



Universitatea Tehnică „Gheorghe Asachi” Iași
Facultatea de Automatică și Calculatoare Iași
Specializarea: Tehnologia Informației



PROIECT

“OPTIMIZAREA UNUI MODEL SVM UTILIZAT PENTRU CLASIFICAREA TUMORILOR MALIGNNE ȘI BENIGNE FOLOSIND UN ALGORITM EVOLUTIV”

Coordonator: Prof. dr. ing. Florin Leon

Studenți: 1. Archir Maria Mirabela, 1409A
2. Postolache Mălina, 1409A

Iași, 2025

Cuprins

- 1. Descrierea problemei**
- 2. Aspecte teoretice privind algoritmul**
- 3. Funcționalitate și mod de rezolvare**
- 4. Rezultate obținute**
- 5. Contribuțiile membrilor**
- 6. Concluzii**
- 7. Bibliografie**

1. Descrierea problemei

În cadrul acestui proiect, s-a implementat o metodă de optimizare evolutivă, cu codare reală, pentru a rezolva Problema Duală a SVM-ului.

SVM-ul este un algoritm de învățare automată, utilizat în probleme de clasificare, iar obiectivul principal al acestuia este să găsească un hiperplan optim, care separă datele în două clase, maximizând distanța față de cele mai apropiate puncte (vectori suport).

Forma duală este o rescriere a problemei primare, care are scopul de a găsi soluții eficiente pentru multiplicatorii Lagrangieni, astfel încât să se maximizeze marja între clase și să se minimizeze erorile de clasificare.

În contextual problemei noastre, algoritmul este aplicat în cadrul clasificării tumorilor (maligne, benigne), pe baza setului de date de la adresa <https://archive.ics.uci.edu/ml/index.php>.

2. Aspecte teoretice privind algoritmul

Problema duală a SVM-ului presupune rescrierea problemei primare, într-o altă formă, astfel încât să se optimizeze hiperplanul de separare, prin intermediul multiplicatorilor Lagrangieni.

Forma echivalentă a problemei este:

$$\max_{\alpha} \left(\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j K(x_i, x_j) \right)$$

În contextul dat, se dorește să se optimizeze funcția obiectiv prezentată mai sus, pe baza multiplicatorilor Lagrangieni, $\alpha[i]$, care reprezintă punctele de date. Dorim să găsim punctele de la margine (vectorii suport) și să construim hiperplanul, fără a calcula w și b în mod direct.

Funcția utilizează în implementare un **Kernel Gaussian RBF**, care are rolul de a măsura similaritatea dintre două puncte date într-un spațiu multidimensional. Scopul său principal este de a permite transformarea datelor non-liniare într-un spațiu unde acestea pot deveni liniar separabile.

$$K(\mathbf{x}, \mathbf{z}) = e^{-\gamma \|\mathbf{x} - \mathbf{z}\|^2}$$

Algoritmul evolutiv este o metodă de optimizare, bazat pe mecanismele selecției naturale, iar esența acestuia este de a găsi valorile optime pentru alpha, care maximizează funcția obiectiv și respectă constrângerile impuse de problema SVM.

Acesta implică utilizarea unui set de soluții potențiale care sunt modificate în mod repetat prin intermediul unor operațiuni aleatorii pentru a găsi cea mai bună soluție posibilă. Membrii acestei populații de soluții reprezintă diferite variante posibile pentru rezolvarea problemei.

Selectăm părinții care se vor reproduce pentru a forma următoarea generație. Criteriul de selecție se bazează pe valoarea funcției de adaptare, iar selecția determină cel mai adaptat membru dintre părinți.

Funcția de adaptare - fitness arată cât de bună este o soluție, adică cât de adaptat este un cromozom.

Încrucișarea are loc cu o anumită probabilitate introdusă din interfață pentru combinarea informațiilor provenite de la doi părinți (mama și tata) care generează copilul. Se realizează apoi mutația ce are ca scop introducerea variație genetice în populație, care ar putea conduce la soluții mai bune pentru problema dată. Mutația este utilizată pentru a evita situațiile în care o anumită valoare a unei gene nu apare în populație. Prin aceasta se poate crește diversitatea populației și se poate evita convergența către un optim local.

