BABES BOLYAI UNIVERSITY, CLUJ NAPOCA, ROMANIA FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

# Evolutia Gripei

- MIRPR report -

Team members Schiopu Adrian, [adrian.schiopu19@gmail.com](mailto:adrian.schiopu19@gmail.com)

Serpe Vlad, [vladvladserpe@gmail.com](mailto:vladvladserpe@gmail.com)

2024-2025

Abstract

Text of abstract. Short info about:

* The main idea of this project is to develop an AI-based clinical decision support system that predicts a patient's immune response to the influenza vaccine. By analyzing various clinical and demographic data, the system will identify immune predictors that discriminate between individuals who have a strong or weak response to the flu vaccine. This system will help healthcare professionals assess the likelihood of post-vaccination complications and provide personalized recommendations for vaccination strategies.
* In this project, a combination of machine learning and data science techniques will be used to identify predictors of immune response to influenza vaccination. Some of the techniques are: **Feature Selection & Dimensionality Reduction, Data Imputation, Normalization/Standardization**. The machine learning algorithms that we use are **Logistic Regression** and **Decision Tree**
* data involved in the numerical experiments; The data involved in the numerical experiments can be classified in the following types: **Demographic Data** (visit\_age, gender, race), **Clinical Data** (bmi, total\_vaccines\_received, vaccinated\_1yr\_prior, vaccine\_type\_1yr\_prior, influenza\_infection\_history, influenza\_hospitalization), **Immunological Data** (cmv\_status, ebv\_status, assay, geo\_mean, d\_geo\_mean), **Vaccine Response Data** (vaccine\_response, statin\_use, flu\_vaccination\_history), **Time-based Data** (visit\_year, visit\_day, visit\_type\_hai, mesurment\_id).
* conclude by the the results obtained.

Please

# Contents

1. [Introduction](#_bookmark0) 1
   1. [What'! Why'! How'!](#_bookmark1) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1
   2. [Paper structure and original contribution(s)](#_bookmark2) . . . . . . . . . . . . . . . . . . . . . . . . 1
2. [Scientific Problem](#_bookmark3) 3
   1. [Problem definition](#_bookmark4) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
3. [State of the art/Related work](#_bookmark5) 4
4. [Investigated approach](#_bookmark6) 5
5. [Application (Study case)](#_bookmark7) 6
   1. [App's description and the main functionalities](#_bookmark8) . . . . . . . . . . . . . . . . . . . . . . . 6
   2. [App's design](#_bookmark9) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
   3. [Implementation](#_bookmark10) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
   4. [Testing of the codebase](#_bookmark11) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
   5. [Numerical validation](#_bookmark12) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
      1. [Methodology](#_bookmark13) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
      2. [Data](#_bookmark14) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
      3. [Results](#_bookmark15) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
      4. [Discussion](#_bookmark16) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
6. [Conclusion and future work](#_bookmark17) 8
7. [Latex examples](#_bookmark18) 9

List of Tables

* 1. [The parameters of the PSO algorithm (the micro level algorithm) used to compute the](#_bookmark20) [fitness of a GA chromosome.](#_bookmark20) 10

# List of Figures

* 1. [The evolution of the swarm size during the GA generations. This results were obtained](#_bookmark19)

[for the *f*2 test function with 5 dimensions.](#_bookmark19) . . . . . . . . . . . . . . . . . . . . . . . . . 9

# List of Algorithms

1. [Logistig](#_bookmark21) Regression
2. Decision Tree 10

Chapter 1

# Introduction

## What? Why? How?

Motivate and abstractly describe the problem you are addressing and how you are addressing it.

* + - What is the (scientific) problem'!
    - Why is it important'!
    - What is your basic approach'!

A short discussion of how it fits into related work in the area is also desirable. Summarize the objectives of the current thesis and the basic results and conclusions that you will present.

## Paper structure and original contribution(s)

The research presented in this paper advances the theory, design, and implementation of several par- ticular models.

The main contribution of this report is to present an intelligent algorithm for solving the problem of *. . .*.

The second contribution of this report consists of building an intuitive, easy-to-use and user friendly software application. Our aim is to build an algorithm that will help *. . .*.

The third contribution of this thesis consists of *. . .*.

The present work contains *xyz* bibliographical references and is structured in five chapters as follows.

The first chapter/section is a short introduction in *. . .*. The second chapter/section describes *. . .*.

Sanatate pe Sistem Project's name

The chapter/section [4](#_bookmark6) details *. . .*.

Chapter 2

# Scientific Problem

## Problem definition

Give a description of the problem. Explain why it must be solved by an intelligent algorithm. Details the advantages and/or disadvantages of solving the problem by a (some) given method(s).

