cu atomi conuri, vom folosi în continuare eroarea (1) ca scor de anomalie. În cazul atomilor gaussieni, putem folosi eroarea de reprezentare, dar și funcția obiectiv (2) sau (3), precum și probabilitatea reprezentării, asociată cu distanța dintre atomii efectivi și cei centrali $\sum_{i=1}^{n} \frac{1}{\sigma_i^2} ||a_i - d_i||^2$. În raportarea performanțelor utilizăm Receiver Operating Characteristic Area under Curve (ROC AUC), utilizat adesea în detecția de anomalii. Curba ROC ilustrează relația dintre rata de fals pozitive și rata de adevărat pozitive, valoarea optimă fiind 1.

Am testat algoritmii noștri pe unul dintre cele mai noi și probabil cel mai important *benchmark* în domeniul detecției de anomalii la momentul actual, ADBench [HHH22]. ADBench conține 57 seturi de date, din domenii diverse (medicină, fizică, sociologie, finanțe, lingvistică ș.a.) și având proprietăți diferite (raport diferit între numărul de semnale și dimensiunea acestora, proporții diferite de anomalii). De asemenea, conține 14 metode de detecție nesupervizată de anomalii, atât consacrate, cât și noi (inclusiv rețele neurale). Autorii propun o taxonomie a anomaliilor în funcție de similaritatea acestora cu semnalele normale: anomalii *global*, *locale*, *dependency* și *cluster*. Pentru fiecare tip, ADBench conține câte o funcție pentru a genera sintetic anomaliile, pornind de la semnalele normale din fiecare bază de date.

Din cele 57 seturi de date am selectat 30 pe care am desfășurat în continuare testele. Într-o primă fază, am verificat performanțele algoritmilor propuși pe fiecare dintre cele 4 tipuri de anomalii. Când seturile de date conțin anomalii de tip *global,* problema este ușor de rezolvat, atât de soluțiile propuse, cât și de metodele din *benchmark*: pentru multe seturi de date se obține un scor de detecție perfect. Anomaliile de tip *local* și *cluster* nu se pretează abordării cu algoritmi DL atunci când scorul de anomalie este bazat pe eroarea de reprezentare. Pentru a putea folosi DL în aceste cazuri sunt necesare alte metode de a eticheta semnalele. Motivul, în cazul anomaliilor *local,* este faptul că acestea sunt foarte similare cu multe semnale normale, ducând la erori mici de reprezentare. Asemănător, în cazul anomaliilor *cluster*, gruparea acestora conduce la o bună reprezentare. În baza concluziilor de mai sus am desfășurat în continuare teste suplimentare

pentru anomaliile de tip *dependency.* De notat observația din [HHH22] că nicio metodă nu are performanțe ridicate pentru toate tipurile de anomalii.

Deoarece performanța s-a îmbunătățit pe măsură ce am lucrat, prezentăm aici rezultate din [A1], mai bune decât cele din [R2,R3]. Tabelul 3 prezintă valorile ROC AUC pe setul de date *dependency*, pentru diverse variante ale DL cu atomi gaussieni, folosind câteva combinații de parametri. În particular, mulțimea deviațiilor standard are valori în intervalul $[\rho_{\min}, \rho_{\max}]$, cu trei distribuții: liniară (grilă echidistantă acoperind intervalul), 50-50 (jumătate dintre deviații au valoarea minimă, restul cea maximă) și 80-20. Folosim mai multe valori pentru raportul n/m dintre numărul de atomi și lungimea semnalului; valorile sunt tipice pentru aplicațiile DL.

