

Универзитет у Београду
Електротехнички факултет

Јелица Н. Цинцовић

**Детектовање неуродегенеративних
поремећаја коришћењем модификоване
метахеуристике за оптимизовање
параметара модела машинског учења**

Докторска дисертација

Београд, 2025.

University of Belgrade
School of Electrical Engineering

Jelica N. Cincović

**Neurodegenerative condition detection using
modified metaheuristic for optimising
parameters of machine learning models**

Doctoral Dissertation

Belgrade, 2025.

Ментор

др Милош Цветановић, ванредни професор
Универзитет у Београду, Електротехнички факултет

Чланови комисије

др Бошко Николић, редовни професор
Универзитет у Београду – Електротехнички факултет

др Владимир Јоцовић, доцент
Универзитет у Београду – Електротехнички факултет

др Милица Ђурић-Јовичић, виши научни сарадник
Универзитет у Београду – Иновациони центар Електротехничког факултета

др Наташа Драгашевић Мишковић, редовни професор
Универзитет у Београду – Медицински факултет

др Саша Стојановић, ванредни професор
Универзитет у Београду – Електротехнички факултет

Датум одбране

Захвалница

Највећу и најискренију захвалност дугујем својим родитељима, мами Љубици и тати Небојши који су ми омогућили безбрижан живот препун могућности, који су увек веровали у мене и подржавали све моје изборе као и тај да постанем професор на Електротехничком факултету. Хвала вам неизмерно на свему, и ова докторска дисертација је посвећена вама.

Захвалила бих се и професорима Милошу Цветановићу и Бошку Николићу, који су ми пружили велику подршку током целих докторских студија, а посебно током писања докторске тезе, као и целој Катедри за рачунарску технику и информатику, која је додатно улепшала мој рад на факултету и због које је сваки дан проведен на послу право уживање.

Захваљујем се и др Милици Ћурић-Јовичић на стручној и научној подршци из области биомедицинског инжењерства, чији су савети и анализе били кључни за обликовање методологије и интерпретацију резултата ове докторске дисертације.

Захвалност дугујем и мом дечку Стефану који је заједно са мном пролазио кроз све фазе и изазове докторских студија, мојим дивним пријатељицама, куми Марији која је свим срцем навијала за мене још од гимназије, и Маријани са којом сам заједно корачала кроз студентске дане, као и свим осталим пријатељима који су ми били велика подршка.

Хвала вам свима!

Наслов докторске дисертације

Детектовање неуродегенеративних поремећаја коришћењем модификоване метахеуристике за оптимизовање параметара модела машинског учења

Сажетак

Рано детектовање неуродегенеративних обољења попут Паркинсонове болести, од кључног је значаја за добре резултате лечења. Велики број непознатих фактора у болести, попут тачног узрока настанка, непостојања коначног теста у дијагностиковању и непостојања лека за заустављање прогресије болести, још више указују на потребу за превазилажењем проблема ране детекције. Додатни отежавајући фактор је постојање болести које подсећају на Паркинсонову болест, попут мултипле системске атрофије и прогресивне супрануклеарне парализе, које се морају другачије лечити. Модели машинског учења могу се искористити у сврхе детектовања болести, решавањем проблема класификације. Сами модели имају велики број хиперпараметара који морају бити оптимизовани како би се дошло до што бољих резултата. Оптимизација хиперпараметара решава се алгоритмима метахеуристике. У овој тези приказани се резултати експеримената који детектују Паркинсонову болест и друга неуродегенеративна обољења користећи скуп података о тапкању прстима и скуп података о ходу. Подаци се анализирају кроз моделе рекурентних неуралних мрежа, рекурентних неуралних мрежа са механизмом пажње и екстремног градијентног појачивача. Хиперпараметри модела су оптимизовани модификованом метахеуристиком хиперболичког синуса и косинуса која додатно користи механизам вештачке колоније пчела и квази-рефлексивног учења. Нова модификована метахеуристика поређена је већ постојећим метахеуристикама и кроз све експерименте је показала завидне резултате.

Кључне речи

машинско учење, метахеуристике, оптимизација хиперпараметара, неуралне мреже, Паркинсонова болест

Научна област

Електротехника и рачунарство

Ужа научна област

Рачунарска техника и информатика

Doctoral dissertation title

Neurodegenerative condition detection using modified metaheuristic for optimising parameters of machine learning models

Abstract

Early detection of neurodegenerative diseases such as Parkinson's disease is crucial for good treatment outcomes. The large number of unknown factors in the disease, such as the exact cause, the lack of a definitive diagnostic test, and the lack of a cure to stop the progression of the disease, further highlight the need to overcome the problem of early detection. An additional complicating factor is the existence of diseases that resemble Parkinson's disease, such as Multiple System Atrophy and Progressive Supranuclear Palsy, which must be treated differently. Machine learning models can be used for the purpose of detecting neurodegenerative diseases by solving classification problems. The models themselves have a large number of hyperparameters that must be optimized in order to achieve the best possible results. Hyperparameter optimization is solved by metaheuristic algorithms. This thesis presents the results of experiments that detect Parkinson's disease and other neurodegenerative conditions using a finger tapping dataset and a gait dataset. The data are analysed through recurrent neural network, recurrent neural network with attention mechanism, and extreme gradient boosting models. The hyperparameters of the model are optimized with a modified hyperbolic sine and cosine metaheuristic that additionally uses the artificial bee colony mechanism and quasi-reflective learning. The new modified metaheuristic was compared to existing metaheuristics and showed impressive results throughout all experiments.

Key words

machine learning, metaheuristics, hyperparameter optimization, neural networks, Parkinsons disease

Scientific field

Electrical Engineering and Computing

Scientific subfield

Computer Engineering and Informatics

Садржај

Садржај.....	I
Списак слика.....	III
Списак табела.....	VII
1. Увод	1
2. Неуродегенеративна обољења: Паркинсонова болест и атипични Паркинсонизми	3
2.1. Паркинсонова болест.....	5
2.2. Атипични Паркинсонизми	6
2.2.1. Прогресивна супрануклеарна парализа	6
2.2.2. Мултипла системска атрофија	7
3. Машинско учење и алгоритми машинског учења.....	8
3.1. Логистичка регресија.....	9
3.2. Наивни Бајесов класификатор	10
3.3. К-најближих суседа.....	11
3.4. Метода потпорних вектора.....	13
3.5. Стабла одлучивања	14
3.6. Ансамбл учење	16
3.6.1. Насумичне шуме	16
3.6.2. Ада појачивач	17
3.6.3. Градијентни појачивач	18
3.6.4. Екстремни градијентни појачивач	19
3.7. Дубоко учење.....	22
3.7.1. Неуралне мреже.....	22
3.7.2. Рекурентне неуралне мреже.....	24
3.7.3. Дуга краткорочна меморија.....	25
3.7.4. Неуралне мреже оплеменењене механизмом пажње.....	26
3.7.5. Конволуционе неуралне мреже.....	27
4. Преглед постојећих решења у области Паркинсонове болести и машинског учења	29
4.1. Скупови података	29
4.2. Преглед литературе	33
5. Оптимизација алгоритама машинског учења	36
5.1. Претрага мреже и насумична претрага	36
5.2. Бајесова оптимизација	37
5.3. Метакхеуристике.....	37

5.3.1.	Оптимизација ројем честица.....	38
5.3.2.	Оптимизација колонијом мрава	39
5.3.3.	Алгоритам свица	41
5.3.4.	Алгоритам сивог вука	42
5.3.5.	Алгоритам грбавог кита	44
5.3.6.	Алгоритам мождане олује	46
5.3.7.	Генетски алгоритам.....	47
5.3.8.	Алгоритам диференцијалне еволуције.....	49
5.3.8.1.	Адаптиван алгоритам диференцијалне еволуције са историјом успеха и линеарном редукцијом популације.....	50
5.3.8.2.	Адаптиван алгоритам диференцијалне еволуције са историјом успеха и линеарном редукцијом популације уз рад са ограничењима.....	51
6.	Метахеуристика хиперболичког синуса и косинуса и њена модификација	54
7.	Експерименти и резултати.....	60
7.1.	Коришћени скупови података	60
7.1.1.	Скуп података о тапкању прстима.....	60
7.1.2.	Скуп података о ходу	63
7.2.	Експерименти	67
7.2.1.	Експеримент1: Рекурентне неуралне мреже	67
7.2.2.	Експеримент2: Рекурентне неуралне мреже са механизмом пажње.....	73
7.2.3.	Експеримент3: Екстремни градијентни појачивач.....	79
7.2.4.	Експеримент4: Екстремни градијентни појачивач.....	86
7.3.	Статистичка доказивања.....	92
8.	Будућа унапређења.....	98
9.	Закључак	101
	Литература	103

Списак слика

Слика 2.1. Структура неурона.....	3
Слика 2.2. МР снимци средњег мозга за особе са ПСП	7
Слика 3.1. Подела машинског учења.....	8
Слика 3.2. Приказ линеарне и логистичке функције	9
Слика 3.3. К-најближих суседа: почетно стање алгоритма.....	11
Слика 3.4. К-најближих суседа: налажење суседа	11
Слика 3.5. К-најближих суседа: класификовање новог податка.....	12
Слика 3.6. Метода потпорних вектора: хиперраван раздвајања	13
Слика 3.7. Метода потпорних вектора: тврда (лево) и мека (десно) маргина.....	13
Слика 3.8. Стабло одлучивања.....	14
Слика 3.9. Илустрација класификације стабла одлучивања	15
Слика 3.10. Насумичне шуме.....	16
Слика 3.11. Ада појачивач	17
Слика 3.12. Градијентни појачивач.....	18
Слика 3.13. Екстремни градијентни појачивач.....	19
Слика 3.14. Екстремни градијенти појачивач: изградња стабла.....	20
Слика 3.15. Екстремни градијентни појачивач: решавање проблема недостајућих података..	21
Слика 3.16. Структура неуралне мреже	22
Слика 3.17. Приказ неурона у неуралној мрежи	23
Слика 3.18. Рекурентна неурална мрежа	24
Слика 3.19. Јединица дуге краткорочне меморије	25
Слика 3.20. Глобални механизам пажње.....	26
Слика 3.21. Локални механизам пажње	26
Слика 3.22. Поређење Луонг и Бахнданау механизма пажње.....	27
Слика 3.23. Архитектура конволуционе неуралне мреже	28
Слика 3.24. Примена филтера на улазни вектор конволуционе неуралне мреже	28
Слика 5.1. Претрага мреже (лево) и насумична претрага (десно).....	36
Слика 5.2. Бајесова оптимизација.....	37
Слика 5.3. Ток алгоритма оптимизације ројем честица	38
Слика 5.4. Понашање мрава	39
Слика 5.5. Ток алгоритма оптимизације колонијом мрава.....	40
Слика 5.6. Ток алгоритма свица.....	41
Слика 5.7. Ток алгоритма сивог вука.....	42

Слика 5.8. Кретање вукова у алгоритму сивог вука.....	43
Слика 5.9. Мрежа мехурића грбавог кита.....	44
Слика 5.10. Ток алгоритма грбавог кита.....	45
Слика 5.11. Ток алгоритма мождане олује.....	46
Слика 5.12. Кластеровање у алгоритму мождане олује.....	46
Слика 5.13. Креирање нове индивидуе у алгоритму мождане олује.....	46
Слика 5.14. Ток генетском алгоритма.....	47
Слика 5.15. Комбиновање јединки у генетском алгоритму.....	48
Слика 5.16. Процес генетског алгоритма.....	48
Слика 5.17. Алгоритам диференцијалне еволуције.....	49
Слика 5.18. Ток L-SHADE алгоритма.....	50
Слика 5.19. Ажурирање историје у L-SHADE алгоритму.....	51
Слика 5.20. Ток COLSHADE алгоритма.....	52
Слика 5.21. Креирање нових решења у COLSHADE алгоритму Левијевим летом.....	53
Слика 6.1. Шематски дијаграм SCHO алгоритма.....	56
Слика 6.2. Дијаграм тока SCHO алгоритма.....	56
Слика 6.3. Псеудокод SCHO алгоритма.....	57
Слика 6.4. Вештачка колонија пчела.....	58
Слика 6.5. Модификовани SCHO алгоритам.....	59
Слика 7.1. Скуп података о тапкању прстима: Сензори прикачени на шаку учесника.....	61
Слика 7.2. Скуп података о тапкању прстима: Подаци са жироскопа за здраву особу.....	61
Слика 7.3. Скуп података о тапкању прстима: Подаци са жироскопа за особу са Паркинсоновом болешћу.....	62
Слика 7.4. Скуп података о тапкању прстима: Подаци са жироскопа за особу са прогресивном супрануклеарном парализом.....	62
Слика 7.5. Скуп података о тапкању прстима: Подаци са жироскопа за особу са мултиплом системском атрофијом.....	63
Слика 7.6. Скуп података о ходу: расподела учесника.....	63
Слика 7.7. Скуп података о ходу: Примери из прве студије.....	64
Слика 7.8. Скуп података о ходу: Примери из друге студије.....	65
Слика 7.9. Скуп података о ходу: Примери из треће студије.....	65
Слика 7.10. Скуп података о ходу: Одузимање у серијама по 7.....	66
Слика 7.11. Експеримент1: Рекурентне неуралне мреже: Voxplot дијаграм функције циља ...	70
Слика 7.12. Експеримент1: Рекурентне неуралне мреже: Виолина дијаграм за индикатор функцију.....	71

Слика 7.13. Експеримент1: Рекурентне неуралне мреже: Конвергенцијске стопе за функцију циља.....	71
Слика 7.14. Експеримент1: Рекурентне неуралне мреже: Конвергенцијске стопе за индикатор функцију.....	72
Слика 7.15. Експеримент1: Рекурентне неуралне мреже: Крива прецизности и опозива	72
Слика 7.16. Експеримент1: Рекурентне неуралне мреже: Конфузиона матрица	73
Слика 7.17. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: <i>Boxplot</i> дијаграм функције циља	76
Слика 7.18. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Виолина дијаграм за индикатор функцију.....	77
Слика 7.19. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Конвергенцијске стопе за функцију циља	77
Слика 7.20. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Конвергенцијске стопе за индикатор функцију.....	78
Слика 7.21. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Крива прецизности и опозива	78
Слика 7.22. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Конфузиона матрица.....	79
Слика 7.23. Експеримент3: Екстремни градијентни појачивач: <i>Boxplot</i> дијаграм функције циља	83
Слика 7.24. Експеримент3: Екстремни градијентни појачивач: Виолина дијаграм за индикатор функцију.....	83
Слика 7.25. Експеримент3: Екстремни градијентни појачивач: Конвергенцијске стопе за функцију циља.....	84
Слика 7.26. Експеримент3: Екстремни градијентни појачивач: Конвергенцијске стопе за индикатор функцију	84
Слика 7.27. Експеримент3: Екстремни градијентни појачивач: Крива прецизности и опозива	85
Слика 7.28. Експеримент3: Екстремни градијентни појачивач: Конфузиона матрица	85
Слика 7.29. Експеримент4: Екстремни градијентни појачивач: <i>Boxplot</i> дијаграм функције циља	89
Слика 7.30. Експеримент4: Екстремни градијентни појачивач: Виолина дијаграм за индикатор функцију.....	89
Слика 7.31. Експеримент4: Екстремни градијентни појачивач: Конвергенцијске стопе за функцију циља.....	90
Слика 7.32. Експеримент4: Екстремни градијентни појачивач: Конвергенцијске стопе за индикатор функцију	90
Слика 7.33. Експеримент4: Екстремни градијентни појачивач: Крива прецизности и опозива	91

Слика 7.34. Експеримент4: Екстремни градијентни појачивач: Конфузиона матрица	91
Слика 7.35. KDE график за први експеримент.....	92
Слика 7.36. KDE график за други експеримент	93
Слика 7.37. KDE график за трећи експеримент	93
Слика 7.38. Средње вредности функције циља за четврти експеримент.....	96
Слика 8.1. Структура ехо мреже стања	99
Слика 8.2. Начини повезивања неурона у резервоару	99
Слика 8.3. Комбиновање модела машинског учења са ехо мрежама стања	100

Списак табела

Табела 1. Преглед скупова података који служе за анализирање немоторних симптома Паркинсонове болести и атипичних Паркинсонизама.....	30
Табела 2. Преглед скупова података који служе за анализирање моторних симптома Паркинсонове болести и атипичних Паркинсонизама.....	31
Табела 3. Преглед радова који се баве анализом немоторних симптома Паркинсонове болести и атипичних Паркинсонизама коришћењем модела машинског учења.....	33
Табела 4. Преглед радова који се баве анализом говора и покрета шаке код Паркинсонове болести и атипичних Паркинсонизама коришћењем модела машинског учења	34
Табела 5. Преглед радова који се баве анализом хода код Паркинсонове болести и атипичних Паркинсонизама коришћењем модела машинског учења	35
Табела 6. Демографски подаци учесника за скуп података о тапкању прстима.....	60
Табела 7. Демографски подаци учесника за скуп података о ходу	64
Табела 8. Експеримент1: Рекурентне неуралне мреже: Резултати функције циља	68
Табела 9. Експеримент1: Рекурентне неуралне мреже: Резултати индикатор функције.....	68
Табела 10. Експеримент1: Рекурентне неуралне мреже: Детаљно поређење најбољих модела	69
Табела 11. Експеримент1: Рекурентне неуралне мреже: Вредности хиперпараметара за најбоље моделе.....	70
Табела 12. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Резултати функције циља	74
Табела 13. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Резултати индикатор функције	74
Табела 14. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Детаљно поређење најбољих модела	75
Табела 15. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Вредности хиперпараметара за најбоље моделе	76
Табела 16. Експеримент3: Екстремни градијентни појачивач: Резултати функције циља.....	80
Табела 17. Експеримент3: Екстремни градијентни појачивач: Резултати индикатор функције	81
Табела 18. Експеримент3: Екстремни градијентни појачивач: Вредности хиперпараметара за најбоље моделе	81
Табела 19. Експеримент3: Екстремни градијентни појачивач: Детаљно поређење најбољих модела.....	82
Табела 20. Експеримент4: Екстремни градијентни појачивач: Резултати функције циља.....	86
Табела 21. Експеримент4: Екстремни градијентни појачивач: Резултати индикатор функције	87

Табела 22. Експеримент4: Екстремни градијентни појачивач: Вредности хиперпараметара за најбоље моделе	87
Табела 23. Експеримент4: Екстремни градијентни појачивач: Детаљно поређење најбољих модела.....	88
Табела 24. П-вредности за Шафир-Вилк тест, за све експерименте	94
Табела 25. П-вредности за Леванов тест, за четврти експеримент.....	94
Табела 26. П-вредности за Вилкоксонов тест, за прва три експерименте	94
Табела 27. Клифове делте за прва два експеримента	95
Табела 28. Клифове делте за трећи експеримент	95
Табела 29. П-вредности за Т-тест, за четврти експеримент	95

1. Увод

Неуродегенеративна обољења се дефинишу као болести изазване дегенерацијом неурона у централном нервном систему [1]. Најзаступљенија неуродегенеративна обољења су Алцхајмерова, Паркинсонова и Хантингтонова болест.

Главним узрочником настанка Паркинсонове болести сматра се одумирање ћелија у црној супстанци, задужених за лучење допамина, а симптоми болести могу бити моторни и немоторни. У моторне симптоме спадају тремор, брадикинезија, ригидност или нестабилност при ходу, док се од немоторних симптома може уочити депресија, проблеми са спавањем или гастроентеролошки проблеми. Лечење болести заснива се на ублажавању симптома лековима задуженим за повећање нивоа допамина.

Око 75% пацијената који имају видне моторне симптоме болести се коректно дијагностикују Паркинсоновом болешћу, међутим око 25% пацијената ће имати атипични Паркинсонов поремећај. У атипичне Паркинсонизме спадају прогресивна супрануклеарна парализа, мултипла системска атрофија, кортикобазална дегенерација и друге. Уколико се атипични Паркинсонизам погрешно дијагностикује Паркинсоновом болешћу, и започне се са лековима за подстицање лучења допамина, симптоми болести се могу додатно погоршати. Ваљано постављање дијагнозе у оваквим ситуацијама је од кључног значаја за успешно третирање пацијената.

Конкретни биомаркери за Паркинсонову болест и атипичне Паркинсонизме још увек не постоје. Утврђивање присуства Паркинсонове болести може се радити магнетном резонанцом и електроенцефалографијом. Поред ових анализа, могу се посматрати и промене хода, покрета и говора пацијента, чија одступања могу указати на постојање болести. Симптоми болести се углавном годинама погоршавају, па је од кључне важности што пре детектовати присуство неуродегенеративних обољења [2].

Машинско учење, као техника која треба да уочи ствари које би потенцијално промакле људском оку, свакако налази примену и у дијагностиковању болести. Анализа слика и сигнала може се унапредити моделима машинског учења, који на основу великог броја података могу произвести сазнања која би помогла лекарима у закључивању. У зависности од врсте података машинско учење може бити надгледано, ненадгледано, полунадгледано и учење подстицајем. Код надгледаног учења, модел се тренира над означеним подацима, а сами алгоритми надгледаног машинског учења решавају проблеме класификације и регресије. Класификацијом се подаци смештају у одређену класу, док се регресијом предвиђа нека нумеричка вредност.

Сами алгоритми машинског учења имају велики број хиперпараметара које је потребно подесити на најбоље вредности, како би сам модел могао да извуче што квалитетније закључке. Метода покушаја и грешке, код подешавања великог броја хиперпараметара може бити веома спора и скупа по погледу ресурса, и због тога је било потребно пронаћи елегантније решење.

Метахеуристике су функције вишег реда, и њихов задатак је да што ефикасније, пронађу довољно добро решење, а могу се користити за оптимизовање хиперпараметара модела машинског учења [3]. Оне не гарантују проналажење најбољег решења, али је и довољно добро решење боље од модела који због лошег избора хиперпараметара није успео да произведе квалитетне резултате. Метахеуристике проналазе инспирацију у окружењу, природи, физици, математици, па тако постоје метахеуристике креиране по посматрању понашања мравца, пчела, китова, вукова, по анализи гравитационе силе, термодинамике и електромагнетизма, као и по анализи тригонометријских функција [4].

Циљ ове докторске дисертације јесте реализација модела машинског учења за детектовање Паркинсонове болести и других атипичних Паркинсонизама, уз оптимизовање

хиперпараметара модела метахеуристикама. Модели на основу временских серија, прикупљених са сензора прикачених на шаку пацијента, предвиђају присуство Паркинсонове болести или других атипичних Паркинсонових обољења. Додатно, модели предвиђају и присуство Паркинсонове болести анализом временских серија о ходу. У дисертацији је поређено више постојећих метахеуристика, а додатно је представљена и нова, модификована верзија једне метахеуристике.

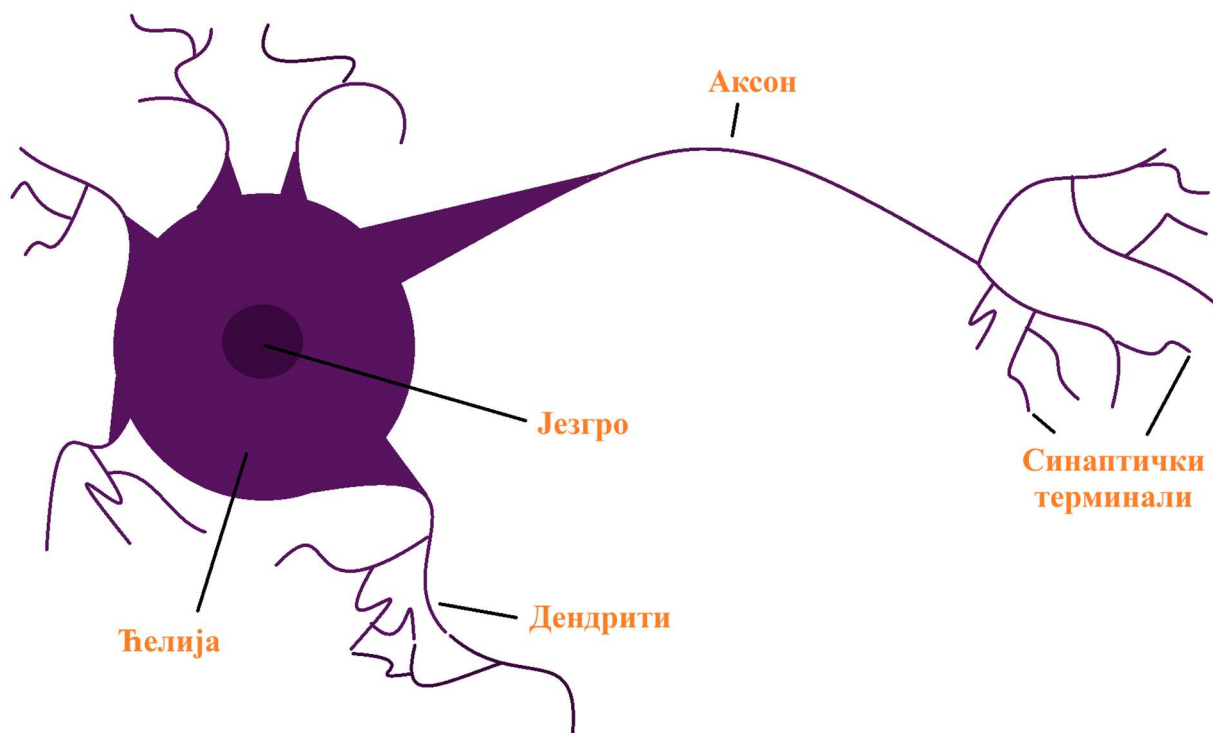
Значај ове докторске дисертације огледа се у томе да се додатно унапреде методе дијагностиковања Паркинсонове болести и атипичних Паркинсонизама, и да се имплементирани модели, који са високом тачношћу потврђују постојање неуродегенеративног поремећаја, користе зарад ефикаснијег откривања болести. Рано детектовање болести је од кључног значаја за што квалитетније резултате лечења пацијената.

Дисертација поред увода који описује предмет, циљ и значај истраживања, има још девет поглавља. У другом поглављу описана су неуродегенеративна обољења, са посебним акцентом на Паркинсову болест и атипична Паркинсонова обољења, који су узроци настанка болести, како се дијагностикују и лече. Треће поглавље говори о машинском учењу, подели машинског учења и различитим алгоритмима машинског учења, и основним принципима функционисања истих. Четврто поглавље прави осврт на преглед скупова података и литературе у домену детектовања Паркинсонове болести и других неуродегенеративних обољења уз коришћење модела машинског учења. У петом поглављу описане су технике оптимизације модела машинског учења, са главним фокусом на анализу рада различитих метахеуристика. Шесто поглавље даје преглед новоимплементиране метахеуристике, и описује коришћене технике за унапређење основне верзије алгоритма. У седмом поглављу даје се преглед коришћених скупова података, извршених експеримената и добијених резултата, као и преглед статистичких валидација. Осмо поглавље говори о потенцијалним будућим унапређењима, док девето поглавље представља закључак дисертације и истиче главне доприносе овог рада.

2. Неуродегенеративна обољења: Паркинсонова болест и атипични Паркинсонизми

Мозак је најзначајнији и најкомплекснији орган у људском телу, и представља главно извориште за покретање свих можданих и телесних функција у нашем организму [5]. Мозак заједно са кичменом мождином чини централни нерви систем. У мозгу се може наћи преко сто билиона ћелија, које сваког човека чине посебном индивидуом, креирајући загонетни сплет знања, сећања и вештина.

У мозгу постоје два типа ћелија: нервне и глијалне. Нервне ћелије, тј. неурони, задужени су за преношење информација, кроз мозак и прикупљање истих, као и за иницирање команди остатку тела. Неурони се састоје од тела неурона и наставка који се протежу из самог тела ћелије. Тело неурона садржи једро (нуклеус), рибозоме и митохондрије. Од наставка из тела неурона препознају се дендрити, који примају сигнале од других ћелија, док аксони прослеђују сигнале ка другим ћелијама. Глијалне ћелије су ћелије које пружају потпору неуронима, омогућавајући им да правилно врше своју улогу. На слици 2.1. може се видети структура неурона.



Слика 2.1. Структура неурона

Синапсе су задужене за преношење сигнала са једног неурона на други. Синаптички терминали, који се налазе на крају аксона, чекају да дође сигнал који је потребно процесуирати, а затим га и прослеђују преко дендрита примајуће ћелије. Наравно и у овом процесу кључна је подршка глијалних ћелија. Унутар синаптичких терминала, синаптичке везикуле пуне су неуротрансмитера неопходних за нормално функционисање процеса преноса сигнала.

Неуротрансмитери откривени су 1921. године од стране Ото Левија, који је за своје откриће добио и Нобелову награду, а први неуротрансмитер који је откривен је ацетилхолин [6]. Због своје централне улоге коју имају у обезбеђивању правилне функције мозга, неуротрансмитерски рецептори и протеини задужени на синтезу неуротрансмитера су у жижи

истраживања научника дуги низ година. Неки од најпознатијих неуротрансмитера су: допамин, серотонин, хистамин, глутамат, адреналин, норадреналин, окситоцин, галанин, аденозин, габа, аспартат и многи други.

Допамин је као неуротрансмитер откривен 1957. године од стране Арвида Карлсона, и од тада се његова улога све интензивније проучава [7]. Задужен је за кретање, спознају, позитиван подстрек, емоције, унос хране, и друге функције [8]. Овај кетахоламин поред наведених функција, игра улогу и у кардиоваскуларном систему, секрецији хормона, као и гастроинтестиналном тракту. Допамин је актер у скоро свим есенцијалним људским активностима, и због тога могу настати разни поремећаји уколико дође до сметњи у његовом синтетисању.

Последице нерегуларности у нивоима допамина могу бити разнолике, и проблеме могу изазвати и прениске дозе допамина, као и превисоке.

Симптоми високог нивоа допамина су:

- Анксиозност,
- Инсомнија,
- Агресивност,
- Халуцинације.

Симптоми ниског нивоа допамина су:

- Недостатак енергије,
- Слабија координација,
- Потешкоће са говором, ходом, уносом хране,
- Грчеви у мишићима и спазми,
- Тремори,
- Промене расположења,
- ... итд.

Било које промене које утичу на мозак, могу изазвати шаренолике последице по људски организам. Под неуродегенеративним обољењима сматрају се све болести које су узроковане одумирањем ћелија у централном нервном систему, и њихов утицај на онеспособљавање појединца може бити огроман. Најзаступљеније неуродегенеративно обољење на свету је Алцхајмерова болест, која погађа око седам милиона становника Земље [9]. Главни симптом Алцхајмерове болести је деменција, која мора бити праћена бар још једном угроженом когнитивном функцијом. Болест је веома прогресивна, остављајући пацијенте онемогућене да воде рачуна о себи.

Друго по реду заступљености неуродегенеративно обољење, после Алцхајмерове болести, је Паркинсонова болест, која утиче на око шест милиона становника. Главни симптоми Паркинсонове болести су моторни симптоми попут тремора, ригидности мишића и нестабилности при ходу, а уколико се појаве и други симптоми попут деменције или атаксије може се указивати на присуство атипичних Паркинсонизама. У наставку поглавља Паркинсонова болест и атипични Паркинсонизми биће детаљно објашњени.

2.1. Паркинсонова болест

Паркинсонова болест (ПБ) је неуродегенеративно обољење, а као главни узрок болести сматра се одумирање ћелија задужених за лучење допамина у црној супстанци и појављивање Левијевих тела [10]. Међутим поред овог фактора, сматра се да постоји још много околности које могу поспешити појаву болести [11].

Старење свакако јесте фактор у настајању Паркинсонове болести, и примећено је генерално одумирање допаминергичких ћелија у старијим добима, а болест је детектована само у 5% случајева за особе млађе од 40 година. Додуше старост сама по себи се ипак не узима као конкретан узрочник болести.

Истраживања последњих година, показују да постоје и генетски фактори који утичу на настанак болести, и детектовано је шест доминантних гена који томе доприносе: SNCA, LRRK2, VPS35, EIF4G1, DNAJC13, и CHCHD2. Додатни гени повезани са Паркинсонизмима су: ATP13A2, C9ORF72, FBXO7, PLA2G6, POLG1, SCA2, SCA3, SYNJ1, RAB39B, и други.

Окружујући фактори се такође морају узети у обзир [12]. Доказани су утицаји једињења попут 1-Methyl-4-Phenyl-1,2,3,6-Tetrahydropyridine (МППП) на настанак болести, а истражује се још о утицају ротенона и параквата. Додатно се анализирају фактори попут понављаних повреда главе, које се повезују са Паркинсоновом болешћу.

Симптоми Паркинсонове болести могу се поделити на моторне и немоторне. Увиђа се да немоторни симптоми могу настати годинама пре дијагностиковања саме болести, а да у каснијим фазама обољења доминирају моторни симптоми.

Моторни симптоми су:

- Брадикинезија – успореност покрета, један је од скоро неопходних присутних симптома за дијагностиковање болести. Пацијенти могу да примете заостајање руке при ходу, изостанак фацијалних експресија, смањена слова приликом писања, спорост при спровођењу теста тапкања прстима или теста отварања/затварања шаке,
- Ригидност – укоченост мишића,
- Тремор – невољно дрхтање целог тела или појединачних екстремитета. Око 70% пацијената са дијагностикованом Паркинсоновом болешћу ће имати овај симптом, док 30% не,
- Нестабилност у ходу – доста присутан симптом код ког ће пацијенти имати суженији корак и згрбљено држање, уз нестабилност могу се јавити и падови који могу и не морају да указују на неки атипични Паркинсонизам.

Немоторни симптоми су:

- Депресија, апатија – неуропсихијатријски симптоми, а пацијенти у каснијим фазама болести најчешће буду дијагностиковани деменцијом,
- Гастроентеролошки проблеми – констипација,
- Проблеми са спавањем – губитак нормалног мишићног тонуса током РЕМ фазе сна, као и поремећај периодичких покрета удова,
- Аносмија – губитак чула мириса, јавља се код преко 70% пацијената,
- Деменција – преко 80% пацијената ће имати барем неку когнитивну сметњу, а деменција се мало чешће јавља код мушкараца. Такође може указивати и на пацијенте са атипичним Паркинсонизмом.

Паркинсонова болест је прогресивна, и годинама квалитет живота пацијента опада, али се не може предвидети када ће особа умрети, и да ли ће ова болест бити главни узрок тога. Примећено је да особе са израженијим симптомима тремора, имају боље коначне прогнозе лекара.

Стандардни неуропатолошки критеријум за постављање дијагнозе болести још увек не постоји. Утврђивање смањеног броја неурона у црној супстанци и присуство Левијевих тела у преосталим неуронима, уз искључивање постојања других обољења је тренутно решење. У ове сврхе користе се магнетна резонанца и електроенцефалографија. Клиничка дијагноза Паркинсонове болести поставља се са присуством брадикинезије, уз који се мора јавити или тремор или ригидност. Такође потребно је да не постоје симптоми који могу указивати на постојање других атипичних Паркинсонизама. Лекар може посумњати на Паркинсонову болест и пре видљивих моторних симптома, али се тада пацијенту саветује праћење и промена животних навика које могу водити здравијем животу (редовна физичка активност и регулисана исхрана).

Конкретан лек за успоравање и нестанак болести не постоји, већ се примењују лекови за ублажавање симптома. Најзаступљенији су лекови који су задужени за повишење нивоа допамина, а један такав лек је Леводопа. Пацијенти га поприлично добро подносе, а често се даје и у комбинацији са допаминским агонистима, који стимулишу допаминске рецепторе. Антихолинергици се могу користити за ублажавање симптома тремора, мада је показано да могу да погоршају друге симптоме попут конфузије. Понекад се примењује и стимулација неурона електродама, а још доста третмана је у фази испитивања.

2.2. Атипични Паркинсонизми

Када се детектује присуство Паркинсонове болести, по претходно описаним правилима, поставља се питање да ли се ради о стандардној Паркинсоновој болести или о атипичним Паркинсонизмима [13]. Атипични Паркинсонизми се узимају у разматрање уколико се поред основних моторних Паркинсонових симптома, јаве и деменција, атаксија, дисаутономија и друге сметње [14]. У атипичне Паркинсонизме спадају најчешће прогресивна супрануклеарна парализа, мултипла системска атрофија, кортикобазална дегенерација и деменција са Левијевим телима. Веома је важно приметити да се ради о неком од ових обољења и не третирати их као стандардну Паркинсонову болест, јер се стање пацијента може погоршати. Такође, ни за ова обољења не постоје биолошки маркери, већ се дијагностика углавном своди на анализу клиничких симптома.

У наставку су детаљније описане прогресивна супрануклеарна парализа и мултипла системска атрофија, које ће поред Паркинсонове болести бити предвиђане моделима машинског учења у овој дисертацији.

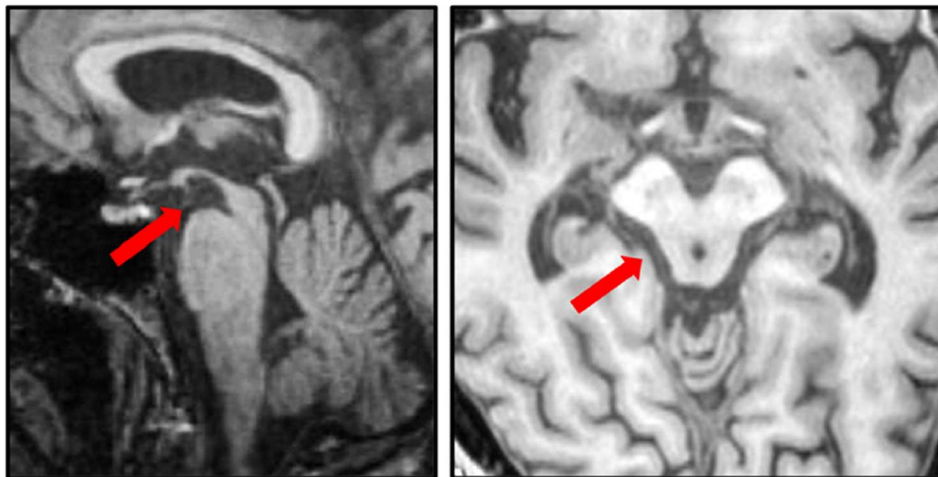
2.2.1. Прогресивна супрануклеарна парализа

Прогресивна супрануклеарна парализа (ПСП) најчешће показује додатне симптоме попут супрануклеарне парализе вертикалног погледа коју додатно прате и абнормалности у хоризонталном погледу, падови уназад, супкортикална и фронтотемпорална деменција, депресија, апатија и слично. У вертикалном погледу најпре се јављају проблеми са погледом надоле, такође је смањен број трептаја. У хоризонталном погледу се јављају проблеми при дужем гледању у предмет. Чести падови се умеју јавити као један од првих симптома, годинама пре дијагнозе болести.

Болест се подједнако јавља и код мушкарца и жена, а просечан број година када крену да се јављају симптоми је 63. Свега око 5% пацијената са симптомима Паркинсонове болести имају заправо ПСП, али је могуће да би овај проценат био већи уз правилнију дијагностику.

Лечењу се углавном приступа мултидисциплинарно, са физикалном и когнитивном терапијом, променом исхране. Може се покушати и месец дана примене Леводопа лека, уз праћење реакције пацијента, док допамински агонисти нису показали довољно успеха. За побољшање расположења примећено је да коензим Q10 даје добре резултате, а у озбиљнијим случајевима антидепресиви.

Магнетна резонанца (МР) може указати на постојање ове болести, што се може видети на слици 2.2. На левом делу слике је приказан сагитални Т1 тежински МР снимак, на ком се уочава карактеристични знак колибирија у средњем мозгу, док се на десном делу слике види аксонални Т1 тежински МР снимак на ком се уочава још један карактеристични знак за ПСП, а то је знак цвета петуније, у средњем мозгу.



Слика 2.2. МР снимци средњег мозга за особе са ПСП [15]

2.2.2. Мултипла системска атрофија

Мултипла системска атрофија (МСА) може се класификовати у два типа. Први тип са доминантним Паркинсоном, а други тип са доминантом церебралном атаксијом. Први симптом који треба да укаже на МСА је аутономна неуропатија. Додатно се врло често јављају поремећаји током РЕМ фазе сна који се ретки за Паркинсонову болест и ПСП. Увиђа се и Рејноов феномен, дистонија, а могу се јавити и чести падови као што је то случај код ПСП.

Болест се подједнако јавља и код мушкара и жена, а просечан број година када крену да се јављају симптоми је од 53 до 55. Случајеви када се болест јавила пре 30-те године не постоје. Болест уме да напредује брже од Паркинсонове болести, па је након дијагностиковања просечна дужина живота од шест до девет година.

Код лечења, од 30% до 60% пацијената добро одреагује на терапију допамином. Треба се исто као код ПСП покушати са Леводопа леком. Стимулација електродама није наишла на успех. Поред уобичајеног приступа преласка на здравији начин живота, остали пропратни симптоми лече се независно.

3. Машинско учење и алгоритми машинског учења

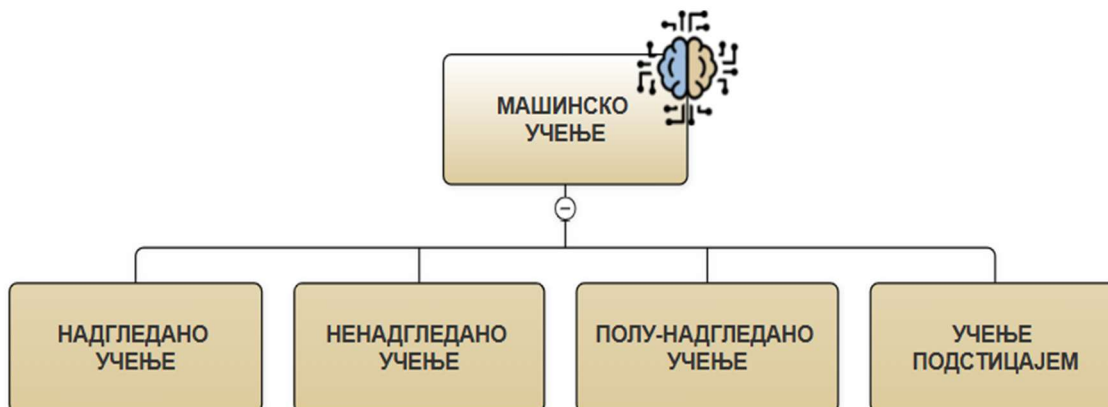
Човек је по природи радознало биће, које је кроз историју дошло до невероватних изума попут ватре, пеницилина, струје, телефона, аутомобила и авиона [16]. Велики број изума је настао како би се човеку олакшао рад. Један такав изум је и рачунар, без ког вероватно не можемо ни да замислимо данашњицу. Рачунари могу да извршавају велики број инструкција у секунди, а инструкције се могу груписати у алгоритме за извршавање неких специфичних задатака. Алгоритми раде са подацима, а када постоји велики број података, људском оку могу промаћи нека увиђања која би на мањем скупу података била уочљива.

Машинско учење постоји управо како би се уз помоћ рачунара анализирао велики број података, и како би се извукли закључци који би можда изостали са стандардним методама евалуације. Данас машинско учење се примењује код развоја софтвера, препознавања говора, управљања роботима [17]. Примењује се и код конзументистичких сервиса за анализу тржишта, за детектовање кварова у сложеним системима, а може да се користи и да нам брже и ефикасније да одговоре на нека питања (*ChatGPT*), за која би нам можда требало и више сати да их пронађемо.

Како је машинско учење постало део наше свакодневице, важно је да разумемо како машинско учење функционише, и да га искористимо на начин који највише доприноси друштву.

Машинско учење можемо поделити у неколико група (слика 3.1.):

- Надгледано – подаци су означени, тј. за улаз x , постоји дефинисан излаз y , а циљ модела машинског учења у овом случају је да за неко ново x^* , пронађе y^* , увиђајући зависност (функцију) између улазних података. Надгледано машинско учење решава проблеме попут класификације (y може да припада двама или више класа) и регресије (y је нумеричког типа),
- Ненадгледано – подаци су неозначени, тј. за улаз x , не постоји дефинисан излаз y , већ се циљ модела машинског учења, да увиди структуру и зависности међу неозначеним подацима. У овом случају се врши кластеризација података, асоцијација и редукција димензија,
- Полу-надгледано – када постоји комбинација означених и неозначених података, може се користити полу-надгледано учење, које представља комбинацију нагледаног и ненадгледаног учења,
- Учење подстицајем – агенти у окружењу функционишу по принципу награде и казне, у случају да се приближавају добром решењу или удаљавају од њега.



Слика 3.1. Подела машинског учења

У овој дисертацији се решава проблем класификације, чији је главни задатак да улазни податак који се може састојати од једне или више одлика, класификује по припадности одређеној класи. У случају надгледаног учења модел добија већ тачне класе за одређени скуп улазних података, и на основу тога увиђа зависности које може да примени на нове улазне податке који нису виђени у процесу тренирања модела. У наставку ће бити описани алгоритми машинског учења који се користе за решавање овог типа задатака.

3.1. Логистичка регресија

Постоји више типова регресије, од којих је најпознатија линеарна регресија [18]. Линеарна регресија анализира излазе који су нумеричке вредности, и покушава да нађе зависност између излаза и улазних података, тако да ту може да се примени линеарна функција. Основна једначина линеарне регресије је:

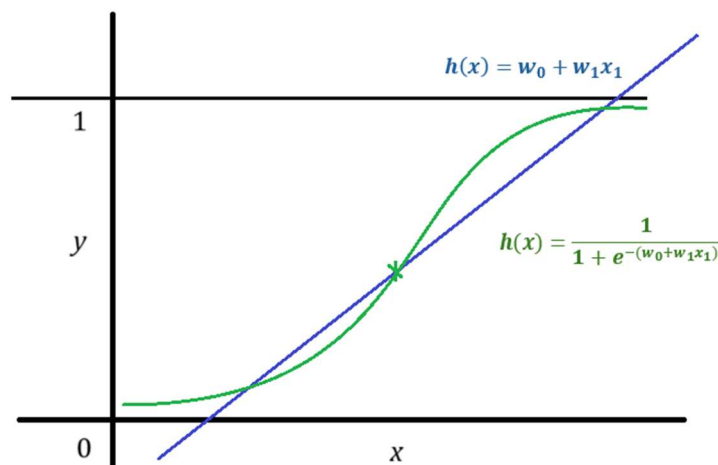
$$h(x) = w_0 + w_1x_1 + \dots + w_nx_n$$

где су $h(x)$ вредност која се предвиђа, w_0 и w_1 параметри модела, а n број одлика које се користе у моделу.

Овај модел се не може применити у свим ситуацијама, на пример када треба да се предвиди бинарни догађај. Тада се може применити логистичка регресија, која уместо линеарне функције покушава да пронађе најбољу логистичку функцију на улазним и излазним подацима. Основна хипотеза логистичке регресије (енг. *Logistic regression (LR)*) је:

$$h(x) = \frac{1}{1 + e^{-(w_0 + w_1x_1 + \dots + w_nx_n)}}$$

Логистичка функција претвара интервал од $(-\infty, +\infty)$ на $[0, 1]$ као што се може видети на слици 3.2. За $h(x) > 0.5$ предвиђа се вредност 1, а у супротном 0.



Слика 3.2. Приказ линеарне и логистичке функције

Приликом учења модела, потребно је пронаћи такве параметре да функција грешке буде минимална. Функције грешке у случају логистичке регресије представља се формулом:

$$J(w) = -\frac{1}{m} \sum_{i=1}^m y^{(i)} \ln(h(x^{(i)})) + (1 - y^{(i)}) \ln(1 - h(x^{(i)}))$$

где m представља број примера у скупу података, а $y^{(i)}$ исправну излазну вредност за пример i . Минимална вредност функције грешке проналази се градијентним спустом, где се

итеративним путем рачунају градијенти (парцијални изводи), а параметри модела се ажурирају вредностима:

$$w_i = w_i - \alpha \frac{\partial J(w)}{\partial w_i}$$

где је α корак учења, који одређује колики скокови у параметрима ће се правити, и мора се одабрати тако да се градијенти спуст не заглави у локалном минимуму, нити да прескочи глобални минимум. Спуст се прекида када вредности параметара престану да се мењају, што значи да је конвергирано ка минимуму функције. Извод функције у минимуму је једнак 0, али ово не може бити једини параметар за заустављање.

Логистичка регресија, поред бинарне класификације, може решавати и мултикласну класификацију, користећи мултиномијалну логистичку регресију, или комбиновањем више бинарних класификатора по принципу: један-против-једног или један-против-свих.

3.2. Наивни Бајесов класификатор

Наивни Бајесов (енг. *Naïve Bayes (NB)*) класификатор заснован је на Бајесовој теореме која рачуна вероватноћу по формули:

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

где је y класа припадности, x податак, $P(y|x)$ апостериорна вероватноћа класе, $P(y)$ априорна вероватноћа класе, $P(x|y)$ функција извесности [19].

$$P(y) = \frac{\text{укупан број појављивања класе } y}{\text{укупан број примера у скупу података}}$$

$$P(x|y) = P(x_1, x_2, \dots, x_n|y)$$

Ако узмемо претпоставку да су одлике $x_1, x_2, \dots, x_n$ међусобно независне, функцију извесности можемо рачунати на начин приказан на формули испод. Због ове хипотезе је класификатор и добио назив „наивни“.

$$P(x|y) = P(x_1|y)P(x_2|y) \dots P(x_n|y)$$

$$P(x|y) = \prod_{i=1}^n P(x_i|y)$$

Уколико би се десила ситуација да се нека одлика x_i никада не јави када се на излазу класа y , онда ће $P(x|y)$ постати 0, само због те чињенице, и због тога је потребно избећи ову ситуацију, што се постиже Лапласовим поравнањем. Лапласово поравнање омогућава да сматрамо да се одређена одлика јавила сигурно једном у посматраној класи.

Сам класификатор може решавати и бинарну и мултикласну класификацију, а бира се она класа која да највећу вероватноћу.

Постоје три врсте наивног Бајесовог класификатора:

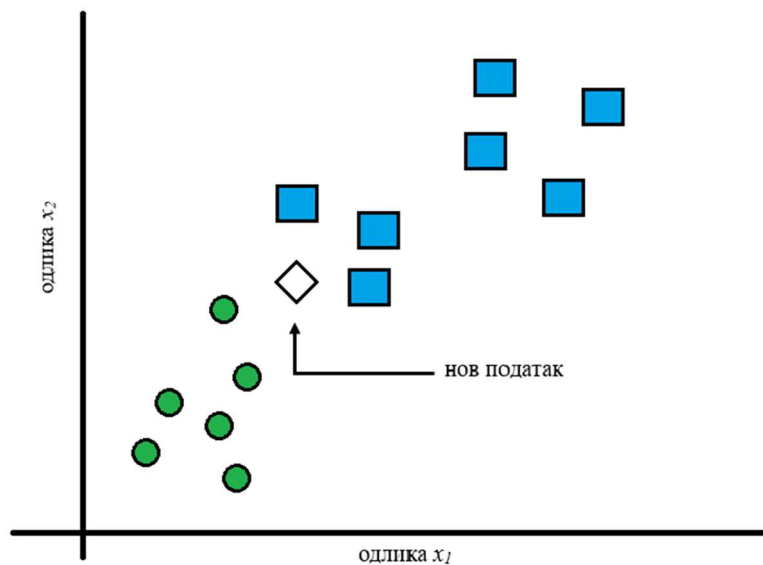
- Гаусов наивни Бајес – одлике прате Гаусову расподелу,
- Мултиномијални наивни Бајес – одлике су фреквенције појављивања,
- Бернулијев наивни Бајес – одлике су бинарне вредности, 0 или 1.

