УНИВЕРЗИТЕТ У КРАГУЈЕВЦУ

Факултет инжењерских наука

Милош Радовић

ТЕХНИКЕ ИСТРАЖИВАЊА ПОДАТАКА И ОПТИМИЗАЦИЈА МАТЕМАТИЧКИХ МОДЕЛА У БИОМЕДИЦИНСКОМ ИНЖЕЊЕРИНГУ

Докторска дисертација

Крагујевац, 2015

I. Аутор
Име и презиме: Милош Радовић
Датум и место рођења: 25.02.1984. Крагујевац
Садашње запослење: Истраживачко-развојни центар за биоинжењеринг БиоИРЦ, Крагујевац
II. Докторска дисертација
Наслов: Технике истраживања података и оптимизација математичких модела у биомедицинском инжењерингу
Број страница: 165
Број слика: 116
Број библиографских података: 109
Установа и место где је рад израђен: Факултет инжењерских наука, Крагујевац
Научна област (УДК): 004.891:519.876.5(043.3)
Ментор: др Ненад Филиповић, редовни професор
III. Оцена и одбрана
Датум пријаве теме: 18.09.2012.
Број одлуке и датум прихватања докторске дисертације: 01-1/2222-9, 29.08.2013.
Комисија за оцену подобности теме и кандидата: 1. др Ненад Филиповић, ред. проф., Факултет инжењерских наука у Крагујевцу, Научне области: Примењена механика, Примењена информатика и рачунарско инжењерство 2. др Вељко Милутиновић, ред. проф., Електротехнички факултет у Београду, Научне области: Рачунарска техника и информатика 3. др Мирко Росић, ред. проф., Факултет медицинских наука у Крагујевцу, Научне области: Физиологија 4. др Весна Ранковић, ред. проф., Факултет инжењерских наука у Крагујевцу, Научне области: Аутоматика и мехатроника, Примењена информатика и рачунарско инжењерство 5. др Милош Ивановић, ванр. проф., Природно-математички факултет у Крагујевцу, Научне области: Рачунарске комуникације
Комисија за оцену докторске дисертације: 1. др Ненад Филиповић, ред. проф., Факултет инжењерских наука у Крагујевцу, Научне области: Примењена механика, Примењена информатика и рачунарско инжењерство 2. др Весна Ранковић, ред. проф., Факултет инжењерских наука у Крагујевцу, Научне области: Аутоматика и мехатроника, Примењена информатика и рачунарско инжењерство 3. др Миљан Милошевић, доц. проф., Универзитет Метрополитан, Научне области: Информационе технологије 4. др Мирко Росић, ред. проф., Факултет медицинских наука у Крагујевцу, Научне области: Физиологија 5. др Данко Милашиновић, доц., Факултет за хотелијерство и туризам у Врњачкој Бањи, Научне области: Информатика
Комисија за одбрану докторске дисертације: 1. др Ненад Филиповић, ред. проф., Факултет инжењерских наука у Крагујевцу, Научне области: Примењена механика, Примењена информатика и рачунарско инжењерство 2. др Весна Ранковић, ред. проф., Факултет инжењерских наука у Крагујевцу, Научне области: Аутоматика и мехатроника, Примењена информатика и рачунарско инжењерство 3. др Миљан Милошевић, доц. проф., Универзитет Метрополитан, Научне области: Информационе технологије 4. др Мирко Росић, ред. проф., Факултет медицинских наука у Крагујевцу, Научне области: Физиологија 5. др Данко Милашиновић, доц., Факултет за хотелијерство и туризам у Врњачкој Бањи, Научне области: Информатика
Датум одбране докторске дисертације:

Технике истраживања података и оптимизација математичких модела у биомедицинском инжењерингу

Радовић Милош

Факултет инжењерских наука, Универзитет у Крагујевцу

Резиме

Атеросклероза је обољење артерија које карактерише смањење лумена крвног суда ограничавајући на тај начин доток крви и кисеоника до одређених органа. Као последица атеросклерозе може доћи до можданог удара, исхемијске болести срца или срчаног удара, па је успешно лечење атеросклерозе од великог значаја како би се избегле евентуалне фаталне последице.

У оквиру ове дисертације су приказани различити математички модели за симулацију настанка и развоја атеросклерозе који су оптимизовани према доступним експерименталним подацима. Оптимизовани математички модели могу бити од велике користи јер могу лекарима пружити увид у даљи развој болести и помоћи им на тај начин у избору најбоље терапије.

Бројна литература показује да на процес настанка атеросклерозе утичу бројни хемодинамички фактори међу којима је најважнији смичући напон на зиду артерије. Зоне артерије са ниским вредностима смичућег напона на зиду су високо ризичне за настанак атеросклерозе. Са друге стране, екстремно високе вредности смичућег напона на зиду могу довести до дестабилизације већ постојећег плака. Према томе, познавање расподеле смичућег напона на зиду артерије може бити од великог значаја. Методом коначних елемената могуће је веома прецизно израчунати расподелу смичућег напона на зиду посматране артерије. Ово међутим захтева у неким ситуацијама много времена па се доводи у питање употреба у реалним клиничким ситуацијама када је резултате потребно приказати у кратком року. Решење може бити базирано на техникама истраживања података које са прихватљивом тачношћу, готово тренутно могу предвиђати расподелу смичућег напона за дату артерију. Ова методологија је верификована за геометријски параметризоване моделе каротидне бифуркације и анеуризме.

Поред атеросклерозе још једна опака болест је канцер. Канцер дојке је један од најчешћих облика канцера који се јавља код жена. Мамографија је неинвазивна, рендгенска метода за преглед дојки који омогућава детекцију маса у раној фази. Међутим, чак и најискуснији лекари понекад имају проблема приликом прегледа мамографа. Применом напредних методологија претпроцесирања, сегментације и техника истраживања података креиран је систем компјутерски помогнуте дијагнозе за детекцију тумора на мамографима.

Data Mining and optimization of mathematical models in biomedical engineering

Radović Miloš

Faculty of Engineering, Uneversity of Kragujevac

Abstract

Atherosclerosis is a disease of the arteries characterized by the lumen decrease of the blood vessel, thus limiting the blood flow and oxygen supply to certain organs. As a result, atherosclerosis can lead to stroke, ischemic heart disease or heart attack, thus the successful treatment of atherosclerosis is of great importance in order to avoid possible fatal consequences.

Within this thesis, different mathematical models for atherosclerosis initiation and development are presented and optimized by the use of available experimental data. These optimized mathematical models can be of a great importance providing to the physicians an overview of the future disease development and assisting them in choosing the best therapy.

Numerous studies show that the process of atherosclerosis initiation is affected by numerous hemodynamic factors among which wall shear stress is the most important. Low wall shear stress areas are the ones with a high risk for atherosclerosis initiation and development. On the other hand, extremely high wall shear stress can cause destabilization of the existing plaque. Thus, the knowledge of the wall shear stress distribution is of a great importance. By using finite element method, it is possible to calculate precisely the wall shear stress distribution of the observed artery. However, this process can be time consuming sometimes calling into question the application in real clinical situations where the results should be presented in a very short period of time. The solution can be seen in data mining methodologies which can accurately enough predict the wall shear stress distribution of the observed artery instantly. This methodology is verified for geometrically parameterized models of carotid artery bifurcation and aneurysm.

In addition to atherosclerosis, another severe disease is cancer. Breast cancer is the most common type of cancer that occurs in women. Mammography is a non-invasive, x-ray method for breast examination, which allows the detection of masses in early stages. However, even the most experienced physicians sometimes have trouble reading the mammogram. By the application of advanced preprocessing, segmentation and data mining methodologies, the computer aided system for tumor detection in mammograms is created.

Предговор

Током основних студија на Машинском факултету у Крагујевцу стекао сам теоријска знања из разних области инжењерства. Међутим, како су се студије приближавале крају све више сам себи постављао питање шта је то чиме бих заиста желео да се бавим. Пред сам крај основних студија одабрао сам као изборни предмет неуронске мреже код професорке Весне Ранковић и то се испоставило као веома важан моменат за даљи избор мог усмеравања. У оквиру овог предмета сам стекао основна знања везана за архитектуру и начин функционисања неуронске мреже под називом вишеслојни перцептрон. Све ми је то било логично и интересантно, посебно из разлога што сам схватио да се неуронске мреже могу примењивати у готово свим областима где постоји потреба за било каквим предвиђањем. Међутим, и поред тога сам био сумњичав према доступности посла у овој области у Крагујевцу, граду у ком сам рођен и у ком сам завршио студије.

Из области неуронских мрежа сам изабрао дипломски рад и 2009. године сам дипломирао са темом под називом "Апроксимација Џоминијеве криве употребом вештачких неуронских мрежа". Исте године се догодио и кључни моменат за даљи избор мог усмеравања када ме је професор Ненад Филиповић питао да ли сам заинтересован за рад на пројектима у оквиру истраживачко развојног центра за биоинжењеринг у Крагујевцу. Невероватна ми је била чињеница да је у овом центру баш тада постојала потреба за запослењем особе која би се бавила неуронским мрежама и другим методама из области техника истраживања података. Задовољно сам прихватио посао и почео активно да учим теорију и имплементацију алгоритама из ове области. Једна од првих реченица коју сам прочитао из ове области се односила на чињеницу да је могуће потрошити огромно време у проналажењу веза имеђу неких података и да од тога на крају не испадне ништа, али када се циљ испуни и веза пронађе, задовољство је велико. Кроз вишегодишњи рад у овој области могу да закључим да је ово заиста истина и могу да се захвалим професору Ненаду Филиповићу који је имао стпљења када тражених резултата није било дуго или их некада није било уопште. Додатно, у оквиру истраживачко развојног центра за бионжењеринг преузео сам и послове везане за оптимизацију па је то била још једна област у којој сам се требао усавршавати. Изазов је био велики јер у оквиру нашег центра није било никога ко се бави поменутим областима и ко би ми могао помоћи у њиховом учењу. Из поменутог разлога ми је било потребно много времена како бих савладао неке алгоритме из области техника истраживања података и оптимизације. Међутим, захваљујући професору Ненаду Филиповићу који је мени и мојим колегама обезбедио одличне услове за рад, што у нашој земљи нажалост није честа појава, ја сам успео да стекнем теоријска и применљива знања везана за бројне алгориме поменутих научних области. Из поменутих разлога се захваљујем ментору Ненаду Филиповићу, редовном професору Факултета инжењерских наука у Крагујевцу.

Велику захвалност дугујем професору Милошу Којићу директору истраживачко развојног центра за биоинжењеринг, дописном члану Српске академије наука и

уметности и научном сараднику института The Methodist Hospital Research Institute у Хјустону. Захваљујем се такође свим својим колегама из истраживачко развојног центра за биоинжењеринг јер су кроз мултидисциплинарност ове дисетрације сви они у мањој или већој мери дали свој допринос.

Захваљујући билатералном пројекту са Факултетом за рачунарство и информатику у Љубљани имао сам прилику да упознам професора Игора Кононенка, професора Зорана Боснића и асистента Петра Врачара који у оквиру лабораторије за когнитивно моделирање постижу врхунске резултате у области техника истраживања података. Захваљујем се поменутим колегама из Љубљане са којима смо објавили један рад у врхунском међународном научном часопису, а ја сам кроз сарадњу са њима имао прилике да стекнем веома важна знања.

Захваљујем се такође и професору Александру Пеулићу, ванредном професору Факултета инжењерских наука у Крагујевцу, са којим сам сарађивао на решавању проблема детекције тумора на дигитализованим мамографима из којих је проистекло једно поглавље у оквиру ове докторске дисертције.

Захваљујем се Министарству просвете, науке и технолошког развоја за учешће на националним научно истраживачким пројектима.

На крају, а никако и најмање важно, захваљујем се својим родитељима Дмитру и Зорици, брату Предрагу и девојци Маријани који су све време били уз мене и који представљају најважнији део мог живота.

Садржај

1. Уводна разматрања

1.1. Предмет рада

Један од најчешћих узрока смрти данас је шлог [1], [2]. О томе колико је шлог опасан говори чињеница да је он трећи узрок смрти у земљама западне Европе. У Србији се сваке године региструје око 25.000 пацијената који су имали мождани удар. Ова поражавајућа статистика сврстала нас је међу водеће земље у свету по броју новооболелих. У 80 до 85 посто случајева реч је о исхемијском можданом удару или инфаркту мозга, који је последица запушења крвног суда који доводи крв до мозга. Најчешћи узрок који доводи до исхемијског можданог удара је атеросклероза, болест великих и средњих мишићних артерија. Ова болест се карактерише дисфункцијом ендотела крвног суда, васкулитисом и накупљањем липида, холестерола, калцијума и ћелијских елемената унутар зида крвног суда. Овај процес за последицу има формирање плака, васкуларно ремоделовање, акутну и хроничну опструкцију лумена крвног суда, поремећен проток крви и смањену оксигенацију циљаних ткива.

У бројној литератури је показано да смичући напон на зиду артерије игра значајну улогу у процесу настанка и развоја атеросклерозе. Показано је да су зоне артерије са ниским вредностима смичућег напона на зиду високо ризичне за настанак плака. Употребом нумеричких метода, које су данас незаменљиве у моделирању физичких феномена, могуће је одредити комплетну расподелу смучућег напона на зиду. Највећи допринос нумеричких метода се огледа у томе што се њиховом применом, уз одређене апроксимације, могу добити задовољавајућа решења за проблеме који се тешко или уопште не могу решити аналитичким путем. Међутим употреба нумеричких метода има и одређене недостатке, често у виду велике рачунарске захтевности и великог времена потребног за добијање тражених резултата. У ситуацијама када су резултати потребни у кратком временском року погоднije је користити интелигентне моделе засноване на техникама истраживања података који резултате пружају брзо, али са одређеном дозом грешке. Употребом техника истраживања података могуће је пронаћи везу између резултата добијених различитим симулацијама и креирати интелигентни систем који ће бити у стању да на основу тог знања врши предвиђање резултата за неке нове примере. Како би се интелигентни системи креирали и обучили за експлоатацију неопходно је постојање што већег броја података у комбинацији улаз-излаз. Прикупљање података за обучавање и тестирање различитих алгоритама техника истраживања података захтева време, али када се тај процес заврши уштеда времена је велика јер експлоатацијом ових модела резултате добијамо готово тренутно. Применом техника истраживања података у овом раду ће бити моделирана веза између геометрије артерије и густине, вискозности и брзине флуида са једне стране и расподеле смичућег напона на зиду као и вредности и положаја његове максималне вредности са друге стране. Ова веза ће бити моделирана на моделима каротидне бифуркације и анеуризме.

Постојање софтверског алата који би лекарима пружио увид у даљи развој болести (атеросклерозе) је од великог значаја. У овом раду ће бити описани математички

модели за симулацију настанка и развоја плака. Симулација атеросклерозе, појаве и раста плака је од великог значаја у медицини. Пружањем увида у даљи развој болести ове симулације могу доста помоћи лекарима приликом избора терапије којом ће пацијент бити подвргнут. Процес раста плака у времену, односно промена концентрација елемената који га сачињавају, може бити описан системима обичних или парцијалних диференцијалних једначина. Ове математичке моделе је потребно оптимизовати према доступним експерименталним подацима како би у што већој мери одговарали реалној ситуацији. За решавање овог проблема могуће је користити различите алгоритме оптимизације као што су Nelder-Mead оптимизација, генетски алгоритми итд.

Канцер дојке је најчешћи облик канцера који се јавља код жена. Мамографија је рендгенска метода која се у клиничкој пракси веома често користи за преглед и евентуално откривање канцера. Међутим, читање мамографа није једноставан задатак па у неким ситуацијама долази до грешака чак и када су најискуснији лекари у питању. У оквиру ове дисертације биће описан систем компјутерски помогнуте дијагнозе за детекцију тумора на мамографима који је базиран на напредним методологијама претпроцесирања, сегментације и техника истраживања података. Опремање радиолога једним оваквим системом може довести до драстичног смањења грешака приликом прегледа.

1.2. Научни циљ рада

У оквиру ове дисертације решавана су три различита проблема. Циљеви дисертације су следећи:

1. Креирање интелигентних система базираних на техникама истраживања података који ће бити у стању да на основу геометријских параметара са великом прецизношћу предвиђају расподелу смичућег напона на зиду за моделе каротидне бифуркације и анеуризме као и вредност и положај максималне вредности смичућег напона на зиду за модел каротидне бифуркације.
2. Развој система компјутерски помогнуте дијагнозе, базираног на напредним методама претпроцесирања, сегментације и техника истраживања података, који ће бити у стању да са великом прецизношћу региструје присутност тумора и његову позицију на мамографима.
3. Оптимизација математичких модела, развијених у оквиру европског оквирног пројекта ARTreat [3], за симулацију процеса атеросклерозе. Ове моделе је потребно оптимизовати према експерименталним подацима који су доступни за зечеве и људе.

1.3. Методе које су коришћене у истраживању

Предвиђање смичућег напона на зиду. За решавање овог проблема тестирани су различити алгоритми техника истраживања података: неуронска мрежа, алгоритам k најближих суседа, метод потпорних вектора, алгортирам случајне шуме и линеарна регресија. Додатно, коришћени су алгоритми за израчунавање поузданости

индивидуалних предвиђања [4], [5] као и методологија за објашњење модела за предвиђање [6]. Базе података за обучавање поменутих модела су добијене из бројних симулација методом коначних елемената.

Детекција тумора дојке. Систем за детекцију тумора на дигитализованим мамографима се састоји од претпроцесирања (алгоритми за побољшање контраста и филтрирање), сегментације (алгоритми за уклањање позадине и пекторалног мишића и издвајање сумњивих регија), израчунавања атрибута (описна статистика, описивање текстуре и метода локалних бинарних шаблона) и класификације (наивни бајесов класификатор, логистичка регресија, метода потпорних вектора, алгоритам k најближих суседа, стабла одлучивања, неуронска мрежа - вишеслојни перцептрон и алгоритам случајне шуме). Оптималне вредности параметара претпроцесирања и сегментације су одређене оптимизацијом применом Nelder-Mead алгоритма. У циљу побољшања резултата класификације коришћени су алгоритми за селекцију атрибута (mRMR [7], [8], Relief [9], [10] и селекција атрибута према информацијском добитку) као и SMOTE алгоритам [11] за решавање проблема неуједначености базе података. Цео систем је обогаћен мером поузданости предвиђања која се израчунава применом алгоритама за израчунавање поузданости индивидуалних предвиђања.

Оптимизација математичких модела за симулацију атеросклерозе. Оптимизација параметара математичких модела за симулацију процеса настанка и раста плака извршена је применом генетских алгоритама и Nelder-Mead оптимизације.

За имплементацију свих поменутих алгоритама коришћени су програмски пакети MATLAB и WEKA[1].

1.4. Преглед садржаја дисертације

Дисертација садржи укупно седам поглавља. У најгрубљим цртама дисертација се састоји из теоријског дела (поглавља 2 и 3), описа истраживања и постигнутих резултата (поглавља 4, 5 и 6) и закључних разматрања (поглавље 7). У наставку текста биће дат нешто детаљнији опис садржаја по поглављима дисертације.