		$\rho_{\min} = 0.01$, $\rho_{\max} = 0.1$			$\rho_{\min} = 0.04$, $\rho_{\max} = 0.12$		
	n/m	linear	min-max 50-50	min-max 80-20	linear	min-max 50-50	min-max 80-20
Gauss-L1	2	0.9511	0.9506	0.9505	0.9511	0.9503	0.9509
	2.5	0.9522	0.9519	0.9517	0.9522	0.9515	0.9519
	3	0.9549	0.9548	0.9542	0.9550	0.9545	0.9545
DL-Gauss-L1	2	0.9531	0.9520	0.9517	0.9538	0.9515	0.9530
	2.5	0.9547	0.9537	0.9529	0.9556	0.9509	0.9543
	3	0.9569	0.9560	0.9549	0.9574	0.9543	0.9561
DLG-L1-adapt (0-norm)	2	0.9557	0.9556	0.9543	0.9567	0.9562	0.9564
	2.5	0.9566	0.9569	0.9548	0.9579	0.9579	0.9570
	3	0.9586	0.9585	0.9570	**0.9598**	*0.9594*	0.9588
DLG-L1-adapt (1-norm)	2	0.9565	0.9559	0.9555	0.9568	0.9566	0.9574
	2.5	0.9572	0.9569	0.9561	0.9579	0.9576	0.9576
	3	*0.9591*	0.9583	0.9584	*0.9596*	0.9585	**0.9598**

Tabel 3. Valori ROC AUC pentru datele din ADBench

Se observă că metodele de adaptare a distribuției deviațiilor obțin rezultate mai bune decât cele cu distribuție fixată de la început (Gauss-L1 și DL-Gauss-L1), indiferent ce măsură folosim pentru a măsura utilizarea atomilor (norma 0 sau norma 1). Pentru comparație, cea mai bună metodă din ADBench dă o valoare de 0.9274, în timp ce algoritmul DL standard AK-SVD dă 0.9329. Se vede deci o clară îmbunătățire.

Tabelul 4 prezintă rangul mediu al metodelor noastre între cele din ADBench. Fiecare metodă a noastră este comparată cu cele 14 din ADBench; pentru fiecare set de date se calculează ROC AUC pentru cele 15 metode, care sunt apoi ordonate după ROC AUC, locul 1 fiind cel mai bun. Se calculează apoi rangul ca medie a locurilor pentru cele 30 de seturi de date. Se vede clar avantajul metodelor adaptive. Rangul AK-SVD standard este 1.93, în timp ce cea mai bună metodă ADBench are un rang mai mare de 2.5.

	n/m	$\rho_{\min} = 0.01$, $\rho_{\max} = 0.1$			$\rho_{\min} = 0.04$, $\rho_{\max} = 0.12$		
		linear	min-max 50-50	min-max 80-20	linear	min-max 50-50	min-max 80-20
Gauss-L1	2	1.90	1.97	1.97	1.90	1.93	1.93
	2.5	1.63	1.67	1.70	1.63	1.63	1.67
	3	1.53	1.63	1.70	1.63	1.63	1.67
DL-Gauss-L1	2	1.83	1.83	1.90	1.73	1.87	1.87
	2.5	1.57	1.60	1.67	1.53	1.63	1.57
	3	1.50	1.60	1.63	1.50	1.60	1.53
DLG-L1-adapt (0-norm)	2	1.60	1.70	1.63	1.63	1.60	1.60
	2.5	1.40	1.43	1.53	1.47	1.47	**1.33**
	3	1.40	1.43	1.43	1.43	1.50	**1.33**
DLG-L1-adapt (1-norm)	2	1.60	1.60	1.60	1.57	1.63	1.57
	2.5	1.40	1.47	1.43	1.43	1.53	1.37
	3	1.43	1.47	1.43	1.40	1.43	**1.33**

Tabel 4. Rangul mediu între metodele din ADBench

Menționăm că am propus o altă metodă de detecție de anomalii de tip DL în [C5], prin modificarea metodei Angle-Based Outlier Detection (ABOD) [KSZ08].

2.3.7. Detecție de anomalii pe date de tip graf

O direcție importantă a acestui proiect este detecția de anomalii pe date provenite din tranzacții bancare și organizate sub formă de graf; nodurile sunt clienți (mai multe conturi pot fi asociate cu aceeași persoană fizică sau juridică) și arcele conțin informații despre tranzacțiile dintre clienți, mai precis suma totală tranzacționată și numărul de tranzacții efectuate. Datele utilizate sunt descrise în [DBB22], cele de interes deosebit fiind date reale provenite de la Libra Internet Bank, etichetate de experții băncii pe baza unor indicatori simpli și ai experienței.