3.3. К-најближих суседа

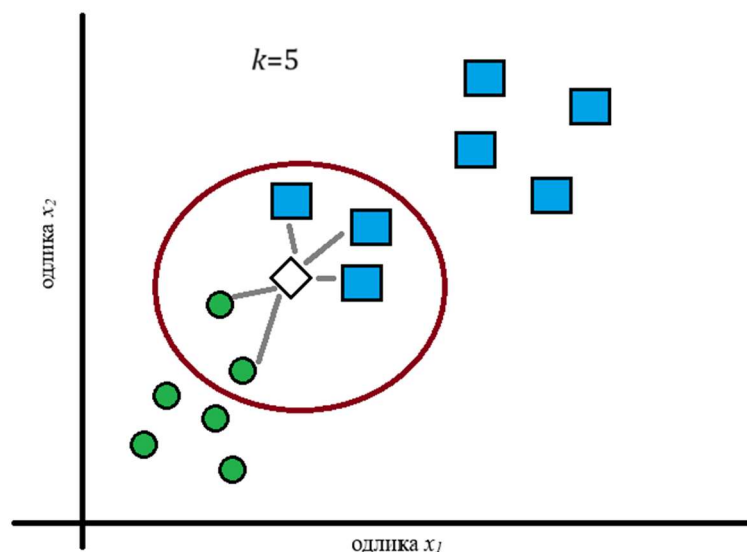
К-најближих суседа (енг. *K-nearest neighbours (KNN)*) је веома једноставан модел класификације, који тражи k најближих суседа одређеног податка, а затим на основу већинске класе суседа, одређује и финалну класу унетог податка [20].

Ток алгоритма је следећи:

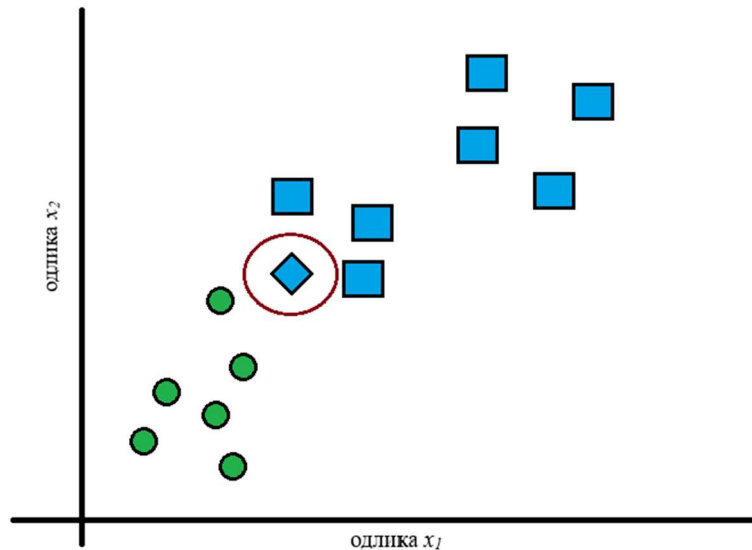
1. Учитавање података (слика 3.3.)
2. Избор почетне вредности параметра k
3. Пролазак кроз све податке у скупу података и рачунање раздаљина сваке тачке у односу на нов податак
4. Сортирање раздаљина и избор k најмањих дистанци (слика 3.4.)
5. Пребројавање класа суседа и избор финалне класе за нов податак (слика 3.5.)



Слика 3.3. К-најближих суседа: почетно стање алгоритма



Слика 3.4. К-најближих суседа: налажење суседа



Слика 3.5. K-најближих суседа: класификовање новог податка

Постоји више начина за рачунање k најближих суседа. Метрике за дистанцу [21] су:

- Еуклидска дистанца

$$\sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

- Менхетен дистанца

$$\sum_{i=1}^n |p_i - q_i|$$

- Чебишева дистанца

$$\max_i (|p_i - q_i|)$$

што је једнако:

$$\lim_{k \rightarrow \infty} \sqrt[k]{\sum_{i=1}^n |p_i - q_i|^k}$$

- Минковски дистанца

$$\sqrt[p]{\sum_{i=1}^n |p_i - q_i|^p}$$

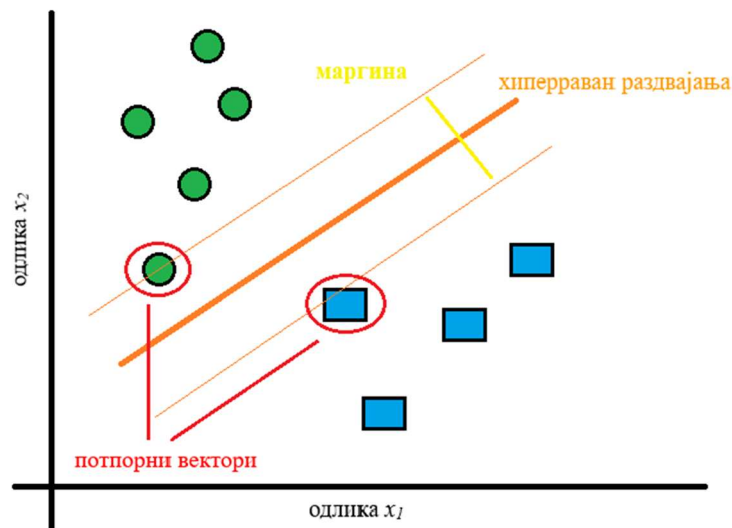
за $p = 1$, Минковски дистанца је Менхетен дистанца,
за $p = 2$, Минковски дистанца је Еуклидска дистанца,
за $p = \infty$, Минковски дистанца је Чебишева дистанца

- и друге...

Још један значајан фактор у овом моделу, је како изабрати најбоље k . Параметар k не сме бити ни превише велики, јер постоји могућност од превелике симплификације модела, а не сме бити ни превише мали, јер онда модел може бити преосетљив на одступања. Почетна вредност параметра углавном се поставља на $\sqrt{n}$, или $\frac{\sqrt{n}}{2}$, где је n број свих примера у скупу података, уз услов да k буде непаран број, како не би био нерешен резултат приликом пребројавања класа суседа. Параметар k може се подешавати и унакрсном валидацијом.

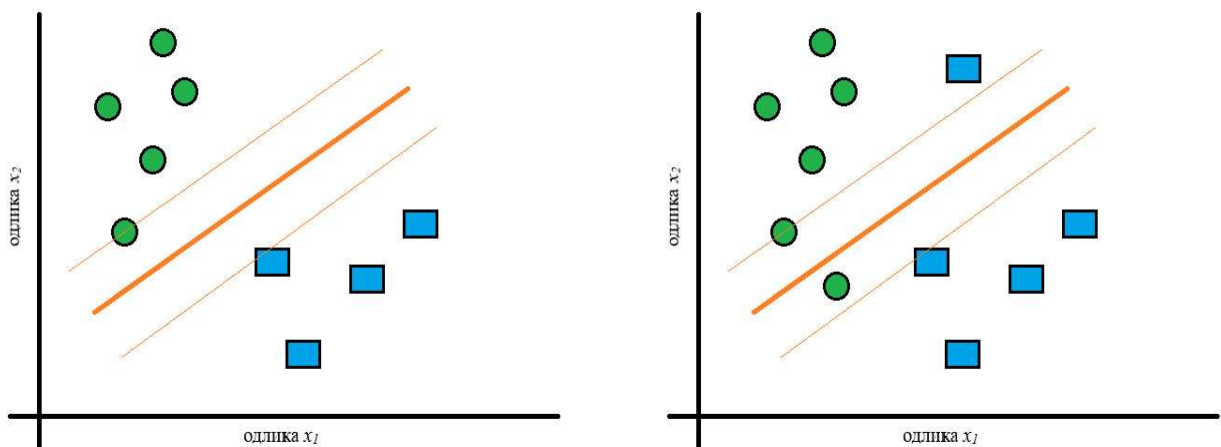
3.4. Метода потпорних вектора

Метода потпорних вектора (енг. *Support vector machines (SVM)*) решава проблем класификације тако што проналази најоптималнију хиперраван која раздваја класе, коју проналази на основу улазних података [22]. Најоптималнија хиперраван је она која проналази највећу маргину, тј. прави највећу дистанцу између потпорних вектора. Потпорни вектори су тачке представнице своје класе, које су најближе хиперравни. Приказ методе потпорних вектора, и поменутих термина, може се видети на слици 3.6.



Слика 3.6. Метода потпорних вектора: хиперраван раздвајања

Разликују се два типа маргине: мека и тврда, што се може видети на слици 3.7. Тврда маргина не дозвољава лоше калкулације, и у потпуности раздваја све податке једне и друге класе. Овакви прорачуни могу бити веома скупи, па се због тога преферира мека маргина која дозвољава одређени степен грешке, који се може подешавати.



Слика 3.7. Метода потпорних вектора: тврда (лево) и мека (десно) маргина

Уколико подаци у скупу података нису линеарно раздвојиви, метода потпорних вектора може применити различита језгра која трансформишу податке у вишедимензионални простор, на математички најједноставнији и ефикаснији начин. Језгра која постоје су [23]:

- Линеарно језгро

$$K(x, x_i) = xx_i$$

- Полиномијално језгро

$$K(x, x_i) = (c + xx_i)^d$$

- Гаусово радијално језгро

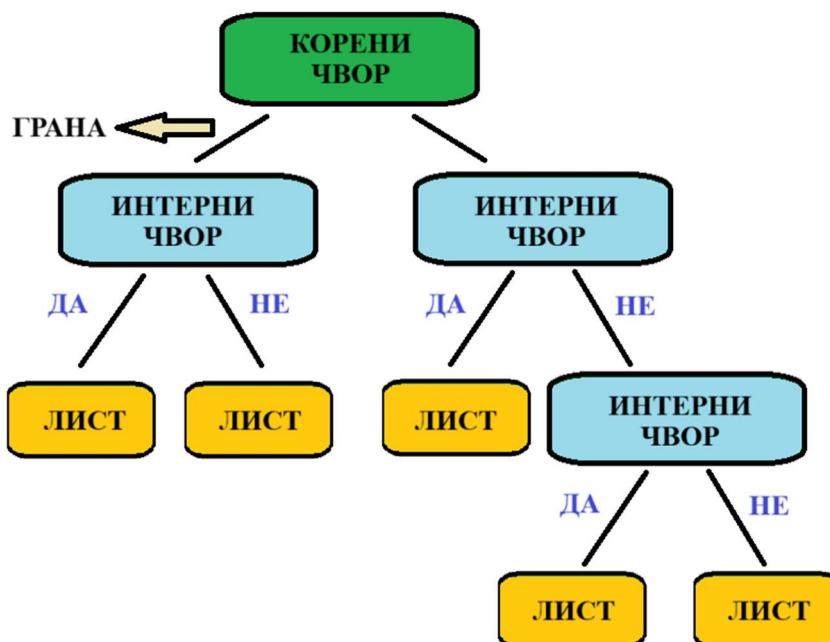
$$K(x, x_i) = e^{-\gamma \|x - x_i\|^2}$$

- Сигмоидно језгро

$$K(x, x_i) = \tanh(\gamma x^T x_i + r)$$

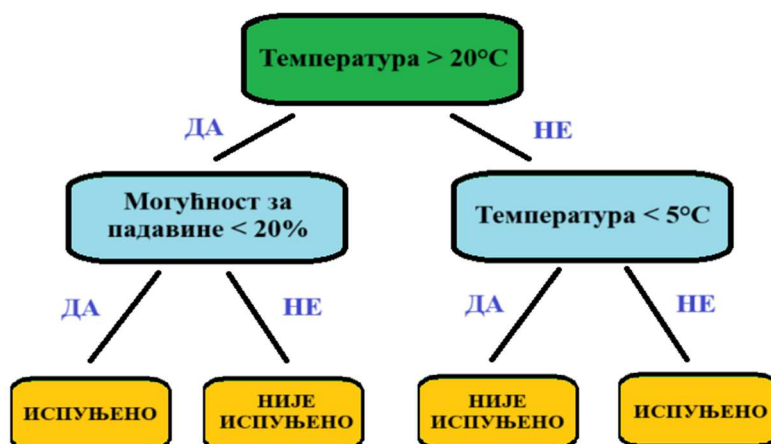
3.5. Стабла одлучивања

Стабло одлучивања (енг. *Decision tree (DT)*) састоји се од корена стабла, грана стабла, интерних чворова стабла и од листова [24]. Класификација се започиње од корена, и у зависности од испуњености услова у кореном чвору наставља се одговарајућом граном ка интерним чворовима, у којима се поново проверавају услови и прате потребне гране, све док се не дође до листа стабла које и представља финалну класу. Пример стабла одлучивања може се видети на слици 3.8.



Слика 3.8. Стабло одлучивања

Ако би се стабло одлучивања применило на конкретном примеру предвиђања испуњености услова за физичке активности на отвореном, за улазни податак (*температура: 28°C, могућност за падавине: 5%*) класификација би се од кореног чвора кретала граном **ДА**, преко интерног чвора у ком би се пут наставио граном **ДА**, и стало би се у чвору у ком се за улазни податак додељује класа **ИСПУЊЕНИ УСЛОВИ ЗА ФИЗИЧКУ АКТИВНОСТ НА ОТВОРЕНОМ**. Илустрација ове класификације може се видети на слици 3.9.



Слика 3.9. Илустрација класификације стабла одлучивања

Веома је важно одредити који услови (атрибути) ће се испитивати у кореном и интерним чворовима, јер управо овај избор одређује успешност предвиђања самог стабла. Могуће је изабрати насумични атрибут, или атрибут који се најмање или највише пута појављује, или бирати атрибут по резултатима одређених метрика. Метрике за бирање атрибута у стаблима одлучивања су:

- Ентропија

$$Entropy(D) = \sum_{i=1}^c -p_i \log_2 p_i$$

где је c број класа, а D скуп примера,

- Добит информација

$$Gain(D, F) = Entropy(D) - \sum_{v \in F} \frac{|D_v|}{|D|} Entropy(D_v)$$

где је F атрибут за који се одређује метрика,

- Гини индекс

$$Gini(D) = 1 - \sum_{i=1}^c p_i^2$$

$$Gini(D, F) = \sum_{v \in F} \frac{|D_v|}{|D|} Gini(D_v)$$

Стабла одлучивања су веома осетљива на било какве промене у скупу података, врло често због тога што се превише добро прилагоде почетном скупу (претренирање), и због тога је важно да се покуша што боља генерализација и смањење комплексности стабала. На комплексност стабла утичу укупан број чворова у стаблу, укупан број листова, максимална дубина стабла и број атрибута који могу да се користе за гранање, и када се достигну максималне вредности за ове параметре престаје се са грађњом стабла. Максималне вредности ових параметара могу се утврдити унакрсном валидацијом.

На само стабло одлучивања не утичу само поменути параметри већ и сами подаци који се уносе у стабло. Подаци се могу и претпроцесирати, и стабло се може градити само над одређеним подскупом података и атрибута, чиме се може смањити комплексност.

Смањење сложености самог стабла може се радити не само пре градње стабла (претпроцесирањем података) и током градње, већ и након што се стабло изгенерисано, техником резивања стабла. Одређене гране и чворови стабла могу бити орезани (одбачени) уколико не испуњавају потребне услове.

3.6. Ансамбл учење

Ансамбл учење је техника у машинском учењу, у којој се предвиђање не ослања само на једног предиктора, већ се упошљава више њих, а онда се узима она предикција коју је највише предиктора изгласало [25]. Овакав вид учења људи примењују већ дуго у свом свакодневном окружењу: гласање на изборима, порота у суду, конзилијум при доношењу медицинских дијагноза или препоруке при запошљавању појединца. У машинском учењу оваквом техником учења добија се веће поверење у сам резултат, решава се проблем избора одлика, небалансираних података и други. У наставку поглавља су описане најчешће коришћене методе ансамбл учења.

3.6.1. Насумичне шуме

Мане стабала одлучивања су да су врло склона претренирању, а у случајевима када се примене технике генерализације стабла, онда не могу да произведу веома високу прецизност. Овакав проблем се може превазићи ансамбл учењем, где се упошљава више стабала одлучивања која креирају шуму стабала (енг. *Random forest (RF)*), а затим се бира предикција коју је највише стабала дало [26].

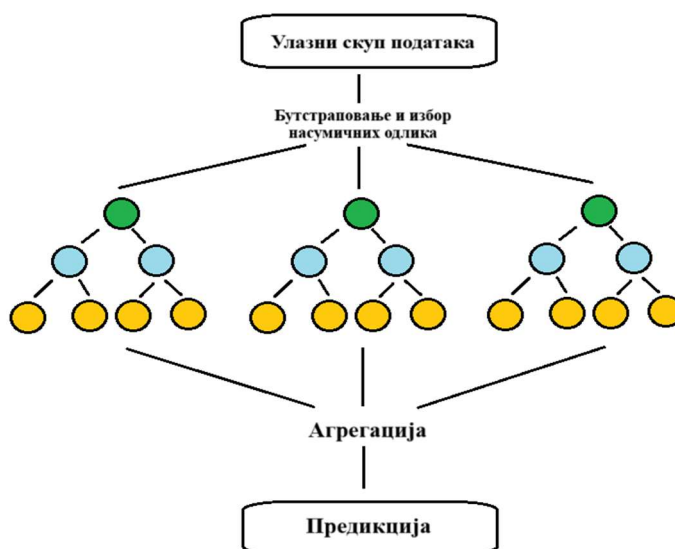
Ток алгоритма је следећи:

1. Креира се више стабала одлучивања користећи насумичан подскуп података из оригиналног скупа података. Овакво креирање подскупова података се назива **бутстраповање**
2. Током креирања стабала над новим скуповима података, за свако стабло се узима насумични скуп одлика, а не све одлике
3. Након креирања стабала, свако стабло даје своју предикцију, а у случају класификације као финална предикција врши се **агрегација** свих предиктора и узима се она класа коју је највише стабала дало

Техника комбиновања бутстрапована и агрегације се назива енгл. *bagging*.

Насумичне шуме креирају стабла која су међусобно потпуно независна и не уче једна од других, а свако стабло има свој подскуп података над којим ради, као и свој подскуп одлика које се узимају у обзир приликом креирања стабла. Овај приступ спречава претренирање модела, и побољшава прецизност предикције што и јесте једна од главних предности ансамбл учења.

Ток алгоритма са бутстраповањем и агрегацијом може се видети на слици 3.10.



Слика 3.10. Насумичне шуме

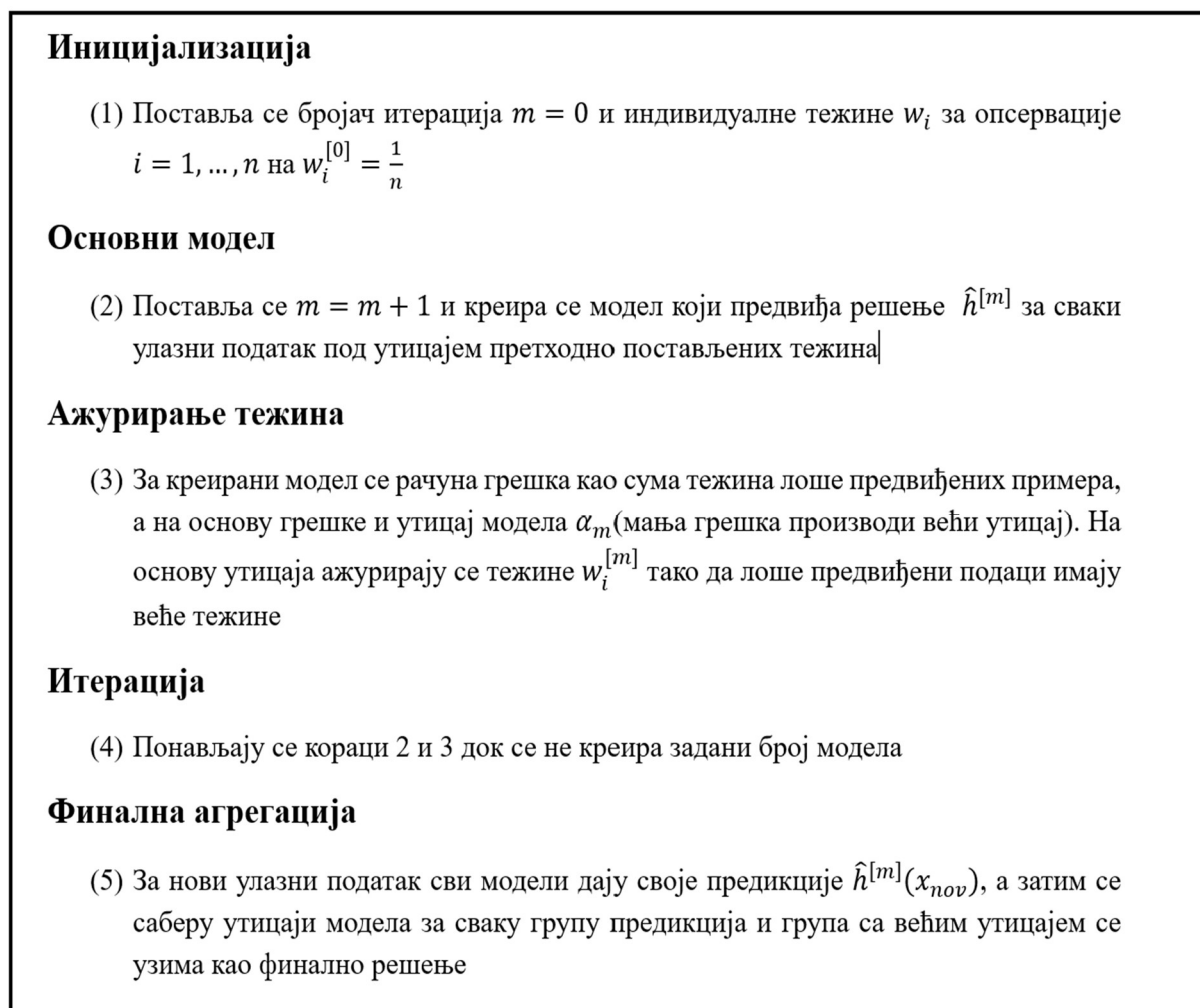
3.6.2. Ада појачивач

Још једна врста ансамбл учења је техника појачивача (енг. *boosting*). Појачивачи исто креирају више предиктора, али у овом случају предиктори су зависни једни од других и уче на претходним грешкама [27].

Први конкретан алгоритам из ове групе је Ада појачивач (енг. *Ada boosting (AB)*), који итеративно прилагођава своје параметре на основу тренутних перформанси.

Алгоритам функционише тако што се најпре сви подаци из скупа података означе тежинским факторима (ω), а затим се креира прво стабло и његова грешка (ϵ), као сума свих тежинских фактора лоше предвиђених примера. Подразумевано стабло је дубине један. На основу грешке рачуна се утицај стабла по формули $\alpha = \eta \log \frac{1-\epsilon}{\epsilon}$, где је η параметар учења. На основу утицаја стабла се прилагођавају (поново израчунавају) тежински параметри. Што је мања грешка већи ће бити утицај. Већи утицаји дају веће тежине за лоше предвиђене примере, како би се следећа стабла више фокусирали на исправљање грешака. На крају се финална предикција врши тако што се стабла поделе по класама које предвиђају, а затим се за сваку групу класа саберу утицаји стабала, и група која има највећи утицај даје финалну предикцију. Алгоритам се завршава када се креира задан број стабала.

Ток алгоритма може се видети на слици 3.11.



Слика 3.11. Ада појачивач

3.6.3. Градијентни појачивач

Градијентни појачивач (енг. *Gradient boosting (GB)*) је алгоритам из групе појачивача, који итеративно гради стабла око одступања правих од предвиђених вредности, а одступања прилагођава градијентним спустом [28].

Алгоритам функционише тако што се направи иницијални погодак који би модел предвиђао, а затим се за свако стабло којих има M , најпре рачунају одступања правих од предвиђене вредности за сваки од N података. Око одступања се прави стабло одлучивања, а листови стабла се означавају са R . За сваки лист се градијентним спустом рачуна вредност γ , која ће дати најмању вредност излаза, јер је циљ да се смањи одступање од праве вредности. Ново предвиђање модела у том тренутку се рачуна као претходно предвиђање на које се додаје излаз стабла одступања за конкретан податак. Алгоритам се завршава када се креира задати број стабала.

Функција губитка у случају класификације се дефинише формулом:

$$L(y_i, F(x)) = -(y_i \log p_i + (1 - y_i) \log(1 - p_i))$$

$$L(y_i, F(x)) = -y_i \log(odds) - \log(1 - p_i)$$

$$\log(odds) = \log \frac{p_i}{1 - p_i}$$

Ток алгоритма може се видети на слици 3.12.

1. Иницијализација $f_0(x) = \arg \min_{\gamma} \sum_{i=1}^N L(y_i, \gamma)$
2. За $m = 1$ до M :
 - (1) За $i = 1, \dots, N$ рачуна се
 - а. $r_{im} = - \left[\frac{\partial L(y_i, f(x))}{\partial f(x)} \right]_{f=f_{m-1}}$
 - (2) Креира се класификационо стабло око r_{im} креирајући регионе око листова који се означавају $R_{jm}, j = 1, 2, \dots, J_m$
 - (3) За $j = 1, \dots, J_m$ рачуна се
 - а. $\gamma_{jm} = \arg \min_{\gamma} \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma)$
 - (4) $f_m(x) = f_{m-1}(x_i) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \in R_{jm})$
3. Излаз $\hat{f}(x) = f_M(x)$

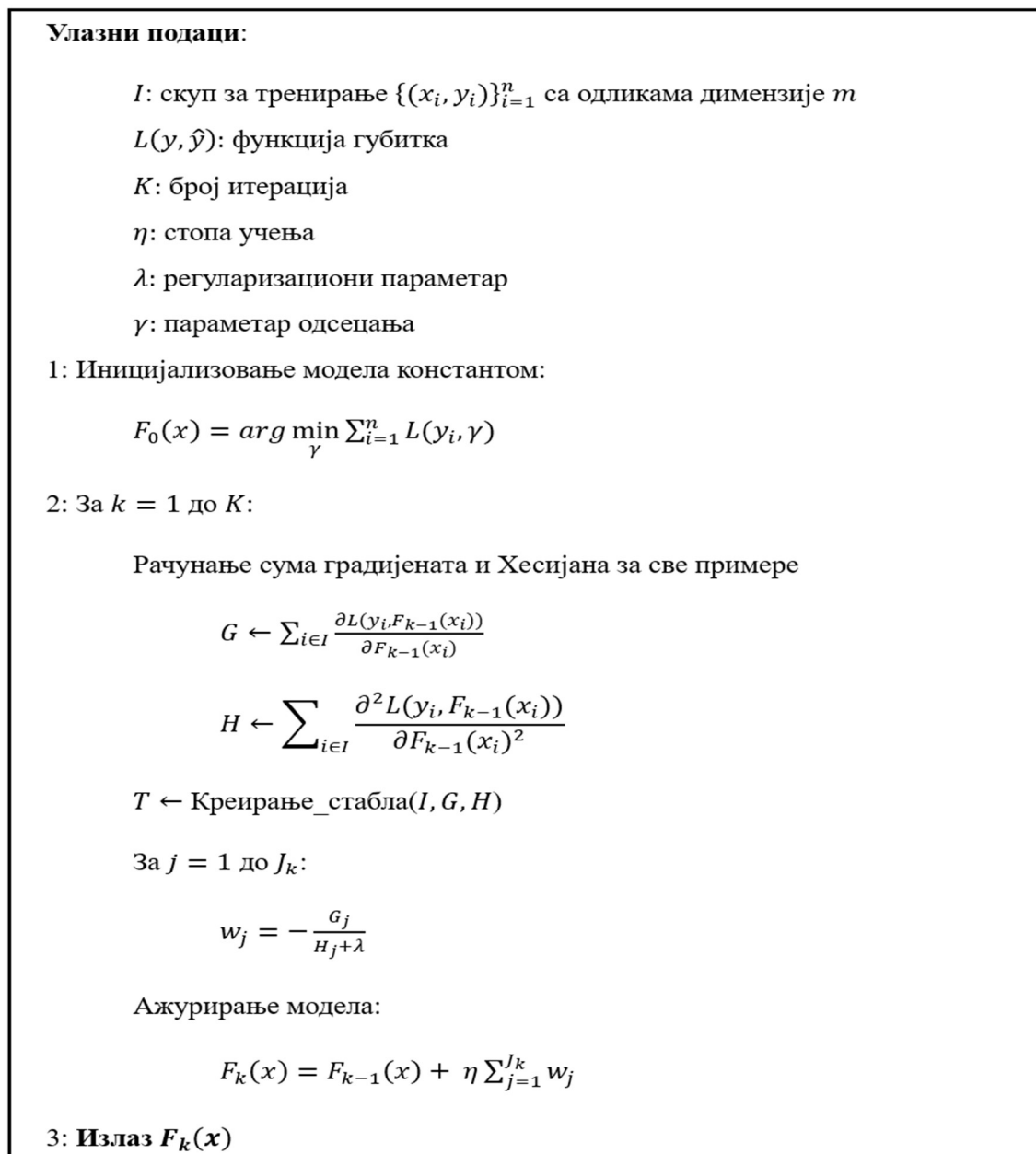
Слика 3.12. Градијентни појачивач

3.6.4. Екстремни градијентни појачивач

Екстремни градијентни појачивач (енг. *Extreme gradient boosting (XGBoost)*) заснован је на идеји градијентног појачивача, који је описан у претходном поглављу, али је додатно оптимизован и доста је скалабилнији.

Алгоритам функционише тако што се направи иницијални погодак који би модел предвиђао, а затим се креирају стабла одступања којих има K . За свако стабло и сваки лист у стаблу (R) рачунају се вредности излаза (ω), где је λ регуларизациони параметар. Ново предвиђање модела у том тренутку се рачуна као претходно предвиђање на које се додаје излаз стабла за конкретан податак. Излаз стабла може бити скалиран стопом учења (η). Алгоритам се завршава када се креира задати број стабала.

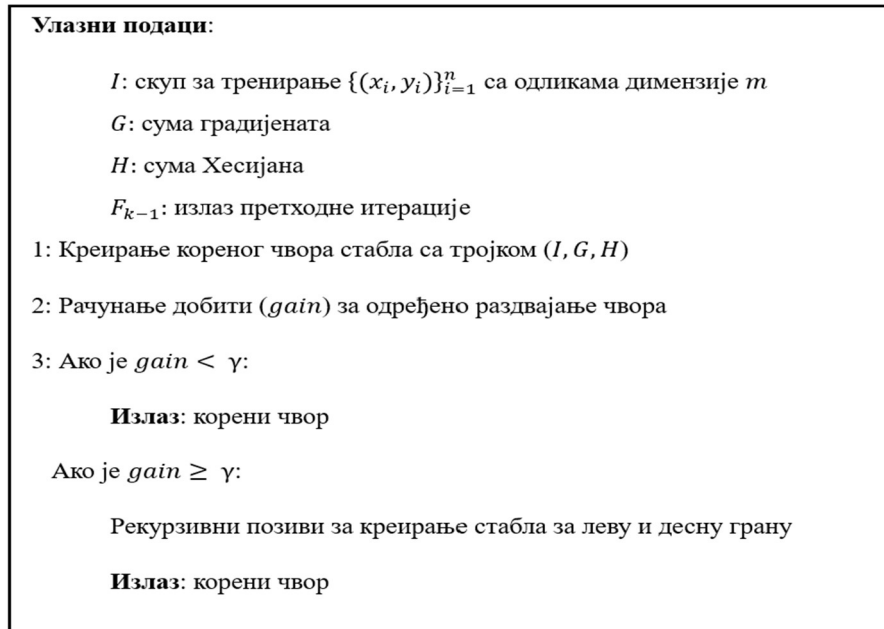
Ток алгоритма може се видети на слици 3.13.



Слика 3.13. Екстремни градијентни појачивач

Приликом креирања стабла покушавају се различити параметри за раздвајање чворова, рачунају се скорови сличности, на основу којих се рачунају добити. Бирају се параметри који дају највећу добит. Затим се врши одсецање (орезивање) стабла на основу добити. Уколико је добит мања од параметра γ одсеца се грана, а уколико није наставља се градњом стабла.

Алгоритам изградње стабла може се видети на слици 3.14.



Слика 3.14. Екстремни градијенти појачивач: изградња стабла

Екстремни градијентни појачивач креира стабла минимизујући излазне вредности листова. [29] Функција која се минимизује је:

$$\sum L(y_i, F(x)) + \frac{1}{2} \lambda \omega^2$$

Екстремни градијенти појачивач у циљу бржег израчунавања апроксимира ову функцију Тејлоровим полиномом другог реда, уместо итеративног градијентног спуста који би могао да буде скуп по ресурсе.

$$L(y_i, F(x)) \approx L(y_i, F(x)) + G \omega + \frac{1}{2} H \omega^2$$

где су G – градијенти (парцијални изводи првог реда) и H – Хесијани (парцијални изводи другог реда). До излазне вредности листа долази се на следећи начин:

$$\sum L(y_i, F(x)) + \frac{1}{2} \lambda \omega^2 = \cancel{L(y_i, F(x))} + G\omega + \frac{1}{2} H \omega^2 + \frac{1}{2} \lambda \omega^2$$

Како $L(y_i, F(x))$ не садржи ω које се оптимизује, може се избацити из прорачуна.

$$= G\omega + \frac{1}{2} H \omega^2 + \frac{1}{2} \lambda \omega^2 \quad \bigg/ \frac{\partial}{\partial \omega}$$

$$G + (H + \lambda)\omega = 0$$

$$\omega = - \frac{G}{H + \lambda}$$

Функција губитка у случају класификације се дефинише формулом:

$$L(y_i, F(x)) = -(y_i \log p_i + (1 - y_i) \log(1 - p_i))$$

па се градијенти и Хесијани рачунају:

$$G = -y_i + \frac{e^{\log(odds)}}{1 + e^{\log(odds)}} = -(y_i - p_i)$$

$$H = \frac{e^{\log(odds)}}{1 + e^{\log(odds)}} \frac{1}{1 + e^{\log(odds)}} = p_i(1 - p_i)$$

а излазна вредност ω постаје:

$$\omega = - \frac{\text{Сума одступања}}{\text{Сума(претходна вероватноћа * (1 - претходна вероватноћа))} + \lambda}$$

Унапређења екстремног градијентног појачивача у односу на основну верзију градијентног појачивача су одсецање стабала током градње, где се могу правити и екстремнији кораци у зависности од γ параметра. Такође λ регуларизациони параметар спречава претренирање модела.

Алгоритам се може и додатно оптимизовати следећим техникама:

- коришћењем квантила – уместо свих података у скупу, који онда приликом налажења најбоље добити могу значајно да успоре креирање стабала, користе се квантили који деле податке у целине исте величине. Што је већи број квантила проналазе се бољи параметри за рачвање стабла,
- коришћењем скеч алгоритма који расподељује податке (квантиле) на више рачунара и после их обједињује чиме се значајно убрзава извршавање,
- чувањем градијената и Хесијана у кеш меморији,
- распоређивањем података на различите партиције диска,
- компресовањем података,
- коришћењем посебног алгоритма

за налажење најбоље поделе у стаблима узимајући у обзир да подаци могу бити оскудни, што подразумева недостајуће податке, или велики број 0 у подацима. По алгоритму (слика 3.15.) подаци се поделе у две групе, групу која нема недостајуће вредности и групу која има. Параметри за поделу се узимају из групе која нема недостајуће вредности, и тражи се најбоља добит у зависности да ли се недостајући податак смести у леви или десни чвор стабла, а из алгоритма излазе најбољи параметри за рачвање и подразумевана дирекција за смештање недостајућих вредности у односу на параметар.

Улазни подаци:

I : подаци тренутног чвора

$I_k = \{i \in I | x_{ik} \neq \text{недостаје}\}$

d : димензија одлика података

$G \leftarrow \sum_{i \in I} g_i, H \leftarrow \sum_{i \in I} h_i$

За $k = 1$ до m :

$G_L \leftarrow 0, H_L \leftarrow 0$

За свако j у растуће сортираним I_k :

$G_L \leftarrow G_L + g_j, H_L \leftarrow H_L + h_j$

$G_R \leftarrow G - G_L, H_R \leftarrow H - H_L$

скор $\leftarrow \max(\text{скор}, \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda})$

$G_R \leftarrow 0, H_R \leftarrow 0$

За свако j у опадајуће сортираним I_k :

$G_R \leftarrow G_R + g_j, H_R \leftarrow H_R + h_j$

$G_L \leftarrow G - G_R, H_L \leftarrow H - H_R$

скор $\leftarrow \max(\text{скор}, \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda})$

Излаз: Параметри за рачвање и подразумевана дирекција

Слика 3.15. Екстремни градијентни појачивач: решавање проблема недостајућих података

3.7. Дубоко учење

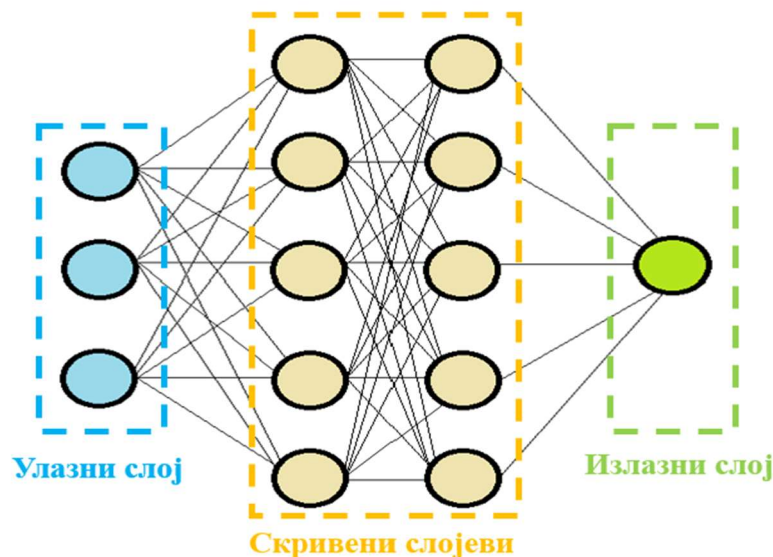
Дубоко учење је врста машинског учења које омогућава да се увиђају веома комплексне зависности међу подацима на основу вишеслојних процесирања [30]. Ова техника је значајно унапредила машинско учење омогућавајући детекцију говора, објеката и анализу многих других података попут слика, аудио и видео сигнала. Дубоко учење открива унутрашњу структуру у веома великом скупу података тако што врши процес враћања грешке уназад (енг. *backpropagation*) чиме омогућава параметрима модела да се прилагоде тако да грешка буде што мања.

У наставку поглавља су описане најчешће коришћене методе дубоког учења.

3.7.1. Неуралне мреже

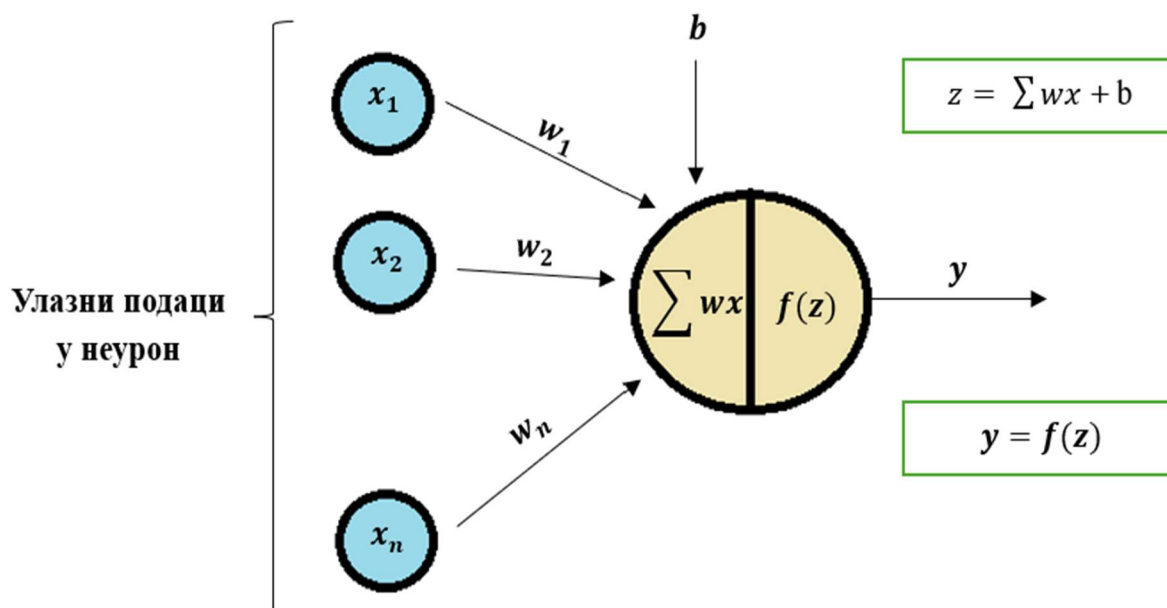
Људски мозак сачињен је у највећој мери од нервних ћелија тј. неурона. Неурони имају наставке, од којих дендрити примају сигнале од других ћелија, а аксони прослеђују сигнале ка другим ћелијама. Синапсе су задужене за пренос сигнала између неурона. Мозак је и данас један од најзагонетнијих органа, а процеси који нам омогућавају да учимо нове вештине, имамо сећања и волимо су послужили као инспирација за креирање модела дубоког учења који називамо неурална мрежа.

Неурална мрежа (енг. *Feedforward Neural networks (FNN)*) има своје чворове, који представљају неуроне и који су међусобно повезани. Чворови имају своје локалне информације, а преко веза им могу стићи и информације из других неурона. Такође, чворови могу да проследе информације другим неуронима. Мрежа има улазни слој, који прима информације, затим се подаци прослеђују скривеним слојевима којих може имати и више од једног за сложене проблеме, и на крају постоји излазни слој који представља предвиђање класе у случају проблема класификације [31]. Пример структуре једноставне неуралне мреже се може видети на слици 3.16.



Слика 3.16. Структура неуралне мреже

Параметри мреже називају се тежине (енг. *weight*) и пристрасности (енг. *bias*). На слици 3.17. може се видети детаљније шта се дешава у сваком неурону, који подаци улазе у њега, а који излазе. Сви подаци који улазе у неурон множе се са својим одговарајућим тежинама, и на ту суму се додаје пристрасност. Затим се примењује активациона функција која представља излаз из неурона.



Слика 3.17. Приказ неурона у неуралној мрежи

Активационе функције се користе да трансформишу улазне податке у неурон у излазни податак из неурона који се прослеђује ка суседима [32]. Најпре се рачуна сума производа улазних података и њима одговарајућих тежина, на ту суму се додаје пристрасност, а затим се на тај резултат (z) примењује активациона функција. Уколико активациона функција не би постојала, излаз неурона би био увек једноставна линеарна функција, а неуралне мреже не би имале снагу да препознају сложене зависности међу подацима. Због тога поред основне линеарне функције која некад и јесте довољна, активационе функције могу бити и нелинеарне.

Неки од најчешће коришћених типова активационих функција су:

- Линеарна функција $f(x) = \alpha x$
- Сигмоидна функција $f(x) = \frac{1}{1+e^{-x}}$
- Хиперболичка тангента функција $f(x) = \frac{2}{1+e^{-2x}} - 1$
- Исправљена линеарна јединица $f(x) = \max(0, x)$
- Експоненцијална линеарна јединица $f(x) = x, x \geq 0$ и $f(x) = \alpha(e^x - 1), x < 0$

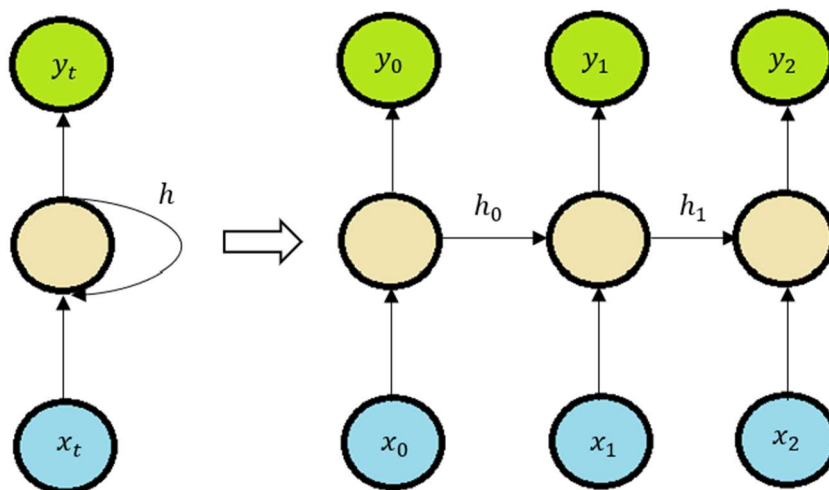
Рачунањем излаза из свих неурона, и пропагацијом информација до излазног слоја мреже врши се предикција модела. Излазни свој мреже може имати један или више неурона у зависности да ли се решава бинарна или вишекласна класификација. Када постоји предикција може се видети колико је одступање од праве вредности, а затим се процесом пропагирања грешке назад кроз мрежу врши прилагођавање тежина и пристрасности да би се грешка што више смањила.

Прилагођавање параметара мреже и умањење грешке се углавном ради методама попут градијентног спуста, па је због тога важно да активационе функције које се примене буду диференцијабилне. Ова метода се врло једноставно имплементира, и чини моделе флексибилним, скалабилним и ефикасним.

Проблеми који могу да се јаве код овог начина прилагођавања параметара су ти да модел може превише да се прилагоди тренинг подацима, потенцијални изузеци имају доста утицаја, уз то да ова метода може изискивати доста времена за само тренирање мреже.

3.7.2. Рекурентне неуралне мреже

Рекурентне неуралне мреже (енг. *Recurrent neural networks (RNN)*) за разлику од обичних неуралних мрежа које искључиво повезују неуроне у смеру од улазног слоја ка излазном слоју, дозвољавају и повратне (рекурентне) везе. Ово им омогућава да памте претходне информације чиме њихова главна употреба постаје управо у анализирању података који су временски међусобно зависни [33]. Рекурентни неурони имају скривена стања која им омогућавају да имају меморију и одржавају информације о претходним улазима у секвенци. На слици 3.18. може се видети развијање (одмотавање) рекурентне мреже кроз време.



Слика 3.18. Рекурентна неурална мрежа

Скривено стање h , рачуна се као излаз активационе функције скривеног стања која прихвата збир производа матрице улазних тежина и улаза, производа матрице скривених тежина и скривених стања и пристрасности, по формули:

$$h_t = \sigma(W_I x_t + W_H h_{t-1} + b_h)$$

где је σ активациона функција скривеног стања, W_I матрица улазних тежина, W_H матрица скривених тежина и b_h пристрасност за скривена стања.

Излаз из рекурентног неурона y_t , рачуна се као излаз активационе функције која прихвата збир производа матрице излазних тежина и скривених стања и пристрасности, по формули:

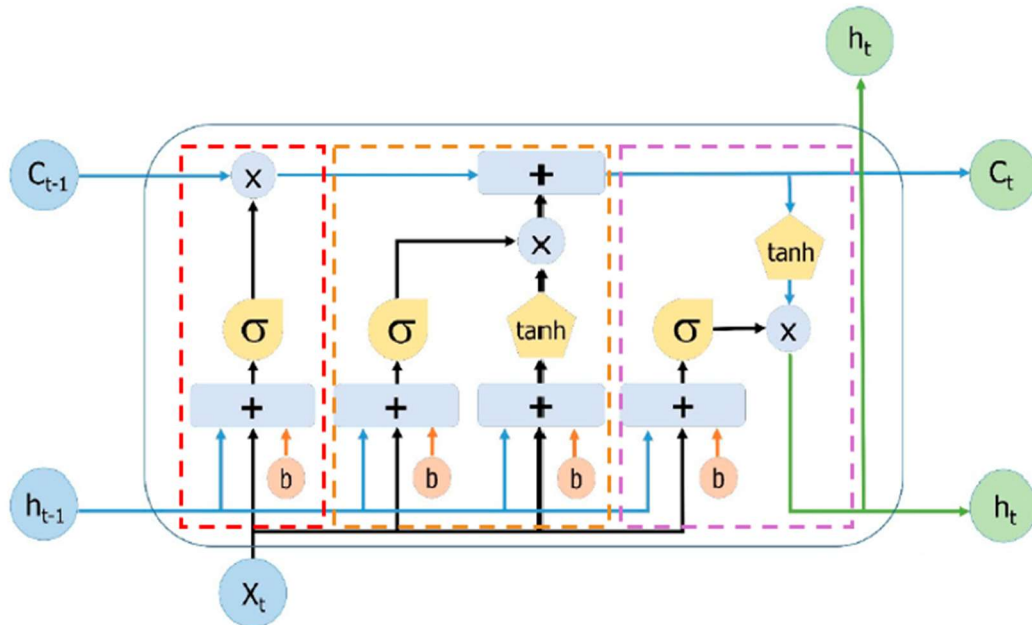
$$y_t = W_O h_t + b_o$$

где је W_O матрица излазних тежина из скривеног слоја, h_t скривено стање и b_o пристрасност за излазни слој. Постоје и имплементације где се y_t добија уз примену активационе функције над $W_O h_t + b_o$.

Главни недостаци рекурентних неуралних мрежа су нестајући и експлодирајући градијент. Оба ова проблема посебно долазе до изражаја за веома дугачке секвенце. Нестајући градијент може да се деси уколико се тежине превише смање и приближе нули, па кораци постану веома мали и не стигне се до минимума функције. Експлодирајући градијент наступа када су тежине веће, али за велики број одмотавања кораци постану веома велики и поново се избегне минимум функције.

3.7.3. Дуга краткорочна меморија

Проблем нестајућег и експлодирајућег градијента који настаје услед веома дугачких секвенци података, може се решити применом дугих краткорочних меморија (енг. *Long short-term memory (LSTM)*) [34]. Дуга краткорочна меморија има ћелије које се састоје од три капије. Капија заборављања контролише који степен дугорочне меморије се памти, на основу краткорочне меморије и новог податка. Улазна капија контролише на који начин се ажурира дугорочна меморија на основу краткорочне меморије и новог податка. Излазна капија контролише које информације излазе из ћелије под утицајем ажурираних дугих и краткорочних меморија. Приказ јединице дуге краткорочне меморије може се видети на слици 3.19.



Слика 3.19. Јединица дуге краткорочне меморије [35]

Од улазних података на слици 3.19. постоје x_t као тренутни улазни податак, c_{t-1} дугорочна меморија од претходне LSTM јединице и h_{t-1} краткорочна меморија од претходне LSTM јединице. Од излазних података постоји дугорочна меморија тренутне јединице c_t и краткорочна меморија h_t . σ представља сигмоидну функцију, а $\tanh$ хиперболички тангенс. b представља пристрасност.

Капија заборављања приказана је црвеним испрекиданим правоугаоником, улазна капија наранџастим испрекиданим правоугаоником, а излазна капија љубичастим. Такође поред пристрасности у урачунавања улазе и тежине које се нису приказане на слици.

Због могућности да увиђа сложене зависности најчешћа примена дуге краткорочне меморије је код: предикције временских серија, обраде природних језика, обраде слика и видео записа, компјутерске визије и препознавања текста.

Мане оваквих модела су:

- Склоност ка претренирању,
- Велики број хиперпараметара модела који је потребно оптимизовати,
- Висока захтевност у виду ресурса.

3.7.4. Неуралне мреже оплемењене механизмом пажње

Механизми пажње се користе да побољшају предвиђања модела који примају секвенце тако што увиђају који делови секвенце имају највише утицаја на сам излаз додељујући им тежине. Једно од најпознатијих имена када се говори о механизмима пажње је сигурно Луонг [36], а његови механизми пажње могу се поделити на локалне и глобалне.