Поглавље 2 садржи теоријске основе оптимизације. У оквиру овог поглавља дат је детаљан опис Nelder-Mead оптимизације и генетских алгоритама.

Поглавље 3 садржи теоријске основе техника истраживања података. У оквиру овог поглавља дат је детаљан опис претпроцесирања података, алгоритама за селекцију атрибута, алгоритама техника истраживања података, поступака тестирања модела и алгоритма за израчунавање поузданости предвиђања.

Поглавље 4 се бави решавањем проблема повезивања података добијених из хемодинамичких симулација применом техника истраживања података. Конкретније, проблем који се решава је предвиђање смичућег напона на зиду за моделе каротидне бифуркације и анеуризме на основу геометријских параметара.

[1] WEKA - софтвер специјализован за имплементацију алгоритама техника истраживања података (http://www.cs.waikato.ac.nz/ml/weka/).

У поглављу 5 описан је систем за детекцију тумора на дигитализованим мамографима.

Тема поглавља 6 је оптимизација математичких модела за симулацију настанка и раста плака. У оквиру овог поглавља дат је детаљан опис математичких модела и њихове оптимизације према доступним експерименталним подацима.

У поглављу 7 су дата закључна разматрања као и будући правци развоја истраживања овухваћеног овом дисертацијом.

2. Оптимизација

2.1. Увод

Оптимизација се може дефинисати као наука која се бави одређивањем најбољег решења одређеног математички дефинисаног проблема. Проблеми налажења оптималног решења срећу се у свакодневном животу и по природи су веома разноврсни. Проблем може бити: план производње, избор рачунарске конфигурације за фирму, планирање транспорта и много тога другог. У свим тим проблемима човек имплицитно тежи проналажењу решења које у највећој могућој мери задовољава његове жеље, односно решење које му ствара највећу корисност. Оптимизацијом се тежи минимизацији негативних ефеката (напора, трошкова итд.) и максимизацији позитивних ефеката (добити).

Предмет оптимизације су превасходно они проблеми за које постоје мање или више добро разрађени математички модели или за које се такви модели могу направити. При томе треба узети у обзир да се у теорији оптимизације превасходно тражи решење постављеног оптимизационог задатка и не разматра питање колико постојећи математички модел одговара реалном задатку. Са аспекта практичне примене резултата управо провера да ли је математички модел добра апроксимација реалног проблема може бити од круцијалног значаја.

Оптимизација је научна дисциплина која је почела да се развија у XVII и XVIII веку и везује се за имена великих математичара Исака Њутна (енг. Isaac Newton), Жозефа-Луја Лагранжа (енг. Joseph-Louis Lagrange), Готфрида Велхелма Лајбница (енг. Gottfried Wilhelm Leibniz), Леонарда Ојлера (енг. Leonhard Euler) и Огистена-Луја Кошија (енг. Augustin-Louis Cauchy). Практична примена је међутим била незнатна, првенствено због превелике захтевности у погледу рачунања. Прави развој математичке оптимизације тече од половине двадесетог века и директно је везан за равој рачунарске технике. Џорџ Данциг (енг. George Dantzig) је 1947. године развио метод за решавање задатака линеарног програмирања [12], Харолд Кун (енг. Harold Kuhn) и Алберт Такер (енг. Albert Tucker) су 1951. године дефинисали услове оптималности за нелинеарне проблеме [12], Ричард Белман (енг. Richard Bellman) је 1957. године дао основне поставке динамичког програмирања [13], а бројне методе мрежног програмирања су развијене 1957. И 1958. године.

У оквиру ове дисертације оптимизовани су различити математички модели за симулацију настанка и раста плака. Применом Nelder-Mead оптимизације и генетских алгоритама, математички модели су оптимизовани према доступним експерименталним подацима. Сходно томе, ово поглавље дисертације садржи теоријске основе везане за опште појмове оптимизације, Nelder-Mead алгоритам оптимизације и генетске алгоритаме.

2.2. Општи појмови

Поступак оптимизације се састоји из следећих пет фаза [14]:

1. формулација проблема,
2. израда математичког модела који репрезентује реални систем,
3. избор методе оптимизације и рачунарског програма за имплементацију,
4. тестирање модела и добијених решења и
5. имплементација.

У оптимизацији постоји неколико јединствених појмова који не зависе од области примене, врсте проблема који се решава или методе која се користи. О овим појмовима биће речи у тексту који следи.

2.2.1 Променљиве оптимизације

Квантитативне величине које је потребно одредити процесом оптимизације називају се променљиве оптимизације. Ове променљиве се представљају вектором променљивих:

$$\mathbf{X} = \{X(1), X(2), \dots, X(N_{var})\} \tag{2.1}$$

где је N_{var} укупан број променљивих оптимизације.

2.2.2 Функција циља

Да би се за неко решење рекло да је најбоље, треба имати меру којом се одређује његов квалитет и која омогућава његово поређење са другим могућим решењима постављеног оптимизационог проблема. У математичком моделу ради тога мора да постоји функција којом се сваком решењу придружује одговарајућа вредност која представља његову меру квалитета. Та функција се назива критеријум, критеријумска функција, функција циља или мера перформансе. Уобичајено се означава са $f(\mathbf{X})$. Задатак оптимизације је налажење решења које даје екстремну вредност критеријума - највећу (задатак максимизације) или најмању (задатак минимизације). Задаци минимизације и максимизације се могу решавати истим методама пошто је је минимизација заправо исто што и максимизација са супротним знаком тј. $-f(\mathbf{X})$.

2.2.3 Скуп ограничења

Скуп ограничења $G(\mathbf{X})$ је дефинисан системом од m једначина и/или неједначина у којима фигуришу променљиве оптимизације. Постојање ограничења и њихова природа битно утичу на избор методе оптимзације.

Ограничења могу бити у облику једнакости:

$$g_i(\mathbf{X}) = 0; \quad i = 1,2, \dots, m_1 \tag{2.2}$$

или неједнакости:

$$g_i(\mathbf{X}) \geq 0; \quad i = m_1 + 1, m_1 + 2, \dots, m \tag{2.3}$$

2.2.4 Математички модел

Правилно дефинисање математичког модела представља предуслов за успешну оптимизацију. Математички модел је неопходно поставити тако да он што веродостојније репрезентује понашање система који се оптимизује. При томе су неопходна два услова:

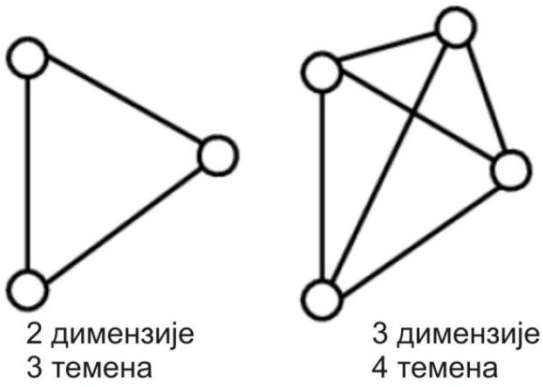
1. да се до решења дође у разумном року како би оно било употребљиво и
2. да трошкови и напори при тражењу решења буду знатно нижи од позитивних ефеката који се остварују поседовањем оптималног решења.

Дефинисање математичког модела обухвата дефинисање функкције циља $f(\mathbf{X})$ и скупа ограничења $G(\mathbf{X})$.

2.3. Nelder-Mead оптимизација

Овај алгоритам је развијен од стране Џона Нелдера (енг. John Nelder) и Роџера Меда (енг. Roger Mead) 1964. године [16], а познат је под називима метод флексибилног полиедра [17] или метод опадајућег симплекса [18]. Велика предност овог алгоритма лежи у чињеници да се у току оптимизације не врши израчунавање извода функције, већ само њене вредности.

Конвексни омотач (енг. *Convex hull*) $k+1$ независних тачака не-нулте запремине у R^k простору назива се k-симплекс (енг. *k-simplex*). Другим речима симплекс садржи једно теме више од димензије простора у коме се налази. На пример, у дводимензионалном простору симплекс је троугао, у тродимензионалном тетраедар итд. (слика 2-1).



Слика 2-1: Дводимензионални и тродимензионални симплекс.

Овај алгоритам се састоји од три основне операције: рефлексије, експанзије и контракције и може се описати следећим корацима.

1. Насумично одабрати прву тачку симплекса $\mathbf{X_1}$.
2. Иницијализовати симплекс. Постоји неколико начина за иницијализацију симплекса. Један од стандардних поступака за иницијализацију симплекса је *axis-by-axis* поступак [18] који је описан следећим једначинама:

$$\begin{aligned} \mathbf{X_2} &= \mathbf{X_1} + k_1\mathbf{e_1} \\ \mathbf{X_3} &= \mathbf{X_1} + k_2\mathbf{e_2} \\ &\vdots \\ \mathbf{X_{N_{var}+1}} &= \mathbf{X_1} + k_{N_{var}}\mathbf{e_{N_{var}}} \end{aligned} \tag{2.4}$$

где је $\mathbf{N_{var}}$ димензија проблема (број променљивих оптимизације), $k_1, k_2, \dots, k_{N_{var}}$ дефинишу величину иницијалног симплекса, а $\mathbf{e_1}, \mathbf{e_2}, \dots, \mathbf{e_{N_{var}}}$ су базни вектори простора у коме се ради.

Други приступ за иницијализацију симплекса заснива се на скалирању иницијалног симплекса према карактеристичним дужинама проблема [19]. Другим речима, омогућава креирање почетног симплекса различите величине у зависности од вредности које има почетна тачка симплекса.

Прво је потребно дефинисати пареметре $\delta_u, \delta_z > 0$, где се δ_z примењује на компоненте почетне тачке симплекса $\mathbf{X_1}$ једнаке нули, док се δ_u примењује на остале компоненте вектора $\mathbf{X_1}$. Уобичајне вредности ових параметара су:

$$\delta_u = 0.05 \quad \text{и} \quad \delta_z = 0.0075$$

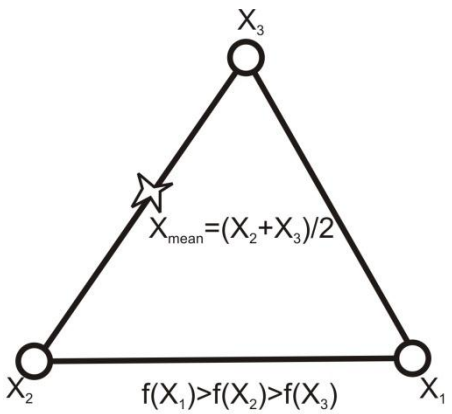
Остала темена иницијалног симплекса се израчунавају на следећи начин:

$$X_i(j) = \begin{cases} X_1(j) + \delta_u \cdot X_1(j) & ako\ je \quad j = i-1, X_1(j) \neq 0 \\ \delta_z & ako\ je \quad X_1(j) = 0 \\ X_1(j) & ako\ je \quad j \neq i-1 \end{cases} \tag{2.5}$$

где је $i = 2, \dots, N_{var}+1$ и $j = 1, \dots, N_{var}$.

3. Израчунати у свакој тачки симплекса $\mathbf{X_i}$ вредност функције $f(\mathbf{X_i})$.
4. Поређати темена симплекса од најгорег ($\mathbf{X_1}$, максимум) до најбољег ($\mathbf{X_{N_{var}+1}}$, минимум) према вредности функције $f(\mathbf{X_i})$.
5. Израчунати $\mathbf{X_{mean}}$ вектор усредњавањем свих вектора осим најлошијег (слика 2-2):

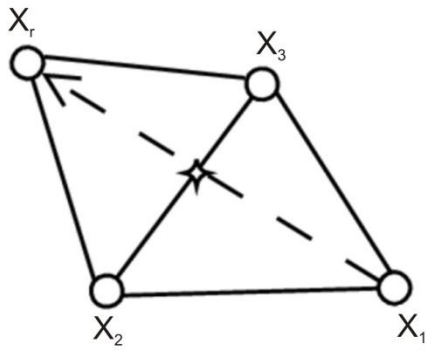
$$\mathbf{X_{mean}} = \frac{1}{N_{var}} \sum_{i=2}^{N_{var}+1} \mathbf{X_i} \tag{2.6}$$



Слика 2-2: Организација темена (енг. *Ordering*) и одређивање $\mathbf{X_{mean}}$.

6. Рефлекција: у овом кораку се израчунава рефлектована тачка (слика 2-3):

$$\mathbf{X_r} = \mathbf{X_{mean}} + \alpha(\mathbf{X_{mean}} - \mathbf{X_1}) \tag{2.7}$$



Слика 2-3: Рефлекција рефлектује најлошију тачку $\mathbf{X_1}$ у односу на $\mathbf{X_{mean}}$.

Ако је рефлектована тачка боља од друге најгоре, али не и од најбоље тј.

$$f\left(\mathbf{X_{N_{var}+1}}\right) \leq f(\mathbf{X_r}) < f(\mathbf{X_2}) \tag{2.8}$$

креирати нови симплекс заменом $\mathbf{X_1}$ са $\mathbf{X_r}$ и вратити се на корак 4.

7. Експанзија: Ако је рефлектована тачка $\mathbf{X_r}$ најбоља за сада тј.

$$f(\mathbf{X_r}) < f(\mathbf{X_{N_{var}+1}}) \tag{2.9}$$

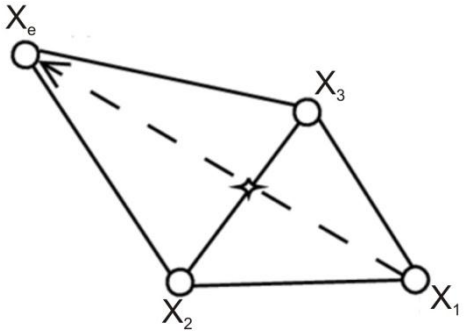
израчунати тачку продужетка (енг. *Expanded point*):

$$\mathbf{X_e} = \mathbf{X_{mean}} + r(\mathbf{X_{mean}} - \mathbf{X_1}) \tag{2.10}$$

Ако је тачка продужетка боља од рефлектоване тачке тј:

$$f(\mathbf{X_e}) < f(\mathbf{X_r}) \tag{2.11}$$

креирати нови симплекс заменом најгоре тачке $\mathbf{X_1}$ тачком продужетка $\mathbf{X_e}$ и вратити се на корак 4. У супротном креирати нови симплекс заменом најгоре тачке $\mathbf{X_1}$ рефлектованом тачком $\mathbf{X_r}$ и вратити се на корак 4. У случају да рефлектована тачка није боља од друге најгоре наставити са кораком 8. Поступак експанзије је приказан на слици 2-4.

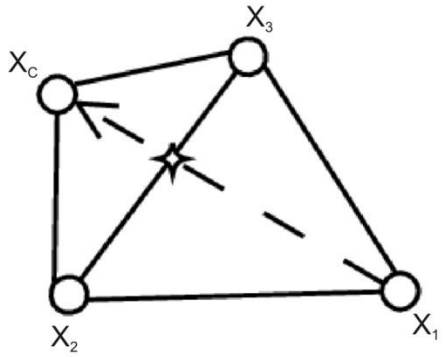


Слика 2-4: Експанзија.

8. Спољашња контракција (слика 2-5): овде је сигурно да важи $f(\mathbf{X_r}) \geq f(\mathbf{X_2})$. Израчунати тачку контракције:

$$\mathbf{X_c} = \mathbf{X_{mean}} + \rho(\mathbf{X_{mean}} - \mathbf{X_1}) \tag{2.12}$$

Ако је контракована тачка боља од друге најгоре, тј. $f(\mathbf{X_c}) < f(\mathbf{X_2})$ креирати нови симплекс заменом најгоре тачке $\mathbf{X_1}$ са тачком контракције $\mathbf{X_c}$, и наставити са кораком 4. У супротном прећи на корак 9.

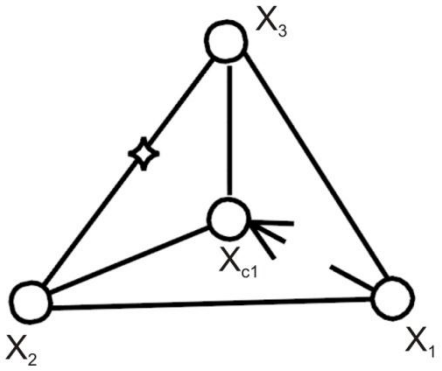


Слика 2-5: Спољашња контракција.

9. Ако је $f(\mathbf{X_c}) > f(\mathbf{X_2})$ (и даље најгора тачка) израчунати тачку унутрашње контракције (слика 2-6):

$$\mathbf{X_{c1}} = \mathbf{X_{mean}} - \rho(\mathbf{X_{mean}} - \mathbf{X_1}) \tag{2.13}$$

Ако је тачка $\mathbf{X_{c1}}$ боља од друге најгоре, тј. $f(\mathbf{X_{c1}}) < f(\mathbf{X_2})$ креирати нови симплекс заменом најгоре тачке $\mathbf{X_1}$ са тачком контракције $\mathbf{X_{c1}}$, и наставити са кораком 4. У супротном закључује се да је симплекс превелики и прези се на корак 10.

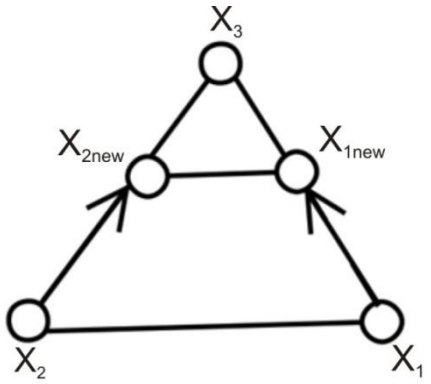


Слика 2-6: Унутрашња контракција.

10. Редукција (слика 2-7): сваку тачку осим најбоље заменити са:

$$\mathbf{X_i} = \mathbf{X_{N_{var}+1}} + \sigma\left(\mathbf{X_i} - \mathbf{X_{N_{var}+1}}\right) \tag{2.14}$$

за свако $i \in \{1, \dots, N_{var}\}$ и вратити се на корак 4.



Слика 2-7: Редукција.

Поступак оптимизације се прекида када се задовољи унапред дефинисани критеријум (да се достигне максимални број итерација, да се одређени број итерација не пронађе боље решење, да се постигне задата вредност функције циља…).

Параметри α, r, ρ и σ су редом коефицијент рефлекције, експанзије, контракције и редукције. Уобичајне вредности ових параметара су дате у табели 2-1.

α	r	ρ	σ
1	2	0.5	0.5

Табела 2-1: Уобичајне вредности коефицијената рефлекције, експанзије, контракције и редукције.

Овај алгоритам се може веома једноставно примењивати употребом програмског пакета MATLAB и функције fminsearch [20].