În [C3] am propus un scor de anomalie numit Community-Augmented Local Link Intensity (CALLI). Metoda utilizează comunități găsite cu algoritmul Louvain și intensități bazate pe medii geometrice, calculând raportul dintre suma intensităților în raport cu toate comunitățile și intensitatea în comunitatea proprie. În [C4] am prezentat un algoritm online de actualizare a trăsăturilor (de tip egonet și egonet redus) pe care se bazează metoda din [DBB22]. Rezultate mai bune am obținut în [C6], prezentate pe scurt în continuare.

Pornind de la neajunsurile Laplacienilor asociați grafurilor orientate, de exemplu utilizarea Laplacianului magnetic [ZHB21] într-o rețea neurală propusă inițial pentru grafuri neorientate [KW16] pentru probleme de clasificare, am construit o rețea de tip autoencoder pentru detecția de anomalii caracterizată de două structuri identice (Figura 5), în care fiecare strat este o convoluție utilizând două matrice rezultând din matricea de adiacență A a grafului: simetrică $A_s = (A + A^T)/2$ și antisimetrică $A_a = (A - A^T)/2$. Ponderile celor două structuri sunt comune. Ieșirile autoencoder-ului sunt introduse într-un singur strat liniar, care are rolul de a reface semnalele de intrare (care sunt semnale asociate nodurilor grafului, precum numărul de tranzacții și sumele totale tranzacționate – separat pentru ieșirile și intrările fiecărui cont). În funcționarea ideală, o eroare mare de refacere semnalează o anomalie.

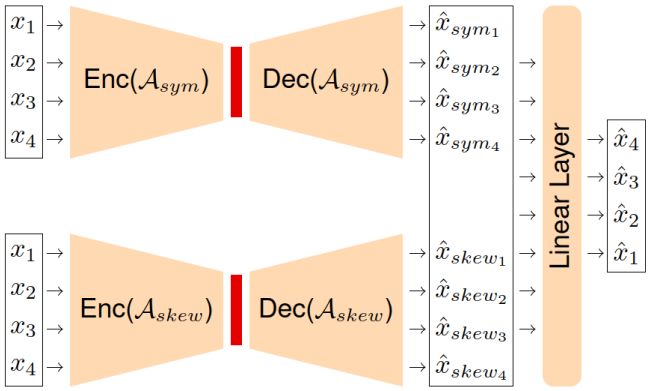

Fig.5 Structura de tip autoencoder a rețelei neurale folosite.

În [C6] sunt prezentate rezultatele pentru mai multe configurații de rețele, hiper-parametrii fiind numărul semnalelor de intrare (2 sau 4), numărul de straturi (1 sau 2) și modul de transformare a semnalelor și ponderilor grafului (fără transformare sau logaritmare, a doua variantă fiind justificată de valorile foarte mare ale sumelor tranzacționate prin unele conturi). Figura 5 prezintă rezultatele pentru câteva dintre cele 16 configurații. DualGCN este rețeaua propusă de noi; SymGCN este tot un autoencoder, dar construit doar cu matricea simetrizată; MagNet reprezintă un autoencoder folosind Laplacianul magnetic [ZHB21]. Indicatorii de performanță sunt True Positive Rate (*tpr*), folosind ponderi pentru noduri în funcție de numărul de tranzacții dubioase în care sunt implicate; procentajul asociat este cel al scorurilor de anomalii cele mai mari considerate. (Având în vedere numărul uriaș de tranzacții bancare, este de dorit ca anomaliile să fie cât mai bine evidențiate, pentru a nu verifica suplimentar un număr mare de fals pozitive.) Prin *an* am notat numărul de noduri (conturi) detectate din cele cu tranzacții dubioase (totalul lor este de 600). Se observă că DualGCN este capabilă să dea rezultate mai robuste decât celelalte rețele. De asemenea, poate bate la unii indicatori metoda EGO din [DBB22], care folosește trăsături de tip egonet, precum și alte metode bazate pe trăsături, GAW [ECL19] și CALLI [C3]. (De notat că aceasta din urmă se bazează pe o singură trăsătură, deci este extrem de simplă.)