Precisely define the problem you are addressing (i.e. formally specify the inputs and outputs).

Elaborate on why this is an interesting and important problem.

Chapter 3

# State of the art/Related work

The theory of the methods utilised until now in order to solve the given problem.

Answer the following questions for each piece of related work that addresses the same or a similar problem.

* What is their problem and method'!
* How is your problem and method different'!
* Why is your problem and method better'!

In order to cite a given work you can use a bib file (see the example) and the cite command: 1[2](#_bookmark23)1, 1[3](#_bookmark24)1, 1[1](#_bookmark22)1, 1[5](#_bookmark26)1, 1[4](#_bookmark25)1.

Chapter 4

# Investigated approach

Describe your approach!

Describe in reasonable detail the algorithm you are using to address this problem. A psuedocode description of the algorithm you are using is frequently useful. Trace through a concrete example, showing how your algorithm processes this example. The example should be complex enough to illustrate all of the important aspects of the problem but simple enough to be easily understood. If possible, an intuitively meaningful example is better than one with meaningless symbols.

Chapter 5

# Application (Study case)

## App's description and the main functionalities

Application description

Main functionalities and their specification Add a fiow diagram (very useful for the client)

## App's design

* use cases
* diagrams (class diagram, sequence diagram, database structure)

## Implementation

## Algoritmii folositi pentru clasificare

**Logistic Regression**

**Logistic Regression** is a classification algorithm commonly used to predict binary outcomes (e.g., high vs. low response). It calculates the probability that a data point belongs to a specific class by using a linear combination of input features and applying a sigmoid function, which transforms the result into a probability between 0 and 1. Based on this probability, the model assigns the data point to one of two classes.

To train the model, logistic regression uses a cost function called cross-entropy, which measures the difference between predicted and actual values. Through an optimization technique called gradient descent, the model iteratively adjusts its parameters to minimize the cost and improve accuracy.

Logistic regression is effective for binary classification tasks, providing clear interpretability of feature importance, which is valuable for understanding factors influencing the immune response to the flu vaccine.

**Decision Tree**

Arborele de decizie este un algoritm de clasificare și regresie care utilizează o structură de tip arbore pentru a lua decizii bazate pe caracteristicile de intrare. Fiecare nod din arbore reprezintă o întrebare sau o condiție asupra unei caracteristici, iar fiecare ramură reprezintă răspunsul (sau decizia) posibilă. Algoritmul împarte datele în subseturi la fiecare nod, până când ajunge la frunze, unde se atribuie o clasă finală sau o valoare de predicție.

Arborii de decizie sunt eficienți și ușor de interpretat, deoarece oferă explicații vizuale pentru clasificări. Aceștia sunt valoroși în proiectul nostru, deoarece permit identificarea factorilor principali care influențează răspunsul la vaccinul antigripal și ajută la dezvoltarea unui sistem de suport decizional în domeniul sănătății.

## Rezultatele obtinute

**Logistic Regression pe 2 dataset-uri de mici dimensiuni:**

Am aplicat Logistic Regression pentru a prezice un outcome ce poate fii low sau high, avănd 75 % train data si 25% test data. Am realizat o acuratete de 0.66 respectiv 0.7 pentru cele 2 dataset-uri.

Am încercat de asemenea combinarea celor 2 dataset-uri pentru un număr mai mare de date, dar deoarece nu toți parametrii au fost identici, au fost excluși câțiva parametrii și am ajuns la o acuratete de 0.525.

Am afișat informați despre rezultatul obținut în fișier, cum ar fii acuratetea, raportul de clasificare si importanța fiecarui parametru.

**Data1:**

Acuratețea modelului: 0.66

Raport de clasificare:

precision recall f1-score support

high 0.33 0.25 0.29 4

low 0.75 0.82 0.78 11

accuracy 0.67 15

macro avg 0.54 0.53 0.53 15

weighted avg 0.64 0.67 0.65 15

A graph with a green line and red line

Description automatically generated

**Data2:**

Acuratețea modelului: 0.7

Raport de clasificare:

precision recall f1-score support

high 0.50 0.33 0.40 3

low 0.75 0.86 0.80 7

accuracy 0.70 10

macro avg 0.62 0.60 0.60 10

weighted avg 0.68 0.70 0.68 10

A graph with green and red lines

Description automatically generated

**Data1+Data2:**

Acuratețea modelului: 0.52

Raport de clasificare:

precision recall f1-score support

high 0.00 0.00 0.00 12

low 0.64 0.75 0.69 28

accuracy 0.53 40

macro avg 0.32 0.38 0.34 40

weighted avg 0.45 0.53 0.48 40

A graph with red and green lines

Description automatically generated

**Decision Tree pe 2 dataset-uri de mici dimensiuni:**

Am aplicat Decision Tree pentru a prezice un outcome ce poate fii low sau high, avănd aceleasi date folosite anterior. Am realizat o acuratete de 0.8 respectiv 0.7 pentru cele 2 dataset-uri.