2.4. Генетски алгоритми

Генетски алгоритам (ГА) је оптимизациони алгоритам базиран на принципима генетике и природне селекције. Код ГА популација састављена од много хромозома еволуира током оптимизационог процеса ка оптималном решењу. Овај алгоритам је развијен од стране Џона Холанда (енг. John Holland) у периоду између 60-тих и 70-их година [21]. Рад Кенета Де Јонга (енг. Kenneth De Jong) је показао корисност ГА за оптимизацију функција и учињени су први напори за одређивање оптималних параметара ГА [22]. Давид Голдберг (енг. David Goldberg), један од Холандових студената, је највише допринео популаризацији генетских алгоритама својом књигом [23].

Неке од предности ГА су:

- могућност оптимизације континуалних и дискретних променљивих,
- не захтевају се изводи,
- могућност оптимизације функција са великим бројем променљивих,
- погодност за паралелизацију,
- могућност оптимизације сложених функција (могућност „искакања” из локалног минимума),
- пружа више решења као резултат оптимизације.

Поменуте погодности често пружају изванредне резултате у ситуацијама када други оптимизациони алгоритми разочарају.

Наравно, ГА није најпогодни алгоритам за решавање свих оптимизационих проблема. На пример, традиционалне методе веома брзо проналазе решење оптимизације једноставних конвексних функција са свега неколико променљивих. У оваквим ситуацијама је погодније користити традиционалне методе јер ГА захтева време да би израчунао вредност функције циља сваког члана популације (потенцијална решења). Велика популација решења која даје предност генетском алгоритму је са друге стране и недостатак због времена које је потребно да би се на серијском рачунару израчунала функција циља свих потенцијалних решења. Међутим, генетски алгоритам је веома погодан за паралелизацију, па је у случају доступности паралелне машине овај проблем превазиђен.

Генетски алгоритам, као и сваки други оптимизациони алгоритам, почиње дефинисањем променљивих оптимизације и функције циља. Завршава као и сваки други оптимизациони алгоритам такође, тестирањем конвергенције. Између међутим, генетски алгоритам је доста другачији у односу на остале оптимизационе методе.

Три основне операције генетског алгоритма (о којима ће касније бити речи) су селекција, укрштање и мутација. Хромозом (енг. *Chromosome*) представља потенцијално решење проблема (вектор променљивих оптимизације). Популација (енг. *Population*) је скуп свих хромозома. Генерација (енг. *Generation*) је скуп свих хромозома у одређеној итерацији генетског алгоритма.

Ако хромозом има N_{var} променљивих (N_{var}-димензионални оптимизациони проблем) $X(1), X(2), \dots, X(N_{var})$, онда се он може представити као:

$$chromosome = \{X(1), X(2), \dots, X(N_{var})\} \tag{2.15}$$

Сваки хромозом има придружену вредност функције циља:

$$f(chromosome) = f\big(X(1), X(2), \dots, X(N_{var})\big) \tag{2.16}$$

У овом поглављу ће бити описане две верзије ГА: бинарни и континуални ГА. Оба алгоритма имају исту основу и исте операције. Разлика је у томе што континуални ГА користи континуалне променљиве, а бинарни ГА представља променљиве као кодиране бинарне стрингове и све операције обавља над стринговима како би минимизовао грешку, тј. решио оптимизациони проблем.

2.4.1 Бинарни генетски алгоритам

Укратко, овај алгоритам се може описати следећим корацима:

1. Дефинисање функције циља, променљивих оптимизације и избор ГА параметара (величина популације, број генерација, степен мутације итд.),
2. Избор иницијалне популације тј. иницијалног скупа хромозома,
3. Кодирање хромозома и израчунавање функције циља,
4. Селекција,

5. Укрштање,
6. Мутација,
7. Провера конвергенције: ако је критеријум за заустављање задовољен оптимизација се завршава, а ако није вратити се на корак 3.

Блок дијаграм бинарног генетског алгоритма је приказан на слици 2-8.

Слика 2-8: Блок дијаграм бинарног генетског алгоритма.

Кодирање и декодирање променљивих оптимизације. Како се код бинарног генетског алгоритма променљиве представљају бинарно, неопходно је обезбедити метод за конвертовање континуалних вредности у бинарне бројеве и обрнуто.

Квантизација је поступак дискретизације континуалног опсега вредности у непреклапајуће подопсеге. Уникатна дискретна вредност се додељује сваком подопсегу. Разлика између стварне функцијске вредности и кватизационог нивоа представља квантизациону грешку. Свака вредност променљиве која припада неком квантизационом нивоу се поставља на минималну, средњу или максималну вредност

тог квантизационог нивоа (слика 2-9). Најчешће се користи средња вредност квантизационог нивоа.

Слика 2-9: Приказ четири континуалне вредности и квантизационих нивоа. Одговарајући хромозом показује квантизациони ниво коме вредност променљиве припада. Сваки хромозом одговара минималној, средњој или највећој вредности одговарајућег квантизационог нивоа.

Математичке формуле за кодирање и декодирање k-те променљиве $X(k)$ дате су једначинама (2.17)-(2.20).

Кодирање:

$$X_{norm}(k) = \frac{X(k) - X_{lo}(k)}{X_{hi}(k) - X_{lo}(k)} \tag{2.17}$$

$$gene[m] = \text{round}\left(0.5 + X_{norm}(k) - 2^{-m} - \sum_{p=1}^{m-1} gene[p] \cdot 2^{-p}\right) \tag{2.18}$$

Декодирање:

$$X_{quant}(k) = \sum_{m=1}^{N_{gene}} gene[m] \cdot 2^{-m} + 2^{-(N_{gene}+1)} \tag{2.19}$$

$$q(k) = X_{quant}(k)\big(X_{hi}(k) - X_{lo}(k)\big) + X_{lo}(k) \tag{2.20}$$

где су:

$X_{norm}(k)$ – нормализована вредност k-те променљиве $X(k)$, $0 \leq X_{norm}(k) \leq 1$,

$X_{lo}(k)$ – најмања вредност k-те променљиве,

$X_{hi}(k)$ – највећа вредност k-те променљиве,

$gene[m]$ – вредност m-тог бита бинарне верзије $X(k)$,

$\text{round}(x)$ – функција која заокружује вредност x на најближи цео број,

$X_{quant}(k)$ – квантификована вредност за $X_{norm}(k)$,

$q(k)$ – квантификована вредност за $X(k)$,

N_{gene} – укупан број бита који се користи за кодирање променљиве.

Како би израчунао вредност функције циља неког хромозома ГА захтева његово декодирање помоћу једначина (2.19)-(2.20). Пример бинарно кодираног хромозома који има N_{var} променљивих, где је свака променљива кодирана са N_{gene} бита је:

$$chromosome = \left[\underbrace{1111001001}_{gene_1} \underbrace{0011011111}_{gene_2} \ldots \underbrace{0001101011}_{gene_{\mathrm{Nvar}}} \right] \tag{2.21}$$

Овај хромозом има укупно $N_{bits} = N_{gene} \times N_{var}$ бита. Заменом сваког гена из хромозома у једначине (2.19)-(2.20) добијамо низ континуалних вредности за променљиве оптимизације.

Популација. ГА започиње процес оптимизације групом хромозома која сачињава иницијалну популацију. Иницијална популација има N_{pop} хромозома и представља $N_{pop} \times N_{bits}$ матрицу. Ова матрица се попуњава јединицима и нулама на следећи начин:

$$population = round\left(rand\left(N_{pop}, N_{bits}\right)\right) \tag{2.22}$$

где функција $rand\left(N_{pop}, N_{bits}\right)$ генерише $N_{pop} \times N_{bits}$ матрицу попуњену случајним бројевима између 0 и 1, а функција $round()$ врши заокруживање сваког члана мартице на ближи цео број (0 или 1). Сваки ред ове матрице представља један хромозом.

Природна селекција. Преживљавање најбољих јединки се своди на одбацивање хромозома са најлошијим вредностима функције циља. Прво се хромозоми сортирају према придруженим вредностима функције циља, а онда се само одређени број најбољих задржава, док се остали одбацују. Степен селекције X_{rate} (енг. *Selection rate*) одређује број хромозома који се задржава у свакој генерацији:

$$N_{keep} = X_{rate} \cdot N_{pop} \tag{2.23}$$

Природна селекција се одвија у свакој итерацији ГА. Од укупно N_{pop} хромозома у свакој генерацији, N_{keep} се задржава и учествује у процесима укрштања и мутације, док се $N_{pop} - N_{keep}$ одбацује остављајући места за нове потомке. Уобичајна вредност степена селекције је 0.5 [24].

Други интересантан приступ природне селекције је базиран на задржавању свих јединки које имају бољу вредност функције циља од неког задатог прага. У почетку се задржава свега неколико јединки, а касније се тај број повећава. Овај приступ је у литератури познат као *thresholding*. Атрактивна карактеристика овог приступа је чињеница да популација не мора бити сортирана.

Селекција. Селекција је део ГА у коме се врши избор хромозома, међу N_{keep} хромозома, који ће учествовати у процесу укрштања. Дакле, бирају се по два родитеља од којих настају по два потомка све док се не попуни празнина настала природном селекцијом (док се не добије нових $N_{pop} - N_{keep}$ хромозома). Овде ће бити наведени само неки, најпопуларнији поступци селекције.

1. Селекција од врха ка дну. Овде се креће од врха (најбољег хромозома) и за родитеље се бирају по два суседна хромозома. То значи да ће се укрштати први и други хромозом, затим трећи и четврти хромозом итд. Овај тип селекције свакако није најбољи, али је веома једноставан за програмирање.
2. Селекција случајним избором. Код овог типа селекције, случајним избором се бирају родитељи међу N_{keep} хромозома.
3. Пропорционална или рулет селекција. Код овог типа селекције вероватноћа избора хромозома за родитеља пропорционална је његовој функцији циља или рангу међу N_{keep} хромозома. Алтернативни назив за пропорционалну селекцију (рулет селекција) је уведен због начина реализације овог оператора: креира се рулет са N_{keep} преграда (енг. *Slot*), при чему су величине преграда пропорционалне квалитету јединки.
   a) Једноставна рулет селекција. Дефинисана је тако да вероватноћа одабира неке јединке буде директно пропорционална вредности њене функције циља. Код ове селекције најпре се врши израчунавање нормализоване вредности функције циља (F). Ово се постиже одузимањем вредности функције циља најбољег одбаченог хромозома (који је отпао природном селекцијом - $f_{N_{keep}+1}$) од функција циља свих хромозома који су „преживели" природну селекцију. На тај начин се обезбеђује да сви хромозоми имају функцију циља истог знака (позитивну или негативну):

$$F_n = f_n - f_{N_{keep}+1} \tag{2.24}$$

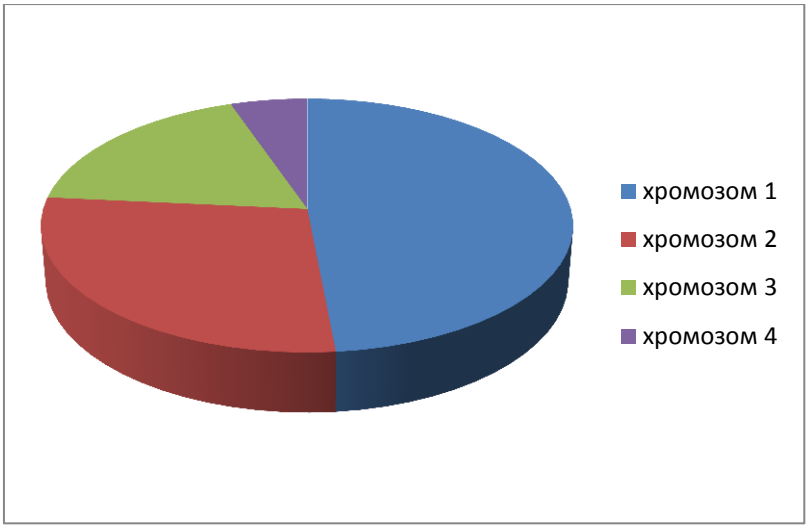
Вероватноћа избора јединке се рачуна на следећи начин:

$$P_n = \frac{F_n}{\sum_{m=1}^{N_{keep}} F_m} \tag{2.25}$$

У табели 2-2 и на слици 2-10 дат је пример једноставне рулет селекције.

n	хромозом	f_n	F_n	P_n	$\sum_{i=1}^{n} P_i$
1	111000100	0.35	0.35-0.99=-0.64	0.485	0.485
2	010100101	0.62	0.62-0.99=-0.37	0.280	0.765
3	110001110	0.75	0.75-0.99=-0.24	0.182	0.947
4	000110110	0.92	0.92-0.99=-0.07	0.053	1.000
5	011001001	0.99			

Табела 2-2: Пример једноставне рулет селекције.



Слика 2-10: Пример једноставне рулет селекције.

Недостатак овог типа селекције је да добре јединке могу бити фаворизоване и више него што је то пожељно, па због тога долази до брзог губитка генетског материјала, а самим тим и преурањене конвергенције.

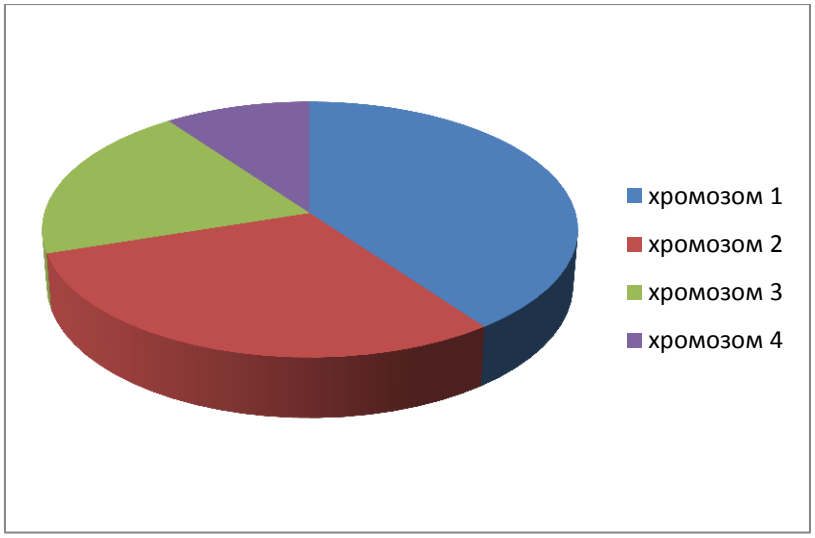
b) Рангирајућа селекција. Дефинисана је тако да вероватноћа одабира неке јединке буде директно пропорционална њеном рангу (позицији) међу N_{keep} хромозома. На овај начин се превазилази проблем претераног фаворизовања неких јединки применом једноставне рулет селекције. Вероватноћа избора јединке се рачуна на следећи начин:

$$P_n = \frac{N_{keep} - n + 1}{\sum_{n=1}^{N_{keep}} n} \tag{2.26}$$

Табела 2-3 и слика 2-11 приказују пример рангирајуће селекције за случај када је $N_{keep} = 4$.

n	хромозом	f_n	P_n	$\sum_{i=1}^{n} P_i$
1	111000100	0.35	0.4	0.4
2	010100101	0.62	0.3	0.7
3	110001110	0.75	0.2	0.9
4	000110110	0.92	0.1	1.0

Табела 2-3: Пример рангирајуће селекције.



Слика 2-11: Пример рангирајуће селекције.

4. Турнирска селекција. Поред рулет селекције, једна од најчешће употребљаваних је свакако турнирска селекција. Код турнирске селекције се случајним избором одаберe неколико јединки (две или три), а онда се за родитеља бира јединка са најбољом вредношћу функције циља. Овај поступак се понавља онолико пута колико је родитеља неопходно издвојити. Турнирска селекција је веома погодна у случају велике популације јер не захтева сортирање хромозома, што у случају великих популација може представљати проблем.

Укрштање. Укрштање је део ГА у коме настаје један или више потомака од родитеља изабраних у процесу селекције. Најчешћи поступци укрштања су:

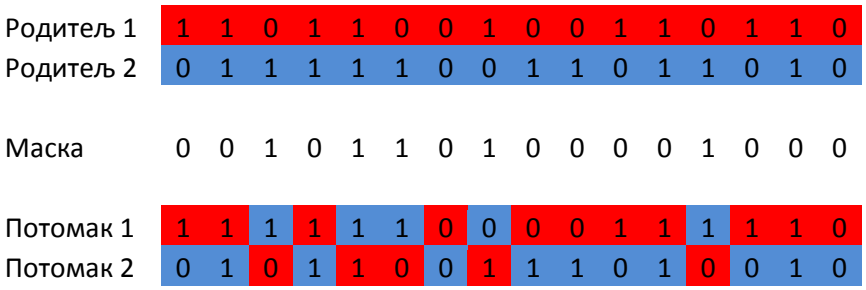
1. Једнопозиционо укрштање (енг. *One-point crossover*). Овде се случајним избором најпре одабере позиција укрштања, а затим креира први потомак копирањем садржаја првог родитеља лево од позиције укрштања и садржаја другог родитеља десно од позиције укрштања. На исти начин, само обрнутим редоследом настаје други потомак. Ово је приказано на слици 2-12.

Слика 2-12: Једнопозиционо укрштање.

2. Двопозиционо укрштање (енг. *Two-point crossover*). Овде се најпре случајним избором одаберу две позиције укрштања, а онда се наизменично копира садржај првог и другог родитеља у потомке. Овај поступак је приказан на слици 2-13.

Слика 2-13: Двопозиционо укрштање.

3. Униформно укрштање (енг. *Uniform crossover*). Код униформног укрштања најпре се креира маска тј. низ дужине N_{bits} који се попуњава нулама. Затим се према задатој вероватноћи укрштања p_x неке вредности претворе у јединицу. Бит маске са вредношћу 1 значи да се на том месту врши замена битова родитеља. Ово је приказано на слици 2-14.



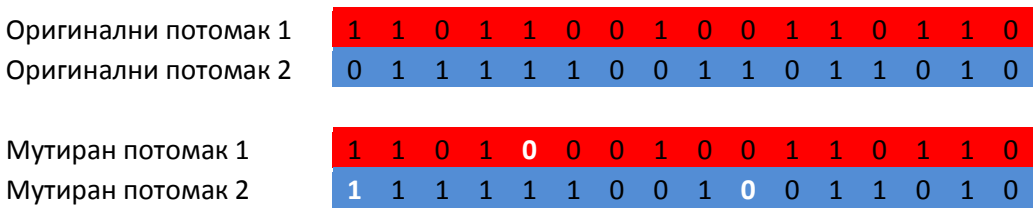
Слика 2-14: Униформно укрштање.

Мутација. Мутација је део ГА у коме се врши промена одређеног броја бита у хромозомима. Ова промена се врши према задатом степену мутације. Најједноставнији облик мутације описан је алгоритмом на слици 2-15.

```
За сваки бит хромозома:
Генериши насумичан број r из интервала [0,1]
Ако је r мањи од вероватноће (степена) мутације
Промени вредност бита
```

Слика 2-15: Алгоритам мутације.

Поступак мутације је илустрован примером на слици 2-16.



Слика 2-16: Мутација.