Method	*tpr*_0.1%	*tpr*_0.2%	*tpr*_0.5%	*tpr*_1%	*auc*_1%	*an*_0.1%	*an*_0.2%	*an*_0.5%	*an*_1%

DualGCN 6	**0.3074**	**0.4588**	**0.6758**	**0.8276**	**0.6130**	**114.80**	**198.80**	**336.90**	**441.90**
SymGCN 6	0.2846	0.4299	0.6416	0.7940	0.5828	108.60	190.00	320.60	423.90
MagNet 6	0.3018	0.4351	0.6510	0.7971	0.5909	110.60	186.20	321.30	420.90
DualGCN 9	**0.3088**	**0.4576**	**0.6760**	**0.8259**	**0.6135**	**115.70**	**199.10**	**337.00**	**440.00**
SymGCN 9	0.1976	0.2953	0.4606	0.6220	0.4270	77.00	128.10	220.20	315.20
MagNet 9	0.0021	0.0028	0.0044	0.0073	0.0045	1.80	2.40	3.90	6.30
DualGCN 10	**0.3086**	**0.4580**	**0.6760**	**0.8263**	**0.6134**	**115.50**	**199.30**	**337.00**	**440.40**
SymGCN 10	0.2734	0.4106	0.6189	0.7697	0.5614	106.30	179.30	307.80	414.00
MagNet 10	0.3069	0.4541	0.6719	0.8197	0.6091	114.30	196.80	334.40	435.80
DualGCN 13	**0.3094**	0.4574	**0.6760**	**0.8259**	**0.6138**	**116.00**	199.00	**337.00**	**440.00**
SymGCN 13	**0.3094**	**0.4578**	**0.6760**	**0.8259**	**0.6138**	**116.00**	**199.20**	**337.00**	**440.00**
MagNet 13	0.0003	0.0019	0.0066	0.0131	0.0064	0.20	1.20	4.50	8.80
DualGCN 14	**0.3094**	0.4574	**0.6760**	**0.8259**	**0.6138**	**116.00**	199.00	**337.00**	**440.00**
SymGCN 14	0.3088	**0.4584**	**0.6760**	**0.8259**	**0.6138**	115.70	**199.50**	**337.00**	**440.00**
MagNet 14	0.3055	0.4484	0.6665	0.8187	0.6062	113.40	194.20	330.80	435.20
GAW_{log}	0.1769	0.3512	0.5871	0.7473	0.5247	71.60	152.50	290.60	406.10
GAW	0.1768	0.3588	0.6150	**0.7601**	0.5402	71.40	156.50	303.90	**413.20**
$CALLI_{log}$	0.0832	0.1412	0.2400	0.3273	0.2173	29.20	56.30	109.60	155.40
CALLI	0.0972	0.1518	0.2399	0.3154	0.2194	35.10	63.70	111.40	153.50
EGO	**0.4004**	**0.5051**	**0.6559**	0.7444	**0.6016**	**160.90**	**225.90**	**317.40**	377.20

Tabel 5. Rezultate obținute pe graful Libra

2.3.8. Analiza datelor de tip graf cu Haar-Laplacian

Cea mai recentă lucrare [A2], publicată pe arXiv în momentul finalizării acestui raport, a extins ideea de mai sus la introducerea unui nou Laplacian, numit Haar-Laplacian, definit astfel. Folosind matricele de adiacență simetrizată și anti-simetrizată, construim matricea Hermitiană $H_h = A_s + iA_a$. Definind matricea diagonală $D_h = diag(|H_h| \cdot \mathbf{1})$, Haar-Laplacianul este $L_h = A_s + iA_a$. Similar cu alte construcții de Laplacieni [ZHB21, FCC23] pentru grafuri orientate, este o matrice Hermitiană, având astfel proprietați spectrale convenabile: valori proprii reale și vectori proprii ortogonali. Am demonstrat în [A2] mai multe proprietăți, dintre care enumerăm:

- Haar-Laplacianul este pozitiv definit iar valorile proprii ale variantei sale normalizate (utilizată în rețelele neurale) se află în intervalul [0,2].
- Haar-Laplacianul este în relație bijectivă cu matricea de adiacență a grafului orientat, funcția ce le leagă fiind continuă. Astfel, poate modela distinct cazul în care există arce în ambele direcții între două noduri. De asemenea, modelează corect un graf după scalarea ponderilor.
- Este corect definit pentru ponderi negative.

Niciunul dintre Laplacienii anteriori nu are simultan toate aceste proprietăți. De asemenea, în [A2] este ilustrat faptul că transformata Fourier pe graf (GFT – Graph Fourier Transform) asociată (formată din vectorii proprii ai Laplacianului) are o comportare naturală pe grafuri simple, utilizate curent pentru caracterul lor intuitiv.