Глобални механизам (слика 3.20.) усмерава пажњу на све улазне тачке у секвенци. Најпре се рачунају скорови између скривених стања сваке тачке у улазној секвенци и излазног скривеног стања, по једној од три формуле:

$$score(h_t, \bar{h}_s) = \begin{cases} h_t^T \bar{h}_s \\ h_t^T \bar{h}_s W_a \\ v_a^T \tanh(W_a [h_t; \bar{h}_s]) \end{cases}$$

Излазно стање може бити стање декодера, а може бити и стање из улазне секвенце чиме се рачуна важност свих других улаза у односу на посматрану.

На основу скорова рачуна се вектор поравнања по формули:

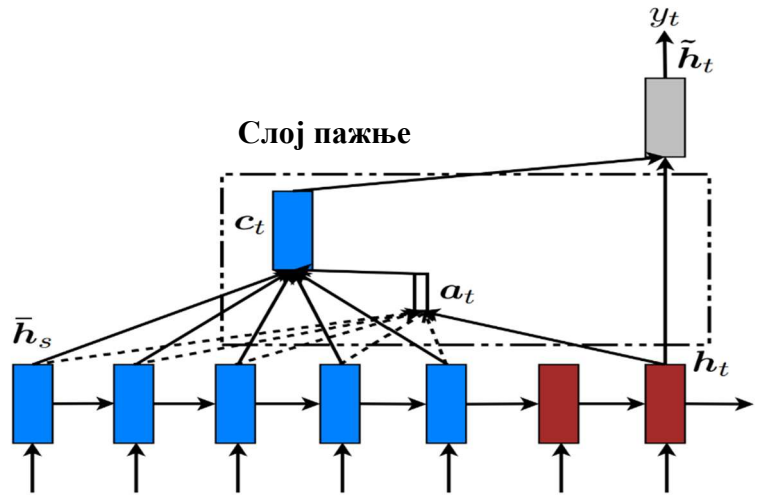
$$a_t = \frac{e^{score(h_t, \bar{h}_s)}}{\sum_j e^{score(h_t, \bar{h}_j)}} \\ = softmax(score(h_t, \bar{h}_s))$$

Контекст вектор c_t , садржи тежински скалирану сваку тачку из улазне секвенце.

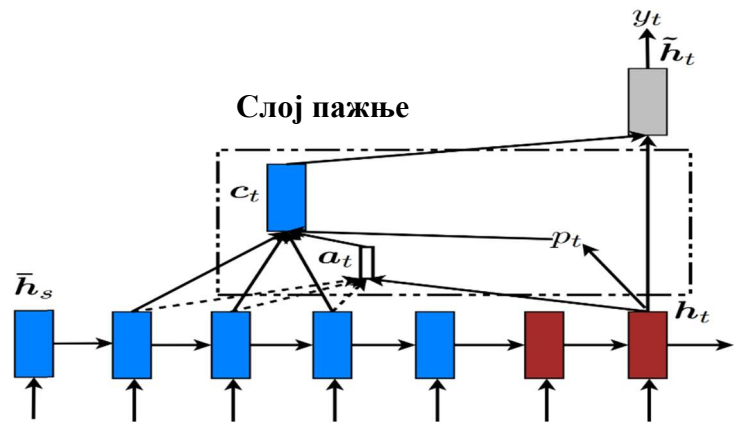
Тежине су добијене из скорова и представљају утицај сваке тачке из улаза на излаз. Контекст вектор добија се по формули:

$$c_t = \sum_{i=0}^n a_{t,i} \bar{h}_{s,i}$$

Локални механизам (слика 3.21.) усмерава пажњу на само неке улазне тачке у секвенци, јер некад глобални механизам може бити скуп по ресурсима. За свако излазно скривено стање се рачуна позиција поравнања p_t , а од улазних стања се узимају из прозора $[p_t - D, p_t + D]$, а D се бира емпиријски.



Слика 3.20. Глобални механизам пажње [36]



Слика 3.21. Локални механизам пажње [36]

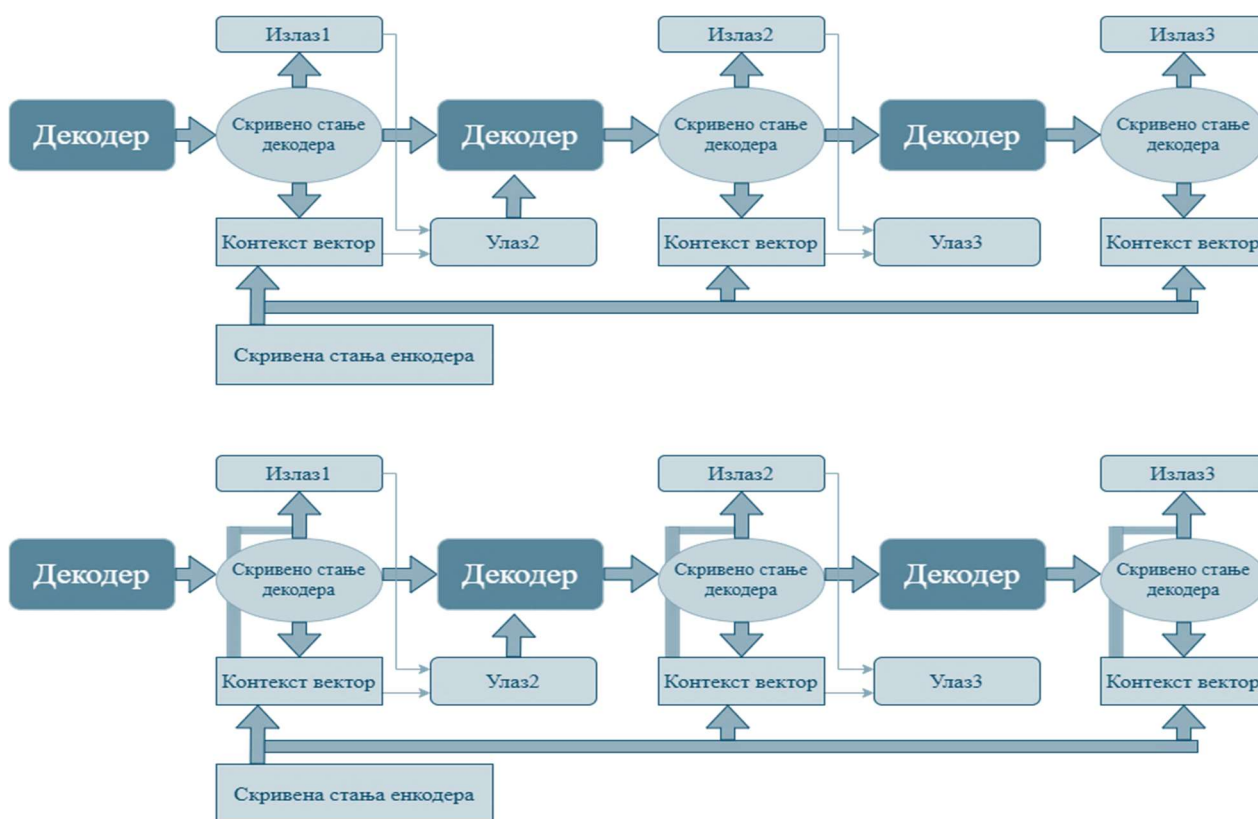
Позиција поравнања може се рачунати:

$$p_t = \left\{ S \operatorname{sigmoid}(v_p^T \tanh(W_p h_t)) \right.$$

где је S дужина секвенце.

Још један познат механизам пажње предложен је од стране Бахндануа [37]. Главна разлика је у томе што се у Луонг моделу посматра тренутно излазно стање h_t , а у овом моделу се посматра h_{t-1} , што се може видети на слици 3.22. (горе Бахндануа, доле Луонг). Такође код рачунања скорова Луонг модел нуди више алтернатива, у односу на Бахндануа механизам који рачуна скорове само по формули:

$$\operatorname{score}(h_t, \bar{h}_s) = v_a^T \tanh(W_a[h_t; \bar{h}_s])$$



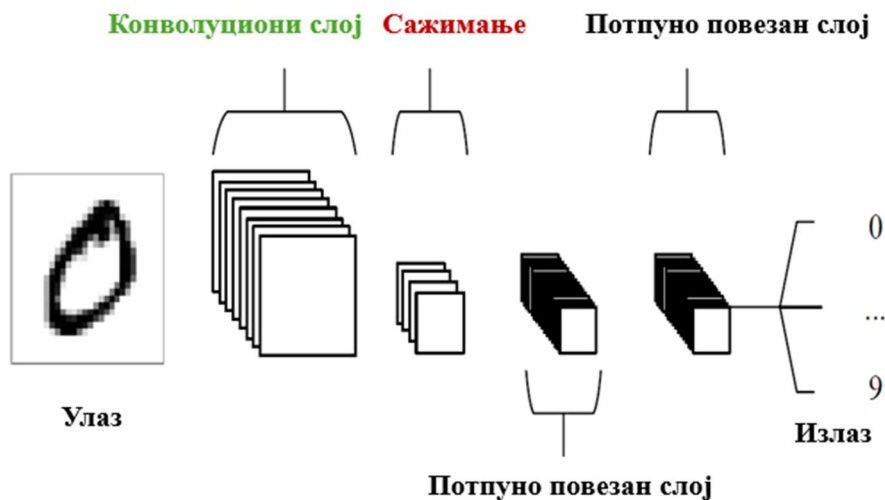
Слика 3.22. Поређење Луонг и Бахндануа механизма пажње

3.7.5. Конволуционе неуралне мреже

Конволуционе неуралне мреже (енг. *Convolutional neural networks (CNN)*) врста су неуралних мрежа које су своју примену нашле најпре у препознавању шема и извлачењу специфичних одлика у сликама [38]. Стандардне неуралне мреже би морале да буду веома комплексне да би могле да обављају овај посао, док конволуционе мреже успевају да са мањим бројем чворова добију врло задовољавајуће резултате. Неурони у конволуционим мрежама су организовани у три димензије: ширина, висина и дубина.

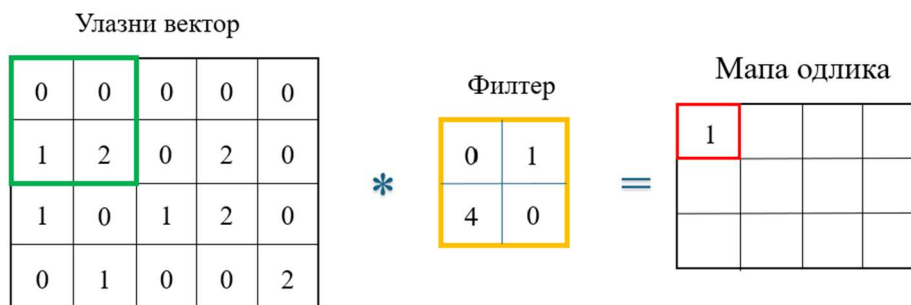
Конволуционе неуралне мреже састоје се од више слојева као што су улазни слој, конволуциони слој, слој за сажимање, један или више потпуно повезаних слојева и излазни слој. Ова архитектура може се видети на слици 3.23.

Улазни слој садржи вредности пиксела слике. Конволуциони слој примењује филтере на улазну слику и издваја битне карактеристике, а слој за сажимање даје увид где је унутар слике филтер дао најбоље поклапање и упрошћава слику да би се смањило време извршавања, а уједно и одржава најбитније податке добијене филтерима. Излаз из слоја сажимања развлачи се у потпуно повезан слој неурона који на крају даје финалну предикцију.



Слика 3.23. Архитектура конволуционе неуралне мреже [38]

Унутар конволуционог слоја дешавају се најзначајније калкулације па ће овај слој бити детаљније анализиран. Филтери (језгра) у овом слоју су задужени за проналажење одлика слика и може их бити више. Сваки филтер пролази кроз целу слику рачунајући скаларни производ вектора (матрица) као на слици 3.24.



Слика 3.24. Примена филтера на улазни вектор конволуционе неуралне мреже

Сваки филтер има своју мапу одлика. Мапе одлика за све филтере пролазе кроз активациону функцију након чега се примењује слој за сажимање. У слоју за сажимање може се вршити издвајање максималних или просечних вредности.

Слој за сажимање се затим поравнава (развлачи) и постаје улаз потпуно повезаног слоја неурона који даје предикцију.

Сви параметри мреже могу се подешавати минимизовањем грешке, укључујући и филтере који се примењују на улазни вектор.

4. Преглед постојећих решења у области Паркинсонове болести и машинског учења

Због непостојања лека за Паркинсонову болест, као и непостојања стандардизованог теста за дијагностиковање болести, Паркинсонова болест и друга неуродегенеративна обољења постала су предмет интересовања за научнике и истраживање широм света. Тражени су различити подаци који могу да послуже за рано откривање болести, спровођени су многобројни експерименти уз примене шареноликог опсега алгоритама машинског учења. У наставку поглавља дат је детаљан преглед постојећих скупова података из ове области, као и попис научних радова.

4.1. Скупови података

Подаци који улазе у моделе машинског учења од великог су значаја за финалне предикције које модел прави. Најчешће анализирани подаци за дијагностиковање Паркинсонове болести и атипичних Паркинсонизама могу бити подељени на: снимке магнетне резонанце, електроенцефалографске снимке, снимке гласа, цртежа, хода, покрета и друге.

У табели 1. види се преглед скупа података који су креирани ради анализе немоторних симптома болести. За сваки скуп података наведена је:

- ознака скупа података (скр. СП),
- извор прикупљања (институција),
- тип података,
- укупан број пацијената,
- број здравих пацијената,
- број са Паркинсоновом болешћу.

За скупове података у табели 1. за које је пронађен начин прикупљања података је наведен и тај податак:

- У скупу СП.1. сви снимци су направљени истим типом скенера и прикупљени су у 20-30 минута видног поља свих снимака укључујући вертекс, мали мозак и Варолијев мост,
- У скупу СП.3. 3Д секвенце осетљиве на контраст неуромеланина су добијене коришћењем 50 резова, са временом аквизиције 4 мин 12,9 с. Такође су прикупљене слике T1, T2, инверзије слабљења течности и дифузионо пондерисане слике како би се елиминисала могућност оштећења мозга,
- У скупу СП.4. подаци се прикупљају за пацијенте који узимају лекове са укљученим и искљученим леком, брзином од 500 узорака у секунди коришћењем синтерованих Ag/AgCl електрода за 64 канала. Пацијенти су први пут дошли на контролу под терапијом, а други пут са терапијом прескоченом за најмање 15 сати,
- У скупу СП.5. вршено је 3 минута снимања са 512 Hz фреквенцијом и 32-каналним *Biosemi ActiveTwo EEG* системом,
- У скупу СП.6. вршено је 5 минута снимања са 128 Hz фреквенцијом и 14-каналним сетом за снимање мозга.

У табели 2. види се преглед скупа података који су креирани ради анализе моторних симптома болести.

Табела 1. Преглед скупова података који служе за анализирање немоторних симптома Паркинсонове болести и атипичних Паркинсонизама

Ознака	Извор	Тип података	Укупан број	Број здравих	Број са ПБ
СП.1.	<i>Parkinson's Progression Markers Initiative</i>	3T T1-тежински снимци магнетне резонанце	406	203	203
СП.2.	<i>Parkinson's Progression Markers Initiative</i>	T2 тежински снимци магнетне резонанце	500	250	250
СП.3.	<i>Department of Neurology, National Institute of Mental Health and Neurosciences</i>	Снимци магнетне резонанце осетљиви на неуромеланин	100	35	45 + 20 атипичних Паркинсонизама
СП.4.	<i>PRED-CT data repository</i>	Електроенцефалографски сигнали	28	/	28
СП.5.	<i>University of San Diego, California</i>	Електроенцефалографски сигнали	31	16	15
СП.6.	<i>Henan Provincial People's Hospital</i>	Електроенцефалографски сигнали	40	20	20
СП.7.	<i>Parkinson's Progression Markers Initiative</i>	Брзи покрети очију (РЕМ) и губитак мириса, подаци о цереброспиналној течности и маркери допаминергичког снимања	584	183	401

Табела 2. Преглед скупова података који служе за анализирање моторних симптома Паркинсонове болести и атипичних Паркинсонизама

Ознака	Извор	Тип података	Укупан број	Број здравих	Број са ПБ
СП.8.	<i>PC-GITA</i>	снимци гласа	100	50	50
СП.9.	<i>Saarbruecken Voice Database</i>	снимци гласа	2042	687	1355
СП.10.	<i>Open Data Commons</i>	снимци гласа	1676	/	/
СП.11.	<i>UCI public repository</i>	снимци гласа	42	/	42
СП.12.	<i>University of Oxford, in collaboration with the National Centre for Voice and Speech</i>	снимци гласа	31	8	23
СП.13.	<i>Department of Neurology in Cerrahpasa Faculty of Medicine, Istanbul University</i>	снимци гласа	252	64	188
СП.14.	<i>Physionet</i>	снимци хода	166	73	93
СП.15.	<i>Department of Medico-Surgical Sciences and Biotechnologies, University of Rome Sapienza</i>	снимци хода	161	80	81
СП.16.	<i>REMPARK project</i>	снимци хода	21	/	21
СП.17.	<i>Department of Neurology in Cerrahpasa Faculty of Medicine, Istanbul University</i>	цртежи	77	15	62
СП.18.	<i>Faculty of Electrical Engineering and Communication, Brno University of Technology</i>	цртежи	75	37	38
СП.19.	<i>Faculty of Sciences - UNESP - Bauru</i>	цртежи	92	18	74
СП.20.	<i>Kaggle</i>	цртежи	66	35	31
СП.21.	<i>Kaggle</i>	цртежи	102	51	51
СП.22.	<i>Federal State Budget Scientific Institution "Scientific Center of Neurology" and Scientific and Educational Medical and Technological Center (SEMTCN) of BMSTU</i>	покрети шаке	32	16	16
СП.23.	<i>Movement Disorders Unit at the Clinic of Neurology, Belgrade</i>	покрети шаке	54	11	14 + 19 атипичних паркинсонизама
СП.24.	<i>Johns Hopkins academic movement disorder clinic</i>	глас, ход, покрети шаке	20	10	10
СП.25.	<i>University of Pittsburgh, Carnegie Mellon University and University of Washington</i>	глас, ход, покрети шаке	5	/	/

За скупове података у табели 2. за које је пронађен начин прикупљања података је наведен и тај податак:

- У скупу СП.8. снимљени су људи који говоре шпански како изговарају самогласнике нормалном интонацијом,
- У скупу СП.9. снимљени су снимци самогласника нормалне, ниске, високе и растуће-падајуће интонације са фреквенцијом од 50Hz,
- У скупу СП.10. снимљени су снимци самогласника без информација о Паркинсоновој болести,
- У скупу СП.11. пацијенти су имали уређаје за даљинско праћење код куће 6 месеци, а снимљено је 200 снимака са 16 гласовних карактеристика за сваког пацијента,
- У скупу СП.12. за сваког појединца снимљене су гласовне мере као што су треперење и друге неправилности,
- У скупу СП.13. појединци су снимљени како три пута изговарају самогласник „а“, испред микрофона са подешеном фреквенцијом од 44.1Hz,
- У скупу СП.14. сваки појединац је носио 8 сензора испод стопала, а вертикална сила реакције тла је мерена на фреквенцији од 100Hz док су ходали уобичајено по два минута,
- У скупу СП.15. почетни скуп података састоји се од 22 карактеристике прикупљене преко сензора инерцијске мерне јединице (ИМЈ),
- У скупу СП.16. ИМЈ сензори постављени на струку узорковали су податке на 200Hz док су појединци обављали неке основне кућне активности као што су ходање од софе до кревета, устајање и седење или приказивање стана,
- У скупу СП.17. појединци су користили *Wacom Cintiq 12WX* таблет и цртали цртеже на екрану дигиталном оловком. Информације о координатама, притиску и углу захвата су сачуване за сваки цртеж,
- У скупу СП.18. цртежи су направљени помоћу *Intuos 4M* оловке, фреквенције 200 Hz,
- У скупу СП.19. појединци су попуњавали образац који је укључивао и спиралне и меандарске цртеже,
- У скупу СП.20. појединци су добили четири различита задатка писања, укључујући круг, спиралу, меандар и сигнале,
- У скупу СП.22. појединци су радили неколико задатака покрета као што су тапкање прстима, пронација-супинација шака, покрети отварања-затварања са максималном могућом амплитудом и брзином. Снимци су трајали 16 секунди за сваки задатак и коришћен је *LeapMotion* сензор,
- У скупу СП.23. сензори причвршћени за палац и кажипрст десне руке појединца снимали су шест пута за сваког пацијента, док су наизменично лупкали палцем и кажипрстом 15 секунди, са паузом од 1 минут,
- У скупу СП.24. појединци су снимали неколико тестова користећи *LG Optimus S* мобилни телефон четири пута дневно, који су укључивали тест гласа, тест држања, тест тапкања прстима и тест времена реакције,
- У скупу СП.25. појединци су носили *Pebble* сат који је био повезан са паметним телефоном.

4.2. Преглед литературе

Како постоје већ покушаји детектовања Паркинсонове болести и других атипичних Паркинсоназама у овом поглављу се даје преглед радова у овој области. У радове који се приказују у овом поглављу ушли су радови написани у претходних 11 година и који су добили тачност детекције преко 80%. У табели 3. наведени су радови који се баве анализом немоторних симптома болести, а за сваки рад наводи се који скуп података је коришћен, које године је рад објављен, који алгоритми машинског учења су употребљени и који резултати су добијени за одговарајуће метрике.

Табела 3. Преглед радова који се баве анализом немоторних симптома Паркинсонове болести и атипичних Паркинсоназама коришћењем модела машинског учења

Рад	Година	Скуп података	Алгоритми	Метрике	Резултати
[39]	2020.	СП.1.	CNN	Тачност Опозив Прецизност Простор испод криве	95.29% 0.943 0.927 0.98
[40]	2018.	СП.2.	CNN	Тачност Простор испод криве	96% 0.995
[41]	2019.	СП.3.	CNN	Тачност Простор испод криве	85.7% 0.911
[42]	2020.	СП.4.	CNN+LSTM	Тачност F1-скор Капа коефицијент	99.22% 0.994 0.984
[43]	2021.	СП.5.	CNN	Тачност	99.7%±0.093
		СП.6.		Прецизност F1-скор	0.99±0.01 0.99±0.0009
[44]	2020.	СП.7	FNN	Тачност Простор испод криве F1-скор	96.60% 0.98 0.97
			RF	Тачност Простор испод криве F1-скор	95.66% 0.98 0.96
			KNN	Тачност Простор испод криве F1-скор	95.16% 0.97 0.96
			SVM	Тачност Простор испод криве F1-скор	95.96% 0.98 0.97

У табели 4. наведени су радови који се баве анализом говора и покрета шаке у области детекције Паркинсонове болести.

Табела 4. Преглед радова који се баве анализом говора и покрета шаке код Паркинсонове болести и атипичних Паркинсонизама коришћењем модела машинског учења

Рад	Година	Скуп података	Алгоритми	Метрике	Резултати
[45]	2022.	СП.8.	CNN	Тачност Простор испод криве	99% 0.89
		СП.9. СП.10.	Коришћени за прилагођавање параметара модела		
[46]	2021.	СП.8.	Резидуална мрежа + LSTM	Тачност Прецизност	97.20% 0.97
[47]	2021.	СП.11.	RNN + LSTM	Тачност Прецизност F1-скор	95.8% 0.92 0.96
		СП.12.		Тачност Прецизност F1-скор	82.2% 0.82 0.90
[48]	2020.	СП.13.	XGBoost	Тачност	85.60%
[49]	2021.	СП.13.	GB	Тачност Прецизност F1-скор	92.95% 0.95 0.95
			XGboost	Тачност Прецизност F1-скор	92.07% 0.93 0.95
			AB	Тачност Прецизност F1-скор	87.22% 0.91 0.92
[50]	2019.	СП.17.	CNN	Тачност F1-скор Простор испод криве	96.5% 0.97 0.99
[51]	2018.	СП.17.	CNN	Тачност	88%
[52]	2021.	СП.18. СП.19. СП.20. СП.21.	Претрениране CNNs	Тачност Тачност Тачност Тачност	62% 90% 90.41% 98.31%
[53]	2020.	СП.22.	KNN SVM DT RF	Тачност Тачност Тачност Тачност	87.5% 89.1% 82.8% 94.5%
[54]	2023.	СП.23.	KNN	Тачност	85.15%

У табели 5. наведени су радови који се баве анализом хода у области детекције Паркинсонове болести, као и постојећа хибридна решења.

Табела 5. Преглед радова који се баве анализом хода код Паркинсонове болести и атипичних Паркинсонизама коришћењем модела машинског учења

Рад	Година	Скуп података	Алгоритми	Метрике	Резултати
[55]	2018.	СП.14.	SVM	Тачност Простор испод криве	94.8% 0.97
[56]	2017.	СП.14.	RF	Тачност Капа коефицијент F1-скор	98.04% 0.95 0.98
[57]	2020.	СП.14.	NB	Тачност F1-скор	69.7% 0.70
			DT	Тачност F1-скор	99.4% 0.99
			SVM	Тачност F1-скор	97.6% 0.97
[58]	2022.	СП.15.	SVM	Тачност F1-скор Простор испод криве	86% 0.85 0.85
			RF	Тачност F1-скор Простор испод криве	82% 0.82 0.82
			DT	Тачност F1-скор Простор испод криве	79% 0.80 0.79
			KNN	Тачност F1-скор Простор испод криве	74% 0.74 0.74
			ANN	Тачност F1-скор Простор испод криве	77% 0.78 0.77
[59]	2018.	СП.16.	CNN	Тачност	89%
[60]	2020.	СП.14.	CNN	Тачност	98.7%
[61]	2015.	СП.24.	RF	Сензитивност Специфичност	96.2% 96.9%
[62]	2014.	СП.25.	/	Тачност	>90%

5. Оптимизација алгоритама машинског учења

Машинско учење нашло је своју примену у разном спектру области, од маркетинга и образовања, преко анализе понашања и видео снимака [63]. За различите врсте проблема примењују се различити модели машинског учења. Да би модел могао да произведе најбоље резултате веома је битно да параметри модела буду подешени на најоптималније вредности.

Постоје две врсте параметара у моделима машинског учења. Параметри који се подешавају током процеса тренирања модела, попут тежина и пристрасности у неуралним мрежама. Друга врста параметара који се подешавају пре самог покретања процеса тренирања модела, се називају хиперпараметри модела. Хиперпараметри модела би били стопа учења, број неурона у улазним и скривеним слојевима неуралних мрежа, фактор за одбацивање података у процесу спречавања претренирања мрежа и слично. Ови параметри се морају подесити пре тренирања јер утичу и на саму архитектуру модела, а сама област подешавања хиперпараметара наишла је на велико интересовање у научном истраживању.

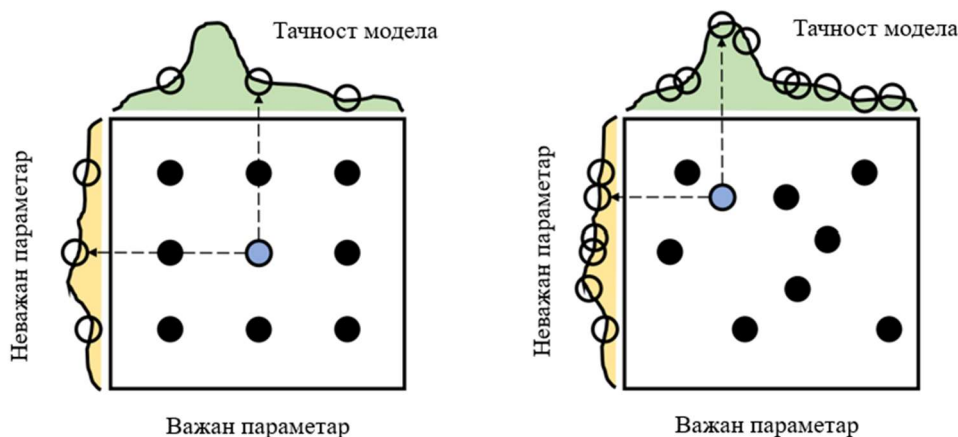
Подешавање хиперпараметара може се радити мануелно међутим ово може бити врло неефикасно када је број хиперпараметара велики, и у погледу финалног резултата и у погледу уложеног времена. Због тога се за решавање овог проблема упошљавају алгоритми оптимизације попут претраге мреже, насумичне претраге, Бајесове оптимизације и метахеуристике.

У наставку поглавља ће поменуте оптимизације бити детаљније описане, са посебним нагласком на оптимизацију хиперпараметара метахеуристикама.

5.1. Претрага мреже и насумична претрага

Претрага мреже покушава све комбинације хиперпараметара у предефинисаној мрежи. За сваки параметар дефинишу се могуће вредности, и након тога се креира мрежа која ће се претражити, а затим се као финална комбинација хиперпараметара бира она која је дала најбољи резултат. Овде се гарантује проналазак најбоље комбинације у оквиру мреже, али постоји могућност да се избегло глобално најбоље решење, и овакво рачунање захтева доста ресурса.

Насумична претрага креира насумичне комбинације хиперпараметара уз дефинисане граничне вредности за сваки параметар. Мање је захтевно од претраге мреже, и може да нађе и боље решење, али није гарантовано. На слици 5.1. може се видети претрага мреже и насумична претрага.



Слика 5.1. Претрага мреже (лево) и насумична претрага (десно) [64]

5.2. Бајесова оптимизација

Бајесова оптимизација користи Бајесово закључивање да пронађе минимум или максимум функције циља. Оптимизациони процес састоји се од две главне компоненте, сурогат модела и функције аквизиције. Сурогат модел представља пробабилистички модел који апроксимира функцију циља, док функција аквизиције на основу сурогат модела усмерава оптимизацију ка бољим решењима. За сурогат моделе може се користити Гаусов процес, насумичне шуме и други модели.

У самом алгоритму се најпре изврши иницијализација хиперпараметара насумичним вредностима, и израчуна се функција циља над тим вредностима хиперпараметара. Почетне вредности заједно са својим функцијама циља стављају се у архиву, и затим се започињу итерације алгоритма. У свакој итерацији се архива убацује у сурогат модел, а затим се на основу функције аквизиције бирају следећи хиперпараметри, који се стављају у архиву заједно са својом прорачунатом функцијом циља. Алгоритам се завршава када се достигне дефинисани број итерација или се конвергира ка најбољем решењу [65]. Ток алгоритма може се видети на слици 5.2.

1. Генерисање $\lambda^{(1)}, \dots, \lambda^{(k)}$
2. Иницијализовање архиве $A^{[0]} = ((\lambda^{(1)}, c(\lambda^{(1)})), \dots, (\lambda^{(k)}, c(\lambda^{(k)})))$
3. За $t = 1, 2, 3, \dots$ до дефинисаног броја итерација:
 - (1) Убацавање $A^{[t-1]}$ у сурогат модел
 - (2) Креирање функције аквизиције $u(\lambda)$
 - (3) Оптимизација функције аквизиције за бирање предложених хиперпараметара λ^+
 - (4) Евалуација функције циља $c(\lambda^+)$
 - (5) Додавање $(\lambda^+, c(\lambda^+))$ у архиву
4. Излаз је λ које је дало најбоље резултате

Слика 5.2. Бајесова оптимизација

Бајесова оптимизација предвиђа следеће вредности за хиперпараметре на основу претходних вредности хиперпараметара, што се разликује од претраге мреже и насумичне претраге где су итерације потпуно независне.

5.3. Метакхеуристике

Велики број оптимизационих проблема је веома комплексан и тежак за решавање, и тешко може бити решен у време које је прихватљиво за чекање [66]. Због тога се често примењују апроксимативни алгоритми који се називају метакхеуристике. Метакхеуристике су функције вишег реда које могу да се прилагоде да решавају било који проблем оптимизације тако што редукују простор претраге решења и ефикасно га претражују. Метакхеуристике због редуковања простора претраге не могу да гарантују проналажење најоптималнијег решења, али су довољно ефикасне да могу да пронађу довољно добро решење.

Основна подела метакхеуристика је по броју кандидат решења које производе, па тако постоје метакхеуристике које производе једно кандидат решење, и оне које производе читаву популацију кандидат решења. Такође још један неизоставан фактор када се говори о метакхеуристикама је балансирање између експлорације и експлоатације.

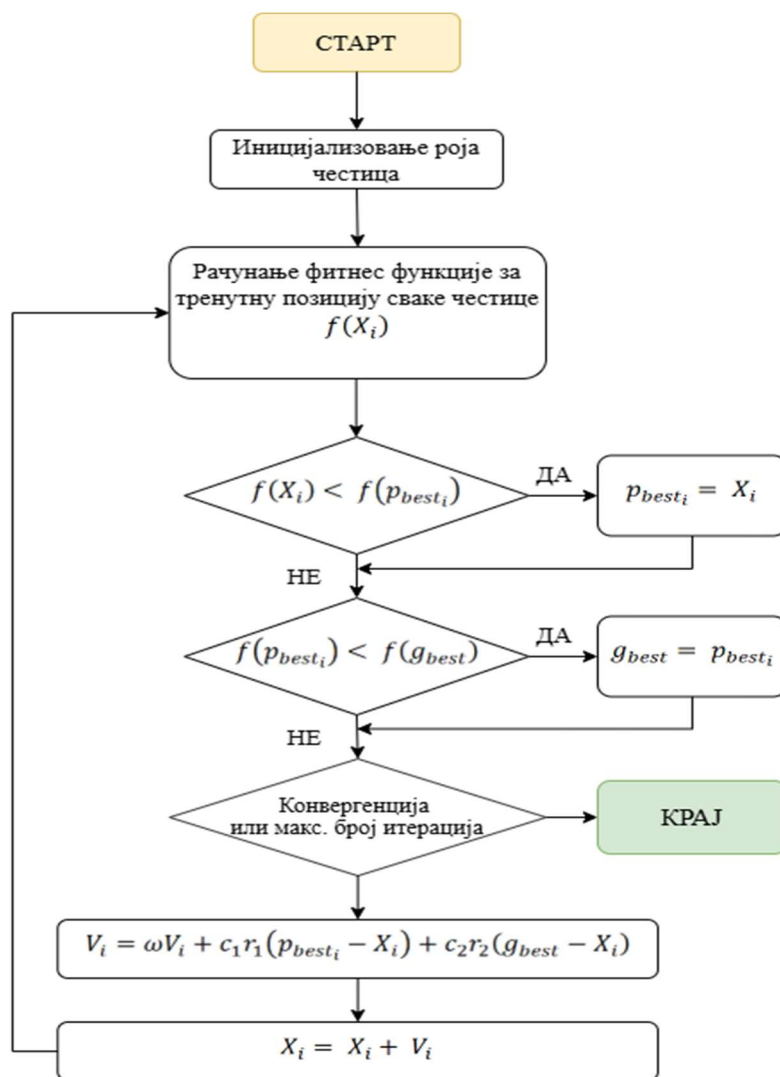
Експлорација или истраживање/диверсификација непознатог простора је од велике важности зарад усмеравања алгоритма ка глобалним минимумима, док је експлоатација/интензификација задужена за локалну претрагу простора у ком већ имамо добра решења у циљу његовог подробног истраживања и налажења локалног минимума. Добри оптимизациони алгоритми имају добар однос између експлорације и експлоатације.

Током процеса усмеравања решења ка најбољим вредностима, кандидат решења се процењују на основу функције циља која је такође значајан фактор у алгоритму метахеуристике. Функција циља формулише који је крајњи циљ оптимизације и уколико се лоше постави може довести до неистинитих решења.

Метахеуристике су неизоставне у процесу оптимизовања хиперпараметара модела машинског учења који су одговорни за постизање што бољих резултата модела. Најчешће, метахеуристике налазе инспирацију у природи, математици, физици и другим природним процесима које увиђамо око нас. У наставку поглавља ће бити описани алгоритми најкоришћенијих метахеуристика.

5.3.1. Оптимизација ројем честица

Оптимизација ројем честица (енг. *Particle swarm optimization (PSO)*) настала је као оптимизација још 1995. године [67], а ток алгоритма може се видети на слици 5.3.



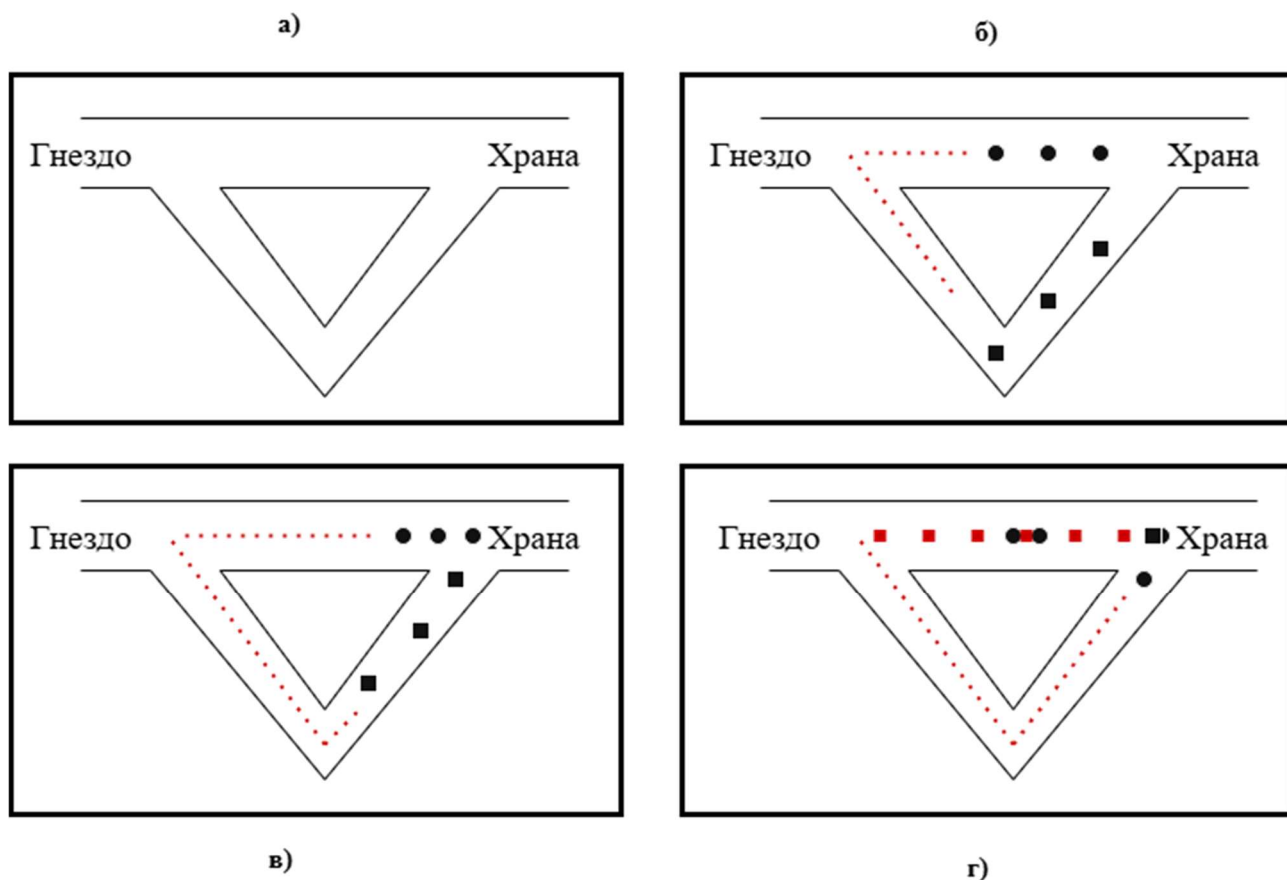
Слика 5.3. Ток алгоритма оптимизације ројем честица

Алгоритам се може описати као процес претраге заснован на ројевима, а свака индивидуа у роју се назива честица и представља потенцијално решење у D-димензионалном простору [68]. При иницијализацији свака честица има своју позицију X и брзину V . Најбоље позиције за сваку честицу су на самом почетку заправо почетне позиције. У свакој итерацији честице рачунају своју фитнес функцију (функцију циља) и налазе најбоље решење за себе, а уједно подешавају и глобално најбоље решење међу свим честицама. Затим се за сваку честицу рачуна нова брзина и позиција. Параметри алгоритма су ω , c_1 и c_2 . c_1 и c_2 врше баланс између експлорације и експлоатације, а подразумеване вредности су 2. r_1 и r_2 су насумичне вредности из опсега $[0,1]$. Алгоритам се извршава док се не прође задат број итерација, или док глобално најбоље решење не конвергира.

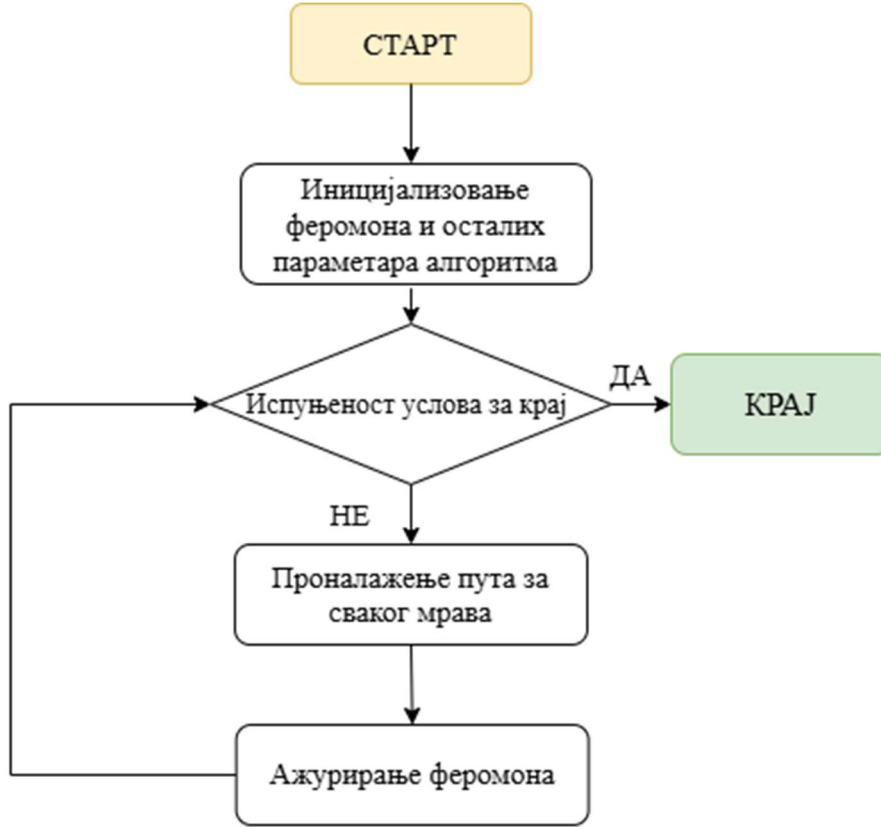
5.3.2. Оптимизација колонијом мрав

Оптимизација колонијом мрав (енг. *Ant colony optimization (ACO)*) настала је као оптимизација још деведесетих година, а ток алгоритма може се видети на слици 5.5. [69]

Мрави су социјални инсекти и међусобно комуницирају приликом проналажења хране [70]. Они насумично претражују простор за храну, а када је пронађу испуштају феромоне које остављају на путу назад. Количина феромона зависи од количине и квалитета хране коју пронађу. Остали мрави прате ове феромоне како би што пре пронашли свој пут до извора хране. Ово понашање се може видети на слици 5.4. Под а) сви мрави су у гнезду, и нема феромона на путевима. Под б) половина мрав бира краћи пут, а половина дужи. Под в) Мрави који су изабрали краћи пут раније стижу до хране и по повратку у гнездо испуштају више феромона. Под г) Краћи пут има више феромона и већина мрав ће у будућности бирати овај пут док ће феромони са дужег пута нестати временом.



Слика 5.4. Понашање мрав



Слика 5.5. Ток алгоритма оптимизације колонијом мрав

Приликом проналажења пута за сваког мрав, уколико је мрав k који у чвору i , вероватноћа да ће ићи у чвор j , је:

$$p_{i,j}^k = \frac{\tau_{i,j}^\alpha \eta_{i,j}^\beta}{\sum_{c_{i,l} \in N(s^p)} \tau_{i,l}^\alpha \eta_{i,l}^\beta}$$

где је $N(s^p)$ скуп изводљивих компоненти, тј. грана (i, l) где је l непосећен од стране посматраног мрав. α и β су параметри модела, $\tau_{i,j}$ феромони на грани (i, j) , а $\eta_{i,j} = \frac{1}{d_{i,j}}$ и $d_{i,j}$ дистанца између (i, j) . Мрави бирају највеће вероватноће приликом креирања пута.

Ажурирање феромона између чворова i и j врши се по формули:

$$\tau_{i,j} = (1 - \rho)\tau_{i,j} + \sum_{k=1}^m \Delta\tau_{i,j}^k$$

где је ρ коефицијент испарења, m број мрав, и $\Delta\tau_{i,j}^k$ количина феромона на грани (i, j) од стране мрав k , која се рачуна као:

$$\Delta\tau_{i,j}^k = \begin{cases} \frac{Q}{L_k}, & \text{ако је мрав користио грану } (i, j) \\ 0, & \text{уколико није користио грану } (i, j) \end{cases}$$

где је Q константа, а L_k дужина пута конструисана од мрав k .

5.3.3. Алгоритам свица

Свици су својим шаблоном понашања инспирисали алгоритам оптимизације. Свици се међусобно привлаче без обзира на свој пол, већ је привлачност пропорционална њиховој светлости, па се свитац са мање светлости приближава свицу са више светлости. Ако постоји свитац који тренутно има највише светлости он ће се кретати насумично [71]. Овакво понашање може се преточити у алгоритам свица (енг. *Firefly algorithm (FA)*), чији ток може да се види на слици 5.6.



Слика 5.6. Ток алгоритма свица

Интензитет светлости опада са повећањем квадрата дистанце између свица [72]. Како се свитац удаљава од извора светлости апсорпција светлости постаје све мања. Овај феномен заправо је аналоган оптимизацији функције циља, па се може рећи да је интензитет светлости пропорционалан фитнес функцији (функцији циља). Привлачност β свица пропорционална је његовом интензитету светлости и може се описати као:

$$\beta = \beta_0 e^{-\gamma r^2}$$

где је β_0 привлачност за $r = 0$, а γ коефицијент апсорпције светлости.

Свици се крећу ка оном свицу који је најпривлачнији по формули:

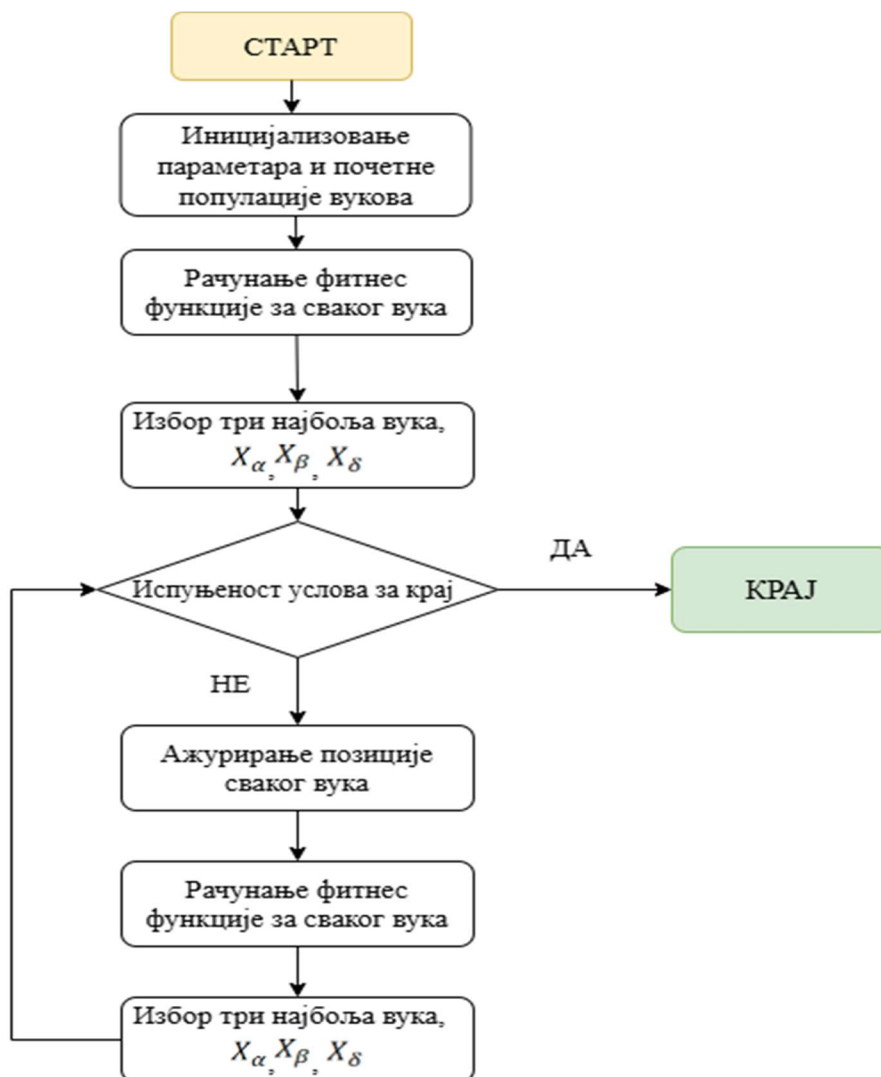
$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \varepsilon_i$$

где је ε_i насумичан број Гаусове дистрибуције, а α насумични параметар.

5.3.4. Алгоритам сивог вука

Сиви вукови су апекс предатори који преферирају живот у чопору који је величине од пет до дванаест вукова у просеку [73]. У самом чопору прати се стриктна социјална хијерархија која се састоји од алфа, бета, делта и омега вукова, а опис њиховог понашања је дат у следећем параграфу.

Алфа вукови су задужени за доношење одлука и сви други вукови показују своју послушност спуштањем репа у присуству алфе. Алфа вук не мора бити и најјачи вук већ вук који најбоље управља чопором. Други ниво у хијерархији чине бета вукови. Бета вук помаже алфа вуку при доношењу одлука и вероватно је кандидат да наследи алфу у случају да се нешто деси. Бета вук управља трећим хијерархијским слојем, а то су омега вукови. Они су последњи у ланцу, последњи једу, али су неизоставни део чопора. Још једна категорија су делта вукови који су испод алфе и бете, али изнад омега вукова. Задужени су за извиђање, лов и бригу о другим вуковима. Поред доминантне социјалне хијерархије интересантан је начин лова чопора, који најпре прате и прилазе плену, окружују га и на крају нападају. Овакво понашање може се преточити у алгоритам сивог вука (енг. *Gray wolf optimizer (GWO)*), чији ток може да се види на слици 5.7.