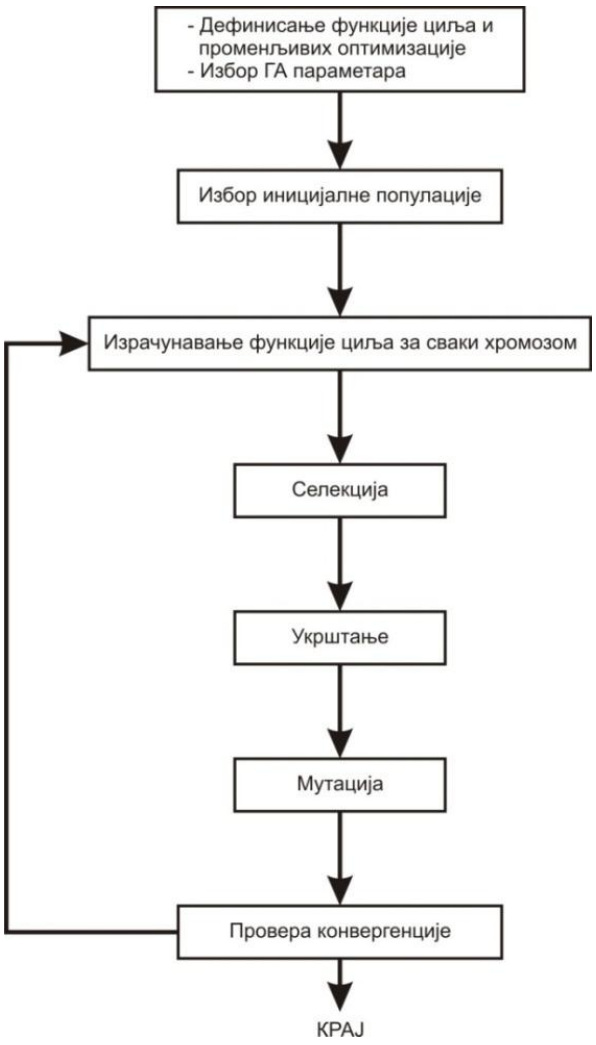
Мутација може спречити ГА од преурањене конвергенције. Она се не извршава у финалној итерацији алгоритма. Уобичајно је да најбољи хромозом (или неколико најбољих хромозома) не подлеже мутацији. Ова појава се назива елитизам и на почетку

алгоритма је потребно дефинисати број елитних решења (хромозома) која не подлежу мутацији.

Конвергенција. Генетски алгоритам прекида оптимизацију оног тренутка када се постигне критеријум за заустављање. Овај критеријум је најчешће задат у виду максималног броја генерација. Такође, критеријум за заустављање може бити дефинисан и у виду минималног напретка (побољшања функције циља) најбољег решења (хромозома) за одређени број генерација. Највећи број генетских алгоритама чува у току оптимизације податке о најбољој и средњој вредности фунцкије циља.

2.4.2 Континуални генетски алгоритам

У случају да су променљиве оптимизације реални бројеви логичније је користити континуални ГА. Разлог је чињеница да бинарно кодирање реалних бројева има ограничење у погледу прецизности. Континуални ГА је бржи од бинарног ГА јер нема потребе за декодирањем хромозома да би се израчунала вредност функције циља. Блок дијаграм континуалног генетског алгоритма је приказан на слици 2-17.



Слика 2-17: Блок дијаграм континуалног генетског алгоритма.

Иницијална популација. Континуални ГА започиње дефинисањем иницијалне популације коју сачињава N_{pop} хромозома. Иницијална популација је представљена

матрицом у којој сваки ред представља по један хромозом. Свака колона ове матрице представља по једну променљиву оптимизације. Дакле ова матрица је димензија $N_{pop} \times N_{var}$ и попуњава се реалним бројевима на следећи начин:

$$population = rand(N_{pop}, N_{var}) \tag{2.27}$$

Све променљиве су нормализоване тако да имају вредност између 0 и 1. Ове вредности су денормализоване приликом израчунавања функције циља. Ако нека променљива $X(k)$ има опсег између $X_{lo}(k)$ и $X_{hi}(k)$, њена денормализована вредност се израчунава на следећи начин:

$$X(k) = \big(X_{hi}(k) - X_{lo}(k)\big)X_{norm}(k) + X_{lo}(k) \tag{2.28}$$

где су:

$X_{lo}(k)$ – доња граница вредности променљиве $X(k)$,

$X_{hi}(k)$ – горња граница вредности променљиве $X(k)$,

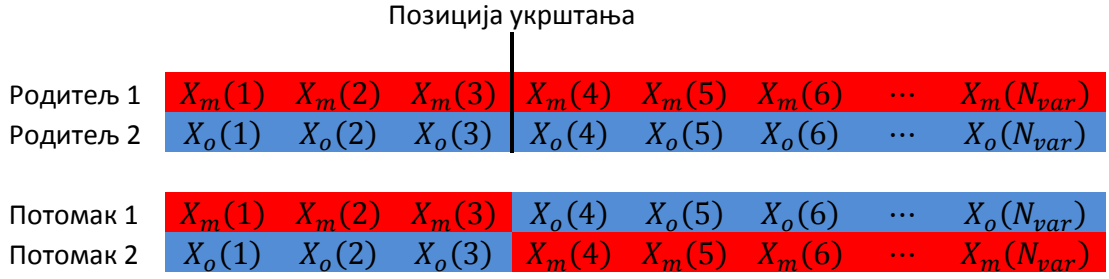
$X_{norm}(k)$ – нормализована вредност променљиве $X(k)$.

Природна селекција. Циљ природне селекције је одабир јединки које ће се задржати у популацији, а затим можда и учествовати у процесу формирања потомака. Као и у случају бинарног ГА, задржава се N_{keep} хромозома (2.23).

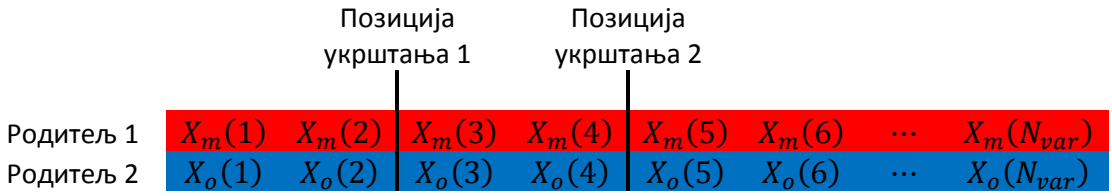
Селекција. Селекција је процес избора хромозома који ће као родитељи учествовати у процесу укрштања и креирати потомке. Код континуалног ГА механизми селекције су исти као и код бинарног ГА (селекција од врха ка дну, селекција случајним избором, једноставна рулет селекција, турнирска селекција итд.).

Укрштање. Укрштање је процес креирања нових хромозома (потомака) од родитеља изабраних селекцијом. Континуални ГА користи велики број различитих поступака укрштања.

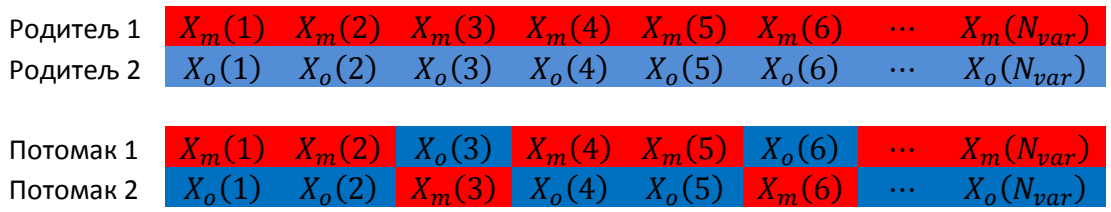
1. Једнопозиционо, двопозиционо и униформно укрштање. ГА одабере једну или више тачака укрштања а затим родитељи размењују вредности променљивих између ових тачака. Може постојати једна тачка укрштања (једнопозиционо укрштање - слика 2-18), две тачке укрштања (двопозиционо укрштање - слика 2-19) или се у екстремном случају одабере N_{var} тачака и потомци насумичним избором узимају вредности промеменљивих од првог или другог родитеља (униформно укрштање - слика 2-20).



Слика 2-18: Пример једопозиционог укрштања.



Слика 2-19: Пример двопозиционог укрштања.



Слика 2-20: Пример униформног укрштања.

Недостатак код оваквог укрштања је што се не уводе нове вредности променљивих, већ се само комбинују већ постојеће. Овакав приступ се потпуно ослања на мутацију када је реч о увођењу новог генетског материјала.

2. Аритметичко укрштање (енг. *Arithmetic crossover*). Овај поступак решава претходно поменути проблем, тј. он комбинује вредности променљивих родитеља како би добио потпуно нове вредности променљивих потомака.

$$X_{new1}(n) = \beta X_m(n) + (1-\beta)X_o(n)$$
$$X_{new2}(n) = (1-\beta)X_m(n) + \beta X_o(n) \tag{2.29}$$

где су $X_m(n)$ и $X_o(n)$ вредности n-те променљиве родитеља (m-мајка, o-отац), $X_{new1}(n)$ и $X_{new2}(n)$ су вредности n-те променљиве потомака, а β је позитиван број из интервала [0, 1].

Постоје различите варијанте аритметичког укрштања. Понекад се укрштају све променљиве, некада само променљиве које се налазе лево или десно од позиције укрштања. Такође, могуће је користити исту или различиту вредност параметра β за различите променљиве. Аритметичко укрштање веома ефикасно комбинује информације које поседују родитељи и креира нове вредности променљивих које леже у границама дефинисаним вредностима променљивих родитеља. Овај метод не дозвољава увођење нових вредности променљивих изван граница дефинисаних родитељским хромозомима. Како би се овај проблем превазишао потребно је користити неки екстраполациони метод укрштања (нпр. Линеарно укрштање).

3. Линеарно укрштање (енг. *Linear crossover*). Овај поступак спада у класу најједноставнијих екстраполационих поступака укрштања [25]. У овом случају се креирају три нове вредности променљивих:

$$\begin{aligned} X_{new1}(n) &= 0.5X_m(n) + 0.5X_o(n) \\ X_{new2}(n) &= 1.5X_m(n) - 0.5X_o(n) \\ X_{new3}(n) &= -0.5X_m(n) + 1.5X_o(n) \end{aligned} \tag{2.30}$$

Свака вредност изван дозвољених граница се одбацује у корист остале 2.

4. Хеуристичко укрштање (енг. *Heuristic crossover*). Овај оператор користи вредности функције циља родитеља како би одредио смер претраге [26]. Вредности променљивих потомака се израчунавају на следећи начин:

$$\begin{aligned} X_{new1}(n) &= X_{better}(n) + \beta \cdot \left(X_{better}(n) - X_{worse}(n)\right) \\ X_{new2}(n) &= X_{better}(n) \end{aligned} \tag{2.31}$$

где су $X_{new1}(n)$ и $X_{new2}(n)$ вредности n-те променљиве потомака, $X_{better}(n)$ је вредност n-те променљиве бољег родитељског хромозома, $X_{worse}(n)$ је вредност n-те променљиве лошијег родитељског хромозома, а β је позитиван број из интервала [0, 1].

Овде се може десити да вредност променљиве $X_{new1}(n)$ буде изван дозвољених граница. Због тога се дефинише параметар r који представља број различитих вредности параметра β са којима се генерише $X_{new1}(n)$. Ако после r покушаја $X_{new1}(n)$ буде и даље изван дозвољених граница, онда је $X_{new1}(n) = X_{worse}(n)$.

Мутација. ГА може веома брзо конвергирати у некој зони површи функције циља. Ово је потпуно у реду када је та зона зона глобалног минимума. Међутим, код реалних проблема функција циља може имати пуно локалних минимума. Зато, ГА може конвергирати унутар зоне локалног минимума. Како би се решио овај проблем уводи се мутација, која случајним избором уводи промене вредности неких променљивих и на тај начин приморава ГА да истражује и друге зоне површи функције циља. Мутација се одвија током еволуције према задатом степену мутације. Ако је нпр. степен мутације 20%, то значи да ће се укупан број мутација добити множењем 0.2 са бројем вредности променљивих које је дозвољено мењати. Овде напомињемо да неки хромозоми не подлежу мутацији услед елитизма. Најчешћи облици мутације су:

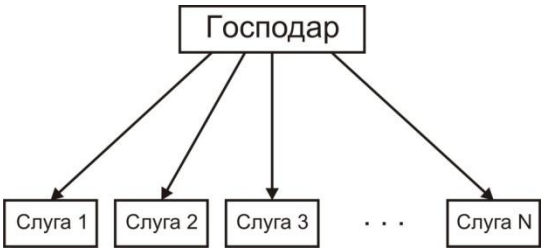
1. Гранична мутација (енг. *Boundary mutation*). Овај оператор врши промену вредности изабраног гена доњом или горњом границом те променљиве.
2. Униформна мутација (енг. *Uniform mutation*). Овај оператор врши промену вредности изабраног гена случајним бројем између доње или горње границе те променљиве.
3. Неуниформна мутација (енг. *Non-uniform mutation*). Код неуниформне мутације степен (вероватноћа) мутације се смањује са бројем генерација. Овај оператор спречава стагнацију популације у раној фази еволуције, а у каснијој фази еволуције омогућава фино тражење коначног решења.
4. Гаусова мутација (енг. *Gaussian mutation*). Овај оператор врши промену вредности изабраног гена додавањем случано изабраног броја према Гаусовој расподели.

2.4.3 Паралелни генетски алгоритам

Употреба сложених компјутерских симулација за решавање инжењерских проблема је данас уобичајена појава. Функције циља које укључују сложене компјутерске симулације могу бити веома прорачунски захтевне. Са друге стране, ГА захтева велики број израчунавања функције циља на свом путу оптимизације. Последица овога је велико време неопходно да би се решио оптимизациони проблем. Овај проблем је могуће превазићи паралелизацијом ГА. Израчунавања функције циља су независна па је самим тим и ГА веома погодан за паралелизацију. Уз минорне измене кода могуће је добити паралелну верзију ГА. Међутим, у случају једноставних функција циља, комуникација међу процесорима може потрошити убрзање добијено употребом паралелног ГА.

Бројна литература показује да је много рада утрошено на развој различитих приступа за паралелизацију ГА [27]-[30]. Избор најбољег метода зависи од природе проблема који се решава као и од архитектуре паралелне машине.

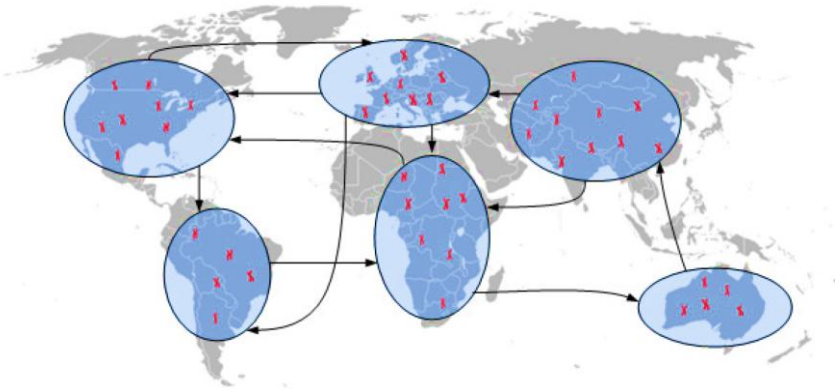
Најједноставнија верзија паралелног ГА је глобални паралелни генетски алгоритам - ГПГА, у литератури познат као господар-слуга (енг. *Master-slave*) алгоритам, који је заправо паралелна имплементација стандардног ГА. Један процесор је главни (господар) и контролише комуникацију, сортирање и упаривање. Главни процесор шаље задатке помоћним процесорима (слугама) ради паралелног израчунавања. Овај принцип паралелизације ГА је најједноставнији за програмирање. Недостатак је чињеница да главни процесор чека док не добије резултате од свих помоћних процесора па је брзина алгоритма ограничена брзином најспоријег помоћног процесора. Већина *master-slave* алгоритама је синхрона: помоћни процесори врше исте операције над делом података, док главни процесор чека. Проблем се јавља када један процесор успорава цео процес оптимизације. Постоје и асинхроне верзије *master-slave* алгоритма. У том случају главни процесор почиње са селекцијом пре него што сви процесори врате резултате. То се постиже применом турнирске селекције над резултатима који пристижу од помоћних процесора. Један од недостатака *master-slave* алгоритма је чињеница да се не користе предности независне еволуције потпопулација. Некада један хромозом доминира целом популацијом још од ране фазе процеса оптимизације. Слика 2-21 показује структуру ГПГА.



Слика 2-21: Глобални паралелни генетски алгоритам – ГПГА.

Представник друге класе паралелних ГА је дистрибуирани паралелни генетски алгоритам – ДПГА који дели популацију на групу потпопулација или острва. Овај алгоритам је у литаратури познат као модел острва. Сваки процесор регулише еволуцију различите потпопулације. Када не би било комуникације међу процесорима

овај модел би био еквивалентан истовременом покретању стандардног ГА над деловима популације. Међутим, уобичајно је да комуникација међу чворовима постоји и да се јавља периодично. Периодично се јавља миграција хромозома међу острвима и на тај начин се повећава разноликост потпопулација. Топологија одређује везе између острва. Који хромозоми мигрирају зависи од алгоритма. Код неких алгоритама процесори размењују случајно одабране хромозоме, код других се размењују најбољи. Потребно је дефинисати степен миграције како би се одредила учесталост миграција. Синхрони алгоритми размењују хромозоме између процесора у исто време, обично након n итерација. Код асинхроних алгоритама сваки процесор независно одређује када ће вршити размену хромозома. Неки алгоритми само размењују хромозоме између процесора, док други клонирају хромозоме и на тај начин омогућавају њихову еволуцију у две одвојене потпопулације. Предност ДПГА је чињеница да је обично бржи од *master-slave* паралелног ГА. На слици 2-22 је илустрована идеја дистрибуираног паралелног генетског алгоритама.



Слика 2-22: Дистрибуирани паралелни генетски алгоритам – ДПГА.

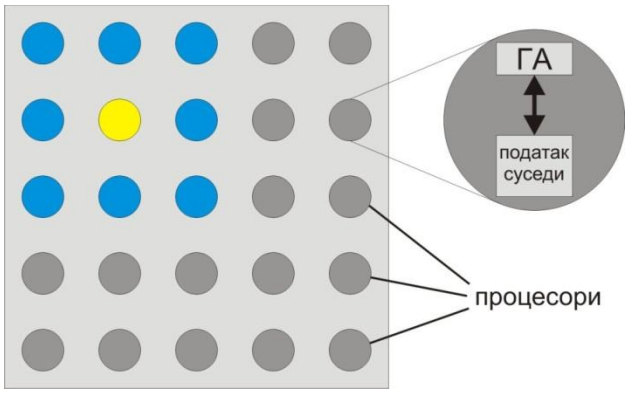
Трећу групу паралелних ГА чини масовно паралелни генетски алгоритам МПГА (енг. *Massively parallel genetic algorithm*). Овај алгоритам се према ситнозрнастој подели састоји од N_p процесора који представљају N_p јединки. Дакле, величина популације је једнака броју процесора. Сваки процесор обавља генетске операторе над својом јединком и над суседним јединкама.