Pentru a testa propritetățile Haar-Laplacianului, l-am substituit în rețeaua neuronală propusă în [ZHB21] și reluată în [FCC23], pentru rezolvarea unor probleme de predicție în grafuri: existența arcelor (indiferent de direcție), 3-class (existența unui arc orientat) și predicția ponderii. Pentru cea de-a treia problemă, netratată în lucrările citate, am modificat blocurile finale ale rețelei,

pentru a realiza regresie în loc de clasificare. Seturile de date utilizate au fost Telegram [BG20], Bitcoin Alpha și Bitcoin OTC [KSS16]. Graful Telegram are 245 de noduri și 8912 arce cu ponderi pozitive, cu valori între 1 și 7934. Bitcoin Alpha are 3783 noduri, 22650 arce cu ponderi pozitive și 1536 cu ponderi negative; pentru Bitcoin OTC valorile sunt 5881, 32029 și, respectiv, 3563. Grafurile Bitcoin sunt rețele de încredere, ponderile având valori între -10 și 10.

Pentru antrenare și testare am scalat ponderile arcelor, liniar pentru grafurile Bitcoin, astfel încât valorile să se situeze în intervalul [-1,1], și cu exp(-1/pondere) pentru Telegram, valorile fiind în [0,1].

Pentru cele două probleme de existență, MagNet [ZHB21] obține cele mai bune rezultate, deși și HaarNet (rețeaua noastră) câștigă pentru unele seturi de date. Surprinzător, SigMaNet [FCC23] se clasează deseori a treia, deși modul de construcție a Laplacianul ar putea să-i ofere unele avantaje. Rezultatele nu sunt surprinzătoare, fiindcă toți cei trei Laplacieni modelează corect structura grafului.

În schimb, pentru problema de predicție a ponderilor, în care modelarea bijectivă a matricei de adiacență în Laplacian este esențială, HaarNet obține clar cele mai bune rezultate. Tabelul 6 prezintă valorile erorii pătratice medii (RMSE) obtinute la testare (detalii privind organizarea simularilor se găsesc în [A2]). Alpha+ și OTC+ sunt seturile Bitcoin în care arcele cu ponderi negative au fost înlăturate. Se observă că pentru acestea rezultatele sunt mai strânse. Însă acolo unde problema a fost mai dificilă, avantajul HaarNet este mai mare.

Model	Telegram	Alpha	OTC	Alpha+	OTC+
Haar	**0.2526** $\pm$ 0.0028	**0.2017** $\pm$ 0.0040	**0.2241** $\pm$ 0.0025	**0.1380** $\pm$ 0.0035	**0.1343** $\pm$ 0.0014
MagNet	0.2544 $\pm$ 0.0036	0.2034 $\pm$ 0.0041	0.2256 $\pm$ 0.0028	0.1380 $\pm$ 0.0034	0.1344 $\pm$ 0.0016
SigMaNet	0.2566 $\pm$ 0.0030	0.2060 $\pm$ 0.0036	0.2317 $\pm$ 0.0031	0.1387 $\pm$ 0.0033	0.1351 $\pm$ 0.0013

Tabel 6. Valori RMSE în predicția ponderilor (medii și variațe)

Haar-Laplacianul a fost utilizat și pentru probleme de prelucrarea semnalelor pe grafuri. De exemplu, într-o problemă tipică de denoising, rezolvată cu ajutorul GFT asociate, s-a dovedit mai bun decât MagNet, SigMaNet și construcția bazată pe SVD din [CCS23].

Bazându-ne pe aceste rezultate foarte bune, precum și pe proprietățile demonstrate, suntem încrezători că lucrarea [A2] va putea fi publicată într-o revistă de vârf. Ne propunem ca în decembrie să trimitem lucrarea la o astfel de revistă.

2.4. Software

Am implementat un repository dedicat și public (https://gitlab.cs.pub.ro/asydil) în care am introdus sub forma unor proiecte codul sursă necesar reproducerii rezultatelor incluse în majoritatea lucrărilor publicate. Acest repository reprezintă un punct de plecare pentru buna gestionare a codului sursă. De asemenea, programele sunt disponibile pe site-ul proiectului, http://asydil.upb.ro/.