Слика 5.7. Ток алгоритма сивог вука

где $j \in$:

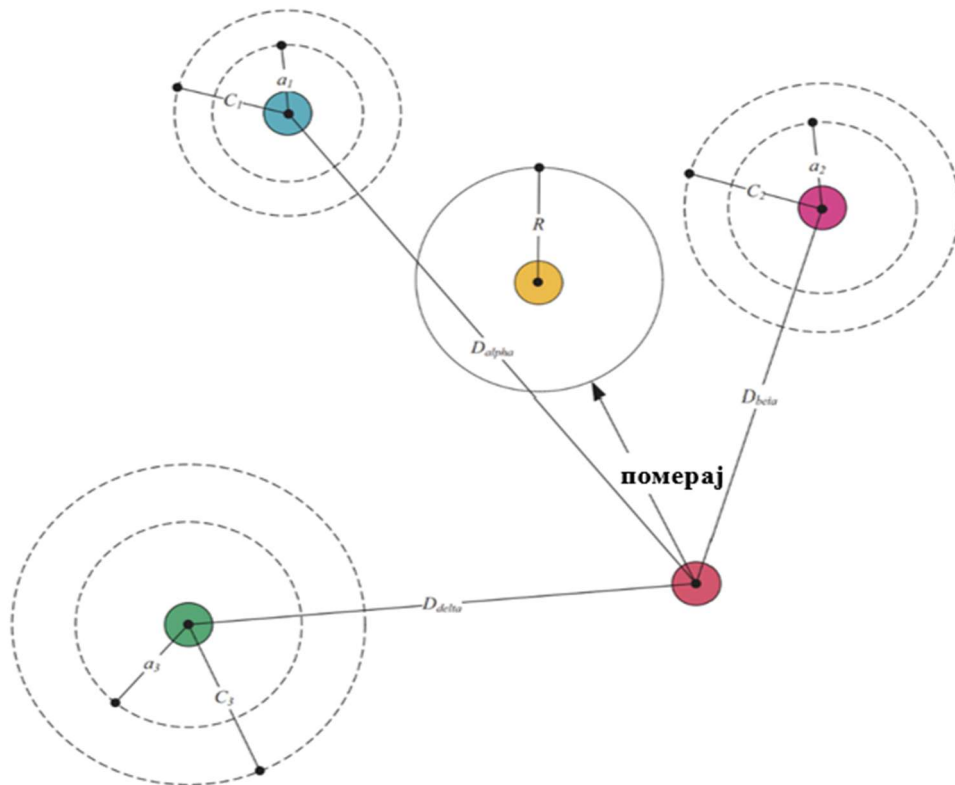
$$X_1 = X_\alpha - A_1 D_\alpha; X_2 = X_\beta - A_2 D_\beta; X_3 = X_\delta - A_3 D_\delta$$

$$D_\alpha = |C_1 X_\alpha - X|; D_\beta = |C_2 X_\beta - X|; D_\delta = |C_3 X_\delta - X|$$

$$A = 2ar_1 - a$$

где a линейно опада од 2 до 0, а r_1 и r_2 су насумичне вредности у скупу $[0, 1]$.

Кретање омега вукова (црвено) на основу позиција алфа (плави), бета (розе) и делта (зелени) вука и усмеравање ка плену (жуто) може се видети на слици 5.8.

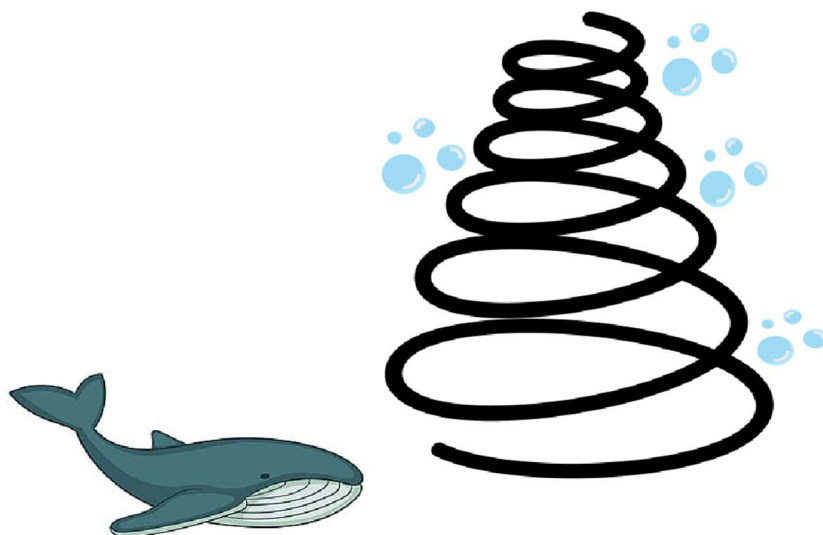


Вектори A и C су веома значајни јер балансирају алгоритам у погледу експлорације и експлоатације.

43

5.3.5. Алгоритам грбавог кита

Грбави кит је највећи сисар на свету и може да достигне дужину од 20 метара и тежину од преко 50 тона [74]. Углавном су предатори и никада не спавају јер морају да удахну ваздух са површине океана. Такође се сматрају једним од најинтелигентнијих животиња са емоцијама. Могу живети сами или у групама и имају врло интересантан метод лова, који се може видети на слици 5.9.



Слика 5.9. Мрежа мехурића грбавог кита

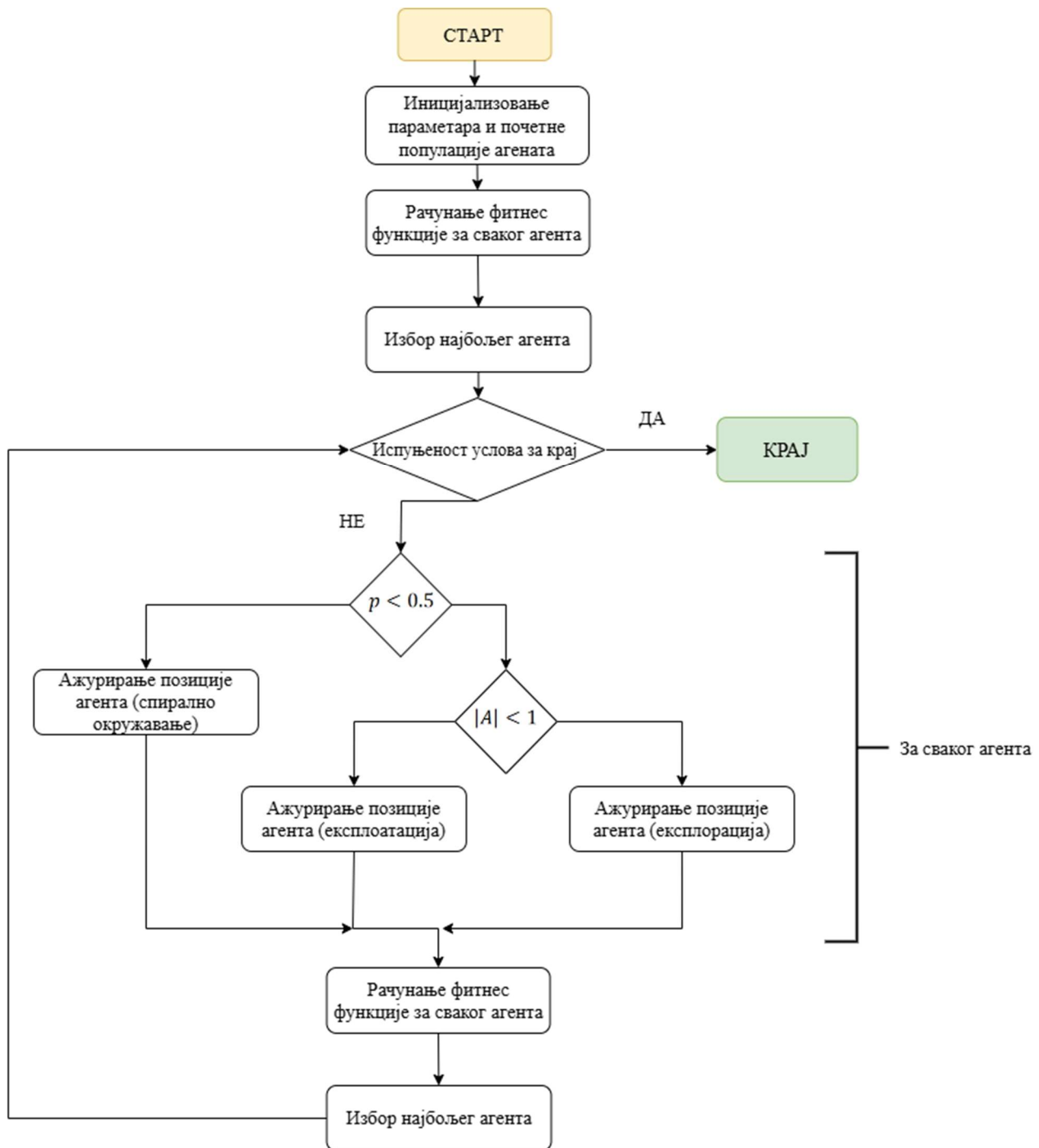
Они најпре пливају у потрази за пленом, а када га пронађу окружују га спиралном мрежом мехурића док се враћају на површину. Оваква врста лова је јединствена међу животињама. Ток алгоритма грбавог кита (енг. *Whale Optimization Algorithm (WOA)*), може да се види на слици 5.10.

У самом алгоритму p је насумичан број између 0 и 1, који води алгоритам или у правцу пливања (тоњења) или у правцу спиралног окруживања плена. У случају да је $p > 0.5$ позиција агента се рачуна као:

$$X = D' e^{bl} \cos(2\pi l) + X^*$$

где је $D' = |X^* - X|$, X^* је плен или најбољи агент до тад, b је константа која дефинише спиралу, а l је насумичан број из опсега $[-1, 1]$.

У случају да је $p < 0.5$ разматра се $A = 2ar - a$, где a линеарно опада од 2 до 0, а r је насумична вредност у скупу $[0, 1]$. За $|A| > 1$, агенти се усмеравају ка другом насумичном агенту чиме се подстиче експлорација, док се за $|A| < 1$, агенти приближавају плену, ка најбољем агенту.



Слика 5.10. Ток алгоритма грбавог кита

У случају да је $|A| < 1$ позиција агента се рачуна као:

$$X = X^* - AD$$

где је $D = |CX^* - X|$, а $C = 2r$.

У случају да је $|A| > 1$ позиција агента се рачуна као:

$$X = X_{rand} - AD$$

где је $D = |CX_{rand} - X|$.

5.3.6. Алгоритам мождане олује

Мождана олуја или брејнсторминг (енг. *Brainstorming optimization (BSO)*) је техника колективног решавања проблема која се често примењује у пословним заједницама. Људи међусобно размењују идеје које се на крају искристалишу у најбоље решење. Ток алгоритма може се видети на слици 5.11., док се на слици 5.12. види алгоритам кластеровања, а на слици 5.13. алгоритам креирања нове индивидуе [75].

1. Иницијализација: Насумично генерисање n потенцијалних решења, и њихова евалуација
2. Док се не нађе довољно добро решење или док се не достигне дефинисан број максималних итерација:
 - (1) Кластеровање n индивидуа у m кластера
 - (2) Креирање нових индивидуа уз насумичан одабир једног или два кластера
 - (3) Селекција боље индивидуе
 - (4) Евалуација свих индивидуа

Слика 5.11. Ток алгоритма мождане олује

1. Кластеровање n индивидуа у m кластера користећи алгоритам кластеризације
2. Рангирање индивидуа у сваком кластеру и узимање најбоље индивидуе као центар кластера
3. Насумично генерисање $r_{\text{кластер}}$ вредности у опсегу од 0 до 1
4. Ако је $r_{\text{кластер}} < \text{предефинисане вероватноће } p_{\text{кластер}}$ онда се насумично бира центар кластера, и генерише се нова индивидуа да замени центар тог кластера

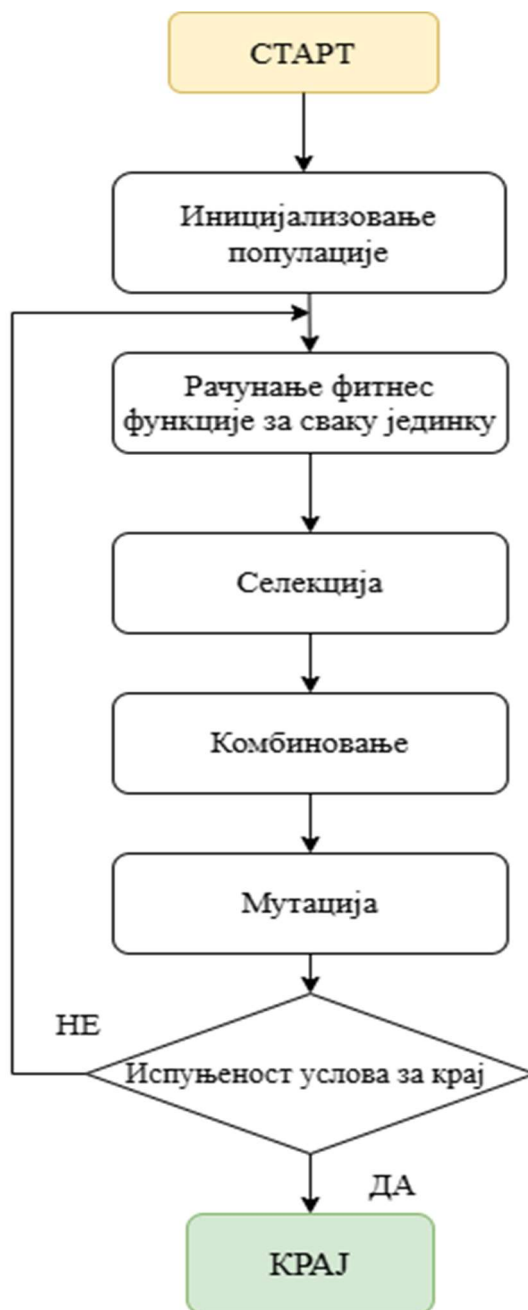
Слика 5.12. Кластеровање у алгоритму мождане олује

1. Насумично генерисање $r_{\text{генерација}}$ вредности у опсегу од 0 до 1
2. Ако је $r_{\text{генерација}} < \text{предефинисане вероватноће } p_{\text{генерација}}$ онда:
 - a. Насумично селектовање једног кластера и генерисање насумичне $r_{\text{једанКластер}}$ вредности у опсегу од 0 до 1
 - b. Ако је $r_{\text{једанКластер}} < \text{предефинисане вероватноће } p_{\text{једанКластер}}$ онда:
 - i. Селектовање центра изабраног кластера и додавање насумичних вредности за креирање нове индивидуе
 - c. Ако је $r_{\text{једанКластер}} \geq \text{предефинисане вероватноће } p_{\text{једанКластер}}$ онда:
 - i. Селектовање насумичне индивидуе изабраног кластера и додавање насумичних вредности за креирање нове индивидуе
3. Ако је $r_{\text{генерација}} \geq \text{предефинисане вероватноће } p_{\text{генерација}}$ онда:
 - a. Насумично селектовање два кластера и генерисање насумичне $r_{\text{дваКластера}}$ вредности у опсегу од 0 до 1
 - b. Ако је $r_{\text{дваКластера}} < \text{предефинисане вероватноће } p_{\text{дваКластера}}$ онда:
 - i. Селектовање центара изабраних кластера, њихово комбиновање и додавање насумичних вредности за креирање нове индивидуе
 - c. Ако је $r_{\text{дваКластера}} \geq \text{предефинисане вероватноће } p_{\text{дваКластера}}$ онда:
 - i. Селектовање две насумичне индивидуе изабраних кластера, њихово комбиновање и додавање насумичних вредности за креирање нове индивидуе
4. Нова индивидуа се пореди са постојећом индивидуом, и задржава се боља

Слика 5.13. Креирање нове индивидуе у алгоритму мождане олује

5.3.7. Генетски алгоритам

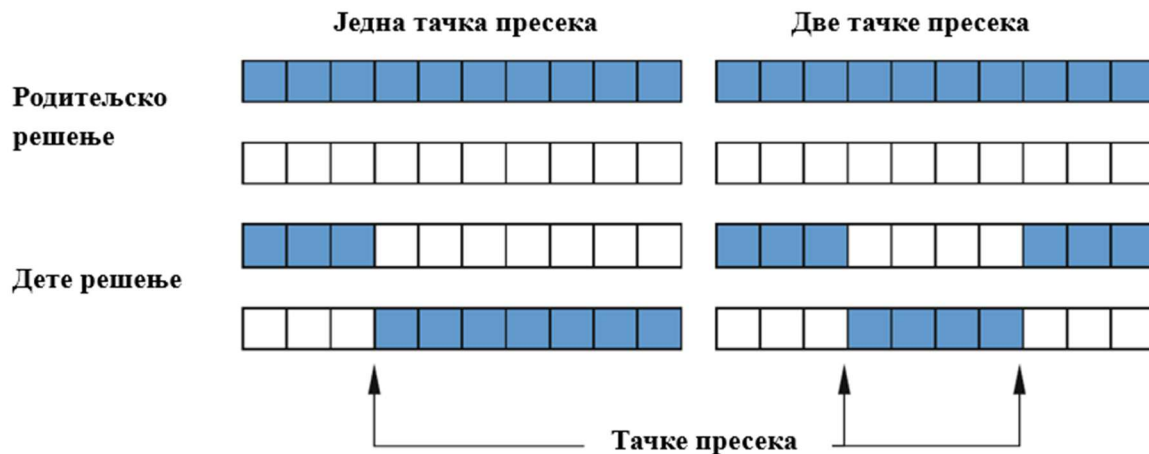
Генетски алгоритам (енг. *Genetic algorithm (GA)*) инспирисан је Дарвиновом теоријом еволуције, у којој се опстанак јединке заснива на идеји да најбоље и најспособније јединке опстају и прослеђују свој генетски материјал [76]. Јединке се процењују фитнес функцијом, а затим се врши насумична селекција, која заправо фаворизује најбоље процењене јединке, које ће дати свој допринос у следећој генерацији. Ток генетског алгоритма може се видети на слици 5.14.



Слика 5.14. Ток генетском алгоритма

Селекција је природна појава у природи, где најспособније јединке имају највише шансе за парење и за проналазак хране. Овде се селекција може симулирати рулет точком, где најспособније јединке имају пропорционално највеће делове точка. И даље постоји могућност да се неке слабије јединке проследе у следећу генерацију, али је доста мала.

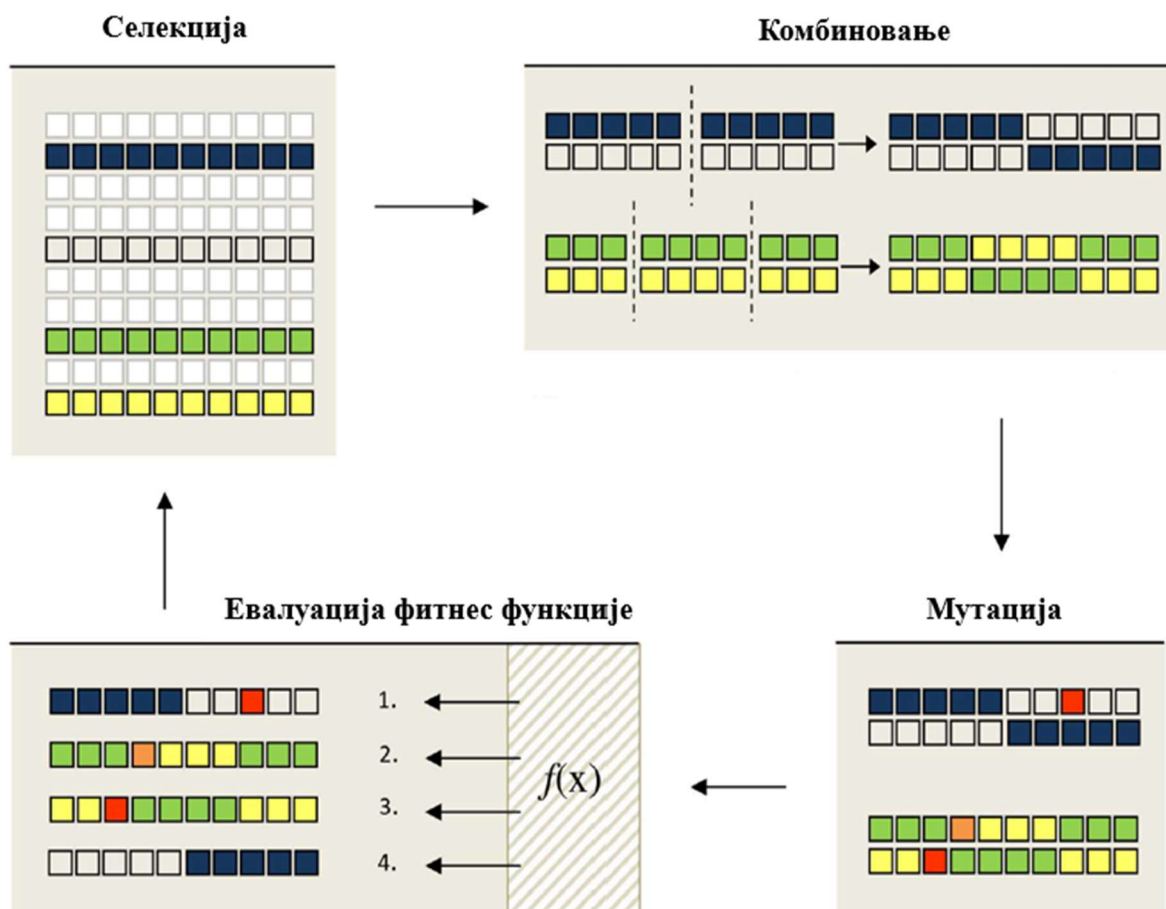
Комбиновање јединки се ради да би се креирале нове јединке, и углавном се ради на једној или две тачке пресека, као што се може видети на слици 5.15.



Слика 5.15. Комбиновање јединки у генетском алгоритму [76]

Мутација додаје ниво насумичности у алгоритам и спречава заглављивање алгоритма у локалном минимуму, тако што мења појединачне гене.

Процес генетског алгоритма може се видети и на слици 5.16.

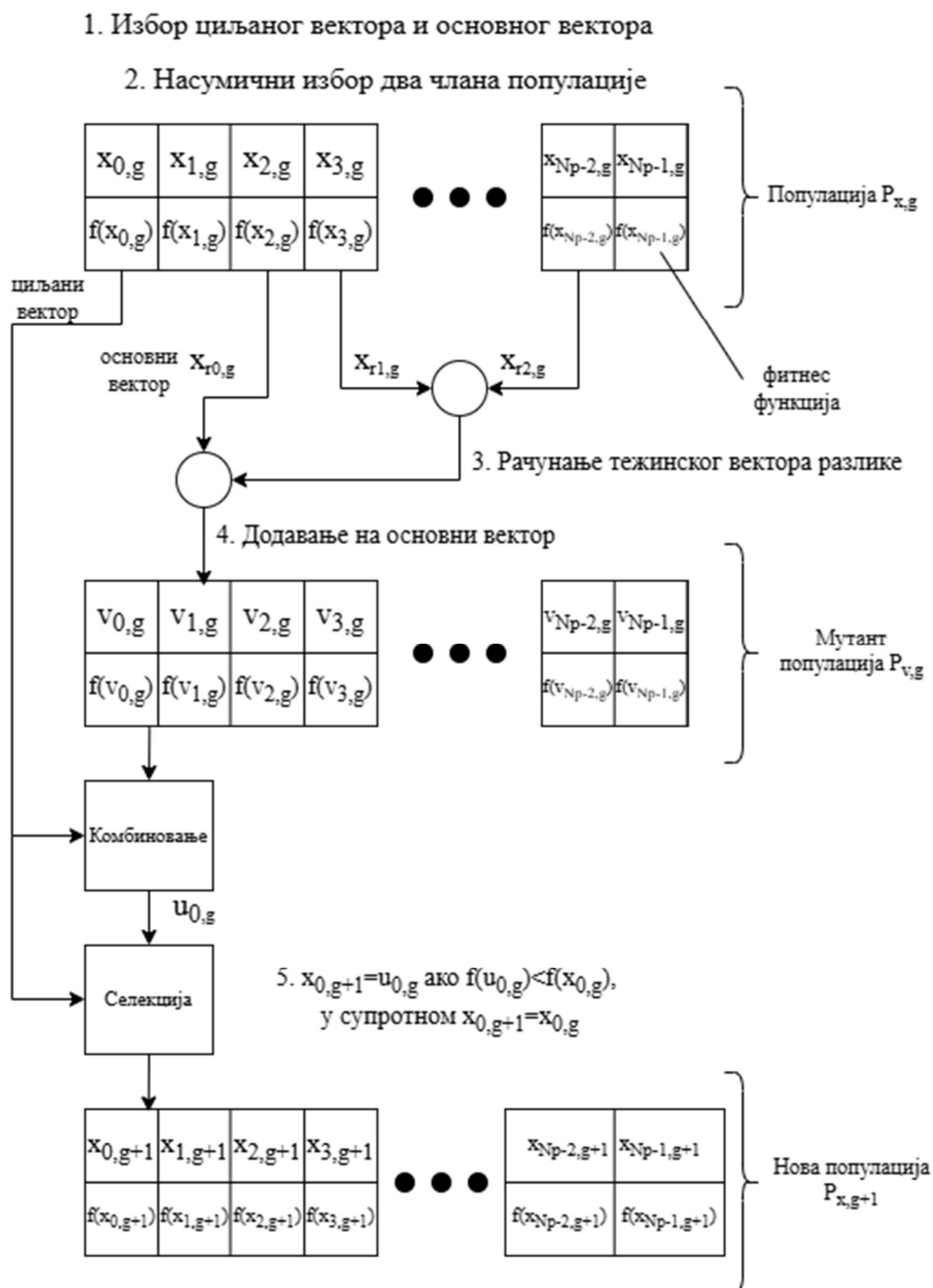


Слика 5.16. Процес генетског алгоритма [77]

5.3.8. Алгоритам диференцијалне еволуције

Алгоритам диференцијалне еволуције (енг. *Differential evolution (DE)*) инспирисан је генетским алгоритмом где се најпре иницијализује популација, а затим се од почетне популације креира мутант популација тако што се изабере основни вектор на који се додаје скалирана (F) разлика друга два насумично одабрана вектора [78]. Затим се за сваки вектор из основне популације ради комбиновање са одговарајућим мутант вектором према CR (*crossover rate*) параметру, и у нову популацију се убацује вектор који има мању фитнес функцију од два поменута. Алгоритам се извршава док се генерише дефинисан број нових генерација, или док се не лоцира минимум.

Скица алгоритма диференцијалне еволуције се може видети на слици 5.17.



Слика 5.17. Алгоритам диференцијалне еволуције

5.3.8.1. Адаптиван алгоритам диференцијалне еволуције са историјом успеха и линеарном редукцијом популације

Адаптиван алгоритам диференцијалне еволуције са историјом успеха и линеарном редукцијом популације (енг. *Linear Population Size Reduction and success-history based adaptive differential evolution (L-SHADE)*) унапређује основни алгоритам диференцијалне еволуције тако што одржава меморију (M_{CR}, M_F) најбољих вредности параметара F и CR диференцијалног алгоритма, а и популације се из генерације у генерацију линеарно смањују и усмеравају ка најбољим решењима. За додатан фактор насумичности додаје се и архива у којој се чувају јединке које нису прешле у следећу генерацију, које учествују у формирању мутант вектора. Ток алгоритма може се видети на слици 5.18., а процес ажурирања меморије се види на слици 5.19. [79]

1. $G = 1, N_G = N^{init}, A$ (архива) = 0
2. Иницијализација популације $P_G = \{x_{1,G}, \dots, x_{N,G}\}$
3. $M_{CR} = 0.5, M_F = 0.5$
4. Док се услови за крај не испуне:
 - (1) $S_{CR} = 0, S_F = 0$
 - (2) За $i = 0, \dots, N$:
 - а. r_i се насумично бира из $[1, H]$
 - б. Ако $M_{CR, r_i} = \perp$, онда $CR_{i,G} = 0$.
 - с. У супротном, $CR_{i,G} = randn_i(M_{CR, r_i}, 0.1)$
 - д. $F_{i,G} = randc_i(M_F, r_i, 0.1)$
 - е. Генерише се $u_{i,G}$ по принципу тренутни_пнајбољи
 - (3) За $i = 0, \dots, N$:
 - а. Ако $f(u_{i,G}) \leq f(x_{i,G})$, онда $x_{i,G+1} = u_{i,G}$
 - б. У супротном, $x_{i,G+1} = x_{i,G}$
 - с. Ако $f(u_{i,G}) < f(x_{i,G})$, онда $x_{i,G} \rightarrow A, CR_{i,G} \rightarrow S_{CR}, F_{i,G} \rightarrow S_F$
 - (4) Подесити величину архиве, ако је потребно обрисати насумичне елементе
 - (5) Ажурирати меморије M_{CR} и M_F
 - (6) Рачунање нове величине популације N_{G+1}
 - (7) Ако $N_G > N_{G+1}$, сортира се популација и брише се вишак елемената, и архива се ставља на исту величину N_{G+1}
 - (8) $G++$

Слика 5.18. Ток L-SHADE алгоритма

Мутант вектор креира се као:

$$v_{i,G} = x_{i,G} + F_i(x_{pbest,G} - x_{i,G}) + F_i(x_{r1,G} - x_{r2,G})$$

где се $x_{pbest,G}$ бира насумично из најбољих $N \times p$ ($p \in [0,1]$) из текуће генерације, а параметар p одлучује баланс између експлоатације и експлорације.

1. Ако $S_{CR} \neq 0$ и $S_F \neq 0$ онда:
 - (1) Ако $M_{CR,k,G} = \perp$ или $\max(S_{CR}) = 0$, онда:
 - a. $M_{CR,k,G+1} = \perp$
 - b. У супротном, $M_{CR,k,G+1} = mean_{WL}(S_{CR})$
 - (2) $M_{F,k,G+1} = mean_{WL}(S_F)$
 - (3) $k++$
 - (4) Ако $k > H$, $k = 1$
 2. У супротном:
 - (1) $M_{CR,k,G+1} = M_{CR,k,G}$
 - (2) $M_{F,k,G+1} = M_{F,k,G}$

$mean_{WL}$ је тежинска Лемерова средина која се рачуна као:

$$mean_{WL} = \frac{\sum_{k=1}^{|S|} \omega_k S_k^2}{\sum_{k=1}^{|S|} \omega_k S_k}$$

$$\omega_k = \frac{\Delta f_k}{\sum_{l=1}^{|S|} \Delta f_l}$$

$$\Delta f = |f(u_{k,G}) - f(x_{k,G})|$$

Слика 5.19. Ажурирање историје у L-SHADE алгоритму

5.3.8.2. Адаптиван алгоритам диференцијалне еволуције са историјом успеха и линеарном редукцијом популације уз рад са ограничењима

Адаптиван алгоритам диференцијалне еволуције са историјом успеха и линеарном редукцијом популације уз рад са ограничењима (енг. *Constrained optimization for Linear Population Size Reduction and success-history based adaptive differential evolution (COLSHADE)*) унапређује основни L-SHADE алгоритам обogaћивањем адаптивним Левијевим летом и динамичком толеранцијом која је укључена у рад са ограничењима [80].

Ограничења се обрађују по принципу изводљивости правила која су поприлично строга, јер индивидуа може да победи уколико крши мање ограничења од осталих, а у случају да има више истих бира се онај који има бољу функцију циља. Не постоји могућност да слабија индивидуа преживи.

Левијеви летови су насумичне шетње са дужином корака симулираном из тешке дистрибуције репа, као што је Левијева дистрибуција вероватноће. У дводимензионалном простору, увиђају се кластери малих корака повезаних спорадичним дужим корацима, док су кластери насумично распоређени у простору. Левијеви летови се уводе у еволуционе алгоритме, као што је случај на пример у алгоритму кукавица које насумично лете до новог гнезда, и остају у њему уколико је боље од претходног. Насумичност кукавица се симулира Левијевим летом.

Ток алгоритма може се видети на слици 5.20., а процес креирања новог вектора Левијевим летом се види на слици 5.21.

1. $N^{init} = round(D \times r^{N^{init}}), |A| = round(N^{init} \times r^{arc})$
2. $N_0 = N^{init}, N_{min} = 4, q_0 = 0.5$
3. $M_{CR} = 0.5, M_F = 0.5, M_{CR_L} = 0.5, M_{F_L} = 0.5$
4. Иницијализација популације $P_0 = \{x_1, \dots, x_{N_0}\}$
5. Постављање иницијалне толеранције, тако да су сва ограничења изводљива

$$\epsilon_{0,j} = \max\{|h_j(x_i)|\}$$
6. Док се услови за крај не испуне:
 - (1) $S_{CR} = 0, S_F = 0, S_{CR_L} = 0, S_{F_L} = 0$
 - (2) $\Delta f = 0, \Delta f_L = 0$
 - (3) За $i = 0, \dots, N_G$:
 - (1) l насумичан број од 0 до 1
 - (2) Ако $l \leq q_G$ онда:
 - (1) CR_i, F_i се добијају позивом методе за генерисање параметара на основу M_{CR_L} и M_{F_L}
 - (2) Генерише се $u_{i,G}$ по принципу Левијевог лета
 - (3) У супротном:
 - (1) CR_i, F_i се добијају позивом методе за генерисање параметара на основу M_{CR} и M_F
 - (2) Генерише се $u_{i,G}$ по принципу тренутни_пнајбољи
 - (4) Ако $f(u_{i,G}) \leq f(x_{i,G})$, онда:
 - (1) $x_{i,G+1} = u_{i,G}$
 - (2) $x_{i,G}$ се ставља у архиву
 - (3) Ако $l \leq q_G$ онда:
 - (1) $CR_i \rightarrow S_{CR_L}, F_i \rightarrow S_{F_L}$
 - (2) Ажурирање Δf_L
 - (4) У супротном:
 - (1) $CR_i \rightarrow S_{CR}, F_i \rightarrow S_F$
 - (2) Ажурирање Δf
 - (4) Ажурирати меморије $M_{CR}, M_F, M_{CR_L}, M_{F_L}$
 - (5) Ажурирање вероватноћа и толеранција
 - (6) $N_{G+1} = round\left[\left(\frac{N^{min} - N^{init}}{MAX_{FE_s}}\right) \times FE_s + N^{init}\right]$
 - (7) Ако $N_G > N_{G+1}$, сортира се популација и брише се вишак елемената, и архива се ставља на исту величину N_{G+1}

Слика 5.20. Ток COLSHADE алгоритма

Свако ограничење има свој вектор толеранције. У свакој генерацији толеранција се ажурира као:

$$\epsilon_{j,G+1} = \begin{cases} \epsilon_{j,G} \times \left(\frac{\epsilon_f}{\epsilon_{j,G}}\right)^{\frac{1}{FE_{S\epsilon} - FE_S}}, & \text{уколико постоји довољан број изводљивих индивидуа} \\ \epsilon_{j,G} & \end{cases}$$

где је ϵ_f финална толеранција за ограничење, а $FE_{S\epsilon}$ максималан број израчунавања фитнес функције.

Параметар q_G одлучује да ли се нове индивидуе које се креирају користе Ливијев лет или не, и рачуна се по формули:

$$q_{G+1} = \begin{cases} q_G, & \text{за } |\Delta f_L| = |\Delta f| = 0 \\ \mu q_G + (1 - \mu) \hat{q}_G & \end{cases}$$

$$\hat{q}_G = \frac{\sum_{i=1}^{|\Delta f_L|} \Delta f_{L_i}}{\sum_{i=1}^{|\Delta f_L|} \Delta f_{L_i} + \sum_{i=1}^{|\Delta f|} \Delta f}$$

где Δf представља унапређење новог решења у односу на индивидуу у текућој генерацији. Унапређење се може гледати из угла фитнес функције или из угла изводљивости.

$$(1) F_{crit} = \sqrt{\frac{(1 - \frac{CR_i}{2})}{N_G}}$$

$$(2) F_i = \max\{F_{crit}, F_i\}$$

$$(3) j_{rand} \text{ је насумичан број у опсегу од } 1 \text{ до } D$$

$$(4) \text{Насумично се бира } x_{\text{пнајбољи}} \text{ из } P_G$$

$$(5) \text{ За } j = 1, \dots, D:$$

$$(1) \text{ Ако је насумичан број из опсега } 0 \text{ до } 1 \leq CR \text{ или } j = j_{\text{насумично}} \text{ онда:}$$

$$(1) F_L = F_i S(\alpha, \beta, \gamma, \gamma + \delta)$$

$$(2) u_j = x_j + F_L(x_{\text{пнајбољи},j} - x_j)$$

$$(2) \text{ У супротном } u_j = x_j$$

Слика 5.21. Креирање нових решења у COLSHADE алгоритму Левијевим летом

6. Метакхеуристика хиперболичког синуса и косинуса и њена модификација

Метакхеуристике налазе инспирацију у многим областима, а једна од њих је и математика. Тако је хиперболичка тригонометрија изнедрила метакхеуристику која користи хиперболички синус и косинус (енг. *A sinh cosh optimization (SCHO)*) [81] у својим израчунавањима позиција решења.

Алгоритам пролази кроз неколико фаза у свом извршавању од којих се најпре врши иницијализација кандидат решења.

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,dim} \\ \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,dim} \end{bmatrix}$$

X представља насумична кандидат решења, којих има N , а свако решење је димензије dim . Свако решење добија се као $x = rand(N, dim) \times (ub - lb) + lb$, где је $rand$ насумичан број из опсега $[0,1]$, ub и lb горња и доња граница за тражену променљиву.

Током рада алгоритам балансира између експлорације и експлоатације на основу параметра A , који се рачуна као:

$$A = (p - q \times (\frac{t}{\max_iteration})^{\frac{\cosh \frac{t}{\max_iteration}}{\sinh \frac{t}{\max_iteration}}}) \times r_{13}$$

r_{13} је насумичан број из опсега $[0,1]$, а p и q су коефицијенти који износе 10 и 9. За $A > 1$ (на почетку алгоритма) извршава се експлорација, а за $A < 1$ (пред крај алгоритма) експлоатација.

У фази експлорације постоје две фазе које се смењују на основу параметра

$$T = floor(\frac{\max_iteration}{ct})$$

ct је 3,6. Ако је тренутни број итерација $t < T$, позиције кандидат решења се ажурирају на следећи начин:

$$x_{i,j}^{t+1} = \begin{cases} x_{best}^j + r_1 \times W_1 \times x_{i,j}^t, & r_2 > 0.5 \\ x_{best}^j - r_1 \times W_1 \times x_{i,j}^t, & r_2 < 0.5 \end{cases}$$

$$W_1 = r_3 \times a_1 \times (\cosh r_4 + u \times \sinh r_4 - 1)$$

$$a_1 = 3 \times (-1.3 \times \frac{t}{\max_iteration} + m)$$

r_1, r_2, r_3 и r_4 су насумични бројеви из опсега $[0,1]$, a монотono опадајућа функција, u је 0.3888 и m је 0.45. Ако је тренутни број итерација $t > T$, позиције кандидат решења се ажурирају на следећи начин:

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^t + |\varepsilon \times W_2 \times x_{best}^j - x_{i,j}^t|, & r_5 > 0.5 \\ x_{i,j}^t - |\varepsilon \times W_2 \times x_{best}^j - x_{i,j}^t|, & r_5 < 0.5 \end{cases}$$

$$W_2 = r_6 \times a_2$$

$$a_2 = 2 \times \left(-\frac{t}{\max_iteration} + n \right)$$

r_5 и r_6 су насумични бројеви из опсега $[0,1]$, a монотono опадајућа функција, ε је 0.003 и n је 0.5.

У фази експлоатације постоје две фазе које се смењују на основу већ дефинисаног параметра T . Ако је тренутни број итерација $t < T$, позиције кандидат решења се ажурирају на следећи начин:

$$x_{i,j}^{t+1} = \begin{cases} x_{best}^j + r_7 \times W_3 \times x_{i,j}^t, & r_8 > 0.5 \\ x_{best}^j - r_7 \times W_3 \times x_{i,j}^t, & r_8 < 0.5 \end{cases}$$

$$W_3 = r_9 \times a_1 \times (\cosh r_{10} + u \times \sinh r_{10})$$

r_7, r_8, r_9 и r_{10} су насумични бројеви из опсега $[0,1]$, a монотono опадајућа функција и u је 0.3888. Ако је тренутни број итерација $t > T$, позиције кандидат решења се ажурирају на следећи начин:

$$x_{i,j}^{t+1} = x_{i,j}^t + r_{11} \times \frac{\sinh r_{12}}{\cosh r_{12}} |W_2 \times x_{best}^j - x_{i,j}^t|$$

r_{11} и r_{12} су насумични бројеви из опсега $[0,1]$.

Како би се додатно истражио потенцијали простор примењује се стратегија ограничене претраге у случају да је тренутни број итерација једнак BS_k .

$$BS_1 = \text{floor}\left(\frac{\max_iteration}{\beta}\right)$$

β је 1,55. Следећи улазак у ограничену претрагу је у тренутку $BS_{k+1} = BS_k + \text{floor}\left(\frac{\max_iteration - BS_k}{\alpha}\right)$, где је α једнако 4,6.

Унутар ограничене претраге границе се рачунају као:

$$ub_k = x_{best}^j + \left(1 - \frac{t}{\max_iteration}\right) \times |x_{best}^j - x_{second}^j|$$

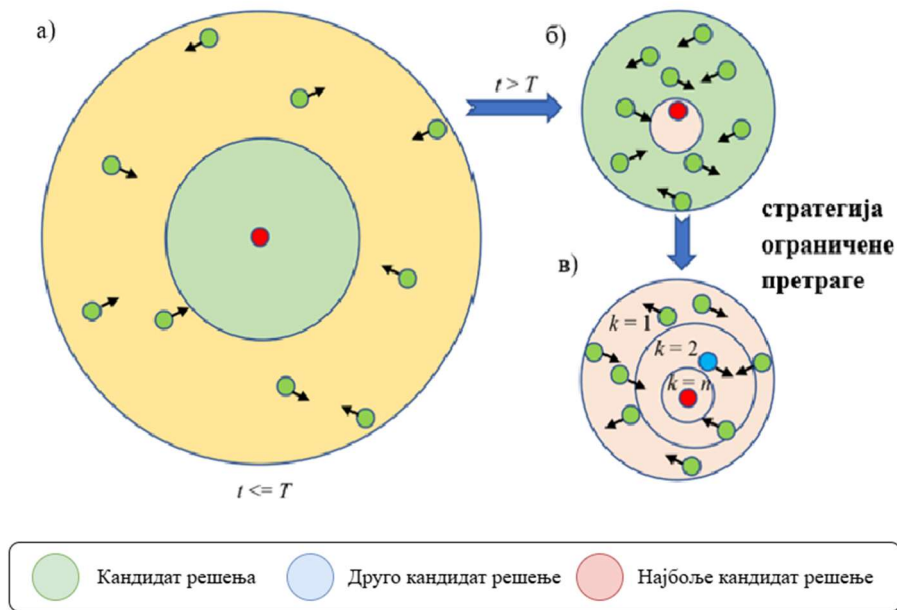
$$lb_k = x_{best}^j - \left(1 - \frac{t}{\max_iteration}\right) \times |x_{best}^j - x_{second}^j|$$

Унутар самог простора кандидат решења се насумично генеришу као у фази иницијализације.

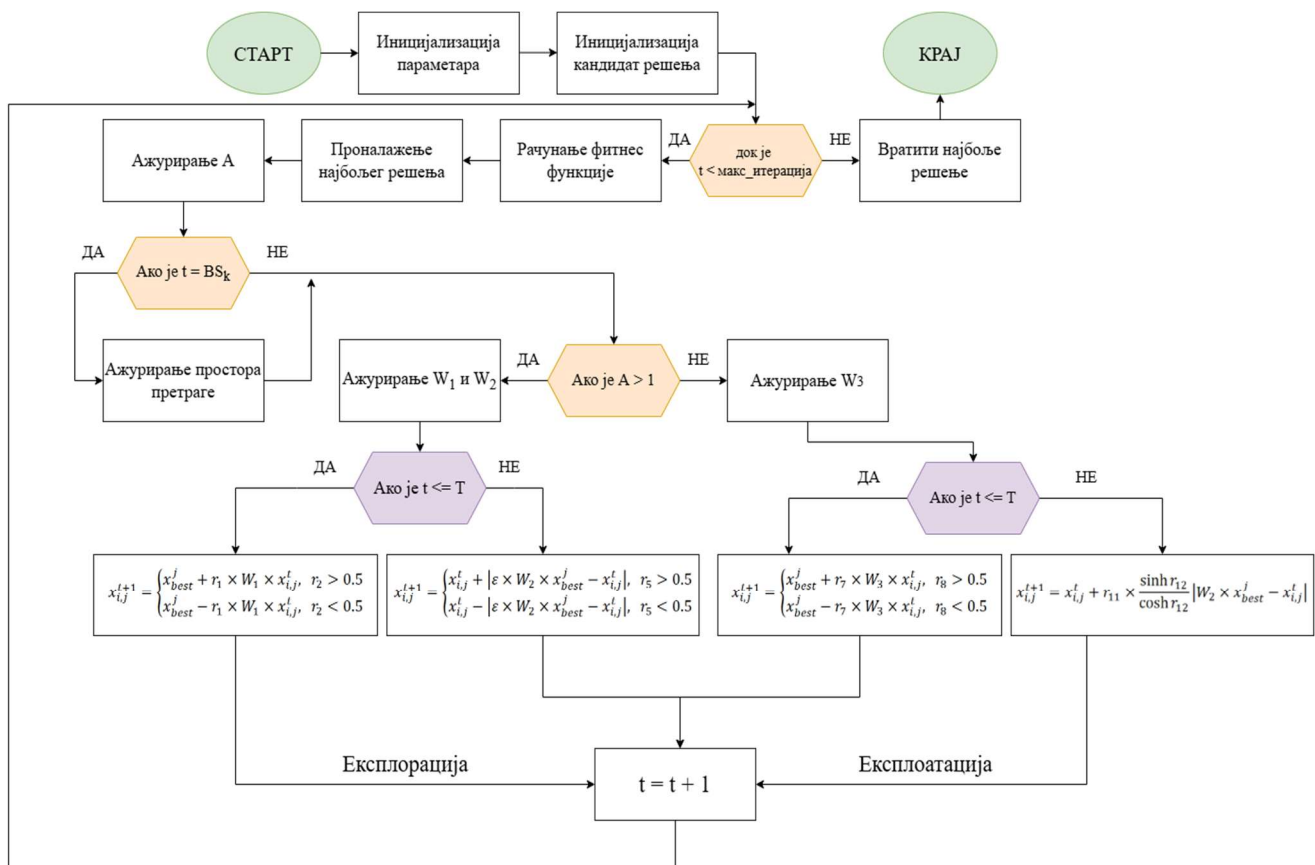
На слици 6.1. приказан је шематски дијаграм алгоритма, где се под а) види прва фаза експлорације и експлоатације, под б) друга фаза експлорације и експлоатације и под в) фаза ограничене претраге.

На слици 6.2. приказан је детаљан дијаграм тока алгоритма.

На слици 6.3. приказан је псеудокод алгоритма.



Слика 6.1. Шематски дијаграм SCHO алгоритма [81]



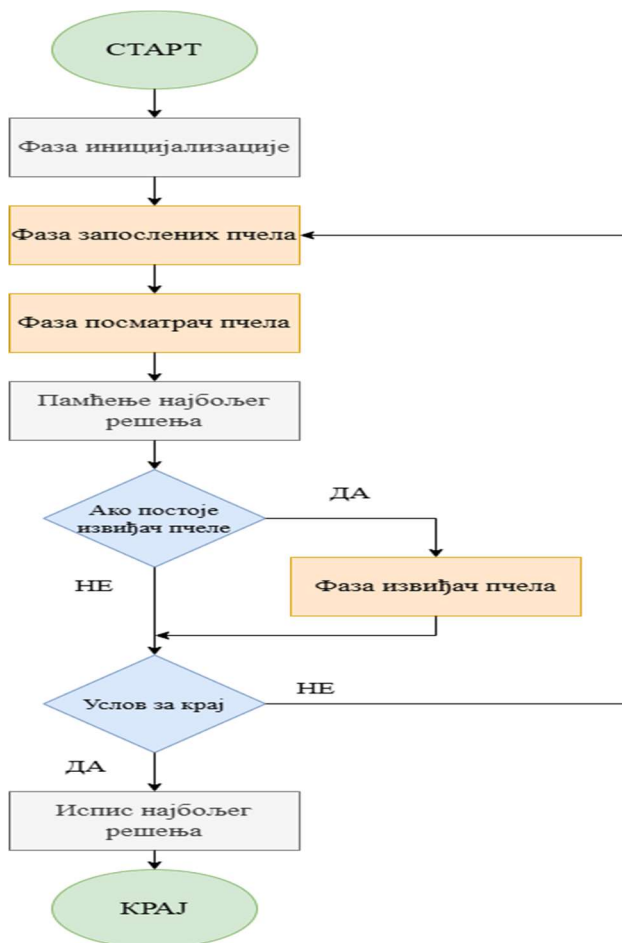
Слика 6.2. Дијаграм тока SCHO алгоритма

- (1) Иницијализовање параметара $ct, T, BS_1, u, m, n, \alpha, \beta, p, q$
- (2) Иницијализовање кандидат решења $X: i = 1, \dots, N$
- (3) Док $t < \text{максималног броја итерација}$:
 - (1) Рачунање фитнес функције кандидат решења
 - (2) Налажење најбољег решења
 - (3) За $i = 1, \dots, N$:
 - (1) За $j = 1, \dots, \text{димензија проблема}$:
 - (1) Ажурира се A
 - (2) Ако је $t = BS_k$:
 - (1) Налази се друго најбоље решење
 - (2) Ажурира се BS_k
 - (3) Ажурира се простор претраге
 - (3) Ако је $A > 1$:
 - (1) Улази се у фазу експлорације
 - (2) Ажурирају се W_1 и W_2
 - (3) Ако је $t \leq T$:
 - а. Ажурирање позиција кандидат решења по формули за прву фазу експлорације
 - (4) У супротном:
 - а. Ажурирање позиција кандидат решења по формули за другу фазу експлорације
 - (4) У супротном:
 - (1) Улази се у фазу експлоатације
 - (2) Ажурира се W_3
 - (3) Ако је $t \leq T$:
 - а. Ажурирање позиција кандидат решења по формули за прву фазу експлоатације
 - (4) У супротном:
 - а. Ажурирање позиција кандидат решења по формули за другу фазу експлоатације
 - (4) $t++$
- (4) Враћа се најбоље решење

Слика 6.3. Псеудокод SCHO алгоритма

Како би се основни алгоритам било које метахеуристике унапредио у контексту ефикасности и проналажења бољег решења, у литератури се могу наћи хибридна решења која комбинују већ постојеће методе у нову модификовану метахеуристику. Уз основни SCHO алгоритам употребљене су још две технике које често оплемењују и друге алгоритме, а у питању су:

1. **Вештачка колонија пчела** [82] [83] састоји се од три врсте пчела: запослених, посматрача и извиђача. Пола колоније се састоји од запослених пчела, а пола од посматрача. Свака запослена пчела има један извор хране, а квалитет тог извора хране је одређен количином нектара у том извору. На почетку алгоритма све запослене пчеле долазе до својих извора хране и одређују количину нектара. Ове информације се деле са пчелама посматрачима које бирају извор хране на основу рулет точка, где бољи извори имају веће изгледе да буду одабрани. Уколико се одређени извори не побољшају после дефинисаног интервала, они се напуштају од стране запослене пчеле и она постаје извиђач. Извиђачи су у потрази за новим изворима хране и немају дефинисано вођство већ насумично долазе до бољих или лошијих решења. Ток алгоритма вештачке колоније пчела може се видети на слици 6.4. Вештачка колонија пчела користиће се у модификованој верзији SCHO алгоритма, тако што ће се одбацивати она кандидат решења која се нису унапредила после две итерације.



Слика 6.4. Вештачка колонија пчела

2. **Квази-рефлексивно учење** [84] засновано је на схватању у психологији да сложени проблеми постоје и да је знање о њима прожето неизвесношћу. Овакво резонување послужиће за проширење иницијалне популације кандидат решења новим решењима, чиме се повећава процес експлорације и на тај начин се спречава да почетна решења остану заглављена у субоптималним доменама.