Евалуација и мутација се обављају над припадајућом јединком која се налази у интерној меморији тог процесора, а укрштање над својом јединком и на некој од суседа. Селекцијом се одабере суседна јединка са којом ће се обавити укрштање. Миграција код МПГА се одвија само између суседних процесора. За разлику од осталих модела паралелних генетских алгоритама, модел масовно паралелних генетских алгоритама захтева вишепроцесорски рачунар који се састоји од много процесора.

Слично као и код модела острва, популација је на неки начин подељена на потпопулације. Сваки процесорски елемент је повезан са својим суседима. Дакле, величина потпопулације је једнака броју суседа плус један (припадајућа јединка). Што је број суседа мањи јединке су више изоловане. Због преклапања потпопулација омогућено је брзо ширење добрих решења по целој популацији. Са порастом броја суседа алгоритам добија све лошија својства, јер се локални оптимуми у почетку

оптимизационог поступка пребрзо прошире по целој популацији. У случају да је број суседа мали и алгоритам у почетку пронађе локални оптимум у једном од процесорских елемената, тај локални оптимум се неће брзо проширити целом популацијом јер су потпопулације удаљених процесорских елемената међусобно изоловане. За време док се локални оптимум споро шири, због малог броја суседа, генетски алгоритам има времена да пронађе неко боље решење у другим подручјима простора решења. Стога је број суседа обично пуно мањи од укупне величине популације.

Предност МПГА је та што се постиже готово линеарно убрзање са порастом броја процесора, али недостаци су што је потребно подешавати нова два параметра (топологију и број суседа) и потреба за рачунаром са великим бројем процесора. У случају превеликог броја суседа може се јавити и проблем да комуникациони канал постане уско грло алгоритма. На слици 2-23 је илустрован масовно паралелни генетски алгоритам.



Слика 2-23: Масовно паралелни генетски алгоритам – МПГА.

2.4.4 Хибридни генетски алгоритам

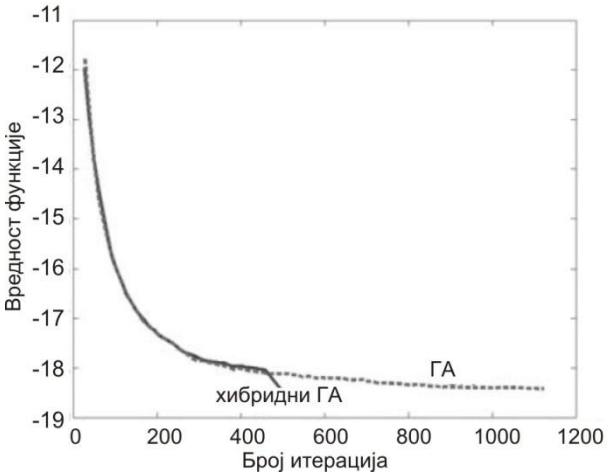
Хибридни генетски алгоритам комбинује предности генетског алгоритма са брзином локалног оптимизатора. Генетски алгоритам веома брзо стиже до зоне глобалног минимума, али није нарочито брз у проналажењу вредности глобалног минимума. Хибридни генетски алгоритам користи генетски алгоритам за проналажење зоне глобалног минимума, а онда локални оптимизатор преузима и проналази вредност глобалног минимума. Хибридни генетски алгоритам може преузети следеће форме:

- Користити генетски алгоритам док не дође до успорења, а онда препустити оптимизацију локалном оптимизатору. У идеалном случају генетски алгоритам долази у зону глобалног минимума.
- Сваких N итерација, локални оптимизатор преузима најбоље или неколико најбољих решења и враћа резултујуће хромозоме у популацију генетског алгоритма.

Континуални генетски алгоритам се веома лако комбинује са локалним оптимизатором, јер локани оптимизатори користе континуалне промењиве. На слици 2-24 су приказани упоредни резултати континуалног и хибридног генетског алгоритма

(Nelder Mead алгоритам је коришћен као локални оптимизатор) за следећи оптимизациони проблем:

$$\begin{array}{ll} \text{минимизовати:} & f(x_1, x_2) = x_1 \sin(4x_1) + 1.1x_2 \sin(2x_2) \\ \text{према ограничењима:} & 0 \leq x_1 \leq 10 \text{ и } 0 \leq x_2 \leq 10 \end{array} \tag{2.32}$$



Слика 2-24: Упоредни резултат хибридног и континуалног генетског алгоритма (преузето из [24]).

Резултати приказани на слици 2-24 су усредњени за 200 независних поступака оптимизације [24]. За ГА је коришћена величина популације 16 са степеном мутације 0.2. Код хибридног ГА Nelder Mead алгоритам преузима поступак оптимизације након 458 итерација. Nelder Mead алгоритам узима најбољи хромозом генетског алгоритма и завршава поступак оптимизације знатно брже од континуалног генетског алгоритма (слика 2-24).

3. Технике истраживања података

3.1. Увод

У савременим условима живота количина података, било о клијентима банака, пацијентима болница, корисницима услуга предузећа (потрошачима) и др., је огромна и расте из дана у дан. Извори тих информација су различити (интерни, екстерни, аналитички), информације могу бити дискретне (атрибутивне) или нумеричке, а количина доступних информација се некада може мерити и терабајтима. Овако сирови подаци немају претерано велику вредност, али адекватно припремљени и анализирани могу довести до откривања знања које потенцијално може бити од великог значаја за остварење бројних успеха. За откривање скривеног знања из података користе се технике истраживања података (енг. *Data mining*).

Технике истраживања података се могу дефинисати као процес проналажења скривених законитости и веза међу подацима. Када се знање „пронађе“ оно може бити употребљено за предвиђање величина од интереса у неким будућим ситуацијама. Проблеми који се решавају применом техника истраживања података су:

1. Класификација (енг. *Classification*) – предвиђање припадности примера некој од постојећих класа на основу података који га описују (атрибута).
2. Регресија (енг. *Regression*) – предвиђање нумеричке вредности жељеног излаза за посматрани пример на основу података који га описују.
3. Асоцијација података (енг. *Association rules*) – Откривање потенцијално важних и интересантних веза међу атрибутима (нпр. који производи се купују заједно у куповини).
4. Груписање (енг. *Clustering*) – Процес откривања група података који су међусобно слични, али различити од осталих група података. У оквиру овог поступка се одређују атрибути на основу којих се најбоље врши груписање.

Области примене техника истраживања података су различите. Неке од њих су:

- медицинска дијагноза,
- процена ризика потенцијалних клијената у области банкарства и осигурања,
- временска прогноза,
- класификација слика,
- препознавање текста и говора,
- кретање берзе,
- и друге.

У оквиру ове дисертације, применом техника истраживања податка решавани су следећи проблеми:

1. Регресиони проблеми - предвиђање максималне вредности смичућег напона на зиду и њеног положаја за модел каротидне бифуркације, као и комплетне

расподеле смичућег напона на зиду за моделе каротидне бифуркације и анеуризме.
2. Класификациони проблем - детекција тумора на дигитализованим мамографима.

Ово поглавље садржи теоријске основе везане за претпроцесирање података, алгоритме за селекцију атрибута, алгоритме техника истраживања података, тестирање модела и поузданост предвиђања.

3.2. Претпроцесирање података

3.2.1 Недостајуће вредности

Подаци из реалног света су често непотпуни и садрже недостајуће вредности (енг. *Missing values*). Један од задатака претпроцесирања података је решавање овог проблема.

Слика 3-1: Недостајуће вредности су уобичајна појава у базама података.

Постоји мноштво методологија за решавање проблема недостајућих вредности, а избор одговарајуће зависи од проблема који се решава. Неке од постојећих методологија су:

1. *Игнорисање примера*: Обично се користи када је за пример непознат излаз или када је непознато неколико атрибута.
2. *Употребљавање глобалне константе за попуњавање непознатих вредности:* Све недостајуће вредности се замене са нпр. *непознато*. Ова методологија се користи у ситуацији када нема никаквог смисла за предвиђање недостајућих вредности.
3. *Употреба средње вредности или медијана за попуњавање недостајућих вредности:* Све недостајуће вредности посматраног атрибута се замене са средњом вредношћу или медијаном тог атрибута из целе базе.
4. *Употреба средње вредности или медијана израчунатих коришћењем само примера који припадају истој класи за попуњавање недостајућих вредности:* Све недостајуће вредности посматраног атрибута се замене средњом вредношћу или медијаном тог атрибута. Средња вредност или медијан се израчунава коришћењем само примера који припадају истој класи.

5. *Употреба неке од техника истраживања података за предвиђање недостајућих вредности:* Све недостајуће вредности се могу одредити употребом нпр. стабла одлучивања или алгоритма к најближих суседа. Ови алгоритми предвиђају недостајуће вредности употребом осталих (познатих) атрибута.

3.2.2 Дискретизација континуалних атрибута

3.2.2.1 1P дискретизација

Овај алгоритам развио је Роберт Холте (енг. Robert Holte) 1993. године [31]. Према овом алгоритму најпре је потребно сортирати све примере за обучавање према вредностима посматраног континуалног атрибута. Након тога се креирају интервали (постављају се границе) при чему је унапред дефинисан мимималан број примера који мора да припада сваком интервалу (ово не важи за последњи интервал). Граница се увек поставља испред примера који припада различитој класи од доминантне класе претходног интервала (што значи да интервали могу имати и више од задатог минималног броја примера). У последњем кораку 1P дискретизације врши се спајање суседних интервала који имају исту доминантну класу. Пример 1P дискретизације је приказан на слици 3-2.

Слика 3-2: Пример 1P дискретизације (мимималан број примера који припада сваком интервалу је 3).

3.2.2.2 Дискретизација заснована на ентропији

Овај тип дискретизације спада у групу дискретизација „одозго-наниже“ (енг. *Top-down*). Овај поступак почиње са једним интервалом. Свака итерација овог алгоритма доводи до креирања једне границе (број интервала се повећава за 1). Овај поступак се наставља до испуњења критеријума за заустављање. Један од критеријума за заустављање је принцип минималне дужине описа (енг. *Minimum Description Length principle* - MDL).

Дакле, за свако потенцијално место поделе потребно је израчунати, ентропију базног скупа примера ($Entropy(S)$), ентропије подскупова примера који настају евентуалном поделом ($Entropy(S_1), Entropy(S_2)$) и информацијски добитак (појмови ентропије и информацијског добитка су детаљније описани у поглављу 3.4.5.1).

Након израчунавања информацијских добитака сваког могућег места поделе бира се највећа вредност и на том месту се поставља граница. Овај поступак се понавља све док се не задовољи критеријум за заустављање (MDL):

$$Information\ gain > \delta$$
$$\delta = \frac{\log(N-1)}{N} + \frac{\Delta(S,T)}{N} \tag{3.1}$$

$$\Delta(S,T) = \log_2\left(3^{m(S)} - 2\right) - \left(m(S) \cdot Entropy(S) - m(S_1) \cdot Entropy(S_1) - m(S_2) \cdot Entropy(S_2)\right) \tag{3.2}$$

где је: N укупан број примера у скупу примера S, T је место поделе, $m(S)$ је број класа унутар скупа примера S, $m(S_1)$ је број класа унутар подскупа S_1, а $m(S_2)$ је број класа унутар подскупа S_2.

3.2.2.3 ChiMerge дискретизација

Овај алгоритам је развијен од стране Рендија Кербера (енг. Randy Kerber) и припада групи дискретизација „одоздо-навише“ (енг. *Bottom-up*) [32]. Дискретизација започиње са онолико интервала колико има примера за учење. За свака 2 суседна интервала потребно је одрадити χ^2 тест који показује да ли је класа независна од интервала. Ако јесте, интервали се спајају. Ако χ^2 тест покаже да је класа зависна од интервала (статистички значајна зависност) интервали остају раздвојени.

Код овог поступка је најпре потребно креирати унакрсну табелу (табела 3-1) на основу које се даље врше израчунавања везана за χ^2 тест.

	Класа 1	Класа 2	Сума
Интервал 1	A_{11}	A_{12}	R_1
Интервал 2	A_{21}	A_{22}	R_2
Сума	$n(C_1)$	$n(C_2)$	N

Табела 3-1: Унакрсно табелирање.

$$\chi^2 = \sum_{i=1}^{2}\sum_{j=1}^{m} \frac{\left(A_{i,j} - E_{i,j}\right)^2}{E_{i,j}} \tag{3.3}$$

$$E_{i,j} = \frac{R_i \cdot n\left(C_j\right)}{N} \tag{3.4}$$

где је:

m – број класа,

$A_{i,j}$ – број примера из i-тог интервала који припадају класи C_j,

R_i – број примера у i-том интервалу,

$n(C_j)$ – број примера који припада класи C_j,

N – укупан број примера у оба интервала.

Дакле, потребно је израчунати χ^2 за свака два суседна интервала и спојити интервале са најмањом вредношћу χ^2. Поступак се понавља све док све вредности χ^2 не буду веће од прага који се одређује у зависности од нивоа значајности (енг. *Significance level*) и броја степени слободе (енг. *Degrees of freedom*).

3.2.3 Бинаризација атрибута

За бинаризацију континуалних атрибута потребно је одрадити само један корак дискретизације „одозго-наниже“.

Дискретни атрибути који могу имати више могућих вредности (више од 2), могу се бинаризовати на један од следећа два начина:

- За сваку могућу вредност атрибута креира се по један бинарни атрибут тако што се све остале вредности атрибута придруже у једну вредност. У овом случају се креира онолико бинарних атрибута колико дати атрибут има могућих вредности.
- За сваку могућу поделу скупа вредности атрибута у два дисјунктна подскупа креира се по један бинарни атрибут. У том случају број креираних бинарних атрибута је:

$$2^{t-1} - 1$$

где је t број могућих вредности атрибута.

3.2.4 Трансформација дискретних атрибута у континуалне

Многи алгоритми из области техника истраживања података (нпр. неуронске мреже) могу радити само са континуалним атрибутима. Из тог разлога се често јавља потреба за трансформацијом дискретних атрибута у континуалне.

Дискретни атрибути који могу имати две вредности могу бити трансформисани у континуалне тако што се прва вредност означи са 0, а друга са 1.

У случају да имамо дискретне атрибуте који могу имати више вредности при чему се те вредности могу поређати по величини (нпр. низак, средње висок, висок) потребно је одрадити нумерацију (низак=0, средње висок=1, висок=2) и онда их третирати као континуалне.

У случају да имамо дискретне атрибуте који могу имати више вредности (t) и при томе се не могу поређати по величини, потребно је креирати t бинарних атрибута (узимајући да је једна вредност атрибута 0, а све остале 1) који се даље третирају као континуални. Недостатак овога је појава потенцијално великог броја атрибута.

3.3. Селекција атрибута

На успешност алгоритама техника истраживања података утичу бројни фактори. Квалитет атрибута је свакако један од њих. Нерелевантни атрибути стварају шум међу подацима па је оптимизација алгоритама истраживања података отежана. Селекција атрибута (енг. *Feature selection*) је поступак откривања и уклањања нерелевантних атрибута. Уклањањем нерелевантних атрибута се повећава брзина алгоритама техника истраживања података. У неким случајевима се постиже и повећање прецизности, а у неким једноставнија интерпретација постигнутог знања.

Постоје бројни критеријуми према којима се могу поделити алгоритми селекције атрибута. Према интеракцији са алгоритмом учења, методе селекције атрибута се могу поделити у две групе:

1. Филтери (енг. *Filters*)
2. Омотачи (енг. *Wrappers*)

Филтери функционишу независно од алгоритма учења. Они једноставно на основу неког критеријума оцењују квалитет атрибута. Са друге стране омотачи чврсто интерагују са алгоритмом учења. У потрази за квалитетним подскупом атрибута омотачи селектују атрибуте тако да се прецизност алгоритма учења повећа.

Алгоритми селекције атрибута се такође могу поделити на групу алгоритама која израчунава квалитет индивидуалних атрибута и на групу алгоритама која израчунава квалитет подскупова атрибута.

3.3.1 Селекција атрибута за класификацију - филтери

3.3.1.1 Рангирање атрибута према информацијском добитку

Ово је један од најједноставнијих (и најбржих) алгоритама за рангирање атрибута. За израчунавање информацијског добитка може се користити ентропија (енг. *Entropy*):

$$Entropy = \sum_{j} -p(C_j)\log_2 p(C_j) \tag{3.5}$$

где $p(C_j)$ представља вероватноћу класе C_j.

Укратко, за израчунавање информацијског добитка једног атрибута, неопходно је најпре поделити скуп примера на онолико подскупова колико тај атрибут има могућих вредности. Информацијски добитак посматраног атрибута се израчунава тако што се од ентропије израчунате над скупом свих примера одузму вредности ентропија подскупова помножени са процентуалним уделом подскупова унутар скупа.

$$Information\ gain\ (i) = Entropy(S) - \sum_{j=1}^{t_i} \frac{n(V_j)}{N} \cdot Entropy(S_j) \tag{3.6}$$

где је $Information\ gain\ (i)$ информацијски добитак i-тог атрибута, $Entropy(S)$ је ентропија скупа свих примера S, t_i је број могућих вредности i-тог атрибута, $n(V_j)$ је

број примера у којима i-ти атрибут има вредност V_j (број примера у подскупу S_j), N је укупан број примера и $Entropy(S_j)$ је ентропија подскупа S_j у коме i-ти атрибут има вредност V_j.

3.3.1.2 Relief и ReliefF

Овај алгоритам је развијен од стране Кенџија Кире (енг. Kenji Kira) и Лерија Рендела (енг. Larry Rendell) [9], а касније је унапређен од стране Игора Кононенка (енг. Igor Kononenko) [10]. Relief функционише тако што случајним избором одабере неки пример из скупа доступних примера, а затим пронађе њему најближе суседе из исте класе и из различитих класа. Вредности атрибута најближих суседа се пореде са вредностима атрибута одабраног примера и користе се за израчунавање квалитета сваког атрибута. Овај поступак се понавља за m случајно одабраних примера. Основа овог алгоритма лежи у чињеници да користан атрибут треба да има различите вредности код примера који припадају различитим класама, и сличне вредности код примера који припадају истој класи.

За $att = 1$ до a // a - бр. атрибута

$W[att] = 0$

Крај { att }

За $r = 1$ до m

случајним избором селектовати пример R

пронаћи k најближих погодака (примера из исте класе) H_{kk}

за сваку класу $c \neq c_R$: //c_R - класа примера R

пронаћи k најближих промашаја (примера из класе $c(\neq c_R)$) $M_{kk}(c)$

За $att = 1$ до a

$$W[att] = W[att] - \frac{\sum_{kk=1}^{k} diff(x(att),R,H_{kk})}{m \cdot k} + \frac{\sum_{c \neq c_R}\left[\frac{p(c)}{1-p(c_R)}\sum_{kk=1}^{k} diff(x(att),R,M_{kk}(c))\right]}{m \cdot k}$$

// $p(c)$ – вероватноћа класе c, $p(c_R)$– вероватноћа класе c_R

Крај { att }

Крај { r }

Слика 3-3: ReliefF алгоритам.