Dupa Crossover și Mutație, se adaptează soluțiile pentru a respecta constrângerile:

$$0 \leq \alpha_i \leq C$$

$$\sum_{i=1}^N \alpha_i y_i = 0$$

Pentru gestionarea celei de a doua constrânger, s-au utilizat metodele de ajustare, prezentate în secțiunea V. C. din articolul:

https://www.researchgate.net/publication/309565420_Evolutionary_Support_Vector_Machines_A_Dual_Approach

Metoda propusă ajustează valorile $\alpha [i]$ astfel încât să aducă suma prezentată în formulă la 0

Pașii metodei de ajustare sunt:

1. Calcularea devierii inițiale s

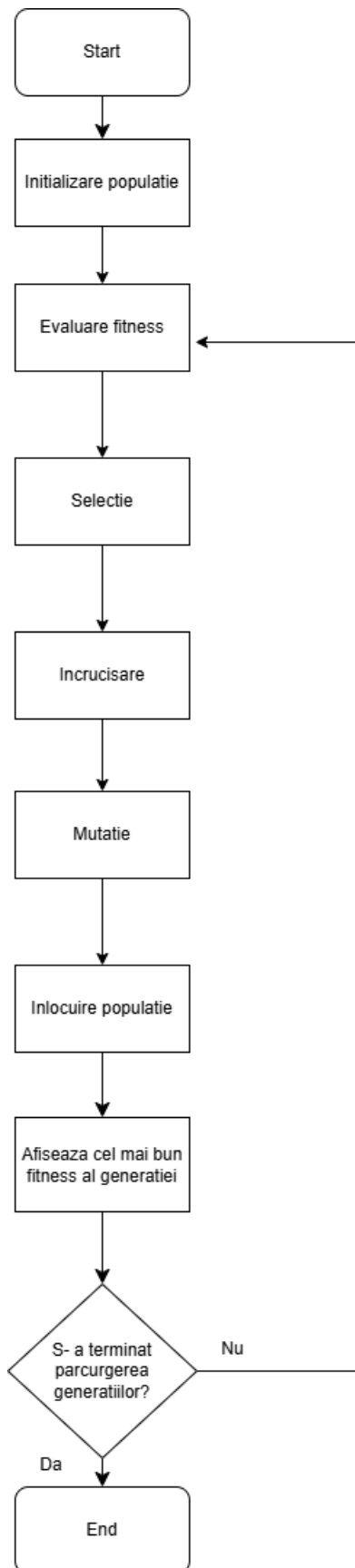
- ✓ Dacă $s=0$, constrângerea este respectată
- ✓ Dacă $s > 0$, atunci instanțele din clasa 1 trebuie ajustate
- ✓ Dacă $s < 0$, atunci instanțele din clasa -1 trebuie ajustate

2. Reducerea treptată a valorilor $\alpha[i]$

- ✓ Pentru a reduce s , se selectează aleatoriu o instanță k cu $y=+1$ sau $y=-1$, în funcție de semnul lui s

3. Se iterează până se respectă constrângerea

- ✓ Procesul se repetă până când s devine suficient de mic



3. Funcționalitate și mod de rezolvare

Aplicația realizată în C# este concepută pentru a fi folosită de un utilizator standard. Aceasta oferă o interfață simplă și intuitivă și permite utilizatorului să introducă datele de intrare, să lanseze procesul principal și să analizeze rezultatele.

Funcționarea aplicației:

1. Lansarea aplicației:

Utilizatorul deschide aplicația prin executabilul generat în Visual Studio.

2. Introducerea datelor de intrare:

Utilizatorul introduce datele necesare folosind un fișier .data, utilizând interfața aplicației. La nivel de cod, este implementată clasa de încărcare și preprocesare a datelor, locul în care ajung cele 569 de instanțe. Se normalizează fiecare instanță, se etichetează conform clasei din care face parte, iar totalul lor este împărțit: 80% date antrenament și 20% date testare.

3. Lansarea procesului (Antrenare SVM):

După ce datele sunt introduse, în clasa ProblemaDuală are loc antrenarea SVM-ului cu ajutorul algoritmului evolutiv cu codare reală. La nivel de interfață, utilizatorul inițiază procesul principal prin apăsarea butonului ANTRENARE.