Иницијална популација X се проширује и квази-рефлективном популацијом у којој се кандидат решења дефинишу као:

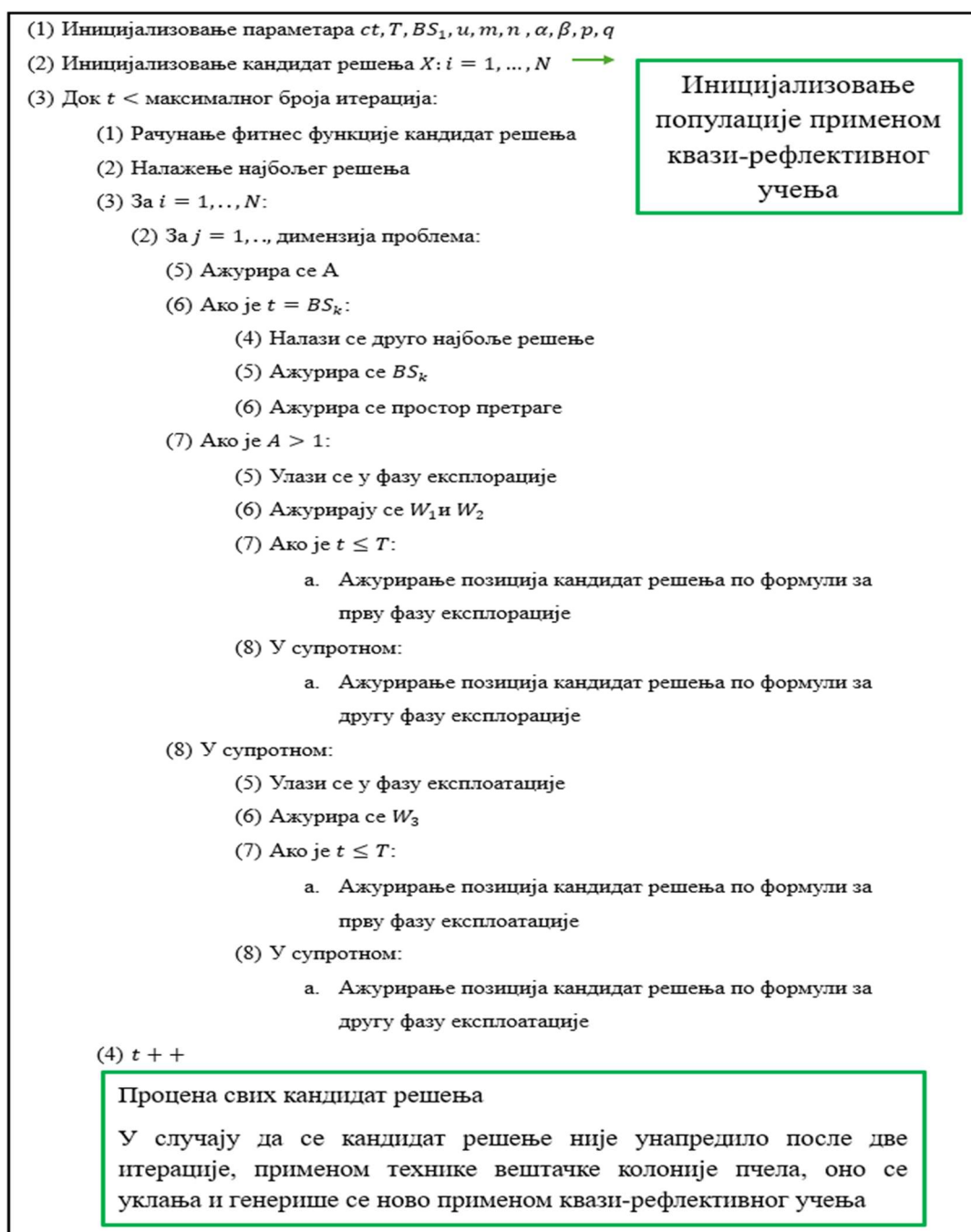
$$X^{qr} = rand(\frac{lb + ub}{2}, x)$$

$rand$ генерише насумичан број у задатом интервалу.

Затим се врши унија X и X^{qr} и евалуација свој почетних кандидат решења, и коначну почетну популацију креирају најбољих N кандидата, чиме се почетна популација већ у самом почетку приближила бољим решењима.

Такође када се неко решење одбаци по алгоритму вештачке колоније пчела, ново решење које би заправо нашле пчеле извиђачи се генерише по квази-рефлективном учењу.

Псеудокод модификованог SCHO (MSCHO) приказан је на слици 6.5.



Слика 6.5. Модификовани SCHO алгоритам

7. Експерименти и резултати

У овом поглављу ће бити детаљно описани коришћени скупови података (скуп података о тапкању прстима и скуп података о ходу), спроведени експерименти и добијени резултати, као и статистичка доказивања резултата и потврђивање хипотеза.

7.1. Коришћени скупови података

Приликом спровођења експеримента сигурност у резултате се повећава када се користе различити скупови података, па се у овој дисертацији користе два скупа података и то скуп података о тапкању прстима, као и скуп података о ходу. Детаљан приказ оба скупа података дат је у следећим потпоглављима.

7.1.1. Скуп података о тапкању прстима

Скуп података настао је на Клиници за неурологију у Београду [54]. Приликом прикупљања података, посматрана су 54 учесника, од којих 14 има Паркинсонову болест, 16 има прогресивну супрануклеарну парализу, 13 има мултиплу системску атрофију, док 11 нема неко неуролошко обољење. Из посматрања су искључени учесници који имају онеспособљене екстремитете, као и учесници који су се у прошлости професионално бавили неким послом којим су ојачали руке, конкретно шаке (атлетичари, музичари и слично). Свим учесницима је углавном десна страна тела више захваћена обољењем.

Приказ демографских података учесника дат је у табели 6.

Табела 6. Демографски подаци учесника за скуп података о тапкању прстима

	Године	Пол [Ж/М]	Дужина трајања болести у годинама
ПБ	62.1 ± 9.4	4/10	4.9 ± 4.5
ПСП	67.1 ± 8.9	5/11	5.23 ± 2.3
МСА	58.4 ± 4.8	9/4	3.47 ± 1.5
Здрави	55 ± 8.4	8/3	-

Критеријуми за утврђивање постојања неуродегенеративног обољења су преузети из следећих студија:

- **За Паркинсонову болест:** Тачност клиничке дијагнозе идиопатске Паркинсонове болести: клиничко-патолошка студија 100 случајева [85],
- **За прогресивну супрануклеарну парализу:** Тачност и поузданост прелиминарних НИНДС неуропатолошких критеријума за прогресивну супрануклеарну парализу и сродне поремећаје [86],
- **За мултиплу системску атрофију:** Други консензусни став о дијагнози мултипле системске атрофије [87].

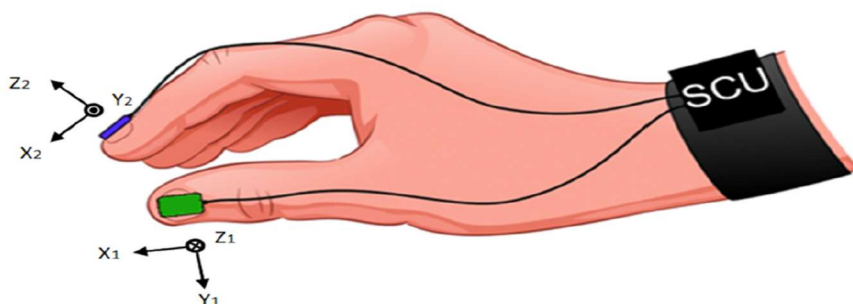
За учеснике са Паркинсоновом болешћу, тестови су рађени ујутру након што је прошло више од 12 сати након узимања терапије.

Приликом прикупљања података коришћена су два инерциона сензора, од којих сваки има троосни жирокоп и акцелерометар. Ови сензори су накачени на врхове кажипрста и палца, и повезани су са централном сензорском јединицом која је прикачена на надлактицу учесника. Приликом касније класификације учесника, коришћени су само подаци прикупљени са жирокопа.

Учесници су требали да се сместе, седећи опуштено и да што је брже могуће и са што већим распоном покрета спајају врх палца и кажипрста десне руке. Задатак је трајао 15 секунди, а између сваког покушаја је прављена пауза од једног минута, како би се учесници одморили. Сваки пацијент је имао по шест покушаја и сваки од њих је поред прикупљених података са сензора и видео документован. Такође, за сваки покушај је покушано да траје тачно 15 секунди, али је у неким случајевима долазило до одступања. Ниједан покушај није трајао краће од 11 секунди и дуже од 20 секунди.

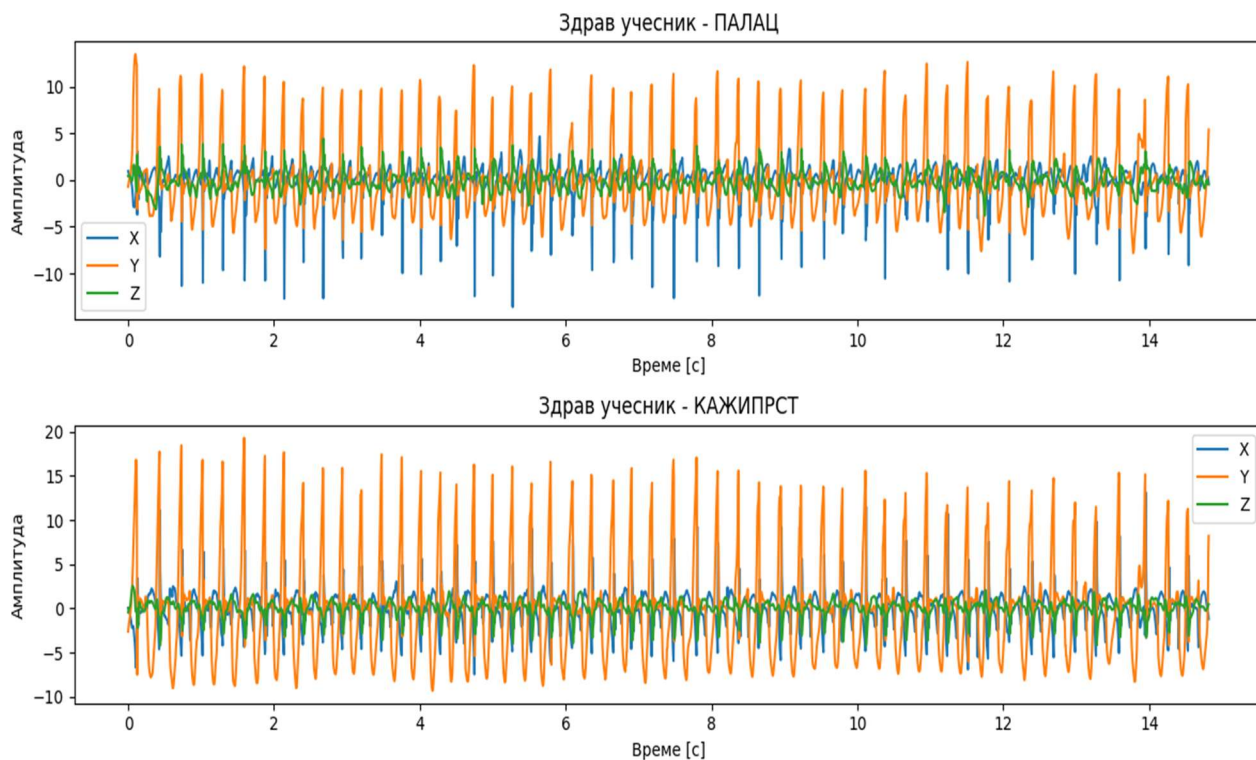
Неки учесници нису били у стању да заврше свих шест покушаја, због замора па је за те учеснике број покушаја смањен. Такође неки покушаји су искључени из финалног скупа због постојања шума и генерално лошег квалитета података. На крају укупан број покушаја који се налази у скупу података је 268, а подаци су прикупљени на фреквенцији од 200 Hz у 12-тобитној резолуцији,

На слици 7.1. види се приказ прикачених сензора на шаку учесника.

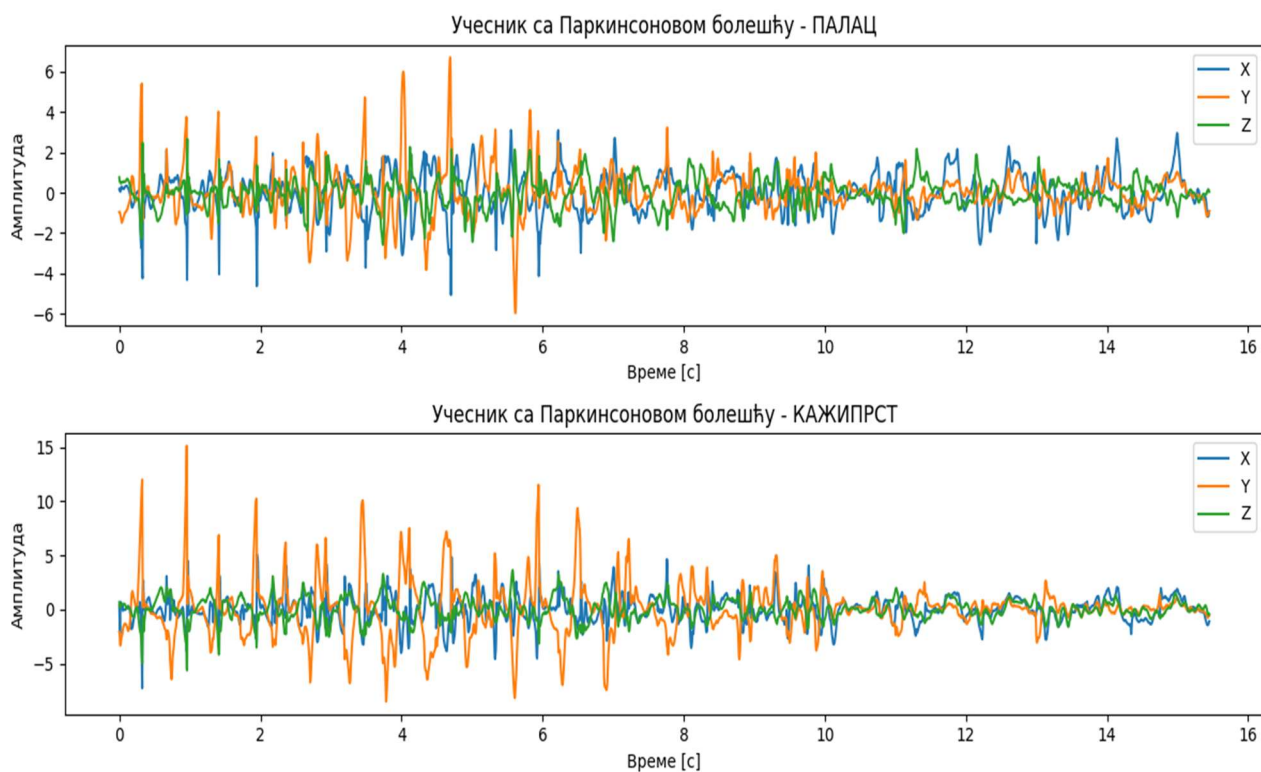


Слика 7.1. Скуп података о тапкању прстима: Сензори прикачени на шаку учесника [54]

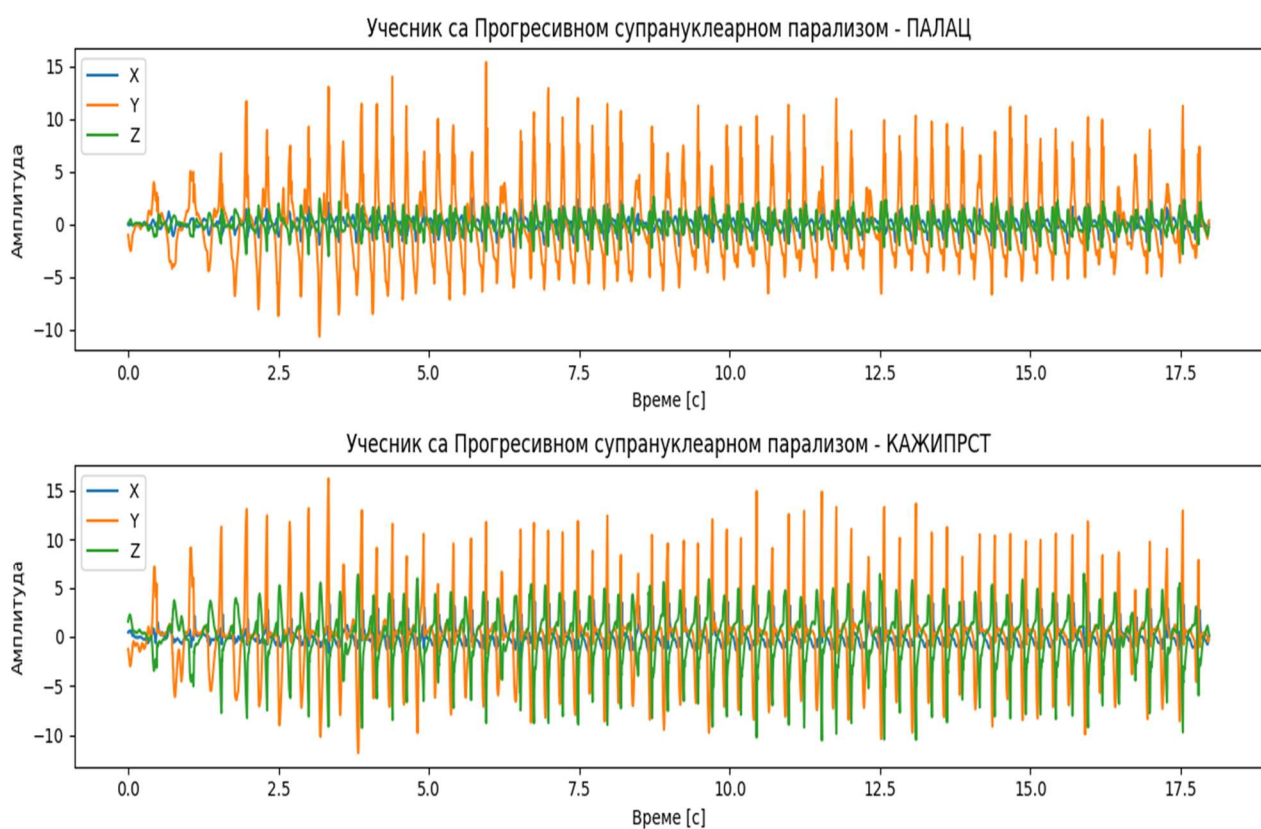
На наредним сликама (7.2., 7.3., 7.4. и 7.5.) приказани су подаци са жироскопа за палац и кажипрст за по један снимак покушаја за сваку групацију учесника.



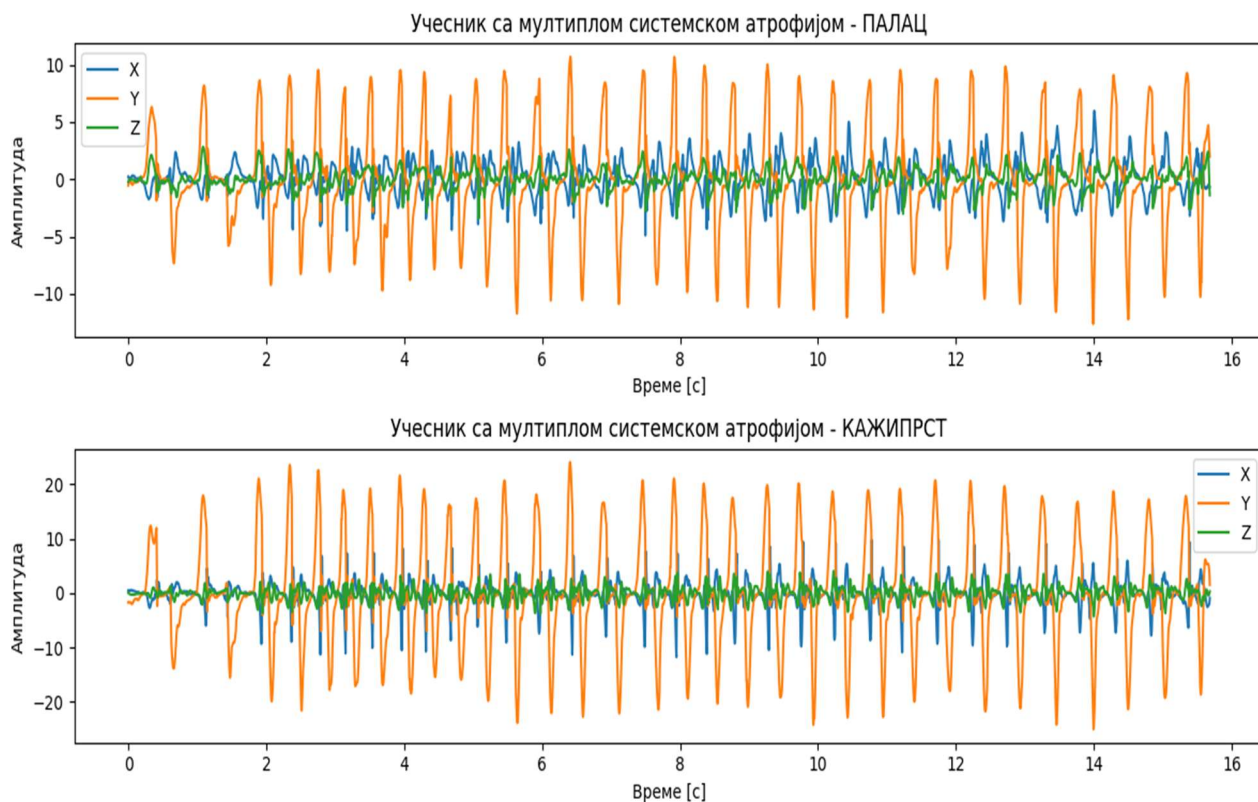
Слика 7.2. Скуп података о тапкању прстима: Подаци са жироскопа за здраву особу



Слика 7.3. Скуп података о тапкању прстима: Подаци са жироскопа за особу са Паркинсоновом болешћу



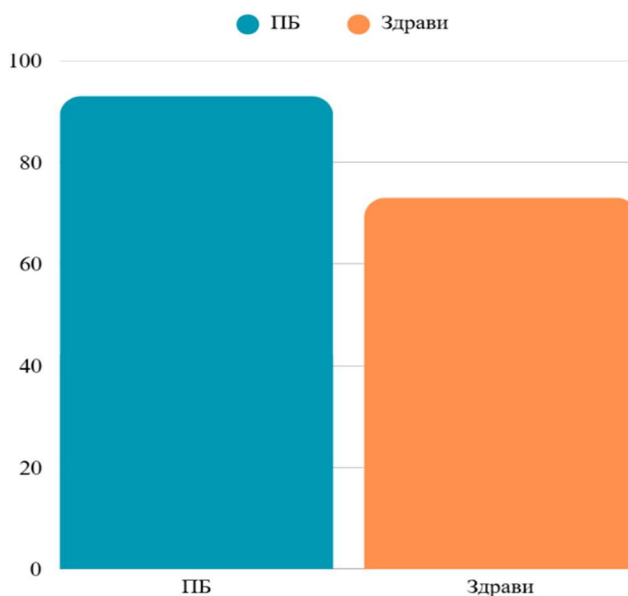
Слика 7.4. Скуп података о тапкању прстима: Подаци са жироскопа за особу са прогресивном супрануклеарном парализом



Слика 7.5. Скуп података о тапкању прстима: Подаци са жироскопа за особу са мултиплом системском атрофијом

7.1.2. Скуп података о ходу

Скуп података садржи информације о ходу за 93 учесника са дијагностикованом идиопатском Паркинсоновом болешћу и 73 здраве особе (расподела се види на слици 7.6.). Подаци представљају вертикалне записе силе реакције на тло, док су учесници ходали својим уобичајеним темпом отприлике два минута по равном тлу.



Слика 7.6. Скуп података о ходу: расподела учесника

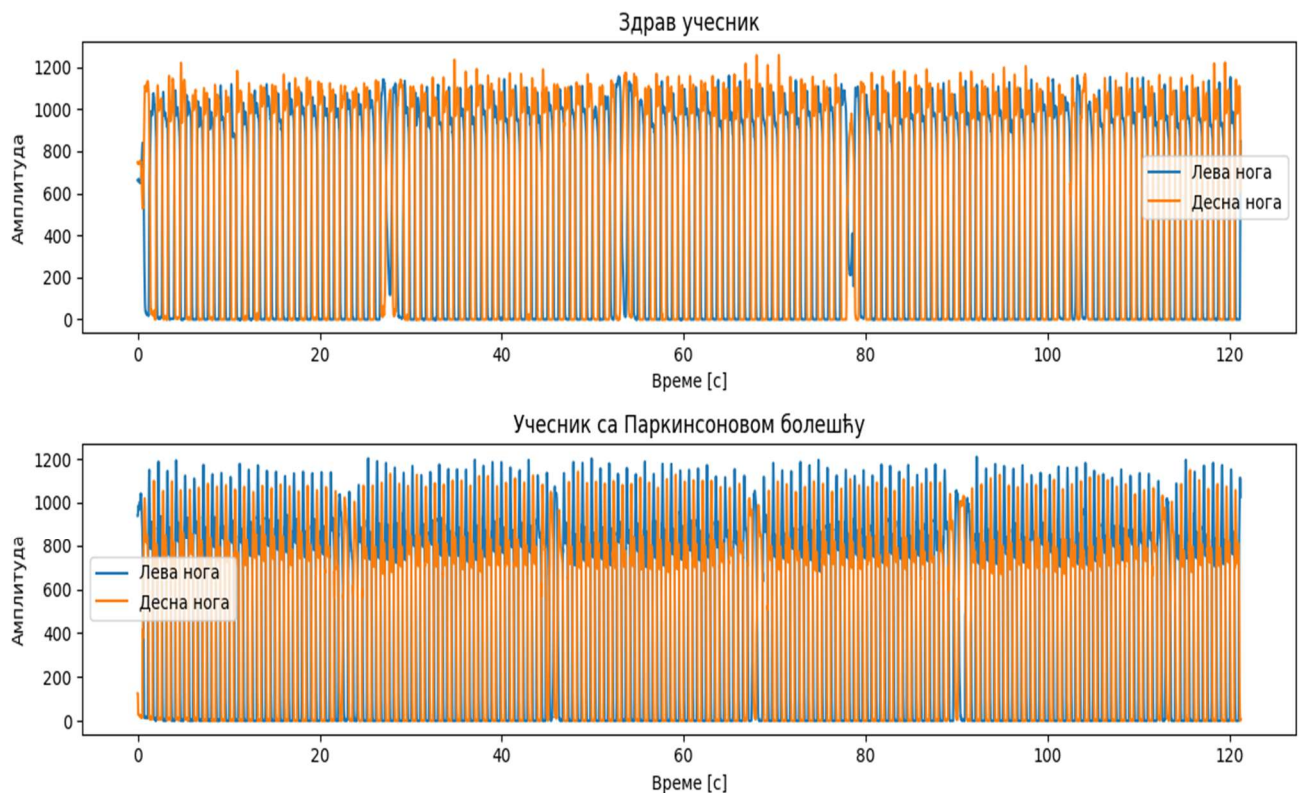
Приказ демографских података учесника дат је у табели 7.

Табела 7. Демографски подаци учесника за скуп података о ходу

Студија	Учесници	Пол [м/ж]	Године (просек)	Висина (просек)	Тежина (просек)
[88]	ПБ - 29	20/9	71.1	1.689	73.81
	Здрави - 18	10/8	71.55	1.673	72.22
[89]	ПБ - 29	16/13	67.2	1.664	70.27
	Здрави - 26	12/14	64.57	1.689	71.5
[90]	ПБ - 35	22/13	61.57	1.671	73.05
	Здрави - 29	18/11	57.93	1.683	74.28

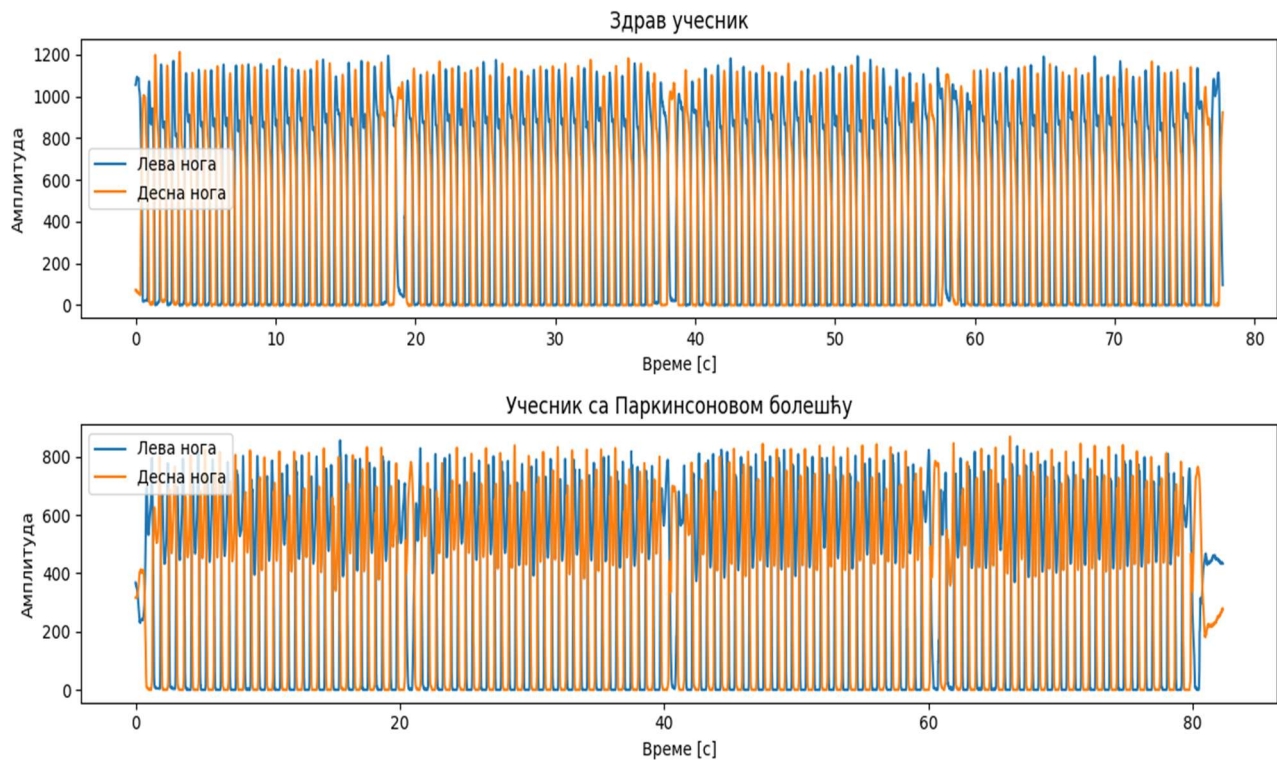
Учесници испод сваког стопала имају по 8 сензора који мере силу у Њутнима, као функцију времена. Излаз сваког од ових 16 сензора је дигитализован и снимљен по фреквенцији од 100 узорака у секунди, а записи такође укључују два сигнала који представљају збир излаза 8 сензора за свако стопало.

На слици 7.7. се види кретање сума по левој и десној ноzi кроз време, за здраву особу и особу са Паркинсоновом болешћу у оквиру студије [88].



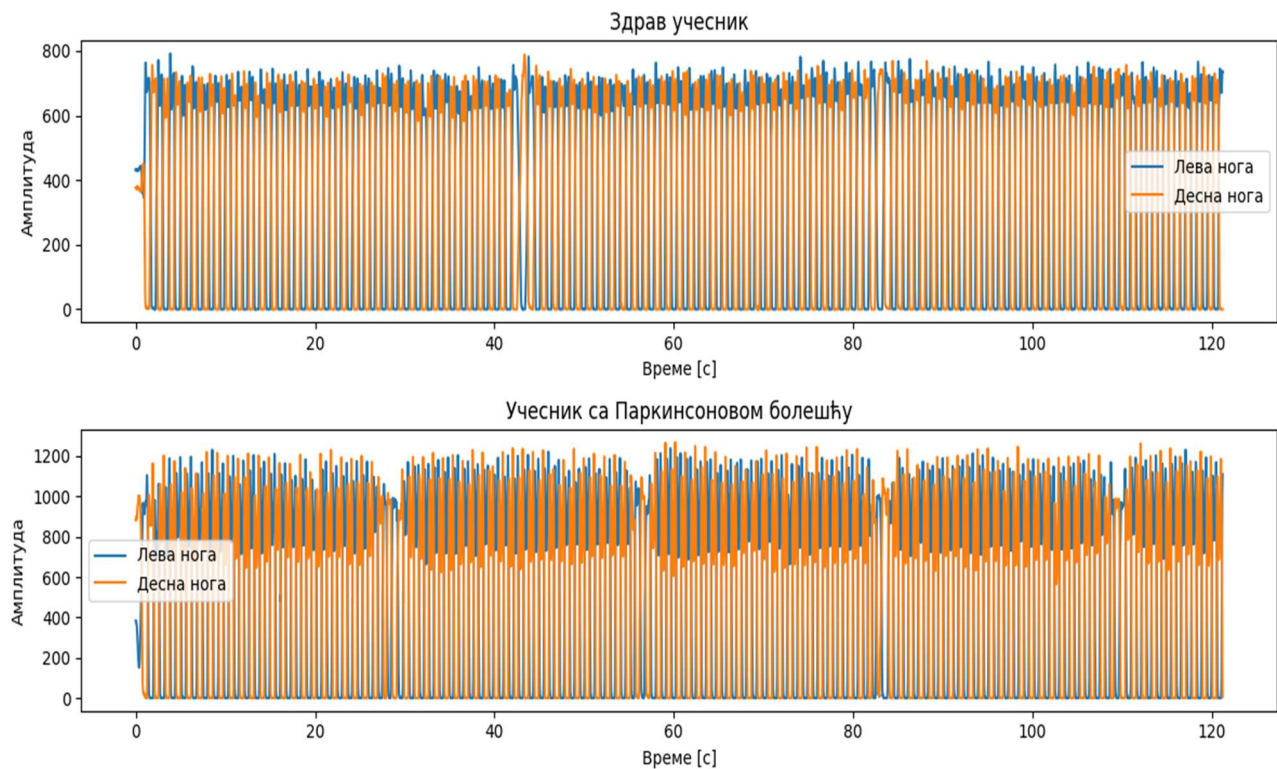
Слика 7.7. Скуп података о ходу: Примери из прве студије

На слици 7.8. се види кретање сума по левој и десној ноzi кроз време, за здраву особу и особу са Паркинсоновом болешћу у оквиру студије [89].



Слика 7.8. Скуп податак о ходу: Примери из друге студије

На слици 7.9. се види кретање сума по левој и десној ноzi кроз време, за здраву особу и особу са Паркинсоновом болешћу у оквиру студије [90].

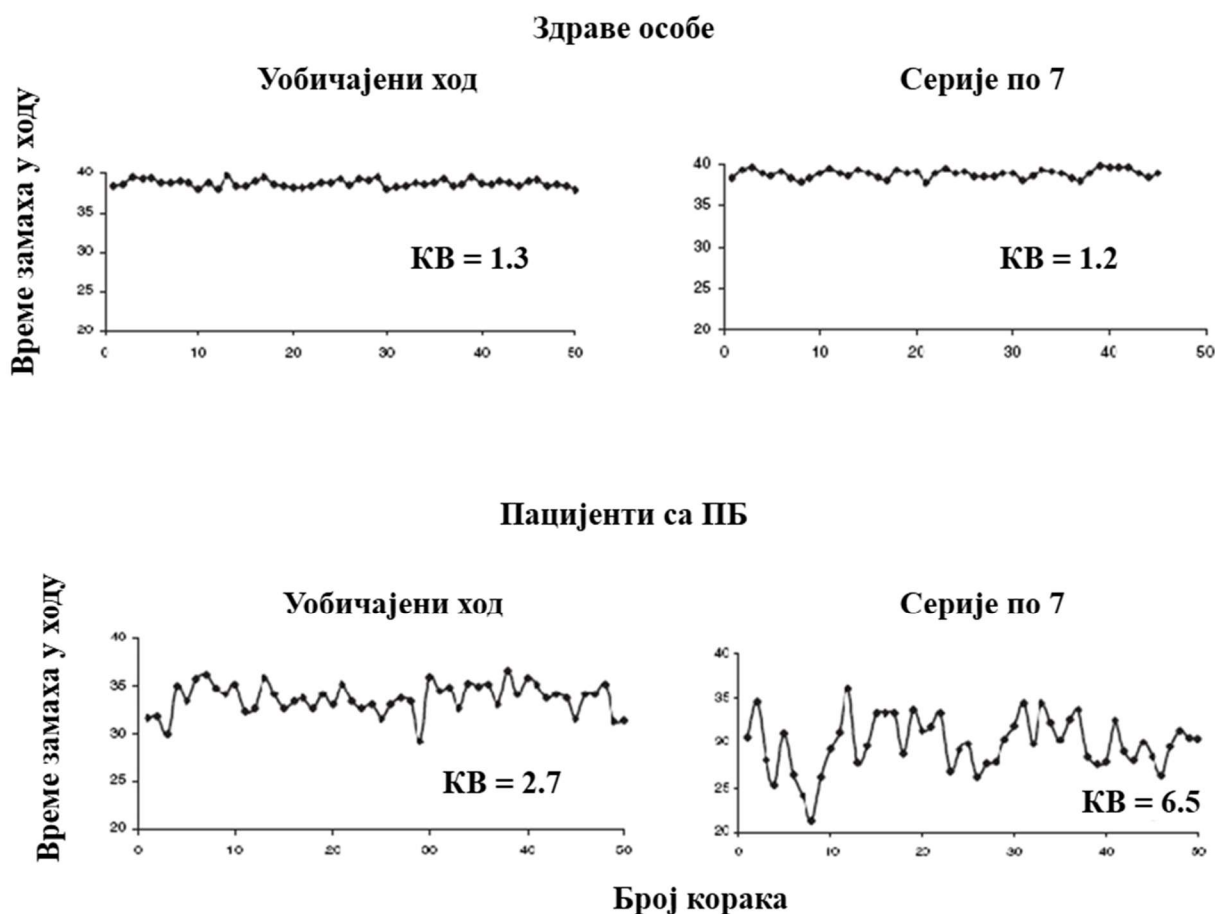


Слика 7.9. Скуп података о ходу: Примери из треће студије

Поред основног задатка где су учесници ходали по равном тлу, један део скупа података односи се на учеснике који су поред ходања спроводили и ментални тест који је подразумевао одузимање од 100 у серијама по 7.

Са слике 7.10. може се приметити да код здравих особа додатни ментални тест не прави скоро никакву разлику, док се код особа са Паркинсоновом болешћу примећују значајно веће флукутације у времену замаху у ходу. Ход делимо на фазу замаху и фазу ослонца, где се у фази ослонца стопало ослања на тло, а у фази замаху је стопало у ваздуху и прави замах при кораку.

Коефицијент варијације (КВ) (енг. *Coefficient of Variation (CV)*) у случају здраве особе при ходању је 1.3%, а уз додати ментални тест је 1.2%. У случају особа са Паркинсоновом болешћу коефицијент варијације при ходању је 2.7%, док се при додатом менталном тесту вредност повећава на 6.5%.



Слика 7.10. Скуп података о ходу: Одузимање у серијама по 7 [88]

7.2. Експерименти

У оквиру ове тезе вршено је више експеримената од којих су три експеримента вршена над скупом података о тапкању прстима, док је још један експеримент рађен над скупом података о ходу, како би се показала применљивост и на друге скупове. У свим експериментима врши се класификација, тачније предикција крајњих класа. У три од четири експеримента врши се бинарна класификација између здраве особе и особе са Паркинсоновом болешћу, док се у једном експерименту ради мултикласна класификација између здраве особе, особе са Паркинсоновом болешћу, особе са мултиплом системском атрофијом и особе са прогресивном супрануклеарном парализом.

У оквиру свих експеримената *MSCHO* поређена је са већ постојећим метахеуристикама: *SCHO*, *GA*, *PSO*, *FA*, *WOA*, *BSO* и *COLSHADE*. Као циљна функција коришћен је степен грешке који је дефинисан као:

$$\text{степен грешке} = 1 - \text{тачност}$$

Поред циљне функције која користи степен грешке, коришћена је и додатна индикатор функција која је максимизовала Коенов капа коефицијент по формули:

$$\kappa = \frac{z_0 - z_e}{1 - z_e}$$

где је z_0 предвиђена, а z_e тачна вредност.

У наставку овог поглавља ће бити детаљно наведени резултати за све експерименте, и то који скуп података се користи, шта се предвиђа, које особине података долазе у modele, како су дељени подаци, који модел се користи, који хиперпараметри модела се подешавају, као и сами резултати симулација.

За све експерименте коришћен је Python (верзија: 3.12.9.), окружење Visual Studio Code (верзија: 1.100.0), на машини са Windows 10 оперативним системом, Intel i9-9900 процесором, са 64 GB радне меморије.

7.2.1. Експеримент1: Рекурентне неуралне мреже

Скуп података: Скуп података о тапкању прстима

Класификација: Бинарна класификација (здрава особа/Паркинсонова болест)

Особине података: Подаци су подељени у временске серије, где је за сваку временску серију дефинисано шест особина (по 3 вредности за 3 осе жироскопа, за оба прста), и свих шест особина улази у модел, а подаци су пре уласка у модел нормализовани

Подела података: Подаци су подељени у односу 70% за тренирање, 30% за тестирање, приликом чега већина записа у скупу за тестирање припада особама које скуп за тренирање није никад видео

Модел: Рекурентна неурална мрежа, чија се архитектура састоји од улазног слоја, затим једног или два рекурентна слоја, и једног излазног слоја. За имплементацију модела коришћен је *Keras API*, који је подигнут над *Tensorflow*-ом.

Хиперпараметри модела: Подешавани хиперпараметри модела су:

- Број рекурентних слојева у мрежи: [1,2],
- Број неурона у сваком слоју: [5,15],
- Фактор одбацивања: [0.05,0.2],
- Стопа учења: [0.0001,0.01],
- Број епоха: [30,60].

Резултати симулације: Табела 8 приказује резултате функције циља за најбоље, најгоре и средње покретање (по 30 покретања) модела за сваку од поређених метахеуристика. Уз то је приказана и медијана функције циља, као и стандардна девијација и варијанса. Табела 9 приказује исте податке за индикатор функцију.

Табела 8. Експеримент1: Рекурентне неуралне мреже: Резултати функције циља

	Најбоље	Најгоре	Средње	Медијана	σ	s
BSO	0.1007345	0.2224554	0.1834558	0.187828	0.0277585	0.000770534
PSO	0.0818468	0.2612802	0.212277	0.218258	0.0353317	0.001248329
FA	0.1458552	0.2581322	0.2126268	0.219832	0.0311908	0.000972866
GA	0.1364113	0.2528856	0.2068905	0.20829	0.024649	0.000607576
MSCHO	0.0640084	0.2392445	0.1809374	0.185729	0.034149	0.001166155
WOA	0.1038825	0.2360965	0.1958027	0.200944	0.0288421	0.000831864
SCHO	0.1448059	0.2455404	0.2017838	0.199895	0.0229097	0.000524853
COLSHADE	0.115425	0.2539349	0.1932494	0.198321	0.0350731	0.001230119

Табела 9. Експеримент1: Рекурентне неуралне мреже: Резултати индикатор функције

	Најбоље	Најгоре	Средње	Медијана	σ	s
BSO	0.7325361	0.3559702	0.4821867	0.484833	0.0840915	0.007071386
PSO	0.7778263	0.2672281	0.4198611	0.397749	0.1050463	0.011034729
FA	0.6796496	0.2208227	0.4322819	0.419294	0.1116067	0.01245606
GA	0.7087106	0.3266274	0.4636778	0.450104	0.0934912	0.008740608
MSCHO	0.8710003	0.3785877	0.5331253	0.516252	0.1065323	0.011349135
WOA	0.8214194	0.284291	0.4652915	0.45887	0.107541	0.011565059
SCHO	0.7694782	0.290179	0.4517118	0.433505	0.0936311	0.008766775
COLSHADE	0.6884695	0.3254779	0.4728525	0.478327	0.0914179	0.008357233

Из табеле 8 се види да *MSCHO* даје најмању стопу грешке кроз тридесет независних покретања, такође даје и најбоље резултате у погледу средње вредности и медијане стопе грешке. *SCHO* дала је најбоље резултате у погледу стандардне девијације и варијансе.

Из табеле 9 се види да *MSCHO* даје најбоље резултате за стопу грешке и за најбоља и најгора покретања свих метахеуристика, као и у погледу средње вредности и медијане. Стандардна девијација и варијанса дале су најбоље резултате за *BSO*.

У табели 10 се види детаљно поређење за најбоље моделе за сваку од поређених метахеуристика.

Табела 10: Експеримент1: Рекурентне неуралне мреже: Детаљно поређење најбољих модела

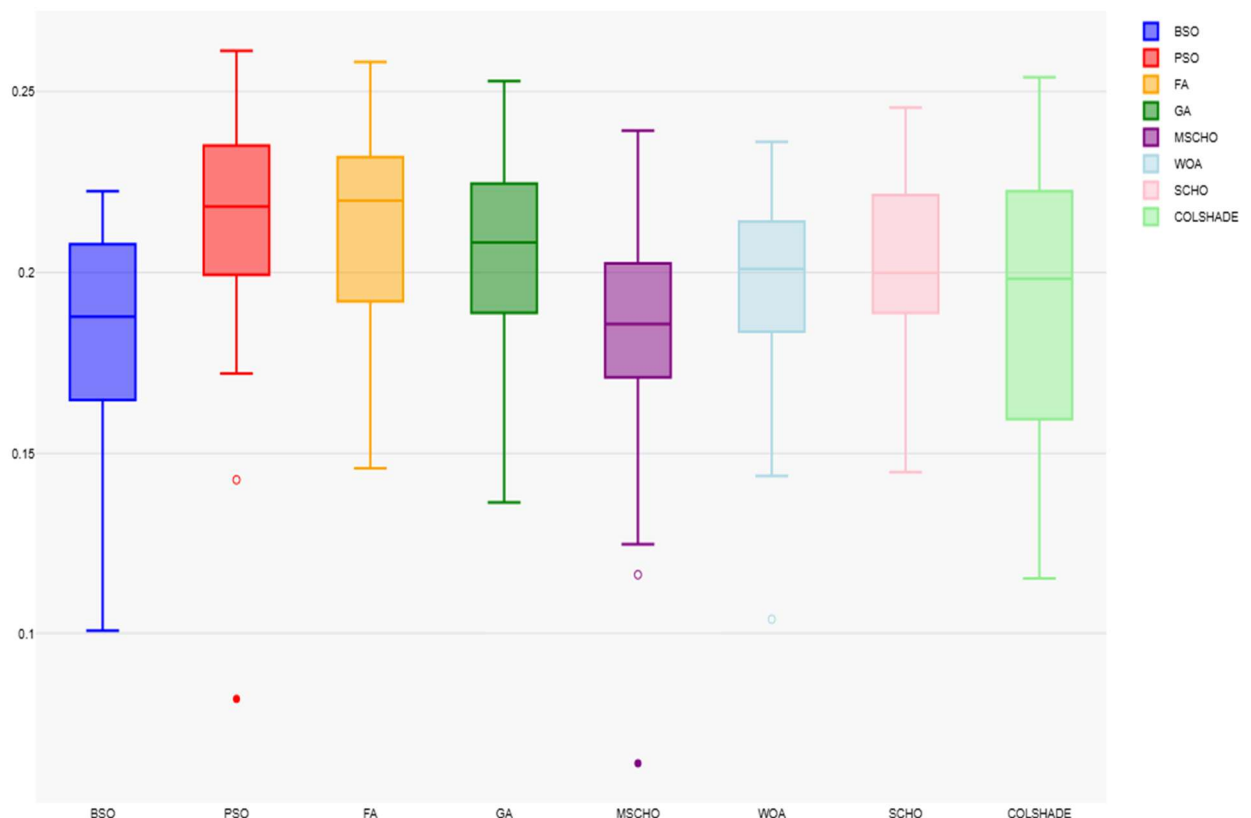
	Метрика	Здрави	ПБ	Тачност	Макро тачност	Тежинска тачност
BSO	Прецизност	0.906977	0.897567	0.899265	0.902272	0.899897
	Опозив	0.661017	0.977685	0.899265	0.819351	0.899265
	Ф1	0.764706	0.935915	0.899265	0.850310	0.893517
PSO	Прецизност	0.846491	0.940690	0.918153	0.893590	0.917362
	Опозив	0.817797	0.951185	0.918153	0.884491	0.918153
	Ф1	0.831897	0.945908	0.918153	0.888903	0.917675
FA	Прецизност	0.699588	0.907042	0.854145	0.803315	0.855669
	Опозив	0.720339	0.898187	0.854145	0.809263	0.854145
	Ф1	0.709812	0.902593	0.854145	0.806202	0.854853
GA	Прецизност	0.778947	0.884666	0.863589	0.831807	0.858486
	Опозив	0.627119	0.941423	0.863589	0.784271	0.863589
	Ф1	0.694836	0.912162	0.863589	0.803499	0.858344
MSCHO	Прецизност	0.851406	0.965909	0.935992	0.908657	0.937554
	Опозив	0.898305	0.948396	0.935992	0.923351	0.935992
	Ф1	0.874227	0.957072	0.935992	0.915650	0.936557
WOA	Прецизност	0.727575	0.973926	0.896118	0.850751	0.912920
	Опозив	0.927966	0.885635	0.896118	0.906800	0.896118
	Ф1	0.815642	0.927684	0.896118	0.871663	0.899938
SCHO	Прецизност	0.697581	0.91063	0.855194	0.804109	0.857877
	Опозив	0.733051	0.895397	0.855194	0.814224	0.855194
	Ф1	0.714876	0.902954	0.855194	0.808915	0.856378
COLSHADE	Прецизност	0.771552	0.920943	0.884575	0.846247	0.883948
	Опозив	0.758475	0.926081	0.884575	0.842278	0.884575
	Ф1	0.764957	0.923505	0.884575	0.844231	0.884242

У табели 11 приказани су параметри архитектуре мреже за најбоље моделе и то број рекурентних слојева (Број слојева), број неурона у првом слоју (Н1), број неурона у другом слоју (Н2), стопа учења (СУ), фактор одбацивања (ФО) и број епоха (Број епоха).