Relief алгоритам је најпре предвиђен за решавање проблема са две класе, али је касније проширен [10] како би могао да решава и проблеме са више класа (ReliefF). Проблеми са више класа се решавају проналажењем најближих суседа за сваку класу различиту од класе изабраног примера. ReliefF алгоритам је детаљно описан на слици 3-3.

Функција $diff(x(att), q_1, q_2)$ израчунава разлику вредности атрибута $x(att)$ између два примера q_1 и q_2. За дискретне атрибуте ова функција даје вредност 1 (ако

су вредности атрибута различите) или 0 (ако су вредности атрибута исте). За континуалне атрибуте функција $diff(x(att), q_1, q_2)$ враћа разлику вредности атрибута између два примера (скалирану на опсег [0,1]):

За дискретне атрибуте:
$$diff(x(att), q_1, q_2) = \begin{cases} 0, & x_1(att) = x_2(att) \\ 1, & x_1(att) \neq x_2(att) \end{cases} \quad (3.7)$$

За континуалне атрибуте:
$$diff(x(att), q_1, q_2) = \frac{|x_1(att) - x_2(att)|}{\max\big(x(att)\big) - \min\big(x(att)\big)} \quad (3.8)$$

3.3.1.3 Minimum Redundancy Maximum Relevance - mRMR

Minimum redundancy maximum relevance (mRMR) алгоритам се заснива на максимизацији релевантности атрибута према класи са једне стране, и минимизацији редундантности међу атрибутима са друге стране. Овај алоритам су развили Ханчуан Пенг (енг. Hanchuan Peng) и Крис Динг (енг. Chris Ding) [7], [8].

mRMR за дискретне атрибуте

За дискретне атрибуте, заједничка информација I два атрибута $x(i)$ и $x(j)$ се дефинише према њиховој заједничкој дистрибуцији вероватноће $p(x(i), x(j))$ и маргиналним вероватноћама $p(x(i))$ и $p(x(j))$ на следећи начин:

$$I(x(i), x(j)) = \sum_{i,j} p(x(i), x(j)) \log \frac{p(x(i), x(j))}{p(x(i))p(x(j))} \quad (3.9)$$

Дакле, за дискретне атрибуте, користи се заједничка информација за израчунавање степена „блискости". Основна идеја је да се одаберу атрибути који су међу собом највише различити. Нека је S подскуп атрибута који тражимо. Услов минималне редундантности је:

$$\min W_I, \qquad W_I = \frac{1}{|S|^2} \sum_{i,j \in S} I\big(x(i), x(j)\big) \quad (3.10)$$

где је $|S|(= a_S)$ број атрибута у подскупу S.

За мерење релевантности атрибута $x(i)$ користи се заједничка информација $I\big(c, x(i)\big)$ између класе $c = \{C_1, \dots, C_m\}$ и атрибута $x(i)$. Према томе, услов максималне релевантности је:

$$\max V_I, \qquad V_I = \frac{1}{|S|} \sum_{i \in S} I\big(c, x(i)\big) \quad (3.11)$$

mRMR подскуп се одређује симултаном оптимизацијом два услова дефинисана једначинама (3.10) и (3.11). Оптимизација оба услова захтева њихово комбиновање у једну функцију циља. Две најједноставније функције циља су дате следећим једначинама:

MID (Mutual Information Difference): $\max(V_I - W_I)$ (3.12)

MIQ (Mutual Information Quotient): $\max\left(\frac{V_I}{W_I}\right)$ (3.13)

Према mRMR алгоритму први атрибут се бира према једначини (3.11), тј. бира се атрибут са највећом вредношћу $I\big(c, x(i)\big)$. Остали атрибути се бирају инкрементално: претходно одабрани атрибути остају у подскупу.

mRMR за континуалне атрибуте

За континуалне атрибуте се као мерило релевантности може користити F вредност између атрибута и класе:

$$F(x(i), c) = \frac{\sum_{k=1}^{m} n(C_k)\big(\bar{x}(i)_k - \bar{x}(i)\big)/(m-1)}{\sum_{k,j}\big(x(i)_{k,j} - \bar{x}(i)_k\big)/(N-m)} \tag{3.14}$$

где је N укупан број примера, m број могућих различитих класа, $\bar{x}(i)$ средња вредност атрибута $x(i)$, $n(C_k)$ број примера који припадају класи C_k, $\bar{x}(i)_k$ средња вредност атрибута $x(i)$ за примере који припадају класи C_k, $x(i)_{k,j}$ је вредност атрибута $x(i)$ j-тог примера који припада класи C_k.

Услов максималне релевантности у случају континуалних атрибута је:

$$\max V_F, \qquad V_F = \frac{1}{|S|}\sum_{i \in S} F(x(i), c) \tag{3.15}$$

Услов минималне редундантности се може изразити помоћу Пирсоновог (енг. Pearson) коефицијента корелације C:

$$\min W_C, \qquad W_C = \frac{1}{|S|^2}\sum_{i,j \in S} C\big(x(i), x(j)\big) \tag{3.16}$$

Као и у случају дискретних атрибута функције (3.15) и (3.16) је потребно објединити. Комбинована функција циља може имати један од следећа два облика:

FCD (F-test Correkation Difference): $\max(V_F - W_C)$ (3.17)

FCQ (F-test Correkation Quotient): $\max\left(\frac{V_F}{W_C}\right)$ (3.18)

3.3.2 Селекција атрибута за регресију - филтери

3.3.2.1 Промена варијансе

Код регресионих проблема се као мерило „нечистоће“ може користити варијанса излазне (континуалне) променљиве:

$$s^2 = \frac{1}{n}\sum_{l=1}^{n}(y_l - \bar{y})^2 \tag{3.19}$$

где је $\bar{y}$ средња вредност излаза n примера унутар посматраног скупа.

$$\bar{y} = \frac{1}{n}\sum_{l=1}^{n} y_l \tag{3.20}$$

За израчунавање квалитета атрибута $x(i)$ користи се промена варијансе која се израчунава на следећи начин:

$$ds^2(x(i)) = \frac{1}{n}\sum_{l=1}^{n}(y_l - \bar{y})^2 - \sum_{j=1}^{t_i}\left(p(V_j)\frac{1}{n(V_j)}\sum_{l=1}^{n(V_j)}(y_{j,l} - \bar{y}_j)^2\right) \tag{3.21}$$

где је t_i број могућих вредности атрибута $x(i)$, V_j је j-та вредност атрибута $x(i)$, $n(V_j)$ је број примера који имају V_j вредност атрибута $x(i)$, $p(V_j)$ је вероватноћа V_j вредности атрибута $x(i)$, $y_{j,l}$ је вредност излаза l-тог примера са V_j вредношћу атрибута $x(i)$, док је $\bar{y}_j$ средња вредност излаза $n(V_j)$ примера који имају V_j вредност атрибута $x(i)$.

$$\bar{y}_j = \frac{1}{n(V_j)}\sum_{l=1}^{n(V_j)} y_{j,l} \tag{3.22}$$

3.3.2.2 RReliefF

Relief алгоритам је адаптиран како би био у могућности да врши рангирање атрибута и код регресионих проблема [32]. Овај такозвани регресиони Relief алгоритам се у литератури означава као RReliefF. Код регресионих проблема излаз је континуална вредност, тако да најближи погоци и промашаји не могу бити коришћени као у случају ReliefF алгоритма. RReliefF (регресиони ReliefF) користи неку врсту „вероватноће“ да два примера припадају „различитим“ класама. Ова „вероватноћа“ се моделира преко удаљености излазних вредности два примера.

RReliefF израчунава квалитет атрибута применом следеће формуле:

$$W(x(i)) = \frac{P_{diffcl|diff}\cdot P_{diff}}{P_{diffcl}} - \frac{(1 - P_{diffcl|diff})\cdot P_{diff}}{1 - P_{diffcl}} \tag{3.23}$$

где је P_{diff} почетна (енг. *Prior*) вероватноћа да два примера имају различиту вредност атрибута, P_{diffcl} је почетна вероватноћа да два примера припадају различитим класама, а $P_{diffcl|diff}$ је постериорна (енг. *Posterior*) вероватноћа да два примера припадају различитим класама под условом да имају различиту вредност атрибута. RReliefF алгоритам мора да апроксимира вероватноће из једначине (3.23) што је детаљно описано на слици 3-4.

За $att = 1$ до a // a - бр. атрибута

$W[att] = 0, N_{dA}[att] = 0, N_{dc \wedge dA}[att] = 0$

Kraj { att }

$N_{dc} = 0$

За $r = 1$ до m

случајним избором одабрати пример R

пронаћи k најближих суседа H_{kk}

За $kk = 1$ до k

$N_{dc} = N_{dc} + diff(y, H_{kk}, R)/k$ // y - излазна променљива

За $att = 1$ до a

$N_{dA}[att] = N_{dA}[att] + diff(x(att), H_{kk}, R)/k$

$N_{dc \wedge dA}[att] = N_{dc \wedge dA}[att] + diff(y, H_{kk}, R) \cdot diff(x(att), H_{kk}, R)/k$

Крај { att }

Крај { kk }

Крај { r }

За $att = 1$ до a

$W[att] = N_{dc \wedge dA}[att]/N_{dc} - (N_{dA}[att] - N_{dc \wedge dA}[att])/(m - N_{dc})$

Kraj { i }

Слика 3-4: RReliefF алгоритам.

Функција $diff(x(att), q_1, q_2)$ је описана једначинама (3.7)-(3.8).

3.3.2.3 CFS

CFS (енг. *Correlation-based Feature Selection*) спада у групу алгоритама која селектује најбољи подскуп атрибута. Алгоритам је развијен од стране Марка Хала (енг. Mark Hall) и базиран је на следећој хипотези [34], [35]:

> *„Добар подскуп атрибута садржи атрибуте који имау високу корелацију са излазном променљивом, а нису корелисани међу собом“*

Квалитет подскупа атрибута S који садржи k атрибута се одређује следећом једначином:

$$W(S) = \frac{k\overline{r_{xy}}}{\sqrt{k + k(k-1)\overline{r_{xx}}}} \tag{3.24}$$

где је $\overline{r_{xy}}$ просечна вредност свих корелација између атрибута и излазне променљиве, а $\overline{r_{xx}}$ је просечна вредност свих корелација међу атрибутима подскупа S.

Најбољи подскуп атрибута се одређује максимизацијом следеће једначине:

$$\max_S W\,,\;\; W = \frac{r_{x_1 y} + r_{x_2 y} + \cdots + r_{x_k y}}{\sqrt{k + 2\left(r_{x_1 x_2} + \cdots + r_{x_1 x_k} + r_{x_2 x_3} + \cdots + r_{x_2 x_k} + \cdots + x_{k-1} x_k\right)}} \tag{3.25}$$

За израчунавање корелација међу атрибутима (r_{xx}) и између атрибута и излазне променљиве (r_{xy}) може се користити нпр. Пирсонов коефицијент корелације. Једна од мера корелације коју др. Хал користи у својој дисертацији је симетрична неизвесност (енг. *Symmetrical uncertainty*):

$$SU(X,Y) = 2 \cdot \frac{E(X) + E(Y) - E(X,Y)}{E(X) + E(Y)} \tag{3.26}$$

где су $E(X)$ и $E(Y)$ ентропије променљивих X и Y, а $E(X,Y)$:

$$E(X,Y) = E(X|Y) + E(Y) = E(Y|X) + E(X) \tag{3.27}$$

где је $E(Y|X)$ условна ентропија:

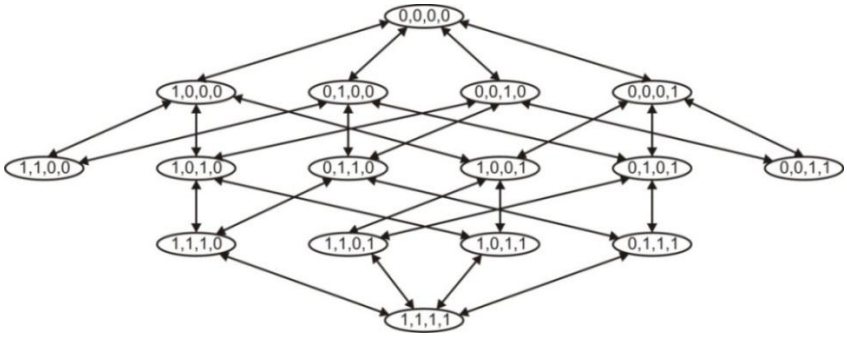
$$E(Y|X) = -\sum_{x \in X} p(x) \sum_{y \in Y} p(y|x) \log_2 p(y|x) \tag{3.28}$$

Како би се израчунала вредност $SU(X,Y)$ потребно је претходно дискретизовати континуалне променљиве $SU(X,Y)$.

3.3.3 Селекција атрибута омотачима

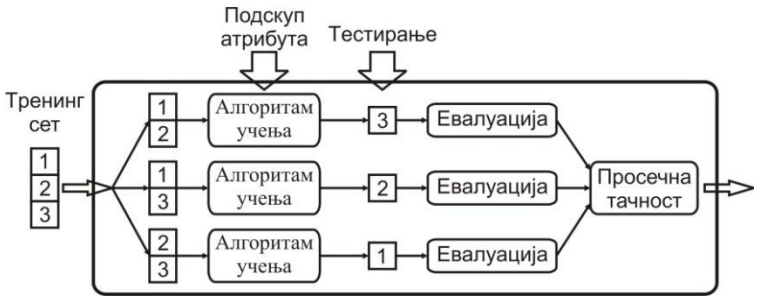
Код омотача, избор подскупа атрибута се врши употребом алгоритма учења као црне кутије (није потребно никакво познавање алгоритма). Омотачи претражују простор атрибута (слика 3-5) у потрази за квалитетним подскупом атрибута употребом алгоритма учења, који је и сам део евалуационе функције (функција која израчунава квалитет подскупа атрибута).

Дакле, омотачи претражују простор атрибута у потрази за најбољим подскупом атрибута. Идеално би било испитати све могуће комбинације атрибута. Међутим тај поступак најчешће захтева много времена, па се зато користе алгоритми претраживања. Алгоритам претраживања захтева простор атрибута (слика 3-5), иницијално стање (иницијални чвор) и критеријум за заустављање претраживања. На слици 3-5 приказан је простор атрибута у коме сваки чвор представља различиту комбинацију атрибута (1 означава присуство, 0 одсуство атрибута).



Слика 3-5: Простор атрибута. Сваки чвор је повезан са чворовима који имају по један атрибут додат или обрисан.

Циљ претраживања је проналажење подскупа атрибута (чвора у простору атрибута) са највећом вредношћу евалуационе функције. За израчунавање евалуационе функције се може користити унакрсна провера (слика 3-6).



Слика 3-6: 3-струка унакрсна провера алгоритма учења над подкупом атрибута (енг. *3-fold cross validation*).

Два најчешће коришћена алгоритма претраживања су:

- Претраживање „пењањем ка врху“ (енг. *Hill climbing*) и
- претраживање „најбољи први“ (енг. *Best-first search*).

Претраживање „пењањем ка врху“, познато и под именом похлепно претраживање (енг. *Greedy search*), је најједноставнији алгоритам претраживања. У основи, овај алгоритам проширује тренутни чвор и врши се премештање на потомка са највећом вредношћу евалуационе функције. Поступак се прекида када ни један потомак не побољшава вредност евалуационе функције у односу на тренутни чвор. Детаљан опис алгоритма је дат на слици 3-7.

1. Нека је v иницијално стање (иницијални чвор).
2. Проширити v; креирати потомке w од v.
3. Израчунати вредност евалуационе функције f за сваки потомак w од v.
4. Нека је v' потомак са највећом вредношћу евалуационе функције $f(w)$.
5. Ако је $f(v') > f(v)$ онда:

   $v \leftarrow v'$

   врати се на корак 2.
6. Врати v као најбољи чвор (најбољи подскуп атрибута).

Слика 3-7: Алгоритам претраживања „пењањем ка врху“.

Претраживање „најбољи први“ је робуснији алгоритам од претраживања „пењањем ка врху“. Основна идеја лежи у селектовању чвора који највише „обећава“, а који при томе не мора бити најбољи до тада. Овај поступак се понавља све док не дође до k узастопних проширења у којима није пронађен бољи чвор. Детаљно је овај алгоритам описан на слици 3-8.

1. Нека је i иницијално стање (иницијални чвор). Поставити:
   $BEST \leftarrow i$
   Додати i у $OPEN\_LIST$
   $CLOSED\_LIST \leftarrow \emptyset$.
2. Нека је v чвор са највећом вредношћу евалуационе функције из $OPEN\_LIST$.
3. Уклонити v из $OPEN\_LIST$, додати v у $CLOSED\_LIST$.
4. Ако је $f(v) - \varepsilon > f(BEST)$ онда: // ε - минимално потребно побољшање
   $BEST \leftarrow v$.
5. Проширити v; креирати потомке w од v.
6. За сваки потомак w од v који није унутар $CLOSED\_LIST$ или $OPEN\_LIST$:
   Израчунати евалуациону функцију и додати га у $OPEN\_LIST$.
7. Ако је $BEST$ промењено у последњих k проширења, врати се на корак 2.
8. Врати $BEST$ као најбољи чвор (најбољи подскуп атрибута).

Слика 3-8: Алгоритам претраживања „најбољи први“.

Термин „селекција унапред“ (енг. *Forward selection*) означава да претраживање почиње празним скупом атрибута и да се у сваком кораку додаје по један атрибут. Са друге стране, тремин „елиминација уназад“ (енг. *Backward elimination*) означава да претраживање почиње чвором који садржи све атрибуте и да се у сваком кораку елиминише по један атрибут.

3.4. Алгоритми техника истраживања података

3.4.1 Алгоритам k најближих суседа

Учење алгоритмом k најближих суседа (енг. *K-Nearest Neighbors* - KNN) је различито од осталих метода машинског учења у смислу да се у овом случају врши само чување података. Када се нови пример доведе на улаз KNN модела, бира се k примера најближих новом примеру и на основу њих се врши предвиђање [32], [35]. Развој овог алгоритма су започели Евелин Фикс (енг. Evelyn Fix) и Џозеф Хоџис (енг. Joseph Hodges) 1951. године [36]. С обзиром да учења скоро да и нема, ова метода се назива лењом методом машинског учења (енг. *Lazy learning method*). KNN алгоритам се може примењивати за решавање класификационих и регресионих проблема.