Funcțiile de bază pentru DL și reprezentare rară cu atomi-mulțime au fost integrate într-o bibliotecă dedicată, astfel:

- funcții pentru reprezentare rară sunt **cone_sparse_encoding** (pentru atomi-con) și **gaussian_sparse_encoding** (pentru atomi gaussieni); metodele gaussiene sunt două: *gauss-omp* and *gauss-omp-l1*, bazate pe abordarea lacomă obișnuită, respectiv pe relaxarea cu normă L1;
- funcții pentru învățarea dicționarelor, numite **cone_dictionary_learning** și **gaussian_dictionary_learning**, versiunea gaussiană oferind metodele *gauss-dl* și *gauss-dl-l1*.

Existența mai multor implementări pentru aceeași problemă permite alegeri în funcție de aplicația avută în vedere.

Funcțiile descrise mai sus au fost incluse într-un toolbox de învățarea dicționarelor creat de noi anterior, **dictlearn**, disponibil la https://pypi.org/project/dictlearn/. Până acum, în toolbox erau doar metode pentru problema DL standard. Metodele noi și permit lucrul cu atomi-mulțime, modul de apel fiind similar.

Explicații suplimentare, suport teoretic și exemple de utilizare pot fi găsite la https://unibuc.gitlab.io/graphomaly/dictionary-learning/dl_with_infinite_set_atoms.html. De asemenea, am inclus informații relevante pe repository-ul proiectului, alături de programele asociate articolelor: https://gitlab.cs.pub.ro/asydil/dictionary-learning-with-infinite-set-atoms.

2.5. Alte rezultate

Între rezultatele indirecte ale proiectului putem menționa susținerea tezei lui Denis Ilie-Ablachim, cu titlul „Classification Methods using Dictionary Learning Algorithms”, care a obținut calificativul excelent. De asemenea, a obținut distincția de cea mai bună teză din România în AI din perioada 1.01.2023—31.07.2024, la Romanian AI Days https://days.airomania.eu/competition.

Alt doctorand, Theodor Badea, a obținut rezultate foarte bune până în acest moment, putând astfel anticipa susținerea cu succes a tezei (probabil în 2026).

2.6. Prezentare succintă a rezultatelor

O prezentare succintă a rezultatelor obținute (pe înțelesul publicului, cu cunoștințe totuși în domeniu) poate fi găsită la https://asydil.upb.ro/results/.

3. Impactul estimat al rezultatelor obţinute, cu sublinierea celui mai semnificativ rezultat obţinut

Câteva dintre rezultatele semnificative obținute în acest proiect sunt:

1. Realizarea unui **algoritm de învățare a dicționarelor pentru cazul atomilor gaussieni**.
2. Realizarea unui algoritm de învățare a dicționarelor, incluzând un algoritm distinct pentru calculul reprezentărilor rare, pentru cazul dicționarelor cu atomi-con.
3. Utilizarea cu succes a acestor algoritmi în detecția de anomalii; în particular, pe anomalii de tip dependency am obținut rezultate superioare celor ale algoritmilor existenți.
4. Implementarea eficientă, în MATLAB și Python, a acestor algoritmi.
5. Realizarea unui algoritm pentru proiectarea frame-urilor incoerente (grassmanniene), care este în acest moment cel mai bun din punctul de vedere al compromisului calitate/viteză.
6. **Propunerea un nou Laplacian pentru grafuri orientate, numit Haar-Laplacian**, cu proprietăți superioare față de Laplacienii existenți, și demonstrarea utilității lui în probleme de învățare (predicția ponderilor arcelor) și prelucrarea semnalelor pe grafuri (denoising).

Cele mai semnificative două rezultate sunt scrise cu caractere îngroșate. În cadrul strict al propunerii inițiale, cel mai mare impact ne așteptăm să-l aibă algoritmul DL pentru atomi gaussieni, deoarece este rapid și obtine rezultate foarte bune în reprezentare și mai ales în detecția de anomalii. Acest algoritm se bazează și pe noțiuni teoretice cu grad de noutate, ceea ce îl poate face și mai atractiv.