Табела 11. Експеримент1: Рекурентне неуралне мреже: Вредности хиперпараметара за најбоље моделе

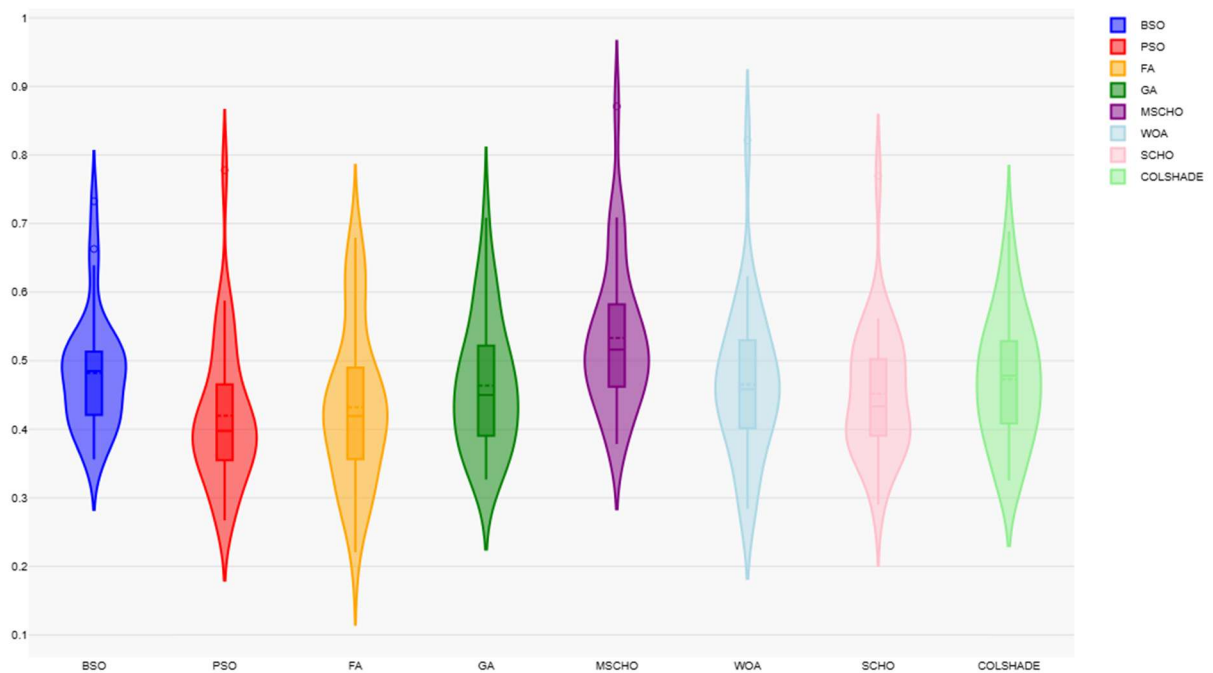
	Број слојева	Н1	Н2	СУ	ФО	Број епоха
BSO	2	9	12	0.01	0.2	44
PSO	2	5	13	0.01	0.2	60
FA	2	15	6	0.009198	0.064	54
GA	2	9	12	0.008054	0.1192	45
MSCHO	2	13	5	0.01	0.2	60
WOA	2	5	15	0.004018	0.05	53
SCHO	2	5	15	0.01	0.2	60
COLSHADE	2	8	12	0.009782	0.1406	47

На слици 7.11. приказан је *boxplot* дијаграм за функцију циља.



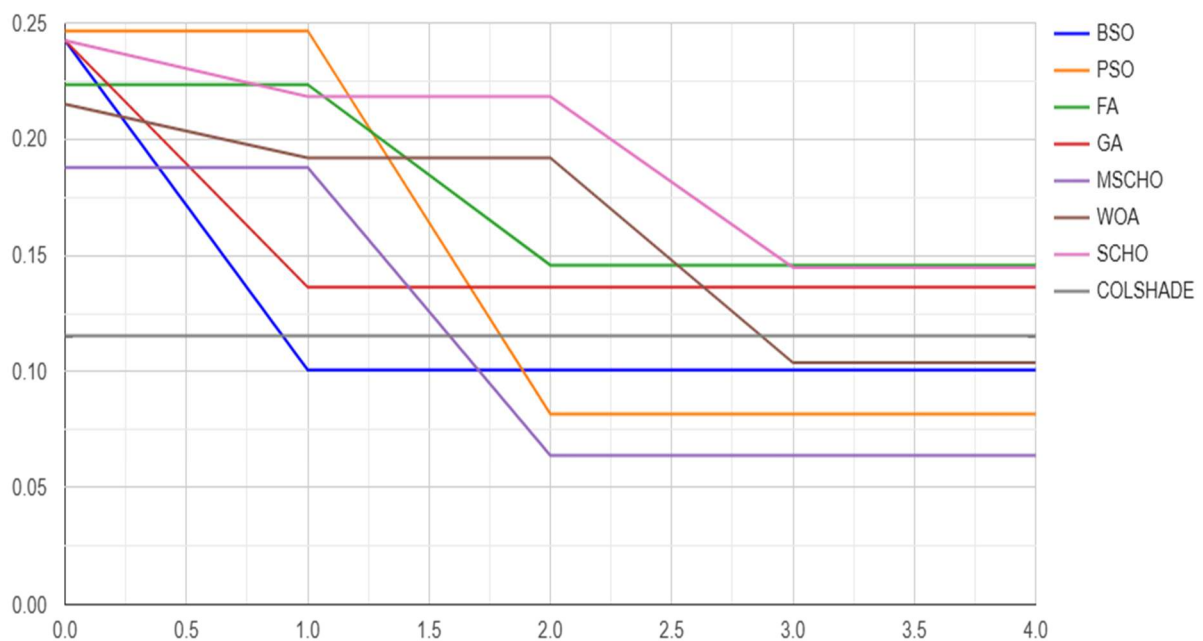
Слика 7.11. Експеримент1: Рекурентне неуралне мреже: Boxplot дијаграм функције циља

На слици 7.12. приказан је виолина дијаграм за индикатор функцију.

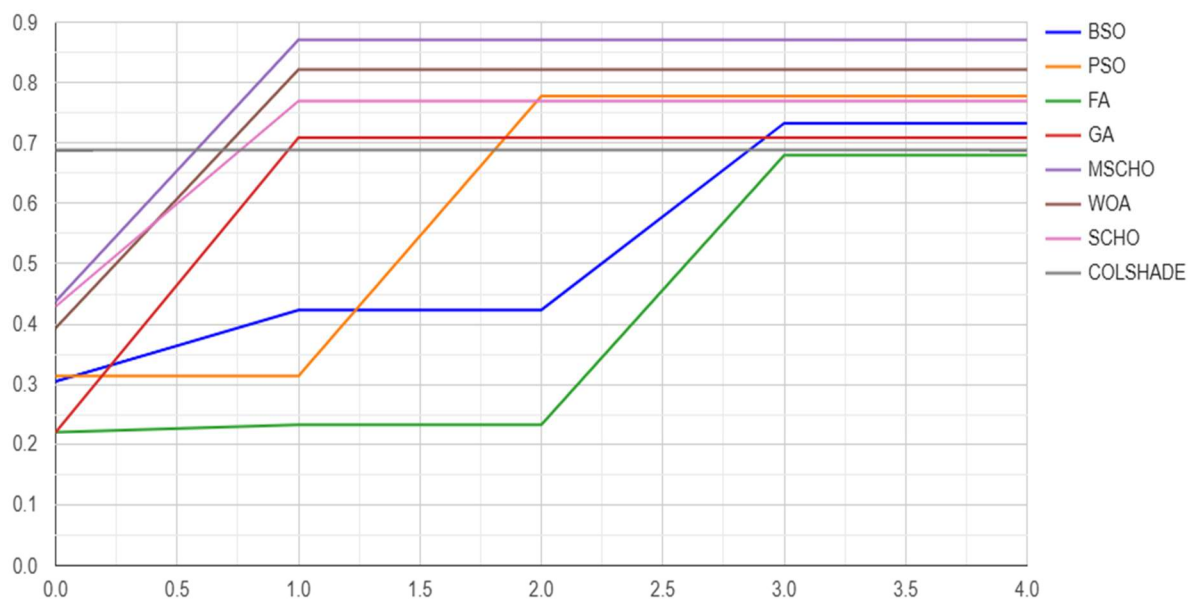


Слика 7.12. Експеримент1: Рекурентне неуралне мреже: Виолина дијаграм за индикатор функцију

На слици 7.13. могу се видети конвергенцијске стопе за функцију циља, док се на слици 7.14. могу видети конвергенцијске стопе за индикатор функцију.

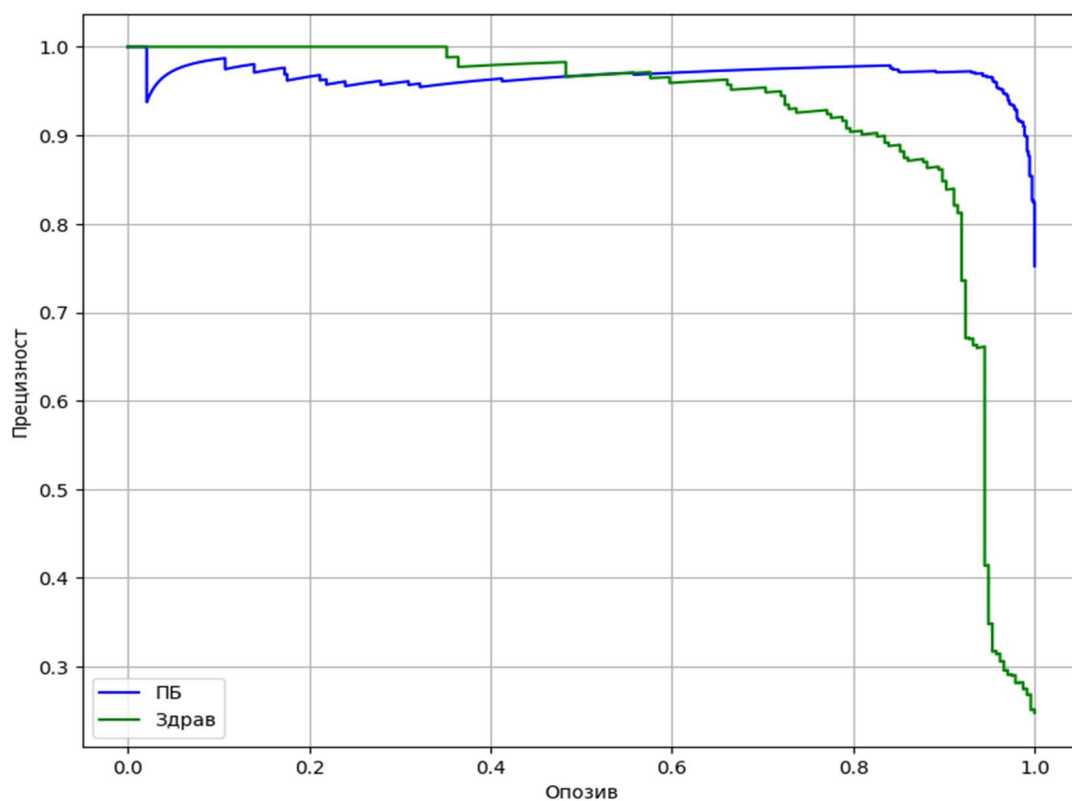


Слика 7.13. Експеримент1: Рекурентне неуралне мреже: Конвергенцијске стопе за функцију циља

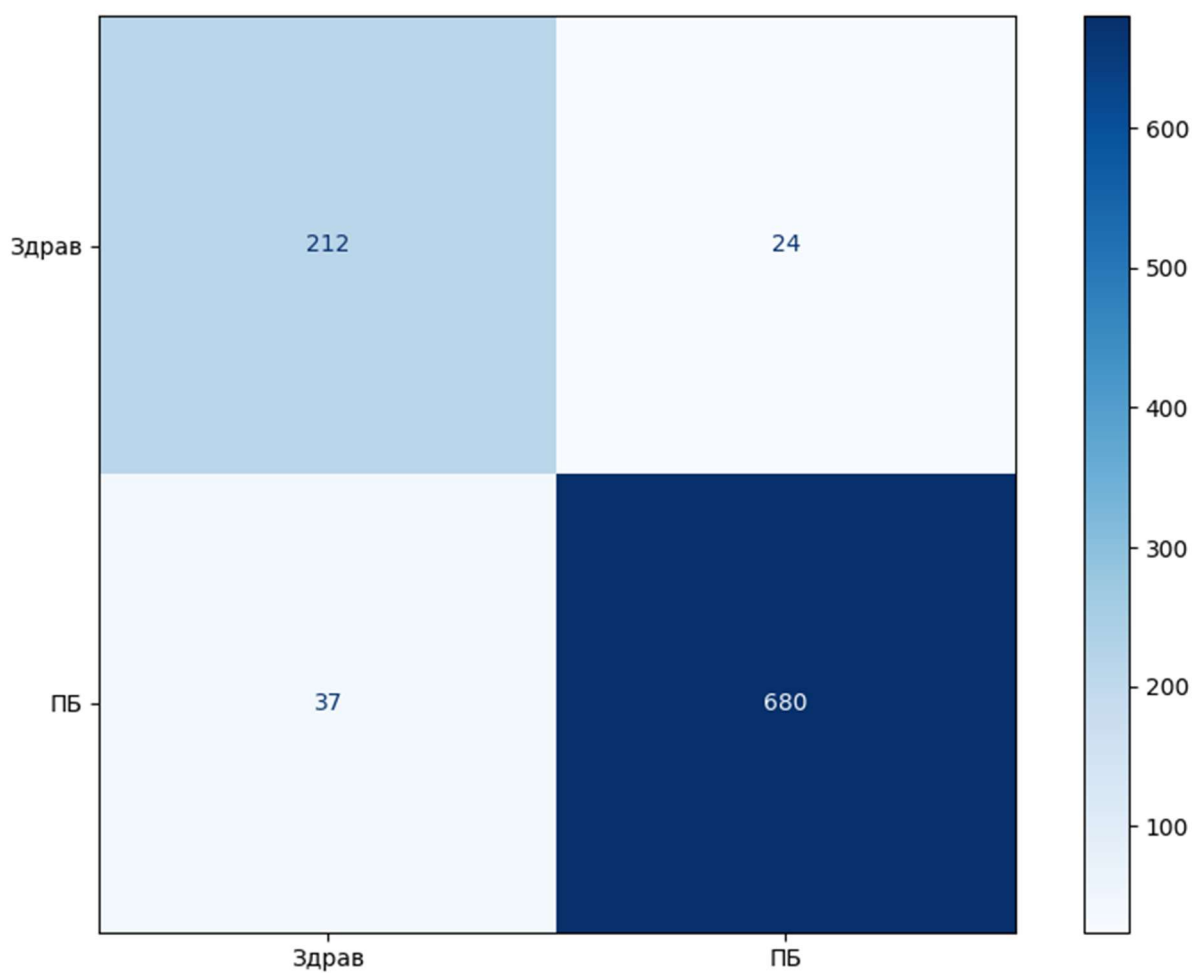


Слика 7.14. Експеримент1: Рекурентне неуралне мреже: Конвергенцијске стопе за индикатор функцију

На слици 7.15. налази се крива прецизности и опозива за најбољи модел који је постигла модификована метахеуристика, док се на слици 7.16. залази конфузиона матрица за исти модел.



Слика 7.15. Експеримент1: Рекурентне неуралне мреже: Крива прецизности и опозива



Слика 7.16. Експеримент1: Рекурентне неуралне мреже: Конфузиона матрица

7.2.2. Експеримент2: Рекурентне неуралне мреже са механизмом пажње

Скуп података: Скуп података о тапкању прстима

Класификација: Бинарна класификација (здрава особа/Паркинсонова болест)

Особине података: Подаци су подељени у временске серије, где је за сваку временску серију дефинисано шест особина (по 3 вредности за 3 осе жirosкопа, за оба прста), и свих шест особина улази у модел, а подаци су пре уласка у модел нормализовани

Подела података: Подаци су подељени у односу 70% за тренирање, 30% за тестирање, приликом чега већина записа у скупу за тестирање припада особама које скуп за тренирање није никад видео

Модел: Рекурентна неурална мрежа са Лоунг механизмом пажње, чија се архитектура састоји од улазног слоја, затим једног или два рекурентна слоја, праћених слојем пажње и једног излазног слоја. За имплементацију модела коришћен је *Keras API*, који је подигнут над *Tensorflow*-ом.

Хиперпараметри модела: Подешавани хиперпараметри модела су:

- Број рекурентних слојева у мрежи: [1,2],
- Број неурона у сваком слоју: [5,15],
- Фактор одбацивања: [0.05,0.2],
- Стопа учења: [0.0001,0.01],
- Број епоха: [30,60],
- Број неурона у слоју пажње: [5,15].

Резултати симулације: Табела 12 приказује резултате функције циља за најбоље, најгоре и средње покретање (по 30 покретања) модела за сваку од поређених метахеуристика. Уз то је приказана и медијана функције циља, као и стандардна девијација и варијанса. Табела 13 приказује исте податке за индикатор функцију.

Табела 12. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Резултати функције циља

	Најбоље	Најгоре	Средње	Медијана	σ	s
BSO	0.0241343	0.1164743	0.0531305	0.047744	0.0216924	0.000470559
PSO	0.0283316	0.1280168	0.0674362	0.062959	0.0285329	0.000814125
FA	0.0209864	0.1941238	0.0862539	0.075551	0.0400911	0.001607295
GA	0.0220357	0.1238195	0.0656873	0.065058	0.0274066	0.000751119
MSCHO	0.0104932	0.0933893	0.0494928	0.048269	0.021086	0.000444617
WOA	0.0115425	0.1458552	0.0647779	0.058762	0.0305529	0.000933481
SCHO	0.0157398	0.13851	0.0690101	0.062959	0.0330956	0.001095319
COLSHADE	0.019937	0.1080797	0.0597062	0.059286	0.0222055	0.000493084

Табела 13. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Резултати индикатор функције

	Најбоље	Најгоре	Средње	Медијана	σ	s
BSO	0.945085	0.7211542	0.852559	0.854431	0.0614518	0.003776328
PSO	0.9173986	0.6065235	0.8110182	0.824896	0.0806795	0.006509176
FA	0.9291992	0.4313567	0.7580787	0.770151	0.1106873	0.012251681
GA	0.9061242	0.6921249	0.8124338	0.809138	0.0593042	0.003516985
MSCHO	0.9494558	0.7200228	0.8488022	0.845945	0.0689037	0.004747718
WOA	0.9402699	0.6856269	0.8292345	0.846963	0.0661773	0.004379441
SCHO	0.9749761	0.6604808	0.8345073	0.860879	0.0861794	0.007426885
COLSHADE	0.942578	0.6929507	0.8316548	0.839124	0.0656503	0.004309959

Из табеле 12 се види да *MSCHO* даје најбоље резултате за стопу грешке и за најбоља и најгора покретања свих метахеуристика, као и у погледу средње вредности и у погледу стандардне девијације и варијансе. За најбољу медијану стопе грешке се показао *BSO*.

Из табеле 13 се види да *SCHO* даје најбоље резултате индикатор функције за најбоље покретање свих метахеуристика, као и за медијану. *BSO* даје најбоље резултате за најгоре покретање свих метахеуристика, као и за средњу вредност. Стандардна девијација и варијанса дале су најбоље резултате за *GA*.

У табели 14 се види детаљно поређење за најбоље моделе за сваку од поређених метахеуристика, где се види да за већину параметара поређења *MSCHO* даје најбоље резултате.

Табела 14. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Детаљно поређење најбољих модела

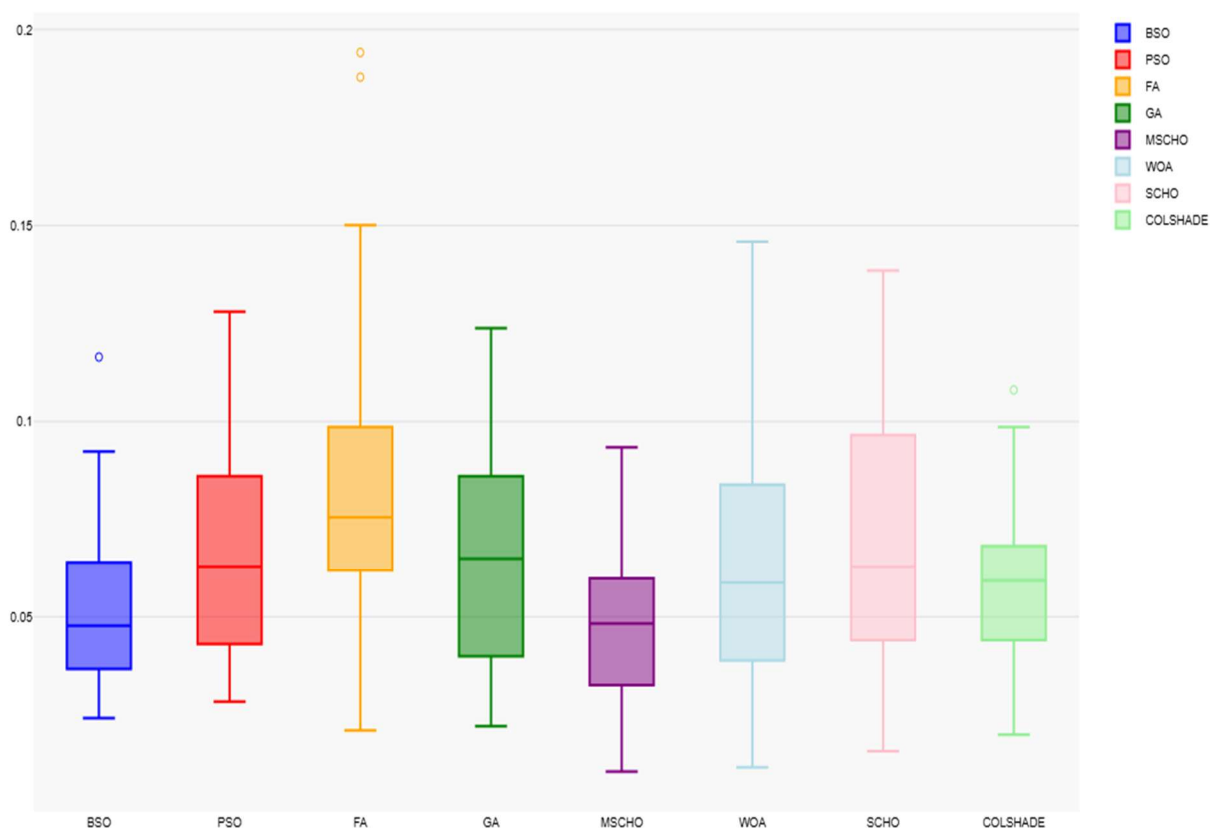
	Метрика	Здрави	ПБ	Тачност	Макро тачност	Тежинска тачност
BSO	Прецизност	0.914397	0.998563	0.975866	0.956480	0.977720
	Опозив	0.995763	0.969317	0.975866	0.982540	0.975866
	Ф1	0.953347	0.983723	0.975866	0.968535	0.976200
PSO	Прецизност	0.897338	1.000000	0.971668	0.948669	0.974577
	Опозив	1.000000	0.962343	0.971668	0.981172	0.971668
	Ф1	0.945892	0.980810	0.971668	0.963351	0.972163
FA	Прецизност	0.942623	0.991537	0.979014	0.967080	0.979424
	Опозив	0.974576	0.980474	0.979014	0.97752	0.979014
	Ф1	0.958333	0.985975	0.979014	0.972154	0.979130
GA	Прецизност	0.935223	0.992918	0.977964	0.964070	0.978630
	Опозив	0.978814	0.977685	0.977964	0.978249	0.977964
	Ф1	0.956522	0.985242	0.977964	0.970882	0.978130
MSCHO	Прецизност	0.982906	0.991655	0.989507	0.987281	0.989488
	Опозив	0.974576	0.994421	0.989507	0.984499	0.989507
	Ф1	0.978723	0.993036	0.989507	0.985880	0.989492
WOA	Прецизност	0.955466	1.000000	0.988458	0.977733	0.988972
	Опозив	1.000000	0.984658	0.988458	0.992329	0.988458
	Ф1	0.977226	0.992270	0.988458	0.984748	0.988544
SCHO	Прецизност	0.940239	1.000000	0.984260	0.970120	0.985201
	Опозив	1.000000	0.979079	0.984260	0.989540	0.984260
	Ф1	0.969199	0.989429	0.984260	0.979314	0.984419
COLSHADE	Прецизност	0.957806	0.987430	0.980063	0.972618	0.980094
	Опозив	0.961864	0.986053	0.980063	0.973959	0.980063
	Ф1	0.959831	0.986741	0.980063	0.973286	0.980077

У табели 15 приказани су параметри архитектуре мреже за најбоље моделе и то број рекурентних слојева (Број слојева), број неурона у првом слоју (Н1), број неурона у другом слоју (Н2), стопа учења (СУ), фактор одбацивања (ФО), број епоха (Број епоха) и број неурона у слоју пажње (БП).

**Табела 15. Експеримент2: Рекурентне неуралне мреже са механизмом пажње:
Вредности хиперпараметара за најбоље моделе**

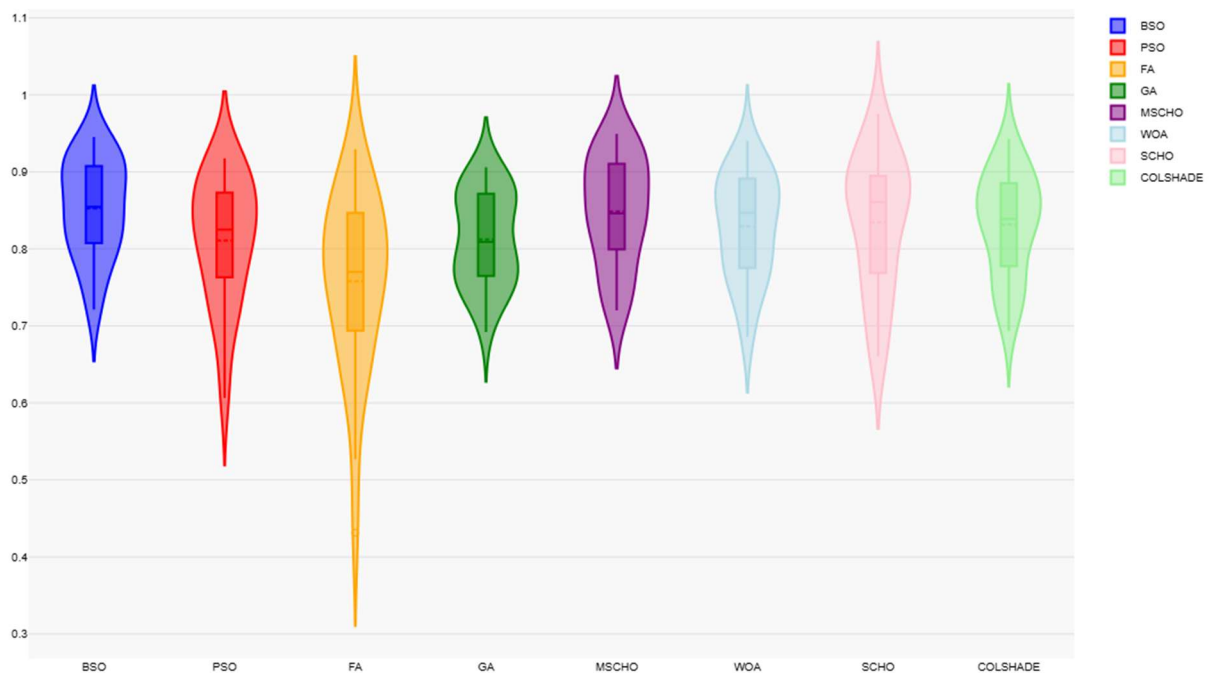
	Број слојева	Н1	Н2	СУ	ФО	Епохе	БП
BSO	2	12	7	0.01	0.2	34	6
PSO	2	10	15	0.01	0.05	30	13
FA	2	10	15	0.0034	0.1934	59	13
GA	2	5	13	0.0087	0.194	40	15
MSCHO	2	8	9	0.01	0.2	50	5
WOA	2	11	7	0.01	0.2	31	15
SCHO	2	15	11	0.01	0.14	60	7
COLSHADE	2	13	14	0.001	0.1952	50	12

На слици 7.17. приказан је *boxplot* дијаграм за функцију циља.



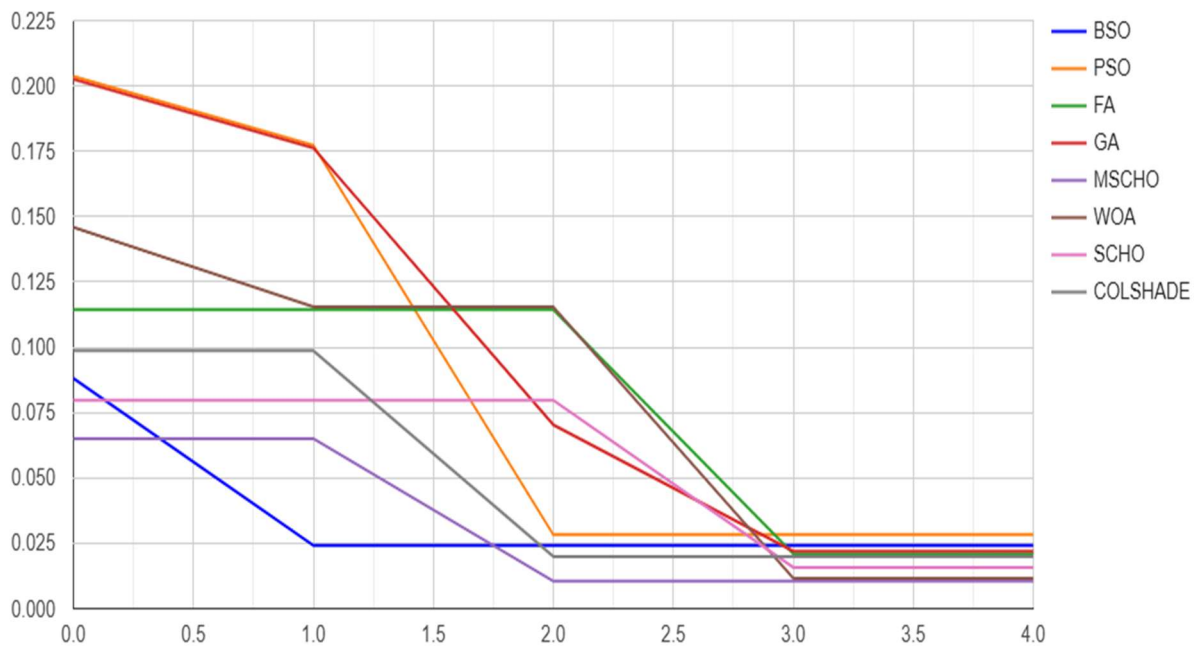
Слика 7.17. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: *Boxplot* дијаграм функције циља

На слици 7.18. приказан је виолина дијаграм за индикатор функцију.

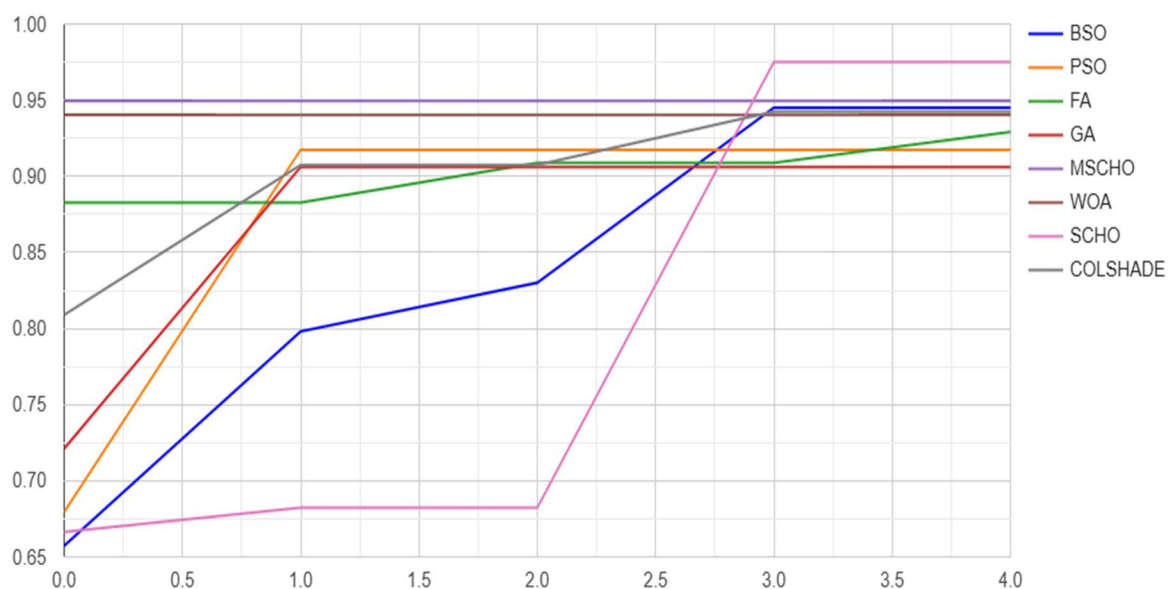


Слика 7.18. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Виолина дијаграм за индикатор функцију

На слици 7.19. могу се видети конвергенцијске стопе за функцију циља, док се на слици 7.20. могу видети конвергенције стопе за индикатор функцију.

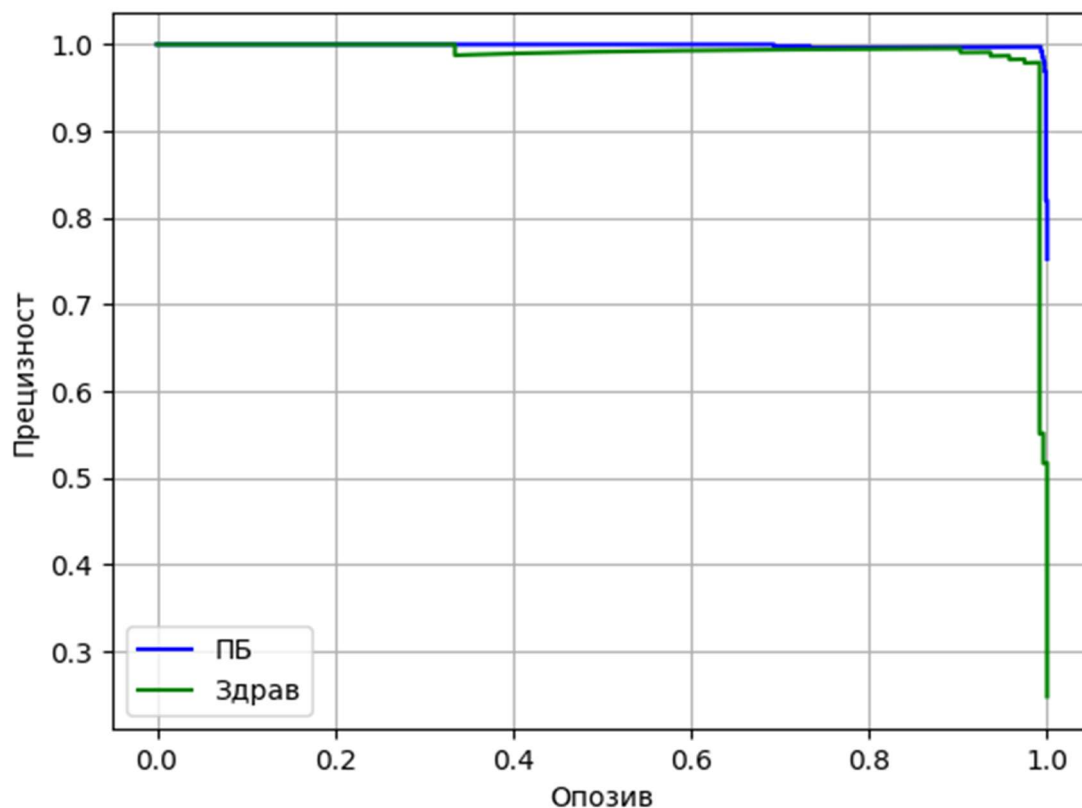


Слика 7.19. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Конвергенцијске стопе за функцију циља

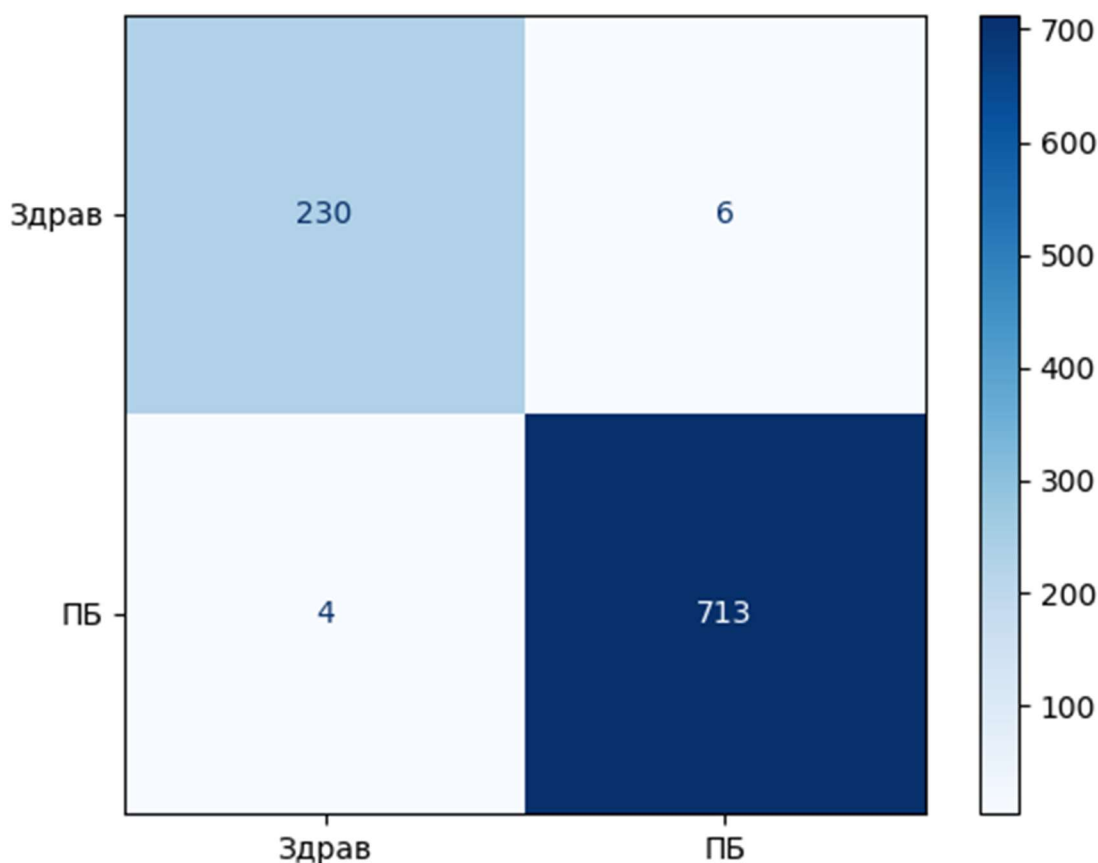


Слика 7.20. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Конвергенцијске стопе за индикатор функцију

На слици 7.21. налази се крива прецизности и опозива за најбољи модел који је постигла модификована метахеуристика, док се на слици 7.22. налази конфузиона матрица за исти модел.



Слика 7.21. Експеримент2: Рекурентне неуралне мреже са механизмом пажње: Крива прецизности и опозива



Слика 7.22. Експеримент2: Рекурентне неуралне мреже са механизмом пажње:
Конфузиона матрица

7.2.3. Експеримент3: Екстремни градијентни појачивач

Скуп података: Скуп података о тапкању прстима

Класификација: Мултикласна класификација (здрава особа/Паркинсонова болест/мултипла системска атрофија/прогресивна супрануклеарна парализа)

Особине података: Подаци су подељени у временске серије, где је за сваку временску серију дефинисано шест особина (по 3 вредности за 3 осе жироскопа, за оба прста). Затим се за сваку временску серију са шест особина, врши подразумевана екстракција додатних особина уз помоћ *Python* библиотеке *tsfresh* (енг. *Time Series Feature extraction based on scalable hypothesis tests*).

Уз помоћ ове библиотеке може се вршити аутоматско екстраховање особина, без мануелног дефинисања жељених особина, и добро је оптимизована за велике скупове података. Екстракција особина може бити подразумевана где се за временску серију извуче око 1000 додатних особина, а може бити и минимална где се за временску серију извуче око 20 додатних особина.

Неке од додатних особина које се могу добити уз помоћ *tsfresh* библиотеке су:

- Средња квадратна вредност временске серије,
- Аритметичка средина апсолутно највећих вредности временске серије,
- Мера нелинеарности у временској серији уз помоћ ц3 статистике,
- Број вредности изнад и испод средње вредности,
- Фуријеови коефицијенти једнодимензионалне Фуријеове трансформације,

- Прве и последње локације минималне и максималне вредности,
- Број дуплих вредности,
- Минимална, максимална, средња вредност,
- Стандардна девијација и варијанса,
- Линеарна регресија методом најмањих квадрата,
- Средња вредност централне апроксимације друге деривације,
- Пермутациона ентропија,
- Q-квантил,
- Статистика симетрије временског обрта,
- и многе друге.

Подела података: Подаци су подељени у односу 70% за тренирање, 30% за тестирање, насумично око свих записа, и нормализовани су пре извлачења додатних одлика

Модел: Екстремни градијентни појачивач. За имплементацију модела коришћена је *xgboost* библиотека, и *XGBClassifier* са подешеном оптимизованом методом за креирање стабла уз помоћ хистограма.

Хиперпараметри модела: Подешавани хиперпараметри модела су:

- Стопа учења: [0.1,0.9],
- Минимална сума Хесијана у чвору: [1,10],
- Проценат тренинг података који се узима: [0.01,1],
- Проценат особина које се случајно бирају при конструкцији новог стабла: [0.01,1],
- Максимална дубина стабла: [3,10],
- Параметар γ : [0,0.8].

Резултати симулације: Табела 16 приказује резултате функције циља за најбоље, најгоре и средње покретање (по 30 покретања) модела за сваку од поређених метахеуристика. Уз то је приказана и медијана функције циља, као и стандардна девијација и варијанса. Табела 17 приказује исте податке за индикатор функцију.

Табела 16. Експеримент3: Екстремни градијентни појачивач: Резултати функције циља

	Најбоље	Најгоре	Средње	Медијана	σ	s
BSO	0.0769231	0.1692308	0.1205128	0.123077	0.0186669	0.000348455
PSO	0.0769231	0.1538462	0.1184615	0.123077	0.0199209	0.000396844
FA	0.1230769	0.2153846	0.1517949	0.153846	0.0259672	0.000674293
GA	0.1076923	0.2461538	0.1579487	0.153846	0.0343858	0.00118238
MSCHO	0.0769231	0.1538462	0.1148718	0.115385	0.0189397	0.000358711
WOA	0.0769231	0.1692308	0.1164103	0.107692	0.0212743	0.000452597
SCHO	0.0923077	0.1692308	0.1194872	0.115385	0.0220035	0.000484155
COLSHADE	0.0923077	0.1846154	0.1369231	0.138462	0.0225932	0.000510454

Из табеле 16. се види да више метахеуристика дели прво место у погледу најбоље вредности стопе грешке и то су: *MSCHO*, *BSO*, *PSO* и *WOA*, док за најгора покретања најбоље вредности

даје *MSCHO* и *PSO*. Најбољу средњу вредност даје *MSCHO*, док за медијану стопе грешке се најбоље показао *WOA*. Најбољу стандардну девијацију и варијансу даје *BSO*.

Табела 17. Експеримент3: Екстремни градијентни појачивач: Резултати индикатор функције

	Најбоље	Најгоре	Средње	Медијана	σ	s
BSO	0.8768551	0.7335225	0.8364781	0.835833	0.0301629	0.000909801
PSO	0.8972819	0.7948880	0.8398800	0.835807	0.0261377	0.000683181
FA	0.8563309	0.7122985	0.7960715	0.804965	0.0392982	0.001544349
GA	0.8561947	0.6716135	0.7946639	0.794823	0.0393354	0.001547276
MSCHO	0.8972819	0.7948232	0.8480832	0.856286	0.0246038	0.000605346
WOA	0.8769328	0.7743056	0.8433013	0.846049	0.0256706	0.000658982
SCHO	0.8768939	0.7535545	0.8398781	0.835807	0.0315209	0.000993565
COLSHADE	0.8768162	0.7535545	0.8227994	0.815370	0.0353068	0.001246570

Из табеле 17. се види да две метахеуристике деле прво место за најбоља покретања за вредности индикатор функције и то су *MSCHO* и *PSO*. За најгоре покретање најбољу вредност даје *PSO*. За средњу вредност, медијану, стандардну девијацију и варијансу најбоље вредности је дала *MSCHO*.

У табели 18 приказани су параметри за најбоље моделе и ти параметри су: стопа учења (CY), минимална сума Хесијана у чвору (Мин_Сум), проценат тренинг података (Тренинг%), проценат особина које се случајно бирају при конструкцији новог стабла (Особине%), максимална дубина стабла (Дубина) и параметар γ .

Табела 18. Експеримент3: Екстремни градијентни појачивач: Вредности хиперпараметара за најбоље моделе

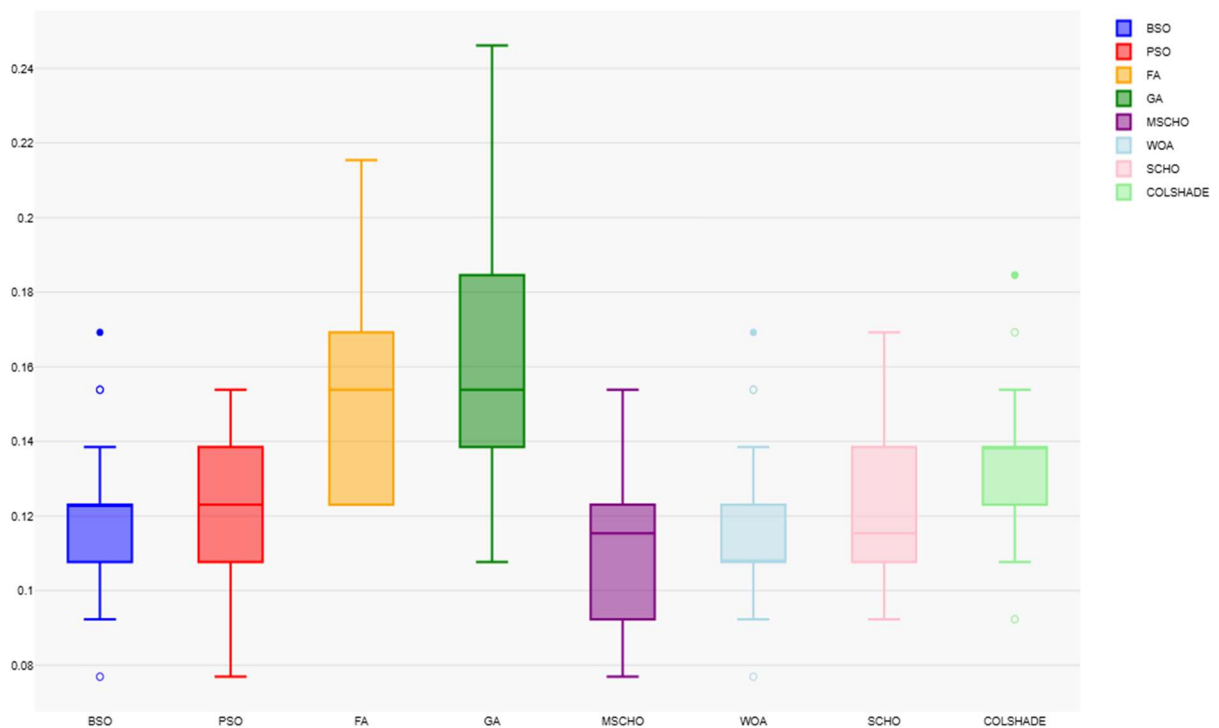
	CY	Мин_Сум	Тренинг%	Особине%	Дубина	γ
BSO	0.3569	5	0.9213	0.6732	3	0.2471
PSO	0.3728	10	0.983	0.7368	7	0.2606
FA	0.7332	6	0.7961	0.77	4	0.7149
GA	0.1627	10	1	0.7805	4	0.4322
MSCHO	0.4455	3	0.7709	0.8025	10	0.4835
WOA	0.3601	5	0.8183	0.9971	10	0.468
SCHO	0.3766	1	0.9547	0.5391	8	0.7677
COLSHADE	0.2292	3	0.8093	0.7751	9	0.2274

У табели 19 се види детаљно поређење за најбоље моделе за сваку од поређених метахеуристика.

Табела 19. Експеримент3: Екстремни градијентни појачивач: Детаљно поређење најбољих модела

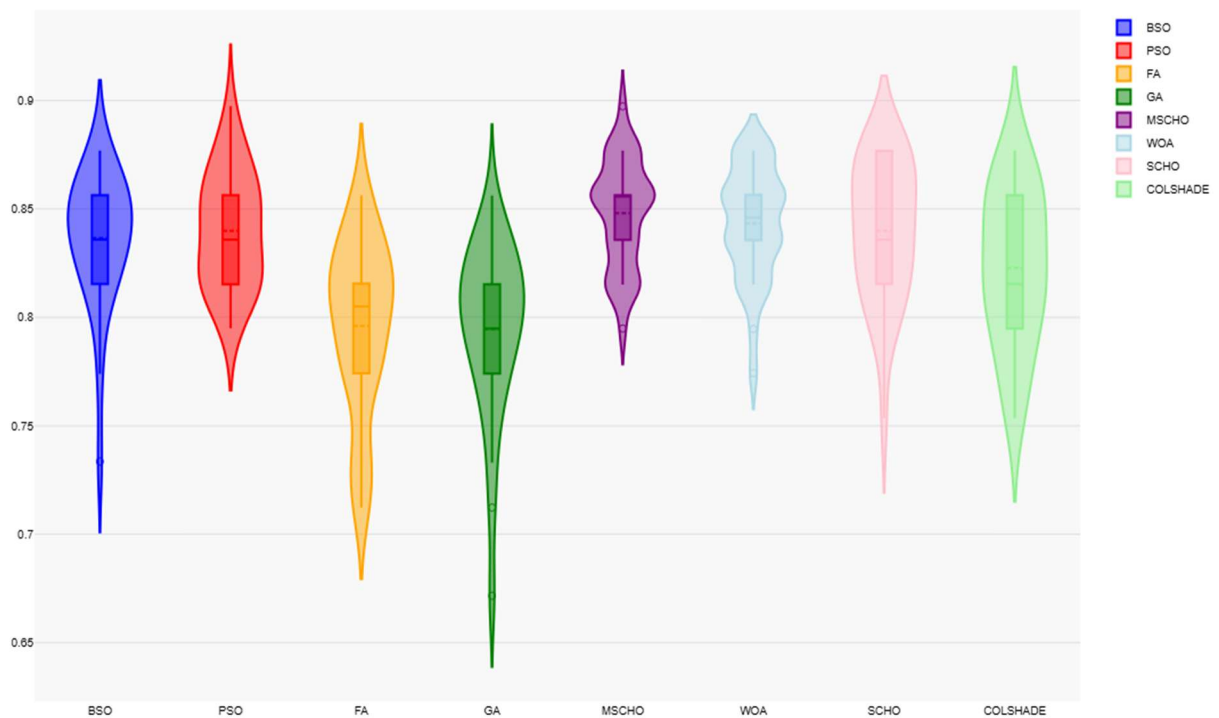
	Метрика	Здрави	ПБ	МСА	ПСП	Тачност	Макро тачност	Тежинска тачност
BSO	Прецизност	1	0.928571	0.888889	0.888889	0.923077	0.926587	0.924298
	Опозив	1	0.8125	0.941176	0.941176	0.923077	0.923713	0.923077
	Ф1	1	0.866667	0.914286	0.914286	0.923077	0.92381	0.922344
PSO	Прецизност	1	0.933333	0.894737	0.875	0.923077	0.925768	0.923367
	Опозив	1	0.875	1	0.823529	0.923077	0.924632	0.923077
	Ф1	1	0.903226	0.944444	0.848485	0.923077	0.924039	0.922022
FA	Прецизност	0.9375	0.866667	0.761905	1	0.876923	0.891518	0.890485
	Опозив	1	0.8125	0.941176	0.764706	0.876923	0.879596	0.876923
	Ф1	0.967742	0.83871	0.842105	0.866667	0.876923	0.878806	0.876686
GA	Прецизност	1	0.857143	0.842105	0.882353	0.892308	0.8954	0.89277
	Опозив	1	0.75	0.941176	0.882353	0.892308	0.893382	0.892308
	Ф1	1	0.8	0.888889	0.882353	0.892308	0.89281	0.89094
MSCHO	Прецизност	1	0.928571	0.842105	0.941176	0.923077	0.927963	0.925737
	Опозив	1	0.8125	0.941176	0.941176	0.923077	0.923713	0.923077
	Ф1	1	0.866667	0.888889	0.941176	0.923077	0.924183	0.922735
WOA	Прецизност	1	0.933333	0.842105	0.9375	0.923077	0.928235	0.925948
	Опозив	1	0.875	0.941176	0.882353	0.923077	0.924632	0.923077
	Ф1	1	0.903226	0.888889	0.909091	0.923077	0.925301	0.923343
SCHO	Прецизност	1	0.833333	0.882353	0.933333	0.907692	0.912255	0.910769
	Опозив	1	0.9375	0.882353	0.823529	0.907692	0.910846	0.907692
	Ф1	1	0.882353	0.882353	0.875	0.907692	0.909926	0.907579
COLSHADE	Прецизност	1	0.933333	0.8	0.933333	0.907692	0.916667	0.913846
	Опозив	1	0.875	0.941176	0.823529	0.907692	0.909926	0.907692
	Ф1	1	0.903226	0.864865	0.875	0.907692	0.910773	0.908143

На слици 7.23. приказан је *boxplot* дијаграм за функцију циља.