За проблеме класификације, доминантна класа из сета најближих суседа се бира као резултат предвиђања:

$$\hat{c}_x = \underset{c\in\{C_1,\dots,C_m\}}{\operatorname{argmax}} \left\{ \sum_{i=1}^{k} \delta(c, c_i) \right\} \tag{3.29}$$

где је $\{C_1, \dots, C_m\}$ сет могућих различитих класа а δ је:

$$\delta(a,b) = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases} \tag{3.30}$$

За регресионе проблеме излаз новог примера се рачуна усредњавањем вредности излаза k најближих суседа:

$$\hat{y}_x = \frac{1}{k} \sum_{i=1}^{k} y_i \tag{3.31}$$

где k задаје корисик (обично се усваја неки непаран број тј. 1, 3, 5…).

При одређивању најближих примера најчешће се користи Еуклидско растојање:

$$D(q_l, q_j) = \sqrt{\sum_{i=1}^{a} d(x_l(i), x_j(i))^2} \tag{3.32}$$

У претходној једначини, $D(q_l, q_j)$ представља Еуклидско растојање између примера q_l и q_j, док a представља укупан број атрибута. Континуалне атрибуте је потребно претходно скалирати на интервал [0,1]. Растојање између две вредности континуалног атрибута $x_l(i)$ и $x_j(i)$ је дефинисано као:

$$d\left(x_l(i), x_j(i)\right) = \left|x_l(i) - x_j(i)\right| \tag{3.33}$$

У случају дискретних атрибута растојање се рачуна на следећи начин:

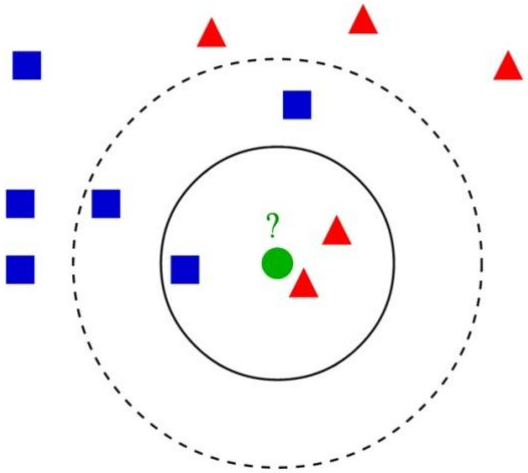
$$d\left(x_l(i), x_j(i)\right) = \begin{cases} 0, & x_l(i) = x_j(i) \\ 1, & x_l(i) \neq x_j(i) \end{cases} \tag{3.34}$$

Претходни обрасци се ослањају на претпоставку да сви атрибути подједнако утичу на излаз. Међутим то у пракси није увек тако па се зато уводи фактор важности атрибута λ. У том случају се растојање рачуна на следећи начин:

$$D(q_l, q_j) = \sqrt{\sum_{i=1}^{a} \lambda(A_i) d(x_l(i), x_j(i))^2} \tag{3.35}$$

где $\lambda(A_i)$ представља фактор важности i-тог атрибута A_i.

Начин функционисања KNN алгоритма визуелно је представљен на слици 3-9.



Слика 3-9: Пример класификације алгоритмом најближих суседа. Пример за тестирање (круг) је потребно класификовати у једну од 2 класе (квадрат или троугао). Уколико је k=3 тестни пример ће бити класификован као троугао. Ако је k=5 тестни пример ће бити класификован као квадрат.

Највећи проблем основног KNN алгоритма је дефинисање параметра k. Овај проблем се може превазићи употребом робуснијег KNN алгоритма који уводи тежинске коефицијенте како би одредио утицај сваког примера из групе најближих (3.36)-(3.37):

- За проблеме класификације:

$$\hat{c}_x = \underset{c\in\{C_1,\ldots,C_m\}}{\operatorname{argmax}} \left\{ \sum_{i=1}^{k} \frac{\delta(c, c_i)}{D(q_x, q_i)} \right\} \tag{3.36}$$

- За регресионе проблеме:

$$\hat{y}_x = \frac{\sum_{i=1}^{k} \frac{y_i}{D(q_x, q_i)^2}}{\sum_{i=1}^{k} \frac{1}{D(q_x, q_i)^2}} \tag{3.37}$$

Употребом KNN алгоритма са тежинским коефицијентима параметар k постаје редундантан.

3.4.2 Линеарна регресија

Линеарна регресија (енг. *Linear Regression* - LR) се може употребити за континуалне атрибуте. Нека је **A** матрица која садржи вредности атрибута свих примера са додатком јединице у првој колони и нека је **B** вектор излазних вредности тј:

$$\boldsymbol{A} = \begin{bmatrix} 1 x_1(1) \cdots x_1(a) \\ \vdots \quad \vdots \quad \ddots \quad \vdots \\ 1 x_N(1) \cdots x_N(a) \end{bmatrix} \tag{3.38}$$

$$\boldsymbol{B} = \begin{Bmatrix} y_1 \\ \vdots \\ y_N \end{Bmatrix} \tag{3.39}$$

Линеарна регресија се у векторском облику може написати као:

$$\mathbf{B} = \mathbf{A} \cdot \mathbf{w} \tag{3.40}$$

где је **w** вектор тежинских коефицијената који садржи $a + 1$ чланова:

$$\mathbf{w} = \begin{Bmatrix} w_0 \\ w_1 \\ \vdots \\ w_a \end{Bmatrix} \tag{3.41}$$

Вектор **w** се одређује применом доступних примера минимизацијом укупне квадратне грешке (енг. *Sum of Squared Errors* - SSE):

$$SSE = \sum_{i=1}^{N} \left(\sum_{j=1}^{a} \left(x_i(j) \cdot w_j \right) + w_0 - y_i \right)^2 \tag{3.42}$$

Након израчунавања вектора **w** (минимизацијом функције (3.42)) предвиђање излаза новог примера q_k се врши према следећем изразу:

$$\hat{y}_k = w_0 + \sum_{j=1}^{a} x_k(j) \cdot w_j \tag{3.43}$$

3.4.3 Логистичка регресија

Иако назив овог алгоритма указује на регресију, логистичка регресија (енг. *Logistic Regression* - LOGR) заправо припада групи класификационих алгоритама. Овај алгоритам је превасходно намењен за решавање проблема припадности једној од две класе, мада може бити употребљен и за решавање проблема вишекласне класификације. Овај алгоритам предвиђа вероватноће припадности првој, односно другој класи према следећим формулама:

$$P(C_1|\mathbf{x_k}) = \hat{y}_k = \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x_k}}} \tag{3.44}$$

$$P(C_2|\mathbf{x_k}) = 1 - \hat{y}_k = \frac{e^{-\mathbf{w}^T \mathbf{x_k}}}{1 + e^{-\mathbf{w}^T \mathbf{x_k}}} \tag{3.45}$$

где је $\mathbf{x_k}$ улазни вектор посматраног примера q_k, а $\mathbf{w}^T$ је вектор тежина који је потребно одредити у циљу постизања максималне ефикасности алгоритма.

Функција веродостојности је:

$$L = \sum_{i=1}^{N} \hat{y}_i{}^{y_i} (1 - \hat{y}_i)^{1 - y_i} \tag{3.46}$$

где је $y_i = 1$ ако пример припада првој класи ($c_i = C_1$) и $y_i = 0$ ако пример припада другој класи ($c_i = C_2$).

Вектор тежина $\mathbf{w}^T$ се одређује градијентном методом минимизацијом негативне вредности логаритма функције веродостојности:

$$E = -\log L = -\sum_{i=1}^{N} [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] \tag{3.47}$$

Промена тежинских коефицијената се у свакој итерацији градијентне методе израчунава на следећи начин:

$$\Delta w_j = -\eta \frac{\partial E}{\partial w_j} = \eta \sum_{i=1}^{N} (y_i - \hat{y}_i) x_i(j) \tag{3.48}$$

Када се одреде вредности тежинских коефицијената, за улазни вектор $\mathbf{x_k}$ логистичка регресија бира као излаз класу $\hat{c}_k \in \{C_1, C_2\}$ са максималном вредношћу следеће функције:

$$\hat{c}_k = \underset{c \in \{C_1, C_2\}}{\operatorname{argmax}} \left\{ \frac{P(c|\mathbf{x_k})}{1 - P(c|\mathbf{x_k})} \right\} \tag{3.49}$$

3.4.4 Наивни Бајесов класификатор

Наивни Бајесов класификатор (енг. *Naive-Bayesian classifier* - NB), као што и сам назив говори, припада групи класификационих алгоритама. За дати тестни пример, овај алгоритам предвиђа вероватноће припадности постојећим класама. Овај алгоритам је базиран на Бајесовој теореми:

$$P(C_i|\mathbf{x_k}) = \frac{P(\mathbf{x_k}|C_i) \cdot P(C_i)}{P(\mathbf{x_k})} \tag{3.50}$$

где је $P(\mathbf{x_k}|C_i)$ вероватноћа улазног вектора $\mathbf{x_k}$ за дату класу C_i, $P(C_i)$ је вероватноћа класе C_i, а $P(\mathbf{x_k})$ је вероватноћа улазног вектора $\mathbf{x_k}$.

С обзиром на чињеницу да $P(\mathbf{x_k})$ има исту вредност за све класе овај члан се може занемарити па наивни Бајесов класификатор бира као излаз класу која максимизује следећи израз:

$$P(C_i|\mathbf{x_k}) = P(\mathbf{x_k}|C_i) \cdot P(C_i) \tag{3.51}$$

У циљу смањења прорачунске захтевности, наивни Бајесов класификатор полази од претпоставке да су вредности различитих атрибута међусобно независане за дату класу, па отуда и потиче назив „наивни“. Сходно томе, $P(\mathbf{x_k}|C_i)$ се математички може представити као:

$$P(\mathbf{x_k}|C_i) \approx \prod_{\mathrm{j}=1}^{a} P(x_k(j)|C_i) \tag{3.52}$$

где је a укупан број атрибута, а $P(x_k(j)|C_i)$ је вероватноћа $x_k(j)$ вредности j-тог атрибута посматраног улазног вектора $\mathbf{x_k}$ за дату класу C_i. У зависности од врсте атрибута $P(x_k(j)|C_i)$ се израчунава на различите начине:

а) за дискретне атрибуте:

$$P(x_k(j)|C_i) = \frac{n(x_k(j), C_i)}{n(C_i)} \tag{3.53}$$

где је $n(C_i)$ број примера који припадају класи C_i, а $n(x_k(j), C_i)$ је број примера који припадају класи C_i и имају вредност j-тог атрибута $x_k(j)$.

б) за континуалне атрибуте:

$$P(x_k(j)|C_i) = \frac{1}{\sqrt{2\pi\sigma_{C_i,x(j)}}} e^{-\frac{\left(x_k(j)-\mu_{C_i,x(j)}\right)^2}{2\sigma^2_{C_i,x(j)}}} \tag{3.54}$$

где су $\mu_{C_i,x(j)}$ и $\sigma_{C_i,x(j)}$ средња вредност и стандардна девијација j-тог атрибута примера који припадају класи C_i.

Наивни Бајесов класификатор израчунава $P(C_i|\mathbf{x_k})$ вредност за сваку класу, а као излаз предвиђа класу са максималном вредношћу функције $P(C_i|\mathbf{x_k})$:

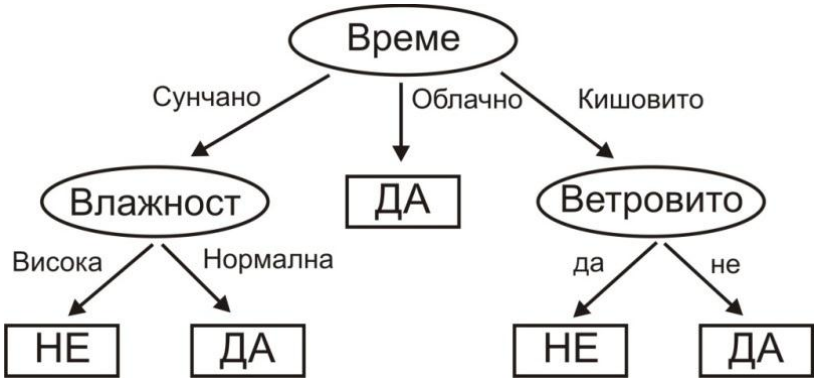
$$\hat{c}_k = \underset{c\in\{C_1,\ldots,C_m\}}{\operatorname{argmax}} \{P(c|\mathbf{x_k})\} \tag{3.55}$$

3.4.5 Стабла одлучивања

Овај алгоритам се може употребљавати за решавање класификационих и регресионих проблема (енг. *Classification And Regression Trees* - CART) [38]. Име овог алгоритма потиче од структуре која подсећа на стабло. Улазне вредности (атрибути) могу бити било ког типа, а излаз (припадност одговарајућој класи у случају класификације или нумеричка вредност у случају регресије) се одређује једноставно, спуштањем низ стабло гранама које одговарају вредностима атрибута. Овај алгоритам је посебно интересантан стручњацима одређених области (нпр. лекарима) јер је изузетно репрезентативан, често се поклапа са њиховим претходним знањима, па чак и пружа нека нова сазнања.

3.4.5.1 Класификациона стабла

Класификациона стабла (енг. *Decision Trees* - DT) се употребљавају за решавање класификационих проблема. Пример класификационог стабла приказан је на слици 3-10 где се на основу временских услова одлучује да ли ће се одиграти тениски меч или не.



Слика 3-10: Пример класификационог стабла.

Стабло одлучивања се састоји од унутрашњих чворова који одговарају атрибутима, грана које представљају одређене вредности атрибута и спољашњих чворова који означавају припадност одређеној класи.

Кључни фактор овог алгоритма је у избору најбољег атрибута кога је потребно поставити у тренутно активни чвор, а на основу кога се стабло даље разграњава. За ову намену се уводе величине које дефинишу степен „нечистоће“ скупа података (базе података): ентропија, гини индекс (енг. *Gini index*) и класификациона грешка које се израчунавају према једначинама (3.56)-(3.58).

$$Entropy = \sum_j -p(C_j)\log_2 p(C_j) \tag{3.56}$$

$$Gini\ Index = 1 - \sum_j p(C_j)^2 \tag{3.57}$$

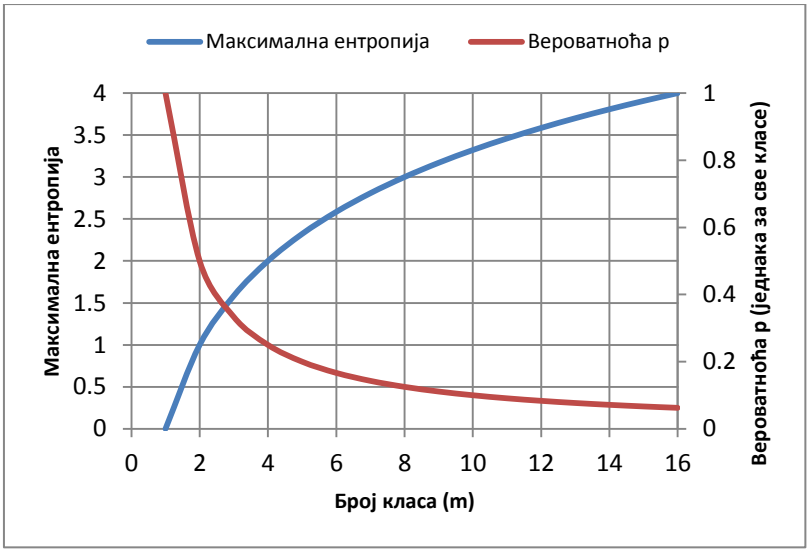
$$Classification\ Error = 1 - \max\{p(C_j)\} \tag{3.58}$$

где $p(C_j)$ представља вероватноћу класе C_j:

$$p(C_j) = \frac{n(C_j)}{N} \tag{3.59}$$

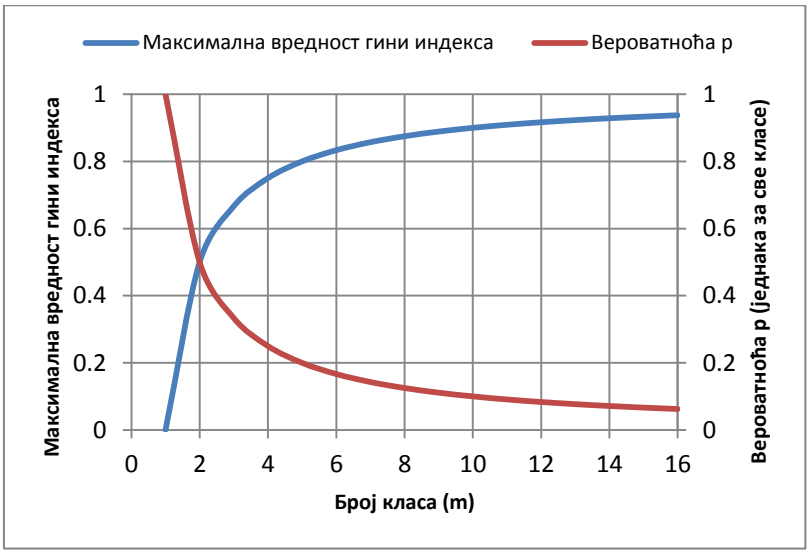
У претходном изразу $n(C_j)$ представља број примера који припадају класи C_j, а N укупан број примера.

Ентропија потпуно „чистог“ скупа података (у коме сви примери припадају једној истој класи) је једнака нули јер у том случају вероватноћа класе једнака јединици, а логаритам од јединице је једнак нули. Слика 3-11 показује максималне вредности ентропије у зависности од броја класа (вероватноћа сваке класе је једнака и износи $p(C_j) = \frac{1}{m}$, где је m број класа).



Слика 3-11: Максимална вредност ентропије у зависности од броја класа.

Гини индекс потпуно „чистог“ скупа података је такође једнак нули ($1 - 1^2$). Слика 3-12 показује максималне вредности гини индекса у зависности од броја класа (вероватноћа сваке класе је једнака и износи $p(C_j) = \frac{1}{m}$, где је m број класа). Гини индекс увек има вредност између 0 и 1 без обзира на број класа.



Слика 3-12: Максимална вредност гини индекса у зависности од броја класа.

Вредност класификационе грешке потпуно „чистог“ скупа података је једнака нули. Класификациона грешка као и гини индекс увек има вредност између 0 и 1.

Сада, када су разјашњени основни појмови везани за израчунавање степена „нечистоће“ скупа података, може бити разјашњен појам информацијског добитка (енг. *Information gain*) на основу кога се доноси одлука о атрибуту који ће бити изабран као најважнији односно најутицајнији у одређеној итерацији овог алгоритма.

За израчунавање информацијског добитка можемо користити ентропију (3.6), гини индекс или класификациону грешку. Укратко, за израчунавање информацијског добитка једног атрибута, неопходно је најпре поделити скуп примера на онолико

подскупова колико тај атрибут има могућих вредности, по један за сваку вредност. После тога се израчунавају степени „нечистоће“ за сваки подскуп. У последњем кораку се израчунава информацијски добитак посматраног атрибута тако што се од степена „нечистоће“ за који смо се одлучили (нпр. ентропија) базног скупа примера одузму степени „нечистоће“ подскупова помножени са процентуалним уделом подскупова унутар скупа (3.6).