Algoritmul SVM efectuează următoarele operațiuni:

- Evaluează performanța pe parcursul a 10 generații. (prin afișarea celui mai bun fitness pentru fiecare generație)
- Afișează evoluția fitness-ului în timp real, observabilă și printr-un grafic Calculează și afișează acuratețea finală a modelului pe setul de date de testare

4. Evaluarea Predicțiilor (Testare SVM):

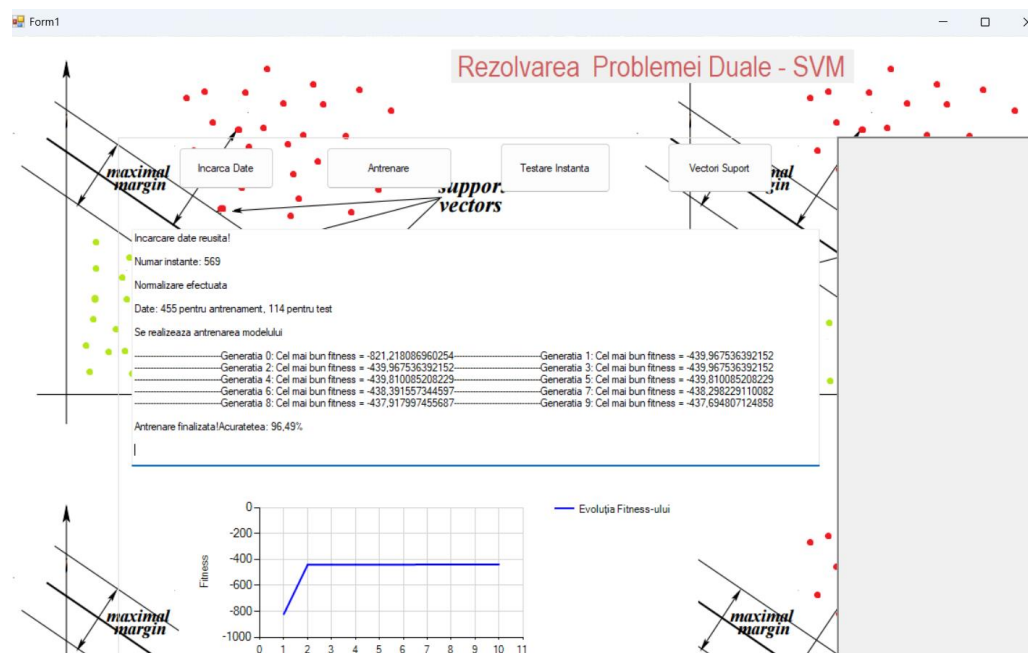
După finalizarea procesului de antrenare, utilizatorul poate verifica performanța modelului pe setul de date de testare, unde sunt observabile următoarele aspecte:

- Clasificarea fiecărei instanțe din setul de test, cu afișarea detaliilor pentru fiecare predicție: Instanțele clasificate corect sunt afișate cu text negru, iar instanțele clasificate greșit sunt marcate cu text roșu
- Calcularea și afișarea acurateței generale a clasificatorului, sub formă procentuală

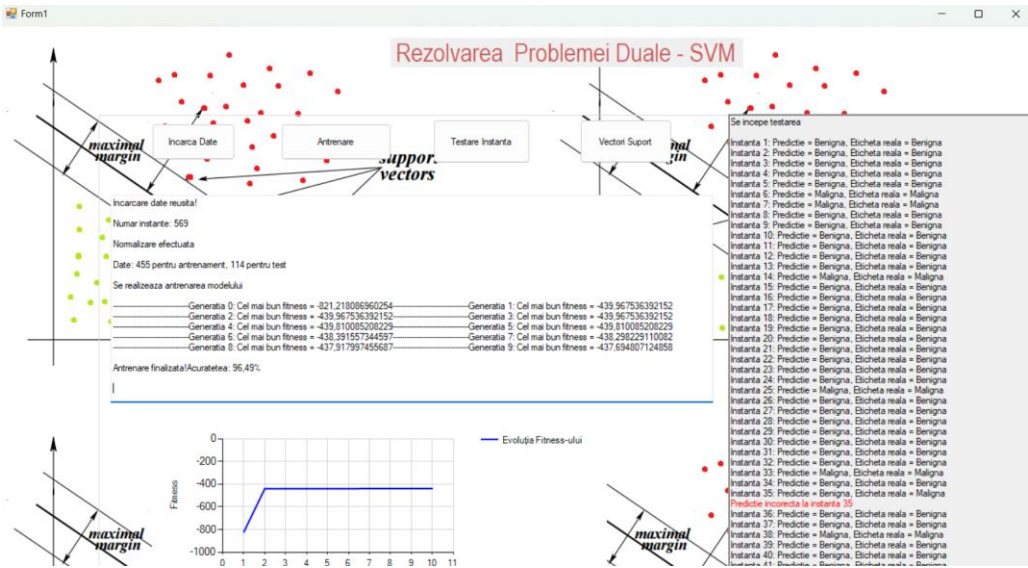
5. Analiza și vizualizarea rezultatelor:

Utilizatorul poate analiza rezultatele în mai multe moduri:

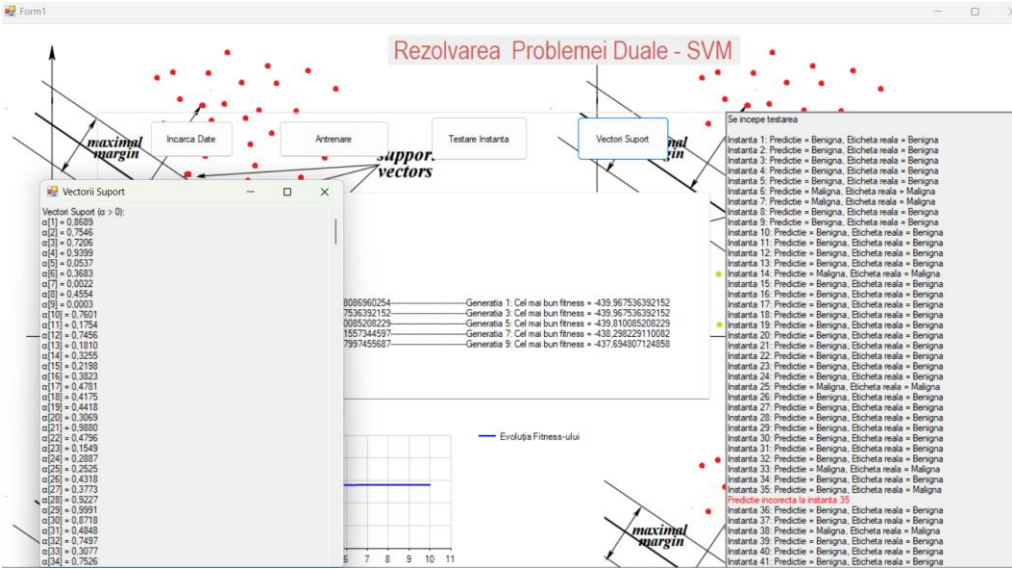
- Graficul fitness-ului: Evoluția fitness-ului este afișată pe parcursul generațiilor, oferind o reprezentare vizuală a progresului algoritmului
- Afișarea rezultatelor, precum cel mai bun fitness pentru fiecare generație, acuratețea generală, și instanțele clasificate corect sau greșit
- Vectorii suport: Utilizatorul poate vizualiza vectorii suport ($\alpha > 0$) și marginea clasificatorului prin apăsarea butonului VECTORI SUPORT.



3. Testarea



4. Vizualizarea Vectorilor Suport



5. Contribuțiile membrilor

Sarcina	Membru
ÎncărcarePreprocesareDate.cs	Mălina
ProblemaDuala.cs	Mira
Constrângeri.cs	Mira
Kernel.cs	Mira
Form.cs	Mălina

6. Concluzii

Proiectul folosește un algoritm evolutiv cu codare reală pentru a rezolva problema duală a unei mașini cu vectori suport (SVM), optimizând multiplicatorii Lagrange și parametrii, inclusiv vectorul de suport și bias-ul, cu un kernel RBF. Algoritmul îmbunătățește optimizarea prin explorarea unui spațiu larg de soluții și convergența la o soluție optimă, având aplicații în clasificare și predicție.

7. Bibliografie

<https://archive.ics.uci.edu/ml/index.php>

https://www.researchgate.net/publication/309565420_Evolutionary_Support_Vector_Machines_A_Dual_Approach

https://edu.tuiasi.ro/pluginfile.php/49457/mod_resource/content/10/IA12_RN2.pdf