Am afișat informați despre rezultatul obținut în fișier, cum ar fii acuratetea, raportul de clasificare si importanța fiecarui parametru.

**Data1:**

Acuratețea modelului: 0.8

Raport de clasificare:

precision recall f1-score support

high 0.57 1.00 0.73 4

low 1.00 0.73 0.84 11

accuracy 0.80 15

macro avg 0.79 0.86 0.78 15

weighted avg 0.89 0.80 0.81 15

A graph with a green line and red line

Description automatically generated

**Data2:**

Acuratețea modelului: 0.7

Raport de clasificare:

precision recall f1-score support

high 0.50 0.67 0.57 3

low 0.83 0.71 0.77 7

accuracy 0.70 10

macro avg 0.67 0.69 0.67 10

weighted avg 0.73 0.70 0.71 10

A graph with a line and a line

Description automatically generated

**Logistic Regression si Decision Tree pe dataset-ul mare:**

Am aplicat Logistic Regression si apoi Decision Tree pentru a prezice un outcome ce poate fii low sau high, avănd 90 % train data si 10% test data. Am realizat o acuratete de 0.62 respectiv 0.64 pe cei 2 algoritmi.

Am afișat informați despre rezultatul obținut în fișier, cum ar fii acuratetea, raportul de clasificare si importanța fiecarui parametru.

**Logistic Regression:**

Acuratețea modelului: 0.62

Raport de clasificare:

precision recall f1-score support

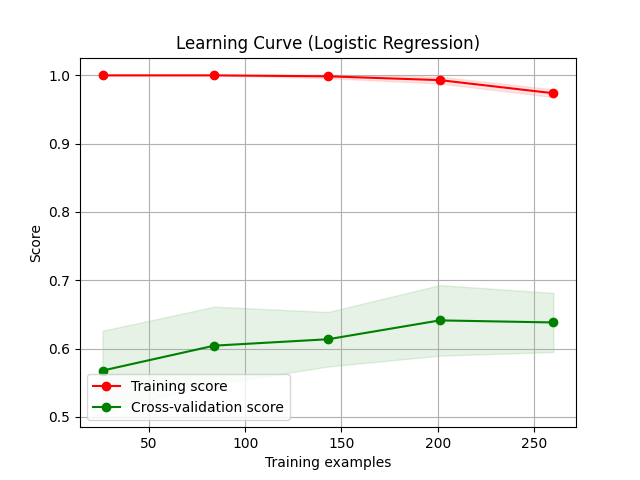
0.0 0.64 0.82 0.72 22

1.0 0.56 0.33 0.42 15

accuracy 0.62 37

macro avg 0.60 0.58 0.57 37

weighted avg 0.61 0.62 0.60 37



**Decision Tree:**

Acuratețea modelului: 0.64

Raport de clasificare:

precision recall f1-score support

0.0 0.67 0.82 0.73 22

1.0 0.60 0.40 0.48 15

accuracy 0.65 37

macro avg 0.63 0.61 0.61 37

weighted avg 0.64 0.65 0.63 37

A graph with a green line and red dotted line

Description automatically generated

**Logistic Regression si Decision Tree pe dataset-ul mare prelucrat:**

Am prelucrat dataset-ul original pentru a obtine un dataset nou cu fiecare rand reprezentand un pacient. Pe dataset-ul rezultat am salvat unul unde raman doar coloanele cu cel mult 75% valori goale, respective 50% valori goale.

Dataset1: 363 pacienti cu 384 de features folositi pentru antrenarea modelului

Dataset2: 363 pacienti cu 102 de features folositi pentru antrenarea modelului

Rezultatele obtinute sunt urmatoarele:

Logistic regression improved pe dataset 75%

Acuratetea modelului rezultat este de 65%

A graph with red and green lines

Description automatically generated

Decision Tree improved pe dataset 75%

Acuratetea modelului este de 75%

Max depth: 100000000000

Min samples split: 35

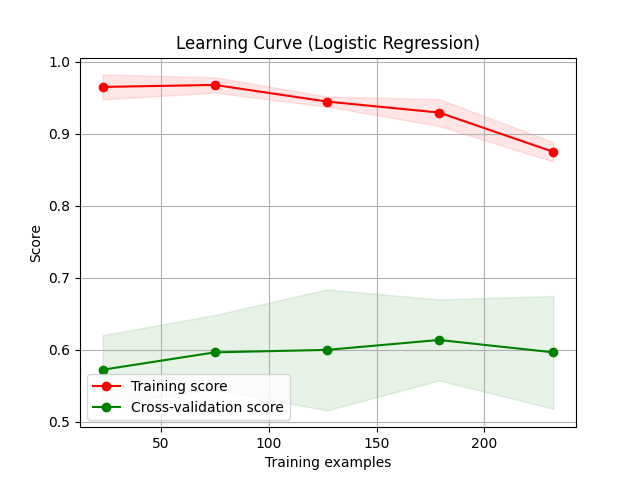
Min samples leaf: 10

A graph with green and red lines

Description automatically generated

Logistic Regression improved cu dataset 50%

Acuratetea modelului este de 65%



Decision Tree improved pe dataset 50%

Acuratetea modelului rezultat este de 67%

Max depth: 100000000000

Min samples split: 35

Min samples leaf: 10

A graph with a red line and green line

Description automatically generated

## Testing of the codebase

## Numerical validation

Explain the experimental methodology and the numerical results obtained with your approach and the state of art approache(s).

Try to perform a comparison of several approaches.

Team's name Project's name

Statistical validation of the results.

* + 1. Methodology
       - What are criteria you are using to evaluate your method'!
       - What specific hypotheses does your experiment test'! Describe the experimental methodology that you used.
       - What are the dependent and independent variables'!
       - What is the training/test data that was used, and why is it realistic or interesting'! Exactly what performance data did you collect and how are you presenting and analyzing it'! Comparisons to competing methods that address the same problem are particularly useful.
    2. Data

Describe the used data.

* + 1. Results

Present the quantitative results of your experiments. Graphical data presentation such as graphs and histograms are frequently better than tables. What are the basic differences revealed in the data. Are they statistically significant'!

* + 1. Discussion
       - Is your hypothesis supported'!
       - What conclusions do the results support about the strengths and weaknesses of your method compared to other methods'!
       - How can the results be explained in terms of the underlying properties of the algorithm and/or the data.

Chapter 6

# Conclusion and future work

Try to emphasise the strengths and the weaknesses of your approach. What are the major shortcomings of your current method'! For each shortcoming, propose additions or enhancements that would help overcome it.

Briefiy summarize the important results and conclusions presented in the paper.

* + - * What are the most important points illustrated by your work'!
      * How will your results improve future research and applications in the area'!

Chapter 7

# Latex examples

Item example:

* + - * content of item1
      * content of item2
      * content of item3 Figure example

*. . .* (see Figure [7.1](#_bookmark19))

Fitness

Figure 7.1: The evolution of the swarm size during the GA generations. This results were obtained for the *f*2 test function with 5 dimensions.

0.049

0.039

Metropolis

Glauber

0.029

0.019

0.009

-0.001

1 5 9 13 17 21 25 29 33 37 41 45

Number of Generations

Table example: (see Table [7.1](#_bookmark20)) Algorithm example

*. . .* (see Algorithm [1](#_bookmark21)).

Team's name Project's name

Table 7.1: The parameters of the PSO algorithm (the micro level algorithm) used to compute the fitness of a GA chromosome.

Parameter Value Number of generations 50

Number of function evaluations/generation 10

Number of dimensions of the function to be 5

optimized

Learning factor *c*1 2

Learning factor *c*2 1.8

Inertia weight 0.5 + *rand*()

2

Algorithm 1 SGA - Spin based Genetic AQlgorithm BEGIN

(g Randomly create the initial GA population.

(g Compute the fitness of each individual. for i=1 TO NoOfGenerations do

for j=1 TO PopulationSize do

p *←* RandomlySelectParticleFromGrid();

n *←* RandomlySelectParticleFromNeighbors(p); (g Crossover(p, n, off);

(g Compute energy ∆*H*

if ∆*H* satisfy the Ising condition then (g Replace(p,off);

end if end for

end for END

Bibliography

|  |  |
| --- | --- |
| 111 | E. Berlekamp, J. Conway, and J. Guy. Winning Ways. Academic Press Inc., London, 1982. |
| 121 | J. Kennedy and R. C. Eberhart. Particle swarm optimization. In International Conference on Neural Networks, pages 1942-1948. IEEE, 1995. |
| 131 | B. Koh, A. George, R. Haftka, and B. Fregly. Parallel asynchronous particle swarm optimization. International Journal for Numerical Methods in Engineering, 67(4):578-595, 2006. |
| 141 | Mozilla. Firefox iOS application. [link](https://github.com/mozilla-mobile/firefox-ios), 2018. |
| 151 | Rainer Storn and Kenneth Price. Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, International Computer Science Institute, Berkeley, CA, March 1995. |