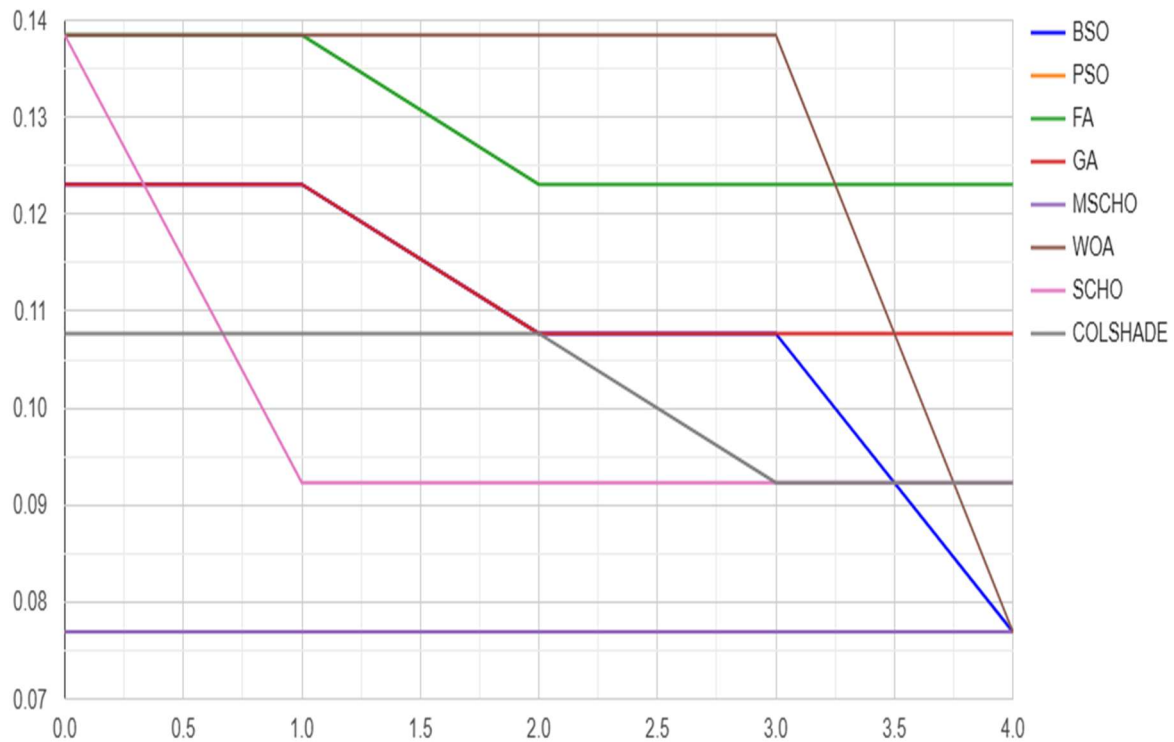
Слика 7.23. Експеримент3: Екстремни градијентни појачивач: *Boxplot* дијаграм функције циља

На слици 7.24. приказан је виолина дијаграм за индикатор функцију.

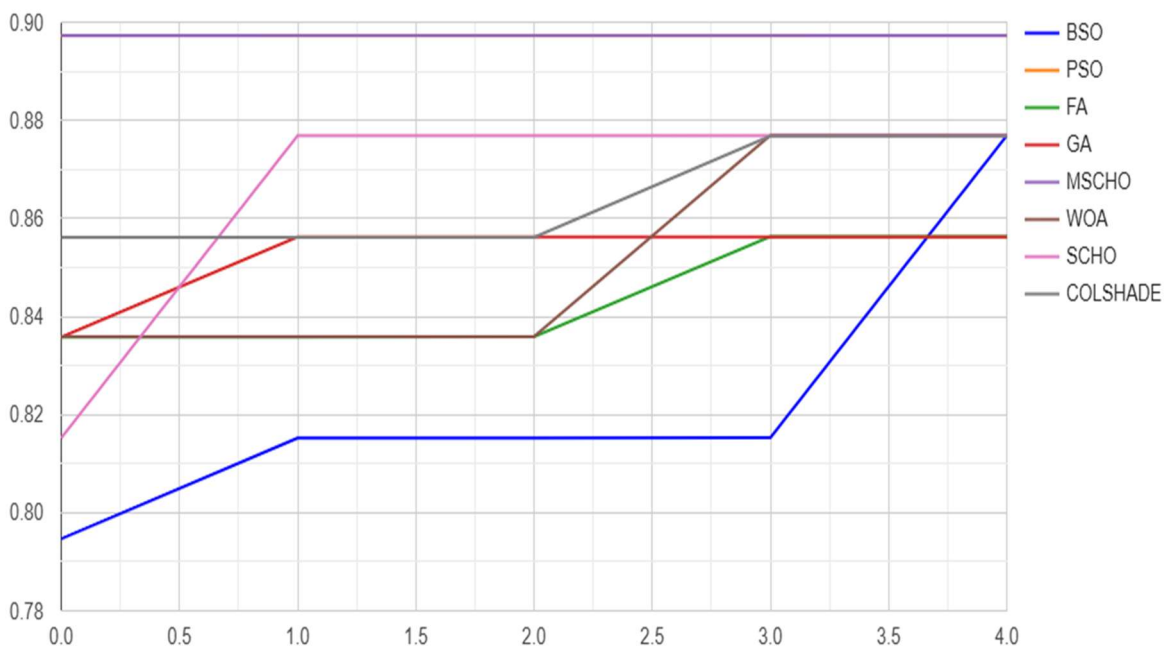


Слика 7.24. Експеримент3: Екстремни градијентни појачивач: Виолина дијаграм за индикатор функцију

На слици 7.25. могу се видети конвергенцијске стопе за функцију циља, док се на слици 7.26. могу видети конвергенције стопе за индикатор функцију.

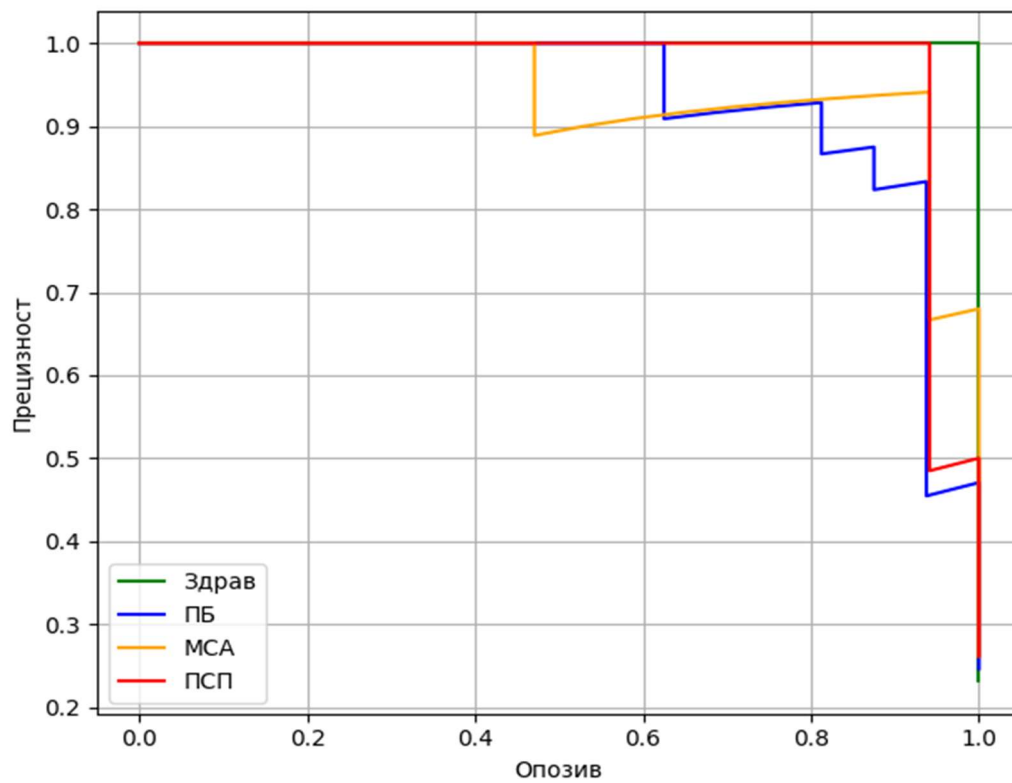


Слика 7.25. Експеримент3: Екстремни градијентни појачивач: Конвергенцијске стопе за функцију циља

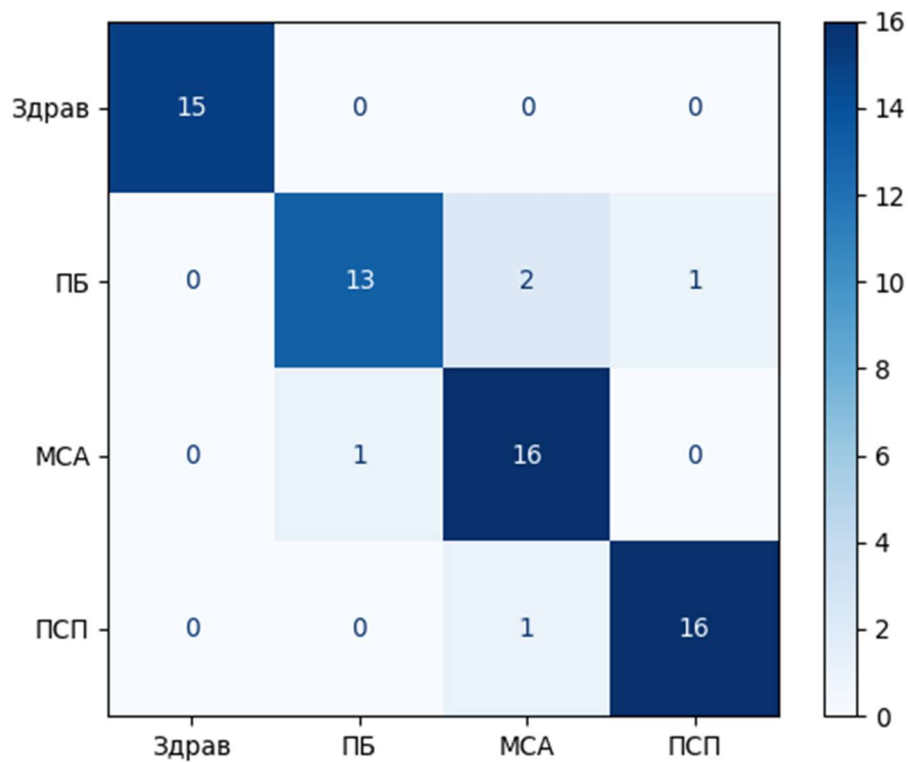


Слика 7.26. Експеримент3: Екстремни градијентни појачивач: Конвергенцијске стопе за индикатор функцију

На слици 7.27. налази се крива прецизности и опозива за најбољи модел који је постигла модификована метахеуристика, док се на слици 7.28. налази конфузиона матрица за исти модел.



Слика 7.27. Експеримент3: Екстремни градијентни појачивач: Крива прецизности и опозива



Слика 7.28. Експеримент3: Екстремни градијентни појачивач: Конфузиона матрица

7.2.4. Експеримент4: Екстремни градијентни појачивач

Скуп података: Скуп података о ходу

Класификација: Бинарна класификација (здрава особа/Паркинсонова болест)

Особине података: Подаци су подељени у временске серије, где је за сваку временску серију дефинисано 16 особина (по 8 вредности сензора за оба стопала). Затим се за сваку временску серију са шеснаест особина, врши минимална екстракција додатних особина уз помоћ *Python* библиотеке *tsfresh* (енг. *Time Series Feature extraction based on scalable hypothesis tests*).

Подела података: Подаци су подељени у односу 70% за тренирање, 30% за тестирање, насумично око особа, и нормализовани су пре извлачења додатних одлика

Модел: Екстремни градијентни појачивач. За имплементацију модела коришћена је *xgboost* библиотека, и *XGBClassifier* са подешеном оптимизованом методом за креирање стабла уз помоћ хистограма.

Хиперпараметри модела: Подешавани хиперпараметри модела су:

- Стопа учења: [0.1,0.9],
- Минимална сума Хесијана у чвору: [1,10],
- Проценат тренинг података који се узима: [0.01,1],
- Проценат особина које се случајно бирају при конструкцији новог стабла: [0.01,1],
- Максимална дубина стабла: [3,10],
- Параметар γ : [0,0.8].

Резултати симулације: Табела 20 приказује резултате функције циља за најбоље, најгоре и средње покретање (по 30 покретања) модела за сваку од поређених метахеуристика. Уз то је приказана и медијана функције циља, као и стандардна девијација и варијанса. Табела 21 приказује исте податке за индикатор функцију.

Табела 20. Експеримент4: Екстремни градијентни појачивач: Резултати функције циља

	Најбоље	Најгоре	Средње	Медијана	σ	s
BSO	0.125	0.2272727	0.1768939	0.181818	0.0220902	0.000487976
PSO	0.1363636	0.25	0.1969697	0.198864	0.0287978	0.000829316
FA	0.1477273	0.25	0.1992424	0.193182	0.0258687	0.000669192
GA	0.1590909	0.2727273	0.2121212	0.204545	0.0315101	0.000992883
MSCHO	0.125	0.2159091	0.1712121	0.170455	0.0221388	0.000490129
WOA	0.0909091	0.25	0.1757576	0.170455	0.0312907	0.000979109
SCHO	0.1363636	0.2272727	0.1810606	0.181818	0.0196678	0.000386823
COLSHADE	0.1590909	0.2272727	0.1893939	0.193182	0.0186339	0.000347222

Из табеле 20 се види да је у погледу стопе грешке најбољи резултат постигао алгоритам *WOA*, док је *MSCHO* на другом месту. За најгоре покретање и за средњу вредност *MSCHO* је на првом месту, док за медијану дели прво место са *WOA*.

Табела 21. Експеримент4: Екстремни градијентни појачивач: Резултати индикатор функције

	Најбоље	Најгоре	Средње	Медијана	σ	s
BSO	0.688946	0.4669897	0.542695	0.541816	0.0499398	0.002493984
PSO	0.6647619	0.4061697	0.5061027	0.498272	0.0717592	0.005149383
FA	0.5890594	0.3853968	0.5032687	0.50877	0.0533706	0.002848419
GA	0.6181029	0.3295238	0.4729187	0.462725	0.0717314	0.005145398
MSCHO	0.6726596	0.4904459	0.575552	0.575835	0.0440879	0.001943739
WOA	0.7709824	0.4136887	0.5549821	0.561397	0.066075	0.004365906
SCHO	0.6231884	0.394012	0.5300292	0.53601	0.052009	0.002704931
COLSHADE	0.6564736	0.4338235	0.523029	0.51928	0.0529718	0.002806009

Из табеле 21 се види да је за најбоље покретање свих метахеуристика *WOA* постигао најбоље резултате, док је *MSCHO* на другом месту. Док је у свим другим случајевима, најгоре покретање, средња вредност, медијана, стандардна девијација и варијанса *MSCHO* дала најбоље резултате.

У табели 22 приказани су параметри за најбоље моделе и ти параметри су: стопа учења (CY), минимална сума Хесијана у чвору (Мин_Сум), проценат тренинг података (Тренинг%), проценат особина које се случајно бирају при конструкцији новог стабла (Особине%), максимална дубина стабла (Дубина) и параметар γ .

Табела 22. Експеримент4: Екстремни градијентни појачивач: Вредности хиперпараметара за најбоље моделе

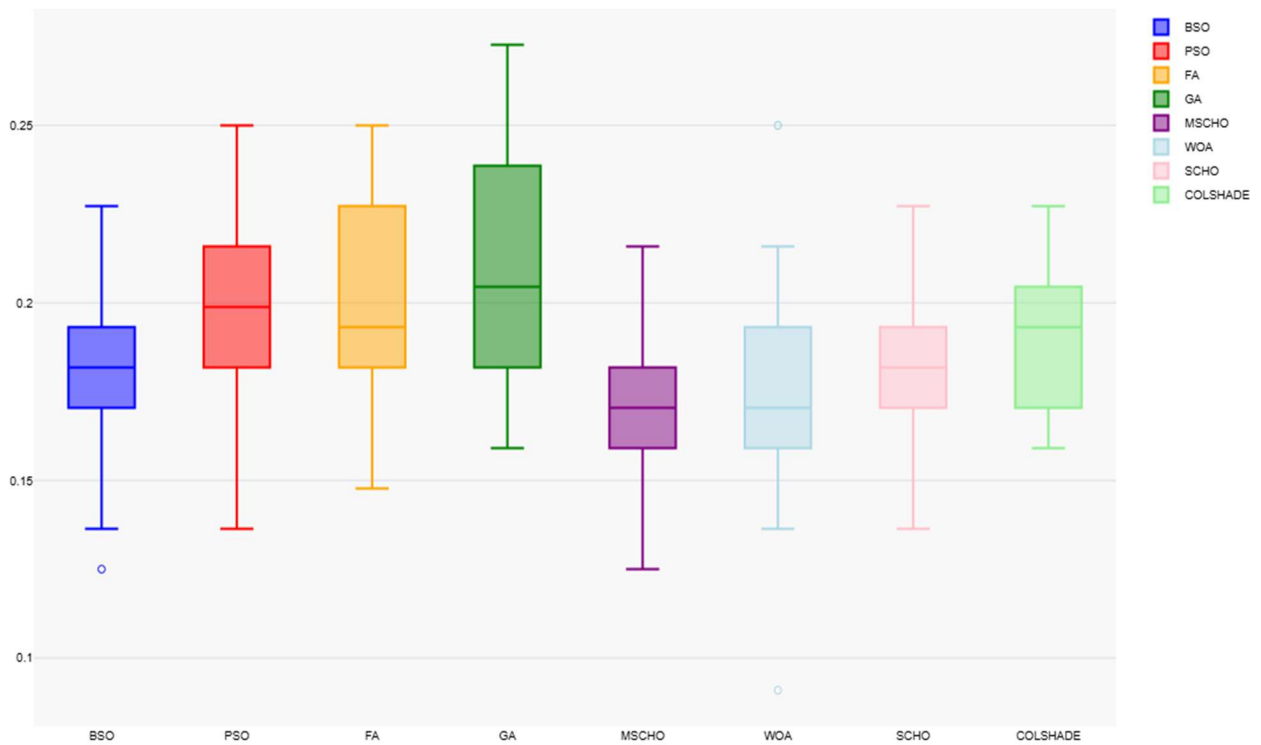
	CY	Мин_Сум	Тренинг%	Особине%	Дубина	γ
BSO	0.7821	10	0.6436	0.4771	4	0.4817
PSO	0.4043	4	0.3844	1	6	0.3076
FA	0.6952	2	0.5197	0.4553	4	0.3698
GA	0.4507	5	0.4543	0.8732	7	0.6917
MSCHO	0.9000	10	1	0.2629	10	0.6863
WOA	0.2722	4	0.1884	1	6	0.4908
SCHO	0.6261	10	0.8361	0.8466	4	0.6220
COLSHADE	0.6280	7	0.5236	0.0678	9	0.7050

У табели 23 се види детаљно поређење за најбоље моделе за сваку од поређених метахеуристика.

Табела 23. Експеримент4: Екстремни градијентни појачивач: Детаљно поређење најбољих модела

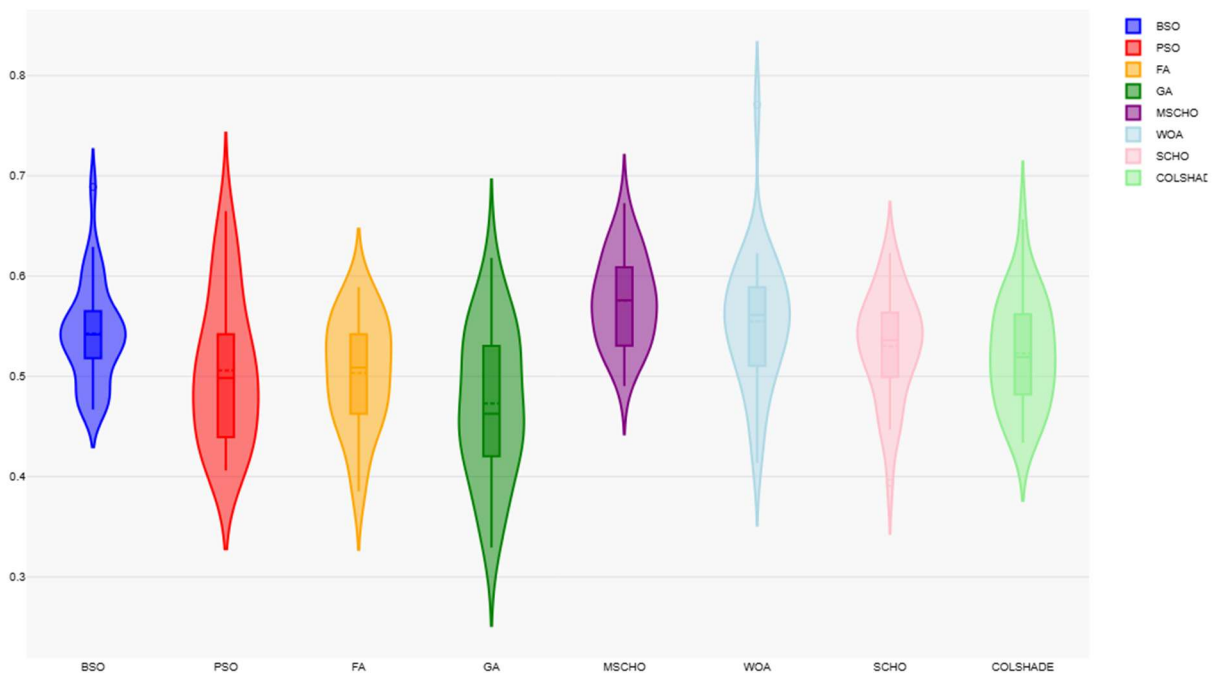
	Метрика	Здрави	ПБ	Тачност	Макро тачност	Тежинска тачност
BSO	Прецизност	0.791667	0.906250	0.875000	0.848958	0.873698
	Опозив	0.760000	0.920635	0.875000	0.840317	0.875000
	Ф1	0.775510	0.913386	0.875000	0.844448	0.874217
PSO	Прецизност	0.760000	0.904762	0.863636	0.832381	0.863636
	Опозив	0.760000	0.904762	0.863636	0.832381	0.863636
	Ф1	0.760000	0.904762	0.863636	0.832381	0.863636
FA	Прецизност	0.772727	0.878788	0.852273	0.825758	0.848657
	Опозив	0.680000	0.920635	0.852273	0.800317	0.852273
	Ф1	0.723404	0.899225	0.852273	0.811315	0.849276
GA	Прецизност	0.703704	0.901639	0.840909	0.802672	0.845408
	Опозив	0.760000	0.873016	0.840909	0.816508	0.840909
	Ф1	0.730769	0.887097	0.840909	0.808933	0.842686
MSCHO	Прецизност	0.818182	0.893939	0.875000	0.856061	0.872417
	Опозив	0.720000	0.936508	0.875000	0.828254	0.875000
	Ф1	0.765957	0.914729	0.875000	0.840343	0.872464
WOA	Прецизност	0.869565	0.923077	0.909091	0.896321	0.907875
	Опозив	0.800000	0.952381	0.909091	0.876190	0.909091
	Ф1	0.833333	0.937500	0.909091	0.885417	0.907907
SCHO	Прецизност	0.842105	0.869565	0.863636	0.855835	0.861764
	Опозив	0.640000	0.952381	0.863636	0.796190	0.863636
	Ф1	0.727273	0.909091	0.863636	0.818182	0.857438
COLSHADE	Прецизност	0.761905	0.865672	0.840909	0.813788	0.836192
	Опозив	0.640000	0.920635	0.840909	0.780317	0.840909
	Ф1	0.695652	0.892308	0.840909	0.793980	0.836440

На слици 7.29. приказан је *boxplot* дијаграм за функцију циља.



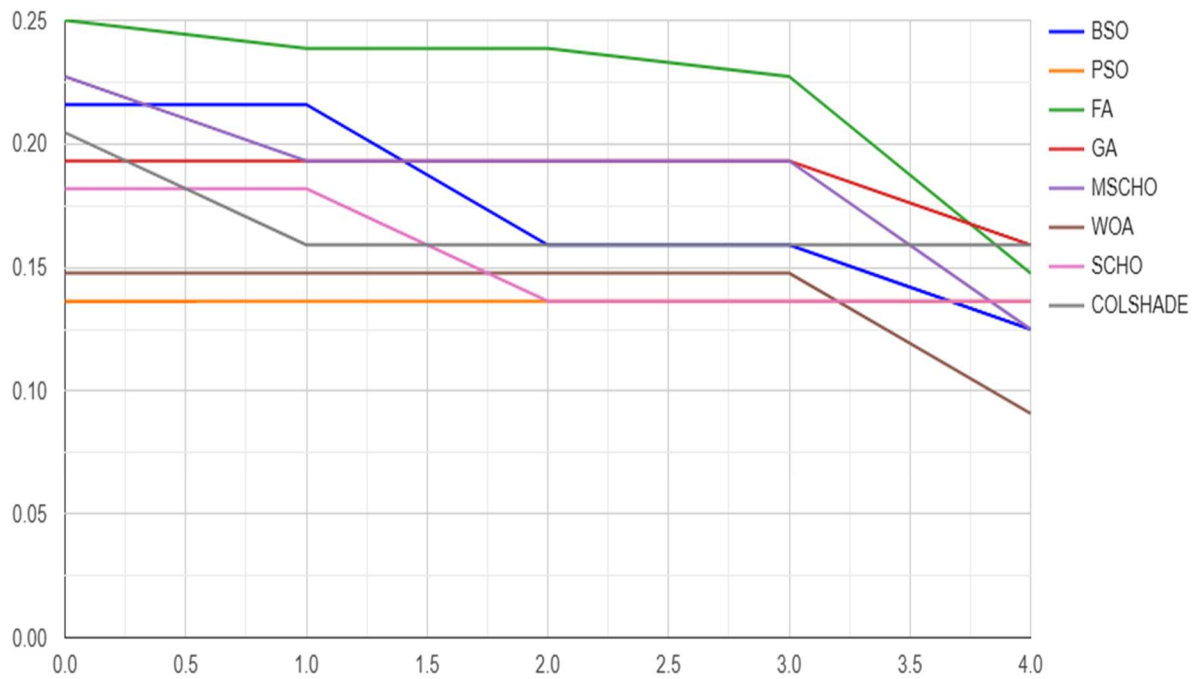
Слика 7.29. Експеримент4: Екстремни градијентни појачивач: *Boxplot* дијаграм функције циља

На слици 7.30. приказан је виолина дијаграм за индикатор функцију.

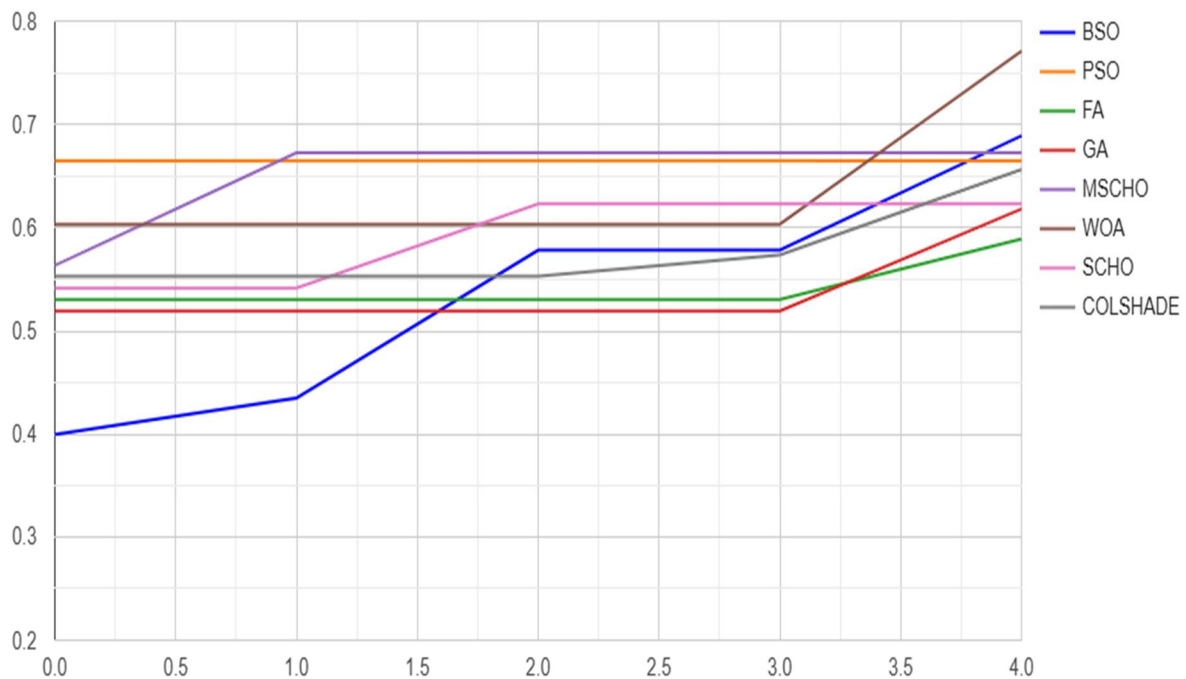


Слика 7.30. Експеримент4: Екстремни градијентни појачивач: Виолина дијаграм за индикатор функцију

На слици 7.31. могу се видети конвергенцијске стопе за функцију циља, док се на слици 7.32. могу видети конвергенције стопе за индикатор функцију.

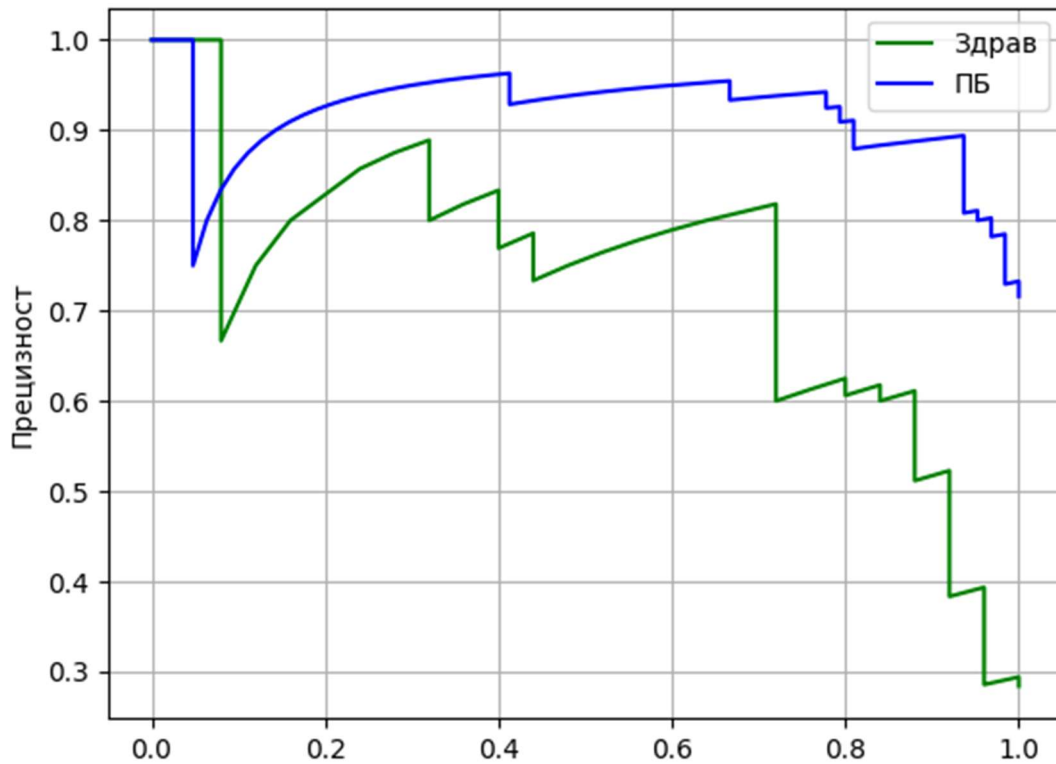


Слика 7.31. Експеримент4: Екстремни градијентни појачивач: Конвергенцијске стопе за функцију циља

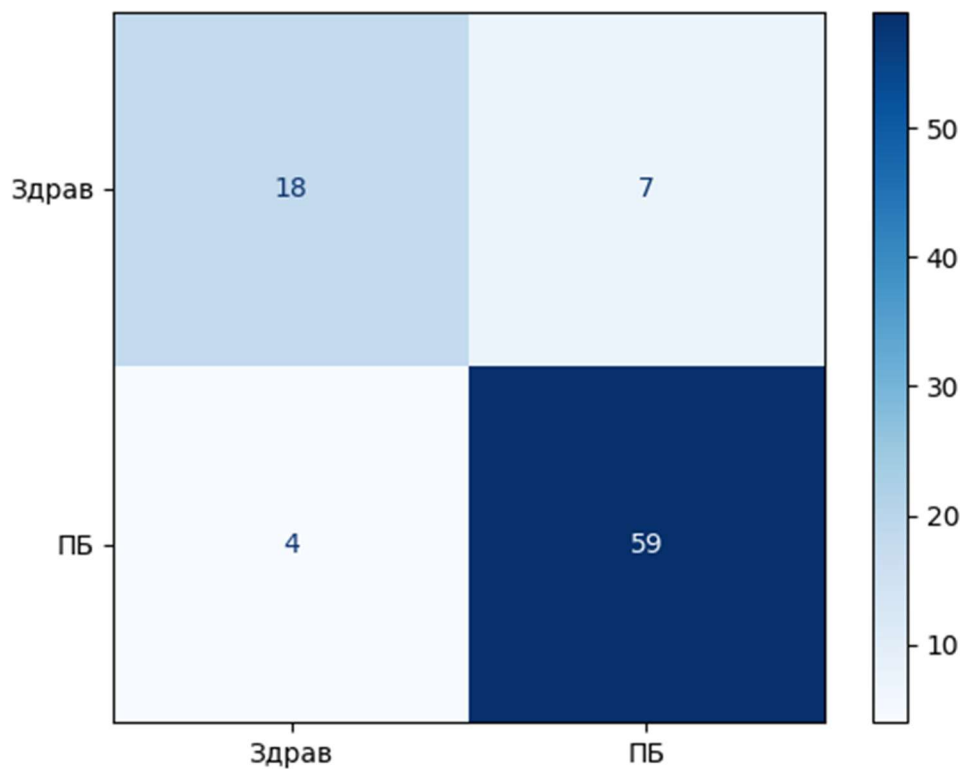


Слика 7.32. Експеримент4: Екстремни градијентни појачивач: Конвергенцијске стопе за индикатор функцију

На слици 7.33. налази се крива прецизности и опозива за најбољи модел који је постигла модификована метахеуристика, док се на слици 7.34. налази конфузиона матрица за исти модел.



Слика 7.33. Експеримент4: Екстремни градијентни појачивач: Крива прецизности и опозива



Слика 7.34. Експеримент4: Екстремни градијентни појачивач: Конфузиона матрица

7.3. Статистичка доказивања

Иако сами резултати експеримената могу бити добар показатељ да је постигнуто задовољавајуће решење, и да је ново унапређење супериорније у односу на претходна решења, неопходно је то потврдити и статистичким тестовима.

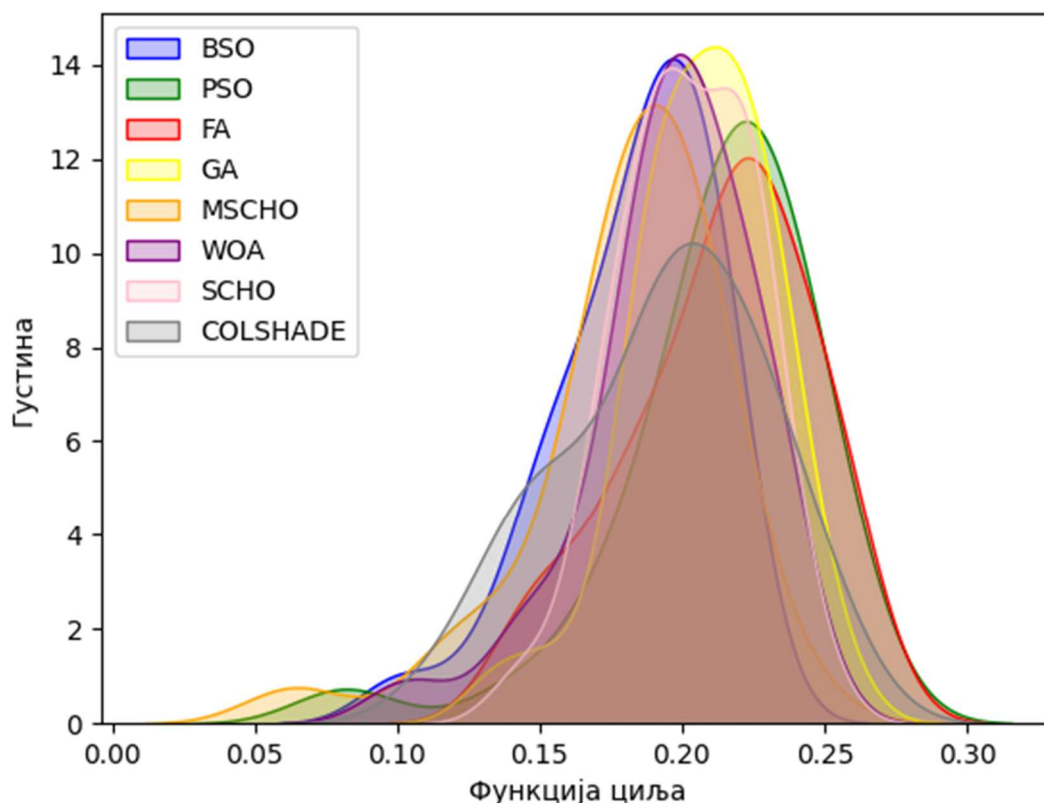
Како би постојало веће поверење у резултате све метахеуристике су имале популацију величине 5 јединки, и максималан број итерација по покретању је 5. Свака метахеуристика је покренута у по 30 независних покретања.

Тестови који се спроводе могу бити параметријски и непараметријски [91]. За извршавање параметријских тестова неопходно је да буду испуњени одговарајући услови, а то су:

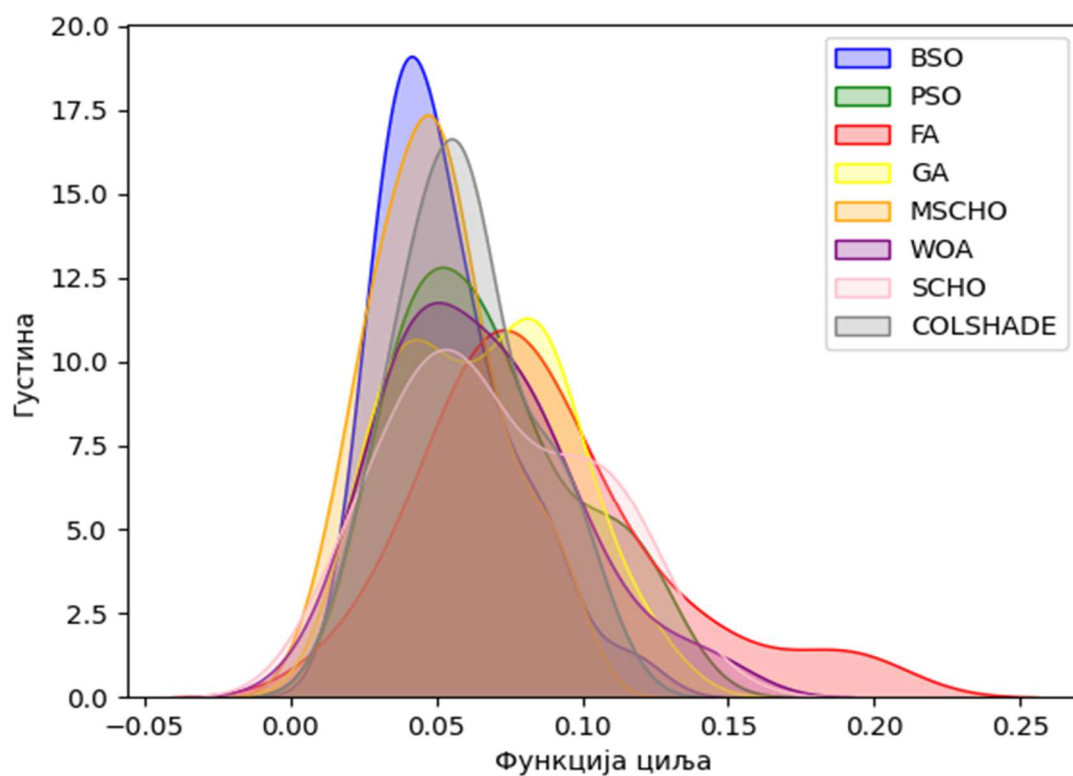
- независност,
- нормалност у погледу расподеле,
- и хомоскедастичност.

Уколико не постоје ови услови онда се извршава непараметријски тест.

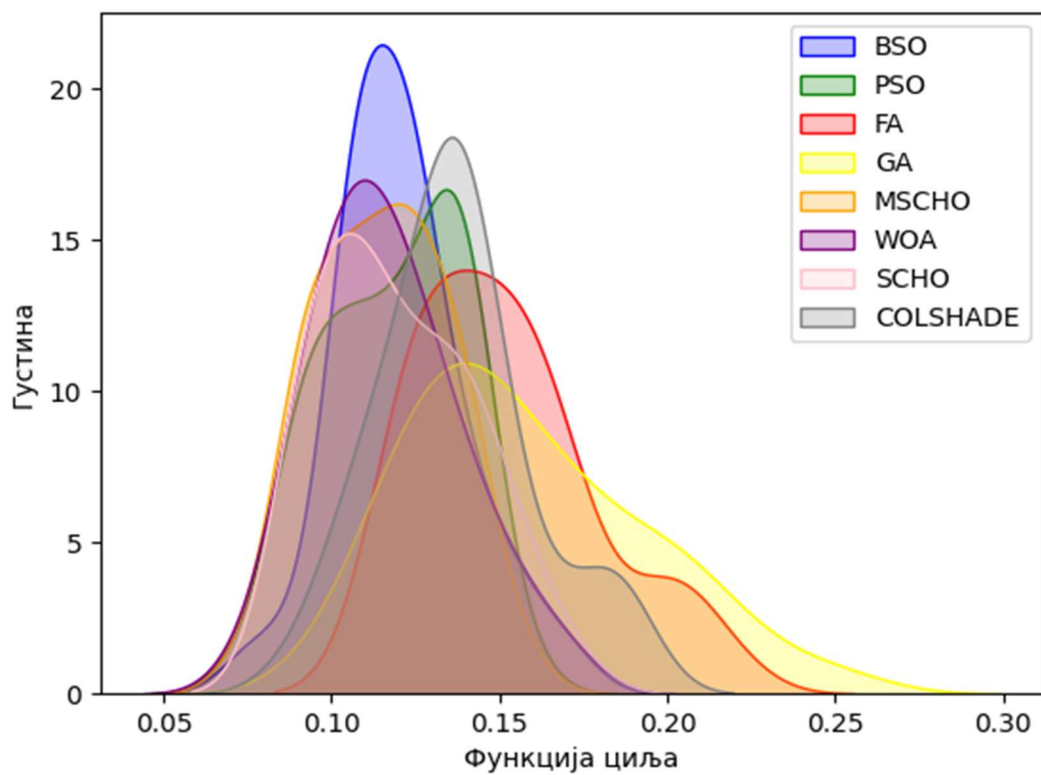
Независност је испуњена за све експерименте јер је свако од 30 покретања метахеуристике иницијализовано посебним псеудонасумичним бројем. Питање нормалне расподеле већ није испуњено за све експерименте, и то се види на сликама 7.35., 7.36. и 7.37. где су приказани графици густине језгра (енг. *Kernel density estimate plot*) за прва три експеримента.



Слика 7.35. KDE график за први експеримент



Слика 7.36. KDE график за други експеримент



Слика 7.37. KDE график за трећи експеримент

Додатно неиспуњење услова нормалности за сваки од експеримената је потврђено и Шафир-Вилк тестом [92], чији се резултати (п-вредности) могу видети у табели 24. За п-вредности мање од 0.05 може се сматрати да подаци не прате нормалну расподелу.

Табела 24. П-вредности за Шафир-Вилк тест, за све експерименте

Експеримент	BSO	PSO	FA	GA	MSCHO	WOA	SCHO	COLSHADE
1	0.0369	0.0006	0.0991	0.3741	0.0040	0.0158	0.6900	0.4845
2	0.0124	0.0517	0.0173	0.2755	0.5932	0.2982	0.2984	0.5657
3	0.0192	0.0093	0.0029	0.0661	0.0501	0.0872	0.0129	0.0176
4	0.2319	0.3183	0.0703	0.1143	0.4799	0.0731	0.0804	0.1385

Из табеле 24 се види да се за прва три експеримента не испуњава услов нормалности, па се самим тим не могу радити параметријски тестови, док се за четврти експеримент мора утврдити и хомоскедастичност.

Хомоскедастичност или једнакост варијанси се испитује Левановим тестом [93]. За п-вредности теста веће од 0.05 сматра се да постоји испуњеност хомоскедастичности. Резултати теста су приказани у табели 25.

Табела 25. П-вредности за Леванов тест, за четврти експеримент

Метахеуристика	П-Вредност
BSO	1.0
PSO	0.92
FA	0.31
GA	0.91
MSCHO	0.77
WOA	0.06
SCHO	1.0
COLSHADE	0.15

Из табеле 25 се види да су за четврти експеримент испуњени сви услови да се изврши параметријски тест, док ће се за прва три експеримента вршити непараметријски тестови.

Од непараметријских тестова за прва три експеримента врши се Вилкоксонов тест [94], где се *MSCHO* пореди са свим осталим, а резултати (п-вредности) се могу видети у табели 26. За п-вредности мање од 0.05 сматра се да постоје статистички докази да је *MSCHO* значајно другачија од друге са којом се пореди и да то није случајно.

Табела 26. П-вредности за Вилкоксонов тест, за прва три експерименте

Експеримент	BSO	PSO	FA	GA	WOA	SCHO	COLSHADE
1	0.227	0.002	0.0002	0.002	0.032	0.002	0.124
2	0.292	0.011	0.000009	0.002	0.0128	0.0064	0.074
3	0.219	0.221	0.000001	0.000001	0.472	0.209	0.0003

Табела 26 показује да се за први и други експеримент *MSCHO* може класификовати у ранг заједно са *BSO* и *COLSHADE* метахеуристикама, док показује статистички значајне доказе да даје боље преформансе од свих осталих метахеуристика. Како би се видело додатно поређење са *BSO* и *COLSHADE* метахеуристикама, у табели 27. дат је приказ Клифове делте [95], која приказује доминацију једног алгоритма у поређењу са другим. Позитивне вредности у табели показују да је *MSCHO* метахеуристика боља.

Табела 27. Клифове делте за прва два експеримента

Експеримент 1		Експеримент 2	
<i>MSCHO</i> против <i>BSO</i>	0.03	<i>MSCHO</i> против <i>BSO</i>	0.064
<i>MSCHO</i> против <i>COLSHADE</i>	0.197	<i>MSCHO</i> против <i>COLSHADE</i>	0.262

Табела 26 показује да се за трећи експеримент *MSCHO* може класификовати у ранг заједно са *BSO*, *PSO*, *WOA* и *SCHO* метахеуристикама, док показује статистички значајне доказе да даје боље преформансе од свих осталих метахеуристика. Како би се видело додатно поређење са *BSO*, *PSO*, *WOA* и *SCHO* метахеуристикама, у табели 28. дат је приказ Клифове делте, која приказује доминацију једног алгоритма у поређењу са другим. Позитивне вредности у табели показују да је *MSCHO* метахеуристика боља.

Табела 28. Клифове делте за трећи експеримент

Експеримент 3	
<i>MSCHO</i> против <i>BSO</i>	0.143
<i>MSCHO</i> против <i>PSO</i>	0.111
<i>MSCHO</i> против <i>WOA</i>	0.015
<i>MSCHO</i> против <i>SCHO</i>	0.095

За параметријски тест за четврти експеримент врши се Т-тест [96], који је заснован на средњим вредностима. П-вредности за поређење *MSCHO* метахеуристике са осталим у Т-тесту, виде се у табели 29. За п-вредности мање од 0.05 сматра се да постоје статистички докази да је *MSCHO* метахеуристика значајно другачија од друге са којом се пореди и да то није случајно.