Deși este un rezultat foarte proaspăt și deci netrecut încă printr-un proces de validare a unor recenzenți externi, noul Haar-Laplacian are potențial mare de a deveni o noțiune de referință. Și în cazul lui există o inovație teoretică palpabilă. Validarea a fost făcută pe probleme de predicție în grafuri pe care se lucrează foarte mult în prezent. Faptul că am obținut rezultatele cele mai bune într-o problemă dificilă și de interes (predicția ponderilor) este extrem de încurajator.

Impactul realizărilor de mai sus se poate manifesta în cel puțin două direcții generale.

- Deoarece articolele principale au fost publicate în reviste bune și noțiunile prezentate au elemente de noutate, ne așteptăm ca ele să obțină citări și să fie utilizate pentru dezvoltări ulterioare.
- Câțiva dintre algoritmii propuși obțin cele mai bune rezultate pentru problemele pe care le rezolvă: i) proiectarea frame-urilor incoerente [R1], ii) detecția de anomalii pentru tipul dependency [R2,R3,A1], iii) predicția ponderilor arcelor în grafuri neorientate. Ne așteptăm deci ca ei să fie utilizați fie care metode de referință, fie chiar introduși în biblioteci de bază în domeniile respective (de exemplu, în ADBench pentru detecția de anomalii).

Dincolo de recunoașterea academică, potențialul aplicativ cel mai mare îl are Haar-Laplacianul, pentru că poate fi folosit în probleme concrete de mare interes. De exemplu, predicția ponderilor arcelor în grafuri orientate poate fi folosită în rețele de încredere sau chiar în rețele sociale, pentru a recomanda persoane mai puțin cunoscute; de asemenea, poate fi utilizată (în conjuncție cu alte

metode) în rețele logistice pentru deschiderea sau nu a unor noi trasee; similar, pentru aplicații de recomandare în comerț.

Nu în cele din urmă, impactul pentru membrii tineri ai echipei este clar: ei și-au adăugat publicatii valoroase la lista de lucrări, în calitate (meritată) de prim autor. Cel puțin doi dintre ei (Andra Băltoiu și Denis Ilie-Ablachim) sunt definitiv câștigați pentru o carieră didactică și de cercetare.

4. Concluzii

Proiectul "Învățarea asimetrică a dicționarelor" (Asymmetric Dictionary Learning—AsyDiL) a fost dus la bun sfârșit, urmând destul de fidel propunerea inițială. Am reușit publicarea a trei articole în reviste foarte bune în tematică principală a proiectului, învățarea dicționarelor cu atomi mulțime (de tip con și gaussian). Au fost obținute 6 articole de revistă, 7 de conferință și 4 articole încărcate pe arXiv sau încă nepublicate. Aplicațiile în detecția de anomalii au fost realizate cu succes. Au fost obținute rezultate suplimentare în analiza datelor pe grafuri orientate. Programe MATLAB și Python disponibile public însoțesc toate articolele semnificative. Considerăm deci că proiectul a fost unul reușit, atât în privința rezultatelor științifice și de implementare, cât și pentru dezvoltarea carierelor în cercetare ale membrilor tineri ai echipei.

Bibliografie

[AEH15] A. Adler, M. Elad, Y. Hel-Or, and E. Rivlin, "Sparse coding with anomaly detection," Journal of Signal Processing Systems, vol. 79, no. 2, pp. 179–188, 2015.

[BG20] A. Bovet, P. Grindrod, The activity of the far right on Telegram v2.11, 2020.

[CCS23] Y. Chen, C. Cheng, Q. Sun, Graph Fourier transform based on singular value decomposition of the directed Laplacian, Sampling Theory, Signal Processing, and Data Analysis 21 (2023).

[DBB22] B. Dumitrescu, A. Băltoiu, Ș. Budulan. Anomaly detection in graphs of bank transactions for anti money laundering applications. IEEE Access, 10:47699–47714, 2022.

[DuIr18] B. Dumitrescu, P. Irofti. Dictionary Learning Algorithms and Applications. Springer, 2018.

[Dumi17] B. Dumitrescu. Designing Incoherent Frames with Only Matrix-Vector Multiplications. IEEE Signal Proc. Letters, 24(9):1265–1269, Sep. 2017.