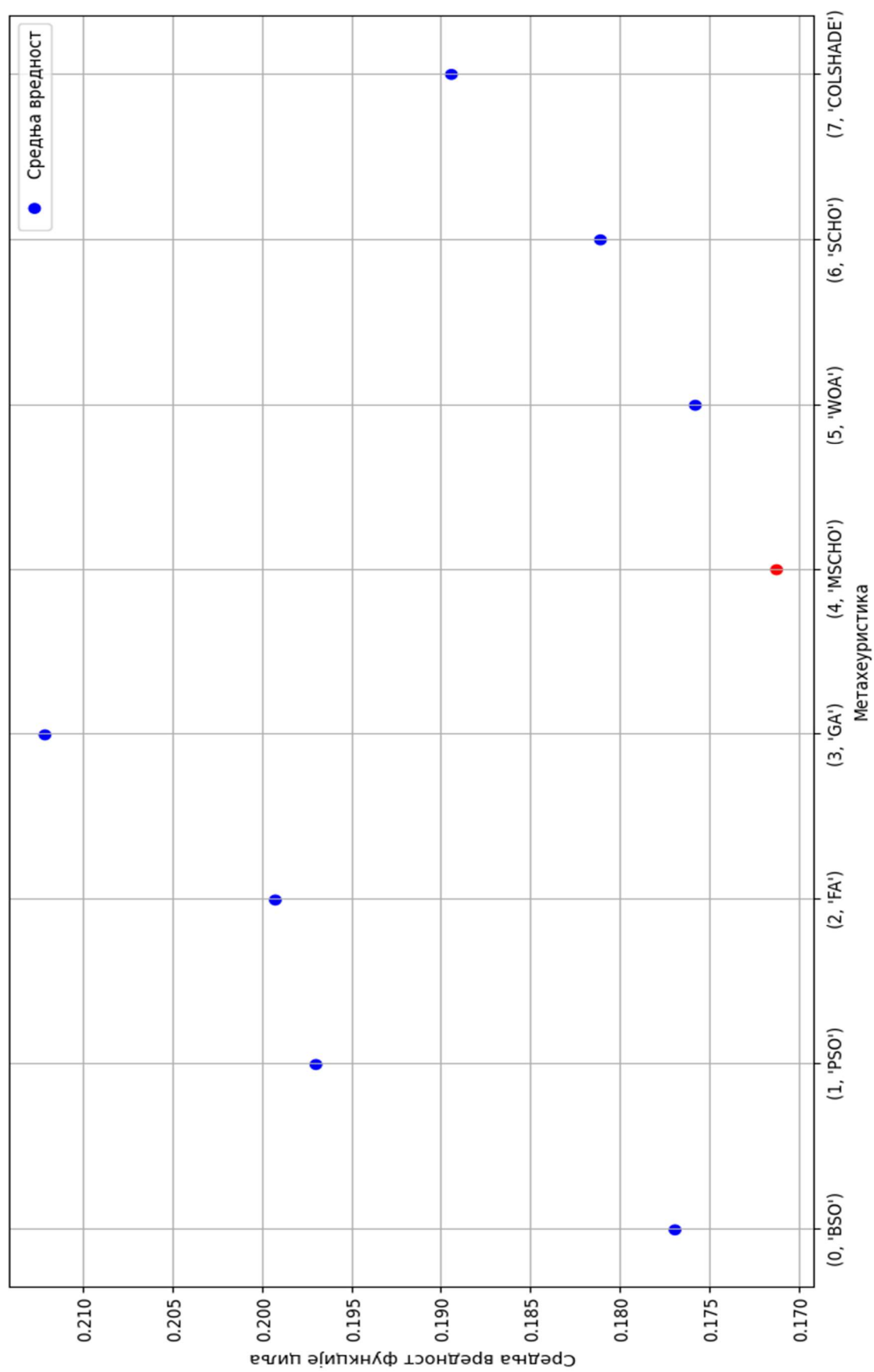
Табела 29. П-вредности за Т-тест, за четврти експеримент

Експеримент	<i>BSO</i>	<i>PSO</i>	<i>FA</i>	<i>GA</i>	<i>WOA</i>	<i>SCHO</i>	<i>COLSHADE</i>
4	0.272	0.0004	0.000008	0.00000007	0.467	0.107	0.0006

За додатно поређење са осталим метахеуристикама за које није показано значајно статистичко одступање, као мера поређења може се узети и средња вредност функције циља кроз сва покретања, у чему је *MSCHO* најбоља са средњом вредношћу функције циља од 0.171, на другом месту је *WOA* са 0.175, на трећем месту *BSO* са 0.176 и на четвртм месту *SCHO* са 0.181.

Средње вредности функција циља за све метахеуристике за четврти експеримент се могу видети на слици 7.38.

Слика 7.38. Средње вредности функције циља за четврти експеримент



Додатно се приликом израде саме докторске дисертације поставило неколико хипотеза које су потврђене у наставку поглавља:

1. Машинско учење се користи у циљу побољшања резултата анализе података у случајевима када су подаци комплексни и зависности међу подацима могу промаћи људском оку

Машинско учење се већ деценијама користи у циљу побољшања резултата у разним научним областима, што се може видети и кроз стручну литературу. Његова примена омогућава ефикасно решавање сложених проблема у образовању [97], медицини [98], енергетици [99] и другим сферама.

2. Детектовање Паркинсонове болести и других неуродегенеративних обољења може бити унапређено применом модела машинског учења

Неуродегенеративна обољења значајно урушавају квалитет живота оболелих пацијената, а научници применом модела машинског учења раде на што ранијој и успешнијој детекцији болести, а преглед литературе у овој области је дат у поглављу 4.

3. Перформансе модела машинског учења могу бити побољшане оптимизовањем хиперпараметара модела, коришћењем алгоритама метахеуристике

За максимално искоришћење модела машинског учења неопходно је оптимизовати хиперпараметре модела, а начини оптимизације описани су у поглављу 5. У резултатима експеримената се види да поређене метахеуристике доводе моделе до високих прецизности, а и у литератури постоји доста примера оптимизације хиперпараметара метахеуристикама, као што су [100] или [101].

4. Метахеуристике се могу имплементационо прилагодити тако да решавају проблем оптимизације хиперпараметара модела машинског учења који детектују неуродегенеративна обољења

Све поређене метахеуристике су могле да се прилагоде да унапреде основне моделе приликом детекције неуродегенеративних обољења, а сами резултати експеримената су приказани у поглављу 7.2.

5. Постојећи алгоритми метахеуристике могу бити модификовани у циљу бољег прилагођавања решавању конкретних проблема, и тако модификовани алгоритми могу се користити и у другим сферама науке у којима је потребно решавати проблем оптимизације хиперпараметара

SCHO метахеуристика унапређена је техником вештачке колоније пчела и квази-рефлексивним учењем, и тако модификована метахеуристика (*MSCHO*) је дала у већини случајева најбоље резултате који су статистички доказиви као што је описано пре, у овом поглављу. *MSCHO* метахеуристика примењена је и на различитим скуповима података, што показује њену разноврсну примену.

8. Будућа унапређења

У сваком раду, као што је то и случај са овом тезом, постоји простор за унапређења. Важно је да се напомене која су то унапређења, како би се отворио пут ка новим, будућим истраживањима. Неки од потенцијалних праваца даљег развоја су следећи:

- **Проширење постојећег скупа података о тапкању прстима**

Квалитетни подаци као што је то случај у коришћеном скупу података о тапкању прстима, су од изузетног значаја, иако је познато да је прикупљање ових података изузетно скупо, још већи број учесника који би учествовао у овом истраживању би додатно унапредио моделе који се користе за класификацију. Са већим бројем пацијената већи број података би се давао моделима, да уче зависности, могао би се већи број података давати и самим метахеуристикама да подешавају хиперпараметре модела, и идеалан сценарио би био да остане и довољан скуп репрезентативних примера за евалуацију.

- **Проширења у погледу метахеуристика**

Имплементиране метахеуристике користе популације од пет јединки, али већи број јединки у популацији би проширио простор претраге и потенцијално би се дошло до бољих решења. Такође и већи број итерација би помогао да се пронађе бољи минимум. Ова проширења захтевају много више ресурса у погледу времена извршавања експеримената, и могућа су уколико би се извршавање пребацило на хардверски јаче конфигурације уз паралелизацију у оквиру самог покретања једне метахеуристике.

Осим поређених метахеуристика додатно би се могле испитивати и друге метахеуристике, како би потенцијално дале боље резултате, или додатно утврдиле постојеће.

- **Истраживање нових модела машинског учења**

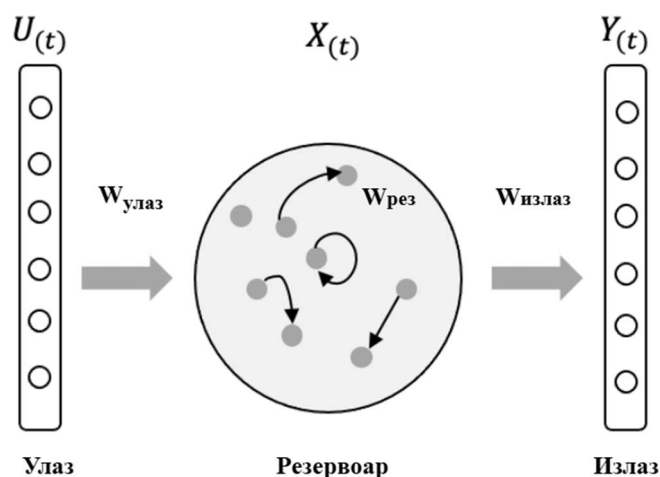
Поред коришћених модела рекурентних неуралних мрежа и рекурентних неуралних мрежа са механизмом пажње, као и екстремног градијентног појачивача, и даље постоје могућности за нове моделе машинског учења, који потенцијално дошли до бољих резултата.

Један од таквих модела су мреже ехо стања. Мреже ехо стања су специфична врста рекурентних неуралних мрежа које су обогачене резервоаром. Главна улога резервоара је да увиђа комплексне зависности међу подацима, што га чини погодним за рад са временским серијама.

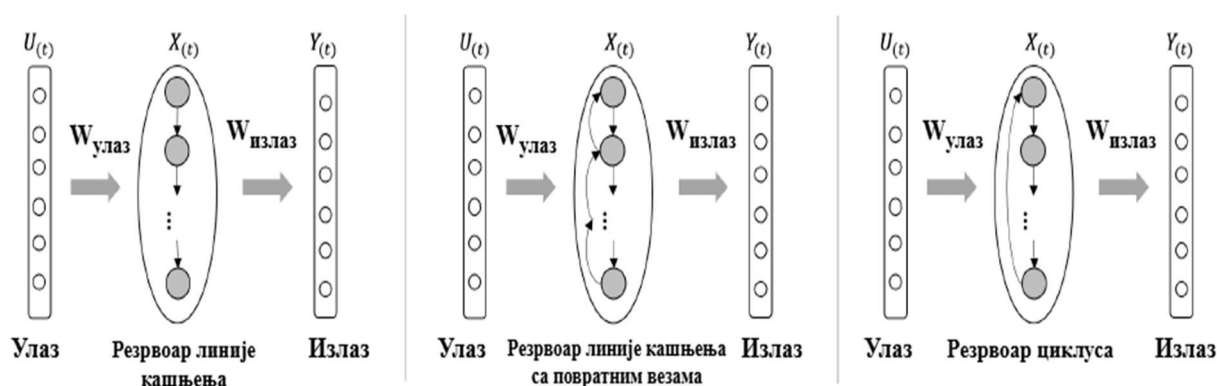
Мрежа ехо стања [102] се састоји из улазног слоја података, резервоара и излазног слоја. Сам резервоар се састоји од повезаних неурона са насумично иницијализованим тежинама. Када се врши предикција улазни подаци се комбинују са стањем резервоара и тако се долази до излазне вредности. Тежине у резервоару се током процеса тренирања не мењају, чиме се на специфичан начин долази до увиђања шаблона у подацима.

Структура мреже ехо стања може се видети на слици 8.1., а неки од начина повезивања неурона у резервоару на слици 8.2. Мрежа ехо стања може да се комбинује са другим моделима и примери таквих структура се виде на слици 8.3.

Предности ехо мреже стања су ефикасност у тренирању, једноставност имплементације и могућност прилагођавања великом броју проблема, док су мане осетљивост на избор хиперпараметара и бојазан од претренирања.



Слика 8.1. Структура ехо мреже стања [102]



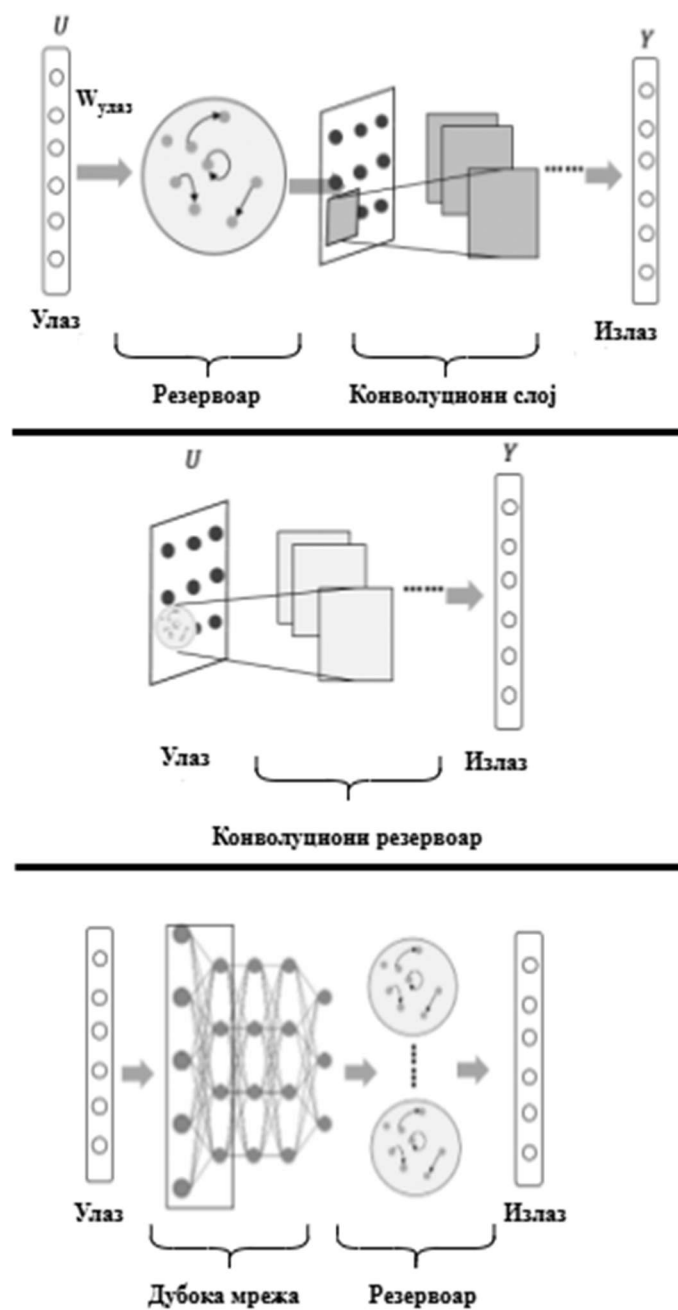
Слика 8.2. Начини повезивања неурона у резервоару [102]

- **Комбиновање више извора података**

Потенцијална унапређења и додатна сигурност у резултате би се постигла уз комбиновање са другим изворима података. За особу за коју се сумња да има Паркинсонову болест, поред теста тапкања прстима може и да одради и тестове хода, и да оба модела дају своје предикције па да се тиме додатно повећа извесност резултата. Овакав приступ је потенцијално компликованији за појединца, па би додатно улагање у приступачност тестирања крајњим корисницима био још један правац могућег унапређења.

Апликације на мобилним телефонима које би купиле податке са сензора би могле да детектују неправилности у ходу. Исте апликације би могле преко својих екрана да спроводе тестове цртања облика и тапкања прстима, што је већ коришћено у претходно наведеним радовима као поуздани начини за детектовање неуродегенеративних обољења. Резултати оваквих апликација могу додатно помоћи лекарима при финалној детекцији болести.

Крајњи циљ је увек рано детектовање обољења и омогућавање појединцима да што једноставније дођу до тестова. Сва унапређења која воде ка овом правцу су добродошла и представљају помак ка квалитетнијем животу.



Слика 8.3. Комбиновање модела машинског учења са ехо мрежама стања [102]

9. Закључак

Неуродегенеративна обољења попут Паркинсонове болести, Алцхајмерове болести и других значајно могу да утичу на квалитет живота појединца. Најчешћи симптоми Паркинсонове болести су моторни попут нестабилности у ходу и дрхтања, али могу бити и немоторни попут несанице или дигестивних проблема. Као главни разлог настанка Паркинсонове болести и сличних обољења сматра се одумирање ћелија које луче допамин, док се тачан узрочник још увек не може утврдити са сигурношћу. Поред генетских предиспозиција, постоје и фактори утицаја из окружења, а чим је узрочник непрецизно дефинисан само спречавање настанка болести је скоро немогуће. Лечење болести спроводи се лековима који су задужени за подстицање лучења допамина, али су они задужени за ублажавање симптома и не могу уклонити болест из организма.

Детектовање болести се врши прегледом лекара, анализом снимака главе, али коначан тест не постоји. Такође поред проблематике детекције Паркинсонове болести, морају се поменути и неуродегенеративна обољења која подсећају на Паркинсову болест, али се никако не смеју исто третирати. Мултипла системска атрофија и прогресивна супрануклеарна парализа су таква обољења која су и обрађена у овој тези. Због великог броја непознаница у овом проблему од великог значаја је рано и тачно детектовање болести.

Иако медицина напредује из дана у дан, мултидисциплинарни приступ детектовању болести је веома важан. Машинско учење је грана у програмирању која се све више развија и постаје наша свакодневица. Уз помоћ модела машинског учења постоји могућност увиђања зависности међу подацима, које би можда природном анализом остале непримећене.

Модел машинског учења могу решавати разнолике проблеме, а најчешће су то проблеми класификације и регресије. У проблему класификације се потребно да се подаци сврстају у одређену групу, док се у проблему регресије предвиђа континуалан број. Постоји велики број алгорита који могу решавати ове проблеме, а најпознатији су стабла одлучивања, неуралне мреже, линеарна и логистичка регресија, метода потпорних вектора, метода k најближих суседа и други. Сваки алгоритам машинског учења има своје хиперпараметре од којих у великој мери зависе сами резултати модела. Број тих хиперпараметара зависи од алгоритма до алгоритма, али може бити велики. Метода ручног подешавања и покушавања оптимизације ових параметара је врло неефикасна, па се због тога све више прелази на оптимизацију ових хиперпараметара метахеуристикама.

Метахеуристике су функције вишег реда, које на што оптималнији начин долазе до што бољег решења. Не могу да гарантују најоптималније решење, али је и довољно добро решење боље него да модел остане заглављен и без доброг решења. Метахеуристике налазе инспирацију у природи, математици, физици, а неке од најпознатијих су инспирисане понашањем мравца, китова, вукова или физичким појавама попут гравитације и центрипеталне силе.

У овој тези се врши детектовање Паркинсонове болести и других неуродегенеративних обољења на основу скупа података о тапкању прстима и скупа података о ходу. Коришћени модели машинског учења су рекурентне неуралне мреже, рекурентне неуралне мреже обогаћене механизмом пажње и екстремни градијентни појачивач. Сваки од модела има своје хиперпараметре који су оптимизовани метахеуристикама.

У случају рекурентних неуралних мрежа оптимизовани су број рекурентних слојева у мрежи, број неурона у сваком слоју, фактор одбацивања, стопа учења и број епоха, док се за рекурентне неуралне мреже са механизмом пажње додатно оптимизује и број неурона у слоју пажње. У случају екстремног градијентног појачивача оптимизује се стопа учења, минимална сума Хесијана у чвору, проценат тренинг података који се узима, проценат особина које се случајно бирају при конструкцији новог стабла, максимална дубина стабла и параметар γ .

Метахеуристике које су оптимизовале моделе су *BSO*, *PSO*, *FA*, *GA*, *WOA*, *SCHO* и *COLSHADE*, уз нову модификовану верзију *SCHO* метахеуристике, *MSCHO*. *MSCHO* метахеуристика унапређује основу верзију алгоритма техникама вештачке колоније пчела и квази-рефлексивног учења.

Све поређене метахеуристике су се извршавале у 30 независних покретања, са популацијом од 5 агената, и са максималним бројем итерација од 5. Приликом оптимизације хиперпараметара метахеуристикама минимизована је стопа грешке (1-тачност), а додатно је и максимизован Коенов капа коефицијент као индикатор функција.

У тези су укупно представљена четири експеримента. Прва три експеримента су рађена над скупом података о тапкању прстима, а четврти експеримент над скупом података о ходу.

Први експеримент представља резултате бинарне класификације, а као модел је коришћена рекурентна неурална мрежа. *MSCHO* метахеуристика дала је најбоље резултате за функцију циља (0.0640084) и за индикатор функцију (0.8710003).

Други експеримент представља резултате бинарне класификације, а као модел је коришћена рекурентна неурална мрежа са механизмом пажње. *MSCHO* метахеуристика дала је најбоље резултате за функцију циља (0.0104932), док је за индикатор функцију најбољи резултат дала *SCHO* метахеуристика (0.9749761), а на другом месту је *MSCHO* са индикатор функцијом од 0.9494558.

Трећи експеримент представља резултате мултикласне класификације, а као модел је коришћен екстремни градијентни појачивач. Из постојећих вредности података уз коришћење *tsfresh* библиотеке су извучене додатне одлике које су улазиле у модел. *MSCHO* метахеуристика дала је најбоље резултате за функцију циља (0.0769231) и за индикатор функцију (0.8972819).

Четврти експеримент представља резултате бинарне класификације, а као модел је коришћен екстремни градијентни појачивач. Главни допринос овог експеримента је показивање примене *MSCHO* метахеуристике и на другачијем скупу података. *WOA* метахеуристика дала је најбоље резултате за функцију циља (0.0909091), а на другом месту је *MSCHO* са функцијом циља од 0.125. *WOA* метахеуристика дала је најбоље резултате за индикатор функцију (0.7709824), а на трећем месту је *MSCHO* са индикатор функцијом од 0.6726596.

Потврђивање резултата експеримената урађено је и статистичким тестовима. За прва три експеримента није испуњен услов нормалности података, па је вршен непараметријски тест, Вилкоксонов тест, који је утврдио постојање статистички значајне разлике *MSCHO* метахеуристике у односу на већину осталих поређених. За метахеуристике за које се то није видело из поменутог теста, додатно су приложене вредности Клифове делте које потврђују доминацију *MSCHO* метахеуристике. За четврти експеримент се испуњавају услови за параметријски тест, па су приказани резултати Т-теста који приказују статистички значајне разлике *MSCHO* метахеуристике у односу на већину осталих поређених.

Као будућа унапређена и правци даљег развоја могућа су додатна проширења у погледу метахеуристике, веће популације и већи број поређених метахеуристике. Додатно је могућа имплементација и других модела машинског учења, као и примена хибридних техника детекције болести.

Рано детектовање неуродегенеративних обољења је од кључног значаја за што боље ефекте лечења болести које утичу на квалитет живота појединаца. Модели машинског учења оптимизовани метахеуристикама могу значајно допринети у овој области, и могу да служе као помоћ лекарима у дијагностици.

Литература

- [1] O. Hardiman and C. P. Doherty Editors, *Neurodegenerative Disorders*. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-23309-3.
- [2] G. Becker *et al.*, “Early diagnosis of Parkinson’s disease,” *J Neurol*, vol. 249, no. 0, pp. 1–1, Oct. 2002, doi: 10.1007/s00415-002-1309-9.
- [3] V. C. Vinod and A. H. S, “Nature inspired meta heuristic algorithms for optimization problems,” *Computing*, vol. 104, no. 2, pp. 251–269, Feb. 2022, doi: 10.1007/s00607-021-00955-5.
- [4] B. Alatas and U. Can, “Physics based Metaheuristic Optimization Algorithms for Global Optimization Physics Based Metaheuristic Algorithms for Global Optimization,” 2015. [Online]. Available: <http://www.aiscience.org/journal/ajiscehttp://creativecommons.org/licenses/by-nc/4.0/>
- [5] P. A. Lewis and J. E. Spillane, “An Introduction to Neurodegeneration,” in *The Molecular and Clinical Pathology of Neurodegenerative Disease*, Elsevier, 2019, pp. 1–23. doi: 10.1016/B978-0-12-811069-0.00001-X.
- [6] S. E. Hyman, “Neurotransmitters,” *Current Biology*, vol. 15, no. 5, pp. R154–R158, Mar. 2005, doi: 10.1016/j.cub.2005.02.037.
- [7] S. D. Iversen and L. L. Iversen, “Dopamine: 50 years in perspective,” May 2007. doi: 10.1016/j.tins.2007.03.002.
- [8] C. Missale, S. R. Nash, S. W. Robinson, M. Jaber, and M. G. Caron, “Dopamine Receptors: From Structure to Function,” 1998. [Online]. Available: www.physiology.org/journal/physrev
- [9] R. J. Castellani, R. K. Rolston, and M. A. Smith, “Alzheimer disease,” 2010, *Mosby Inc*. doi: 10.1016/j.disamonth.2010.06.001.
- [10] L. V. Kalia and A. E. Lang, “Parkinson’s disease,” Aug. 29, 2015, *Lancet Publishing Group*. doi: 10.1016/S0140-6736(14)61393-3.
- [11] A. Samii, J. G. Nutt, and B. R. Ransom, “Parkinson’s disease,” *The Lancet*, vol. 363, no. 9423, pp. 1783–1793, May 2004, doi: 10.1016/S0140-6736(04)16305-8.
- [12] P. A. Lewis and J. E. Spillane, “Parkinson’s Disease,” in *The Molecular and Clinical Pathology of Neurodegenerative Disease*, Elsevier, 2019, pp. 83–121. doi: 10.1016/B978-0-12-811069-0.00003-3.
- [13] M. Stamelou and K. P. Bhatia, “Atypical Parkinsonism,” *Neurol Clin*, vol. 33, no. 1, pp. 39–56, Feb. 2015, doi: 10.1016/j.ncl.2014.09.012.
- [14] N. R. McFarland, “Diagnostic Approach to Atypical Parkinsonian Syndromes,” *CONTINUUM: Lifelong Learning in Neurology*, vol. 22, no. 4, pp. 1117–1142, Aug. 2016, doi: 10.1212/CON.0000000000000348.
- [15] A. Quattrone *et al.*, “Magnetic Resonance Planimetry in the Differential Diagnosis between Parkinson’s Disease and Progressive Supranuclear Palsy,” Jul. 01, 2022, *MDPI*. doi: 10.3390/brainsci12070949.
- [16] B. Mahesh, “Machine Learning Algorithms - A Review,” *International Journal of Science and Research (IJSR)*, vol. 9, no. 1, pp. 381–386, Jan. 2020, doi: 10.21275/art20203995.

- [17] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science* (1979), vol. 349, no. 6245, pp. 255–260, Jul. 2015, doi: 10.1126/science.aaa8415.
- [18] J. C. Stoltzfus, "Logistic regression: A brief primer," *Academic Emergency Medicine*, vol. 18, no. 10, pp. 1099–1104, Oct. 2011, doi: 10.1111/j.1553-2712.2011.01185.x.
- [19] D. Berrar, "Bayes' theorem and naive bayes classifier," in *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, vol. 1–3, Elsevier, 2018, pp. 403–412. doi: 10.1016/B978-0-12-809633-8.20473-1.
- [20] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "LNCS 2888 - KNN Model-Based Approach in Classification," 2003.
- [21] K. Chomboon, P. Chujai, P. Teerarassammee, K. Kerdprasop, and N. Kerdprasop, "An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm," in *The Proceedings of the 2nd International Conference on Industrial Application Engineering 2015*, The Institute of Industrial Applications Engineers, 2015, pp. 280–285. doi: 10.12792/iciae2015.051.
- [22] D. A. Pisner and D. M. Schnyer, "Support vector machine," in *Machine Learning: Methods and Applications to Brain Disorders*, Elsevier, 2019, pp. 101–121. doi: 10.1016/B978-0-12-815739-8.00006-7.
- [23] A. Patle and D. S. Chouhan, "SVM kernel functions for classification," in *2013 International Conference on Advances in Technology and Engineering (ICATE)*, IEEE, Jan. 2013, pp. 1–9. doi: 10.1109/ICAdTE.2013.6524743.
- [24] S. B. Kotsiantis, "Decision trees: A recent overview," Apr. 2013. doi: 10.1007/s10462-011-9272-4.
- [25] R. Polikar, "Ensemble Learning," in *Ensemble Machine Learning*, C. Zhang and Y. Ma, Eds., New York, NY: Springer New York, 2012, pp. 1–34. doi: 10.1007/978-1-4419-9326-7_1.
- [26] L. Breiman, "Random Forests," 2001. doi: <https://doi.org/10.1023/A:1010933404324>.
- [27] A. Mayr, H. Binder, O. Gefeller, and M. Schmid, "The evolution of boosting algorithms: From machine learning to statistical modelling," *Methods Inf Med*, vol. 53, no. 6, pp. 419–427, 2014, doi: 10.3414/ME13-01-0122.
- [28] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. in Springer Series in Statistics. New York, NY: Springer New York, NY, 2009. doi: 10.1007/978-0-387-84858-7.
- [29] T. Chen and C. Guestrin, "XGBoost," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.
- [30] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," May 27, 2015, *Nature Publishing Group*. doi: 10.1038/nature14539.
- [31] A. D. Dongare, R. R. Kharde, and A. D. Kachare, "Introduction to Artificial Neural Network," 2008.
- [32] S. Sharma, S. Sharma, and A. Athaiya, "ACTIVATION FUNCTIONS IN NEURAL NETWORKS," 2020. [Online]. Available: <http://www.ijeast.com>
- [33] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, "Recent Advances in Recurrent Neural Networks," Dec. 2017, [Online]. Available: <http://arxiv.org/abs/1801.01078>

- [34] G. Van Houdt, C. Mosquera, and G. Nápoles, “A review on the long short-term memory model,” *Artif Intell Rev*, vol. 53, no. 8, pp. 5929–5955, Dec. 2020, doi: 10.1007/s10462-020-09838-1.
- [35] X. H. Le, H. V. Ho, G. Lee, and S. Jung, “Application of Long Short-Term Memory (LSTM) neural network for flood forecasting,” *Water (Switzerland)*, vol. 11, no. 7, 2019, doi: 10.3390/w11071387.
- [36] T. Luong, H. Pham, and C. D. Manning, “Effective Approaches to Attention-based Neural Machine Translation,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, PA, USA: Association for Computational Linguistics, Aug. 2015, pp. 1412–1421. doi: 10.18653/v1/D15-1166.
- [37] D. Bahdanau, K. Cho, and Y. Bengio, “Neural Machine Translation by Jointly Learning to Align and Translate,” Sep. 2014, [Online]. Available: <http://arxiv.org/abs/1409.0473>
- [38] K. O’Shea and R. Nash, “An Introduction to Convolutional Neural Networks,” Nov. 2015, [Online]. Available: <http://arxiv.org/abs/1511.08458>
- [39] S. Chakraborty, S. Aich, and H. C. Kim, “Detection of Parkinson’s disease from 3T T1 weighted MRI scans using 3D convolutional neural network,” *Diagnostics*, vol. 10, no. 6, Jun. 2020, doi: 10.3390/diagnostics10060402.
- [40] P. M. Shah, A. Zeb, U. Shafi, S. F. A. Zaidi, and M. A. Shah, “Detection of Parkinson Disease in Brain MRI using Convolutional Neural Network,” in *2018 24th International Conference on Automation and Computing (ICAC)*, IEEE, Sep. 2018, pp. 1–6. doi: 10.23919/ICoNAC.2018.8749023.
- [41] S. Shinde *et al.*, “Predictive markers for Parkinson’s disease using deep neural nets on neuromelanin sensitive MRI,” *Neuroimage Clin*, vol. 22, Jan. 2019, doi: 10.1016/j.nicl.2019.101748.
- [42] S. A. A. Shah, L. Zhang, and A. Bais, “Dynamical system based compact deep hybrid network for classification of Parkinson disease related EEG signals,” *Neural Networks*, vol. 130, pp. 75–84, Oct. 2020, doi: 10.1016/j.neunet.2020.06.018.
- [43] S. K. Khare, V. Bajaj, and U. R. Acharya, “PDCNNNet: An Automatic Framework for the Detection of Parkinson’s Disease Using EEG Signals,” *IEEE Sens J*, vol. 21, no. 15, pp. 17017–17024, Aug. 2021, doi: 10.1109/JSEN.2021.3080135.
- [44] W. Wang, J. Lee, F. Harrou, and Y. Sun, “Early Detection of Parkinson’s Disease Using Deep Learning and Machine Learning,” *IEEE Access*, vol. 8, pp. 147635–147646, 2020, doi: 10.1109/ACCESS.2020.3016062.
- [45] M. Hireš, M. Gazda, P. Drotár, N. D. Pah, M. A. Motin, and D. K. Kumar, “Convolutional neural network ensemble for Parkinson’s disease detection from voice recordings,” *Comput Biol Med*, vol. 141, Feb. 2022, doi: 10.1016/j.combiomed.2021.105021.
- [46] M. B. Er, E. Isik, and I. Isik, “Parkinson’s detection based on combined CNN and LSTM using enhanced speech signals with Variational mode decomposition,” *Biomed Signal Process Control*, vol. 70, Sep. 2021, doi: 10.1016/j.bspc.2021.103006.
- [47] H. A. A. El Aal, S. A. Taie, and N. El-Bendary, “An optimized rnn-lstm approach for parkinson’s disease early detection using speech features,” *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, pp. 2503–2512, Oct. 2021, doi: 10.11591/eei.v10i5.3128.

- [48] G. Abdurrahman and M. Sintawati, "Implementation of xgboost for classification of parkinson's disease," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2020. doi: 10.1088/1742-6596/1538/1/012024.
- [49] M. M. Nishat, T. Hasan, S. M. Nasrullah, F. Faisal, M. A. A. R. Asif, and M. A. Hoque, "Detection of Parkinson's Disease by Employing Boosting Algorithms," in *2021 Joint 10th International Conference on Informatics, Electronics and Vision, ICIEV 2021 and 2021 5th International Conference on Imaging, Vision and Pattern Recognition, icIVPR 2021*, Institute of Electrical and Electronics Engineers Inc., 2021. doi: 10.1109/ICIEVICIVPR52578.2021.9564108.
- [50] M. Gil-Martín, J. M. Montero, and R. San-Segundo, "Parkinson's disease detection from drawing movements using convolutional neural networks," *Electronics (Switzerland)*, vol. 8, no. 8, Aug. 2019, doi: 10.3390/electronics8080907.
- [51] P. Khatamino, I. Canturk, and L. Ozyilmaz, "A Deep Learning-CNN Based System for Medical Diagnosis: An Application on Parkinson's Disease Handwriting Drawings," in *2018 6th International Conference on Control Engineering & Information Technology (CEIT)*, IEEE, Oct. 2018, pp. 1–6. doi: 10.1109/CEIT.2018.8751879.
- [52] I. Kamran, S. Naz, I. Razzak, and M. Imran, "Handwriting dynamics assessment using deep neural network for early identification of Parkinson's disease," *Future Generation Computer Systems*, vol. 117, pp. 234–244, Apr. 2021, doi: 10.1016/j.future.2020.11.020.
- [53] A. Moshkova, A. Samorodov, N. Voinova, A. Volkov, E. Ivanova, and E. Fedotova, "Parkinson's Disease Detection by Using Machine Learning Algorithms and Hand Movement Signal from LeapMotion Sensor," in *2020 26th Conference of Open Innovations Association (FRUCT)*, IEEE, Apr. 2020, pp. 321–327. doi: 10.23919/FRUCT48808.2020.9087433.
- [54] M. Belić, Z. Radivojević, V. Bobić, V. Kostić, and M. Đurić-Jovičić, "Quick computer aided differential diagnostics based on repetitive finger tapping in Parkinson's disease and atypical parkinsonisms," *Heliyon*, vol. 9, no. 4, Apr. 2023, doi: 10.1016/j.heliyon.2023.e14824.
- [55] E. Abdulhay, N. Arunkumar, K. Narasimhan, E. Vellaiappan, and V. Venkatraman, "Gait and tremor investigation using machine learning techniques for the diagnosis of Parkinson disease," *Future Generation Computer Systems*, vol. 83, pp. 366–373, Jun. 2018, doi: 10.1016/j.future.2018.02.009.
- [56] K. Açılcı, Ç. B. Erdaş, T. Aşuroğlu, M. K. Toprak, H. Erdem, and H. Oğul, "A random forest method to detect parkinson's disease via gait analysis," in *Communications in Computer and Information Science*, Springer Verlag, 2017, pp. 609–619. doi: 10.1007/978-3-319-65172-9_51.
- [57] B. E., B. D., and B. R., "Supervised machine learning based gait classification system for early detection and stage classification of Parkinson's disease," *Applied Soft Computing Journal*, vol. 94, Sep. 2020, doi: 10.1016/j.asoc.2020.106494.
- [58] D. Trabassi *et al.*, "Machine Learning Approach to Support the Detection of Parkinson's Disease in IMU-Based Gait Analysis," *Sensors*, vol. 22, no. 10, May 2022, doi: 10.3390/s22103700.
- [59] J. Camps *et al.*, "Deep learning for freezing of gait detection in Parkinson's disease patients in their homes using a waist-worn inertial measurement unit," *Knowl Based Syst*, vol. 139, pp. 119–131, 2018, doi: 10.1016/j.knosys.2017.10.017.

- [60] I. El Maachi, G. A. Bilodeau, and W. Bouachir, "Deep 1D-Convnet for accurate Parkinson disease detection and severity prediction from gait," *Expert Syst Appl*, vol. 143, Apr. 2020, doi: 10.1016/j.eswa.2019.113075.
- [61] S. Arora *et al.*, "Detecting and monitoring the symptoms of Parkinson's disease using smartphones: A pilot study," *Parkinsonism Relat Disord*, vol. 21, no. 6, pp. 650–653, Jun. 2015, doi: 10.1016/j.parkreldis.2015.02.026.
- [62] V. Sharma *et al.*, "LNCS 8519 - SPARK: Personalized Parkinson Disease Interventions through Synergy between a Smartphone and a Smartwatch," 2014.
- [63] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/j.neucom.2020.07.061.
- [64] K. E. S. Pilario, Y. Cao, and M. Shafiee, "A Kernel Design Approach to Improve Kernel Subspace Identification," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 7, pp. 6171–6180, Jul. 2021, doi: 10.1109/TIE.2020.2996142.
- [65] B. Bischl *et al.*, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *WIREs Data Mining and Knowledge Discovery*, vol. 13, no. 2, Mar. 2023, doi: 10.1002/widm.1484.
- [66] E.-Ghazali. Talbi, *Metaheuristics : from design to implementation*. John Wiley & Sons, 2009.
- [67] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, IEEE, 1995, pp. 1942–1948. doi: 10.1109/ICNN.1995.488968.
- [68] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview," *Soft comput*, vol. 22, no. 2, pp. 387–408, Jan. 2018, doi: 10.1007/s00500-016-2474-6.
- [69] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput Intell Mag*, vol. 1, no. 4, pp. 28–39, Nov. 2006, doi: 10.1109/MCI.2006.329691.
- [70] C. Blum, "Ant colony optimization: Introduction and recent trends," Dec. 2005. doi: 10.1016/j.plrev.2005.10.001.
- [71] X.-S. Yang and X. He, "Firefly algorithm: recent advances and applications," 2013. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/29693-firefly-algorithm>
- [72] I. Fister, X. S. Yang, and J. Brest, "A comprehensive review of firefly algorithms," *Swarm Evol Comput*, vol. 13, pp. 34–46, 2013, doi: 10.1016/j.swevo.2013.06.001.
- [73] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.
- [74] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, May 2016, doi: 10.1016/j.advengsoft.2016.01.008.
- [75] S. Cheng, Q. Qin, J. Chen, and Y. Shi, "Brain storm optimization algorithm: a review," *Artif Intell Rev*, vol. 46, no. 4, pp. 445–458, Dec. 2016, doi: 10.1007/s10462-016-9471-0.
- [76] S. Mirjalili, "Genetic algorithm," in *Studies in Computational Intelligence*, vol. 780, Springer Verlag, 2019, pp. 43–55. doi: 10.1007/978-3-319-93025-1_4.
- [77] U. Schroeders, O. Wilhelm, and G. Oлару, "Meta-Heuristics in short scale construction: Ant colony optimization and genetic algorithm," *PLoS One*, vol. 11, no. 11, Nov. 2016, doi: 10.1371/journal.pone.0167110.

- [78] Kenneth V. Price, Rainer M. Storn, and Jouni A. Lampinen, “The Differential Evolution Algorithm,” in *Differential Evolution*, Berlin/Heidelberg: Springer-Verlag, 2005, pp. 37–134. doi: 10.1007/3-540-31306-0_2.
- [79] R. Tanabe and A. S. Fukunaga, “Improving the search performance of SHADE using linear population size reduction,” in *2014 IEEE Congress on Evolutionary Computation (CEC)*, IEEE, Jul. 2014, pp. 1658–1665. doi: 10.1109/CEC.2014.6900380.
- [80] J. Gurrola-Ramos, A. Hernandez-Aguirre, and O. Dalmau-Cedeno, “COLSHADE for Real-World Single-Objective Constrained optimization Problems,” in *2020 IEEE Congress on Evolutionary Computation (CEC)*, IEEE, Jul. 2020, pp. 1–8. doi: 10.1109/CEC48606.2020.9185583.
- [81] J. Bai *et al.*, “A Sinh Cosh optimizer,” *Knowl Based Syst*, vol. 282, Dec. 2023, doi: 10.1016/j.knosys.2023.111081.
- [82] D. Karaboga and B. Basturk, “On the performance of artificial bee colony (ABC) algorithm,” *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 687–697, Jan. 2008, doi: 10.1016/j.asoc.2007.05.007.
- [83] E. Garoudja *et al.*, “Artificial bee colony algorithm: a novel strategy for optical constants and thin film thickness extraction using only optical transmittance spectra for photovoltaic applications,” *Optik (Stuttg)*, vol. 241, Sep. 2021, doi: 10.1016/j.ijleo.2021.167030.
- [84] N. Bacanin *et al.*, “Quasi-reflection learning arithmetic optimization algorithm firefly search for feature selection,” *Heliyon*, vol. 9, no. 4, Apr. 2023, doi: 10.1016/j.heliyon.2023.e15378.
- [85] A. J. Hughes, S. E. Daniel, L. Kilford, A. J. Lees, D. Society Brain Bank, and L. A. WCIN IPJ J Hughes S E Daniel L Kilford A J Lees, “Accuracy of clinical diagnosis of idiopathic Parkinson’s disease: a clinico-pathological study of 100 cases,” 1992.
- [86] I. Litvan *et al.*, “Validity and Reliability of the Preliminary NINDS Neuropathologic Criteria for Progressive Supranuclear Palsy and Related Disorders,” *J Neuropathol Exp Neurol*, vol. 55, no. 1, pp. 97–105, Jan. 1996, doi: 10.1097/00005072-199601000-00010.
- [87] S. Gilman *et al.*, “Second consensus statement on the diagnosis of multiple system atrophy,” *Neurology*, vol. 71, no. 9, pp. 670–676, Aug. 2008, doi: 10.1212/01.wnl.0000324625.00404.15.
- [88] G. Yogev, N. Giladi, C. Peretz, S. Springer, E. S. Simon, and J. M. Hausdorff, “Dual tasking, gait rhythmicity, and Parkinson’s disease: Which aspects of gait are attention demanding?,” *European Journal of Neuroscience*, vol. 22, no. 5, pp. 1248–1256, Sep. 2005, doi: 10.1111/j.1460-9568.2005.04298.x.
- [89] J. M. Hausdorff, J. Lowenthal, T. Herman, L. Gruendlinger, C. Peretz, and N. Giladi, “Rhythmic auditory stimulation modulates gait variability in Parkinson’s disease,” *European Journal of Neuroscience*, vol. 26, no. 8, pp. 2369–2375, Oct. 2007, doi: 10.1111/j.1460-9568.2007.05810.x.
- [90] S. Frenkel-Toledo, N. Giladi, C. Peretz, T. Herman, L. Gruendlinger, and J. M. Hausdorff, “Treadmill walking as an external pacemaker to improve gait rhythm and stability in Parkinson’s disease,” *Movement Disorders*, vol. 20, no. 9, pp. 1109–1114, Sep. 2005, doi: 10.1002/mds.20507.
- [91] A. LaTorre, D. Molina, E. Osaba, J. Poyatos, J. Del Ser, and F. Herrera, “A prescription of methodological guidelines for comparing bio-inspired optimization algorithms,” *Swarm Evol Comput*, vol. 67, p. 100973, Dec. 2021, doi: 10.1016/j.swevo.2021.100973.

- [92] S. S. Shapiro and R. S. Francia, "An approximate analysis of variance test for normality," *J Am Stat Assoc*, vol. 67, no. 337, pp. 215–216, 1972, doi: 10.1080/01621459.1972.10481232.
- [93] B. B. Schultz, "LEVENE'S TEST FOR RELATIVE VARIATION," 1985. [Online]. Available: <http://sysbio.oxfordjournals.org/>
- [94] S. M. Taheri and G. Hesamian, "A generalization of the Wilcoxon signed-rank test and its applications," *Statistical Papers*, vol. 54, no. 2, pp. 457–470, May 2013, doi: 10.1007/s00362-012-0443-4.
- [95] K. Meissel and E. S. Yao, "Using Cliff's Delta as a Non-Parametric Effect Size Measure: An Accessible Web App and R Tutorial," vol. 29, 2024, [Online]. Available: <https://cliffdelta.shinyapps.io/calculator;>
- [96] T. K. Kim, "T test as a parametric statistic," *Korean J Anesthesiol*, vol. 68, no. 6, p. 540, 2015, doi: 10.4097/kjae.2015.68.6.540.
- [97] K. Milenkovic, D. Draskovic, and B. Nikolic, "Educational software system for reasoning and decision making using Bayesian networks," in *2014 IEEE Global Engineering Education Conference (EDUCON)*, IEEE, Apr. 2014, pp. 1189–1194. doi: 10.1109/EDUCON.2014.7096840.
- [98] R. C. Deo, "Machine Learning in Medicine," *Circulation*, vol. 132, no. 20, pp. 1920–1930, Nov. 2015, doi: 10.1161/CIRCULATIONAHA.115.001593.
- [99] C.-G. Haba, "Monitoring Solar Panels using Machine Learning Techniques," in *2019 8th International Conference on Modern Power Systems (MPS)*, IEEE, May 2019, pp. 1–6. doi: 10.1109/MPS.2019.8759651.
- [100] N. Bacanin, T. Bezdan, E. Tuba, I. Strumberger, and M. Tuba, "Optimizing Convolutional Neural Network Hyperparameters by Enhanced Swarm Intelligence Metaheuristics," *Algorithms*, vol. 13, no. 3, p. 67, Mar. 2020, doi: 10.3390/a13030067.
- [101] P. R. Lorenzo, J. Nalepa, M. Kawulok, L. S. Ramos, and J. R. Pastor, "Particle swarm optimization for hyper-parameter selection in deep neural networks," in *Proceedings of the Genetic and Evolutionary Computation Conference*, New York, NY, USA: ACM, Jul. 2017, pp. 481–488. doi: 10.1145/3071178.3071208.
- [102] C. Sun, M. Song, S. Hong, and H. Li, "A Review of Designs and Applications of Echo State Networks," Dec. 2020, doi: 10.48550/arXiv.2012.02974.

Биографија аутора

Цинцовић Јелица рођена је 04.12.1994. у Београду. Основну школу завршила је као ђак генерације и носилац дипломе „Вук Караџић“. Потом је завршила Трећу београдску гимназију, такође као носилац дипломе „Вук Караџић“.

Након завршене средње школе 2013. уписала је Електротехнички факултет Универзитета у Београду, студијски програм Софтверско инжењерство. Током студија завршила је две стручне праксе у компанијама Asecco SEE и Avisto Eastern Europe. Такође је била демонстратор на неколико предмета при Катедри за рачунарску технику и информатику. Дипломирала је 2017. године са просечном оценом 8,93. Дипломски рад „Развој 2D видео-игре комбиновањем трке на табли са квизом знања“ одбранила је са оценом 10.

Дипломске академске - мастер студије на Електротехничком факултету у Београду, модул Софтверско инжењерство уписала је у октобру 2017. године. Мастер рад „Развој друштвене вишеплатформске 3Д видео-игре са интегрисаним квизом знања“ је одбранила 2019. године са оценом 10. Просечна оцена на мастер студијама је 10,00.

Докторске академске студије на Електротехничком факултету у Београду уписала је школске 2019/2020, на модулу Софтверско инжењерство, и положила је све испите са просечном оценом 10,00. Током мастер и докторских студија објавила је један рад као први аутор у часопису са SCI листе и 12 научних радова на домаћим и страним конференцијама као аутор или коаутор.

Од школске 2017/2018. године почела је са радом на Електротехничком факултету у Београду као сарадник у настави, а од школске 2019/2020. као асистент. Тренутно је ангажована на предметима: Интернет програмирање, Програмирање интернет апликација, Практикум из пословне комуникације и презентације, Програмирање корисничких интерфејса, Веб дизајн и Проналажење скривеног знања.

Током периода у ком је запослена на Електротехничком факултету учествује на више научних и комерцијалних пројеката. Учесница је или је била учесница на неколико научних пројекта: 1) „*Software for Text Offenses Prevention in Serbian: AI-driven Hate Speech Detection (STOP)*“ који финансира Фонд за науку Републике Србије; 2) „*Advancing novel textual similarity-based solutions in software development (AVANTES)*“ који је финансирао Фонд за науку Републике Србије; и 3) „*Belgrade Data Innovation Hub (BELDIH)*“ - HORIZON 2020. У периоду од претходних пет година учествовала је као предавач у оквиру УНДП програма за преквалификације у ИТ сектору, у организацији Електротехничког факултета у Београду. На комерцијалним пројектима је сарађивала са неколико домаћих и страних фирми и колегама са других катедри Електротехничког факултета. Говори течно енглески и немачки језик.

Изјава о ауторству

Име и презиме аутора: Јелица Цинцовић

Број индекса: 5032/2019

Изјављујем

да је докторска дисертација под насловом

Детектовање неуродегенеративних поремећаја коришћењем модификоване
метахеуристике за оптимизовање параметара модела машинског учења

- резултат сопственог истраживачког рада;
- да дисертација у целини ни у деловима није била предложена за стицање друге дипломе према студијским програмима других високошколских установа;
- да су резултати коректно наведени и
- да нисам кршила ауторска права и користила интелектуалну својину других лица.

Потпис аутора



У Београду, 24.06.2025.

Изјава о истоветности штампане и електронске верзије докторског рада

Име и презиме аутора: Цинцовић Јелица

Број индекса: 5032/2019

Студијски програм: Софтверско инжењерство

Наслов рада: Детектовање неуродегенеративних поремећаја коришћењем модификоване метахеуристике за оптимизовање параметара модела машинског учења

Ментор: проф. др Милош Цветановић

Изјављујем да је штампана верзија мог докторског рада истоветна електронској верзији коју сам предала ради похрањивања у **Дигиталном репозиторијуму Универзитета у Београду**.

Дозвољавам да се објаве моји лични подаци везани за добијање академског назива доктора наука, као што су име и презиме, година и место рођења и датум одбране рада.

Ови лични подаци могу се објавити на мрежним страницама дигиталне библиотеке, у електронском каталогу и у публикацијама Универзитета у Београду.

Потпис аутора

У Београду, 24.06.2025.



Изјава о коришћењу

Овлашћујем Универзитетску библиотеку „Светозар Марковић“ да у Дигитални репозиторијум Универзитета у Београду унесе моју докторску дисертацију под насловом:

Детектовање неуродегенеративних поремећаја коришћењем модификоване
метахеуристике за оптимизовање параметара модела машинског учења

која је моје ауторско дело.

Дисертацију са свим прилозима предала сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигиталном репозиторијуму Универзитета у Београду и доступну у отвореном приступу могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (*Creative Commons*) за коју сам се одлучила.

1. Ауторство (CC BY)
2. Ауторство – некомерцијално (CC BY-NC)
- ③. Ауторство – некомерцијално – без прерада (CC BY-NC-ND)
4. Ауторство – некомерцијално – делити под истим условима (CC BY-NC-SA)
5. Ауторство – без прерада (CC BY-ND)
6. Ауторство – делити под истим условима (CC BY-SA)

(Молимо да заокружите само једну од шест понуђених лиценци.

Кратак опис лиценци је саставни део ове изјаве.)

Потпис аутора

У Београду, 24.06.2025.



1. **Ауторство.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце, чак и у комерцијалне сврхе. Ово је најслободнија од свих лиценци.
2. **Ауторство – некомерцијално.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела.
3. **Ауторство – некомерцијално – без прерада.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела. У односу на све остале лиценце, овом лиценцом се ограничава највећи обим права коришћења дела.
4. **Ауторство – некомерцијално – делити под истим условима.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца не дозвољава комерцијалну употребу дела и прерада.
5. **Ауторство – без прерада.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца дозвољава комерцијалну употребу дела.
6. **Ауторство – делити под истим условима.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца дозвољава комерцијалну употребу дела и прерада. Слична је софтверским лиценцама, односно лиценцама отвореног кода.