[ECL19] A. Elliott, M. Cucuringu, M.M. Luaces, P. Reidy, G. Reinert. Anomaly detection in networks with application to financial transaction networks, arXiv, 2019.

[FCC23] S. Fiorini, S. Coniglio, M. Ciavotta, E. Messina, SigMaNet: One Laplacian to rule them all, in: Proc. AAAI Conference on Artificial Intelligence, volume 37, 2023, pp. 7568–7576.

[HHH22] S. Han, X. Hu, H. Huang, M. Jiang, and Y. Zhao. ADBench: Anomaly detection benchmark. Advances in Neural Information Processing Systems, 35:32142–32159, 2022.

[HZS21] X. Han, H. Zhang, W. Sun, Spectral anomaly detection based on dictionary learning for sea surfaces, IEEE Geoscience and Remote Sensing Letters vol.19, pp. 1–5, 2021.

[JBS21] R. Jyothi, P. Babu, P. Stoica. Design of high-dimensional grassmannian frames via block minorization maximization. IEEE Communications Letters, 25(11):3624–3628, 2021.

[JyBa22] R. Jyothi, P. Babu. TELET: A monotonic algorithm to design large dimensional equiangular tight frames for applications in compressed sensing. Signal Processing. 2022 Jun 1;195:108503.

[KSS16] S. Kumar, F. Spezzano, V. Subrahmanian, C. Faloutsos, Edge weight prediction in weighted signed networks, in: IEEE 16th international conference on data mining (ICDM), 2016, pp. 221–230.

[KSZ08] H.P. Kriegel, M. Schubert, A. Zimek. Angle-based outlier detection in high-dimensional data. In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 444–452, 2008.

[KW16] T. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, arXiv preprint arXiv:1609.02907 (2016).

[MeDa14] A. Medra and T.N. Davidson. Flexible codebook design for limited feedback systems via sequential smooth optimization on the Grassmannian manifold. IEEE Transactions on Signal Processing, 62(5):1305–1318, 2014.

[PRK93] Y.C. Pati, R. Rezaiifar, P.S. Krishnaprasad, “Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition,” in 27th Asilomar Conf. Signals Systems Computers, Nov. 1993, vol. 1, pp. 40–44.

[PBT20] B. Pilastre, L. Boussouf, S. d’Escrivan, J. Tourneret, Anomaly detection in mixed telemetry data using a sparse representation and dictionary learning, Signal Processing, vol. 168, art. 107320, 2020.

[RuGo16] C. Rusu and N. Gonzalez-Prelcic. Designing Incoherent Frames Through Convex Techniques for Optimized Compressed Sensing. IEEE Trans.Signal Proc., 64(9):2334–2344, May 2016.

[RZE08] R. Rubinstein, M. Zibulevsky, and M. Elad. Efficient Implementation of the K-SVD Algorithm Using Batch Orthogonal Matching Pursuit. Technical Report CS-2008-08, Technion Univ., Haifa, Israel, 2008.

[StHe03] T. Strohmer and R.W. Heath. Grassmannian frames with applications to coding and communication. Appl. Comput. Harmon. Anal., 14(3):257–275, 2003.

[TDHS05] J.A. Tropp, I.S. Dhillon, R.W. Heath Jr., and T. Strohmer. Designing structured tight frames via an alternating projection method. IEEE Trans.Info. Th., 51(1):188–209, Jan. 2005.

[TSR19] B. Tahir, S. Schwarz, and M. Rupp. Constructing Grassmannian frames by an iterative collision-based packing. IEEE Signal Processing Letters, 26(7):1056–1060, 2019.

[YMW19] Y. Yuan, D. Ma, Q. Wang, Hyperspectral anomaly detection via sparse dictionary learning method of capped norm, IEEE Access, vol.7, pp.16132–16144, 2019.

[ZHB21] X. Zhang, Y. He, N. Brugnone, M. Perlmutter, M. Hirn, “MagNet: A neural network for directed graphs,” Advances in neural information processing systems, vol. 34, pp. 27003–27015, 2021.

[ZoBo15] H. Zorlein and M. Bossert. Coherence Optimization and Best Complex Antipodal Spherical Codes. IEEE Trans. Signal Proc., 63(24):6606–6615, Dec. 2015.

Director proiect,
Prof. Bogdan Dumitrescu