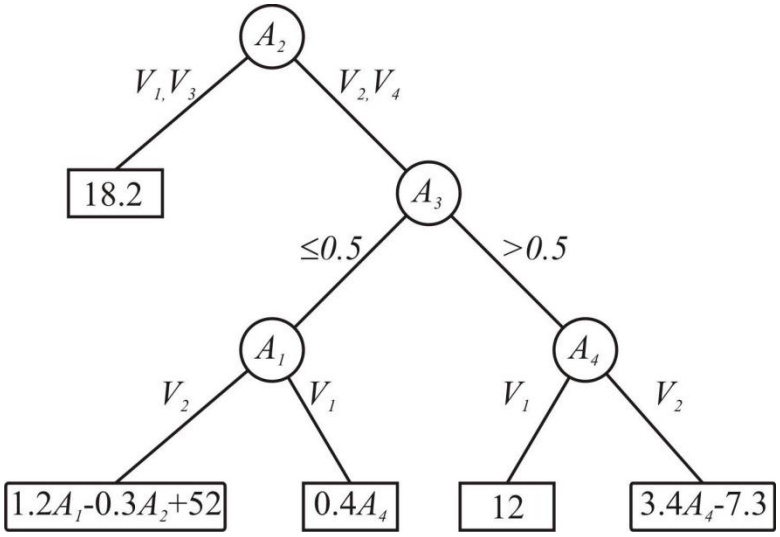
Атрибут са највећим информацијским добитком се бира као најутицајнији и везује за чвор стабла, стабло се разгранава и у следећој итерацији се поступак понавља за следећи чвор.

3.4.5.2 Регресиона стабла

Регресиона стабла (енг. *Refression Trees* - RT) се употребљавају за решавање регресионих проблема. Као и класификациона стабла регресиона стабла се састоје од унутрашњих чворова који одговарају атрибутима, грана које представљају одређене вредности атрибута и спољашњих чворова (листова). Разлика је у томе што је за сваки спољашњи чвор везана функција која може бити:

1. Константна функција – средња вредност излаза свих примера у чвору.
2. Линеарна функција – излазна вредност се моделира као линеарна функција једног или више атрибута.
3. Произвољна функција – излазна вредност се израчунава помоћу неке произвољне функције.

Пример регресионог стабла приказан је на слици 3-13.



Слика 3-13: Пример регресионог стабла.

Као и код класификационих стабала кључни фактор овог алгоритма је у избору најбољег атрибута кога је потребно поставити у тренутно активни чвор, а на основу кога се стабло даље разграњава. Код регресионих стабала се за избор атрибута често користи промена варијансе излазне променљиве (поглавље 3.3.2.1).

Након креирања регресионог стабла оно се употребљава за предвиђање излаза нових примера (који нису учествовали у креирању стабла). Вредност излаза се добија

спуштањем низ стабло, почевши од чвора на врху, низ одговарајуће гране према вредностима атрибута, па све до спољашњег чвора који садржи функцију помоћу које се израчунава излаз.

3.4.6 Алгоритам случајне шуме

Алгоритам случајне шуме (енг. *Random Forest* - RF), развио је Лео Брајман (енг. Leo Breiman), који је такође творац и CART алгоритма [38]. Основа овог алгоритама су стабла одлучивања. Укратко алгоритам се заснива у следећем:

1. *Bagging*: од укупног броја примера случајним избором се бира одређени број (обично око 63%). Преосталих 37 % примера се у литератури обично означава као OOB (енг. *Out of bag*).
2. За сваки чвор из стабла се узима случајним избором m_{try} атрибута и од њих се по неком критеријуму (нпр. информацијски добитак) одабира најбољи.
3. Креира се N_{trees} различитих стабала.
4. Резултат предвиђања је просечна вредност излаза свих N_{trees} стабала у случају регресије или класа која је доминантна међу излазима свих N_{trees} стабала у случају класификације.

У случају регресије уобичајно је да се усваја:

$$m_{try} = \frac{a}{3} \tag{3.60}$$

док се случају класификације обично усваја:

$$m_{try} = \sqrt{a} \tag{3.61}$$

где је a број атрибута.

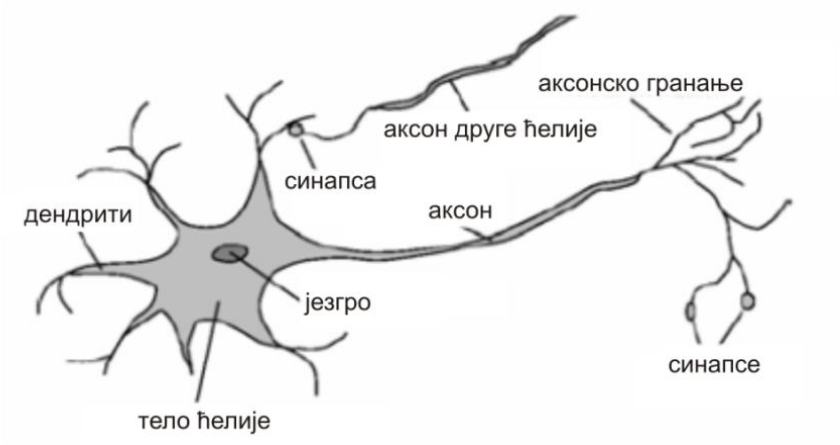
Код овог алгоритма је потребно нагласити да тестирање није неопходно одрадити унакрсном провером (енг. *Cross validation*), нити је потребно издвојити део примера за тестирање. Потребно је сваки пример који је у скупу OOB приликом креирања k-тог стабла пропустити кроз стабло и добити излазе стабла. На тај начин за свако креирано стабло добијамо предвиђања стабла за око једну трећину примера (који су у скупу OOB). На крају се узме да је $\hat{c} = C_j$ класа која је добила највише „гласова“ сваки пут када је посматрани пример био у скупу OOB. Однос укупног броја случајева када је $\hat{c}$ различито од тачне вредности излаза посматраног примера и укупног броја примера преставља OOB грешку (енг. *OOB error estimate*).

3.4.7 Неуронска мрежа: вишеслојни перцептрон

3.4.7.1 Појам неуронских мрежа

Још педесетих година двадесетог века човек је дошао на идеју да креира систем који ће решавати бројне проблеме по узору на човеков начин рамишљања. Управо тада је и настао појам вештачких неуронских мрежа. Постоје две категорије неуронских мрежа: биолошке и вештачке.

Представник биолошких неуронских мрежа је нервни систем живих бића. Мозак човека поседује 10^{11} нервних ћелија-неурона. Неурони су изузетно међусобно повезани и управо та повезаност омогућава бројне процесе као што су читање, размишљање, дисање итд. Начин на који биолошке мреже функционишу није детаљно разјашњен, али се зна да све неуронске функције обављају неурони и њихове међусобне везе. На слици 3-14 је приказана основна структура биолошког неурона.



Слика 3-14: Основна структура биолошког неурона.

Као што се са слике види неурон се састоји од:

- тела и
- две врсте наставака: дендрита и аксона.

Дендрити су кратки, разгранати наставци који доводе сигнал до тела неурона. Са друге стране аксон је дугачак продужетак који одводи сигнал од тела неурона до следећег неурона. Тачка контакта аксона једне ћелије и дендрита друге ћелије зове се синапса.

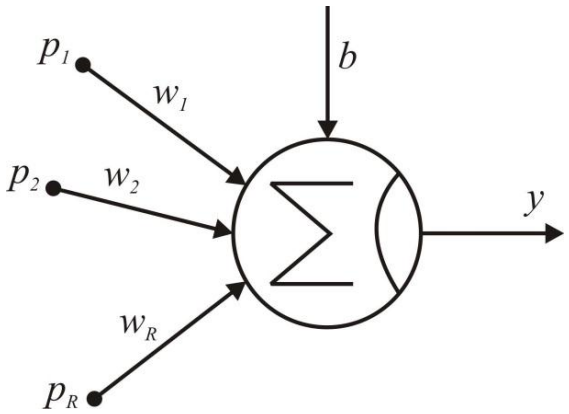
Вештачке неуронске мреже (енг. *Artificial Neural Networks* - ANN) су по структури, функцији и обради информација сличне биолошким неуронским мрежама, али се ради о вештачким творевинама. Неуронска мрежа у рачунарским наукама представља веома повезану мрежу елемената који обрађују податке. Оне су способне да изађу на крај са проблемима који се традиционалним поступцима тешко решавају, као што су говор и препознавање облика. Једна од најважнијих особина неуронских мрежа је њихова способност да уче на ограниченом скупу примера.

Већина вештачких неуронских мрежа има неку врсту правила за „обучавање“, чиме се коефицијенти веза између неурона подешавају на основу улазних података. Другим речима вештачке неуронске мреже „уче“ преко примера (као што деца уче да препознају конкретан предмет, објекат, процес или појаву преко одговарајућих примера).

Неуронска мрежа се може дефинисати као вештачки ћелијски систем способан да прихвати, упамти и примени експериментално (емпиријско) знање. Овде се под знањем подразумева способност да неуронска мрежа у посматраној улазној ситуацији реагује на одговарајући начин.

3.4.7.2 Модел вештачког неурона

Вештачки неурони као и биолошки, имају једноставну структуру и имају сличне функције као и биолошки неурони. Тело неурона се назива чвор или јединица. Најједноставнији неурон сабира улазе, помножене са тежинским коефицијентима и шаље резултат кроз нелинеарну функцију. На слици 3-15 је приказан модел вештачког неурона.



Слика 3-15: Модел вештачког неурона.

Излаз из неурона је:

$$y = f(g) \tag{3.62}$$

$$g = \sum_{i=1}^{R} p_i w_i - b \tag{3.63}$$

где су:

$p_1, p_2, \ldots, p_R$ - улазни сигнали,

$w_1, w_2, \ldots, w_R$ - тежински коефицијенти (појачања по синапсама),

b - праг активације и

f - активациона функција.

Синапсе којима биолошки неурони регулишу проходност одређене путање између аксона и дендрита, код вештачких неурона се остварују преко прилагодљивих тежинских коефицијената (енг. *Weight*) или тежина веза.

3.4.7.3 Вишеслојни перцептрон

Вишеслојне неуронске мреже са унапредном пропагацијом (енг. *Multilayer feedforward neural networks*) представљају важну класу неуронских мрежа. Мрежа се састоји од улазног слоја, једног или више скривених слојева и излазног слоја процесних елемената (неурона). Улазни сигнал се шири кроз мрежу унапред слој по слој. Ове неуронске мреже се често називају и вишеслојни перцептрони (енг. *Multi-Layer Perceptrons* - MLP).

Вишеслојни перцептрони се успешно примењују за решавање неких тешких и разноликих проблема и то кроз поступке надгледаног учења (учење са учитељом) алгоритмом пропагације грешке уназад кроз мрежу (енг. *Error back-propagation algorithm*). Алгоритам се заснива на правилу учења корекцијом грешке.

У основи, учење пропагацијом грешке уназад састоји се од два пролаза кроз различите слојеве мреже: пролазак унапред и пролазак уназад. У проласку унапред, активни узорак (улазни вектор примера) се поставља на улазне чворове мреже, и његов учинак се шири даље кроз сваки слој мреже. Коначно, генерише се скуп излаза као конкретан одзив мреже. Током проласка унапред сви тежински коефицијенти мреже су фиксни, непроменљиви. Током проласка уназад, тежински коефицијенти се подешавају у складу са правилом учења корекцијом грешке. Детаљније, разлика жељеног (циљаног) одзива и тренутног конкретног одзива мреже представља сигнал грешке. Тај сигнал грешке се даље шири уназад кроз мрежу - отуда и име „пропагација грешке уназад“. Тежински коефицијенти се подешавају тако да реални одзив мреже буде што ближи жељеном одзиву. Процес учења који се изводи алгоритмом назива се учење пропагацијом уназад (енг. *Back-propagation learning*).

Вишеслојни перцептрон одликују три карактеристике:

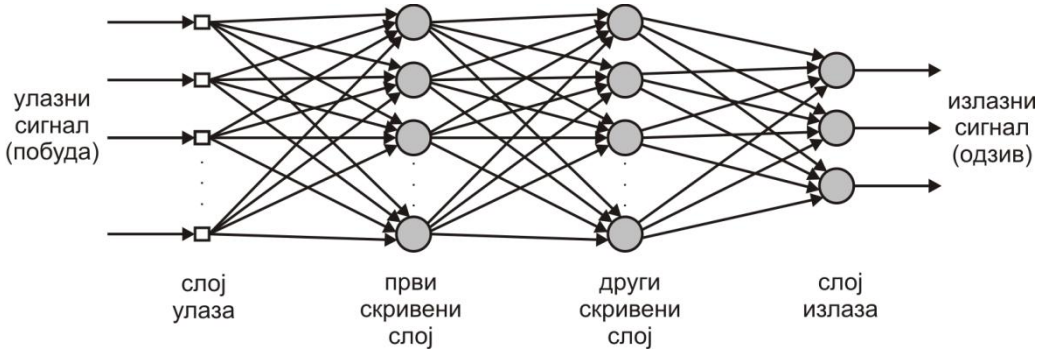
1. Модел сваког неурона у излазном и скривеним слојевима мреже укључује активациону функцију. Често коришћени облици активационе функција су дати у табели 3-2.

Назив функције	Формула
Линеарна	$f(g) = g$
По деловима линеарна	$f(g) = \begin{cases} 1 & za \quad g > 1 \\ g & za \quad -1 \le g \le 1 \\ -1 & za \quad g < -1 \end{cases}$
Функција прага (унуполарна)	$f(g) = \begin{cases} 1 & za \quad g \ge 0 \\ 0 & za \quad g < 0 \end{cases}$
Функција прага (биполарна)	$f(g) = \begin{cases} 1 & za \quad g \ge 0 \\ -1 & za \quad g < 0 \end{cases}$
Униполарна сигмоидална функција	$f(g) = \frac{1}{1 + e^{-g}}$
Биполарна сигмоидална функција	$f(g) = \frac{2}{1 + e^{-2g}} - 1$

Табела 3-2: Најчешћи облици активационе функције.

2. Мрежа садржи један или више слојева скривених неурона који нису део ни улаза ни излаза мреже. Ови скривени неурони омогућавају мрежи учење комплексних задатака.
3. Мрежа показује висок степен повезаности одређен тежинским коефицијентима (синапсама) мреже.

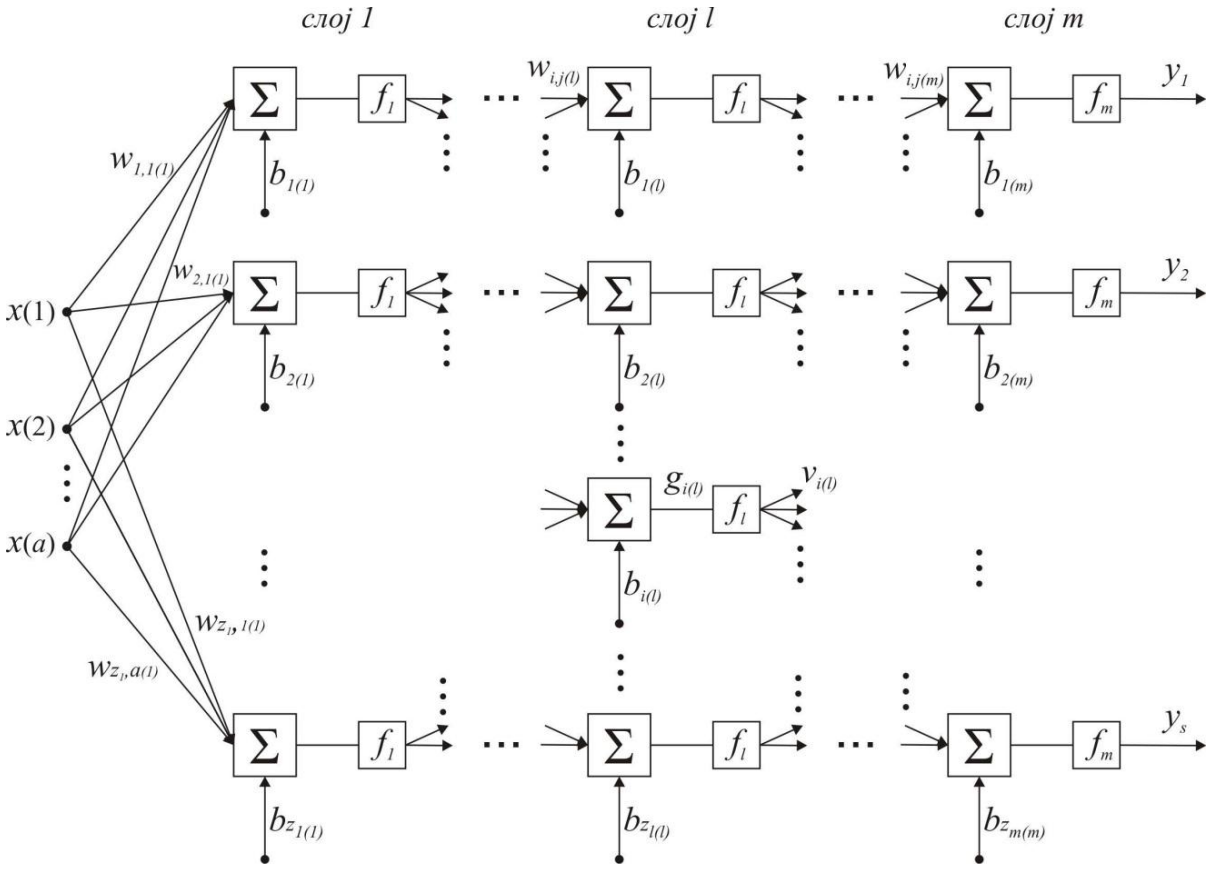
Вишеслојни перцептрон се састоји од више слојева међусобно повезаних процесних елемената (чворова или неурона). Код ове структуре су сви неурони у сваком слоју (изузев улазног) повезани са свим неуронима претходног слоја (слика 3-16). Сигнал пролази кроз мрежу унапред, с лева на десно слој по слој.



Слика 3-16: Вишеслојни перцептрон са два скривена слоја.

На слици 3-16 је приказан вишеслојни перцептрон са два скривена слоја. Број улазних и излазних неурона је дефинисан проблемом који је потребно решити: број улазних неурона је једнак броју независнио променљивих, а број излазних неурона одговара броју зависно променљивих.

Излазни неурони чине излазни слој мреже. Први скривени слој се пуни подацима из улазног слоја кога чине улазни чворови, резултујући излази првог скривеног слоја представљају улазе у следећи скривени слој, и тако даље за остатак мреже.



Слика 3-17: Вишеслојни перцептрон са m слојева.

На слици 3-17 је приказан вишеслојни перцептрон са m слојева где су:

$x(1), x(2),\ldots, x(a)$ - улазни сигнали,

m - број слојева у мрежи,

z_l - број неурона у l-том слоју,

$w_{i,j(l)}$ - тежина везе између i-тог неурона у l-том слоју и j-тог неурона у (l-1)-ом слоју,

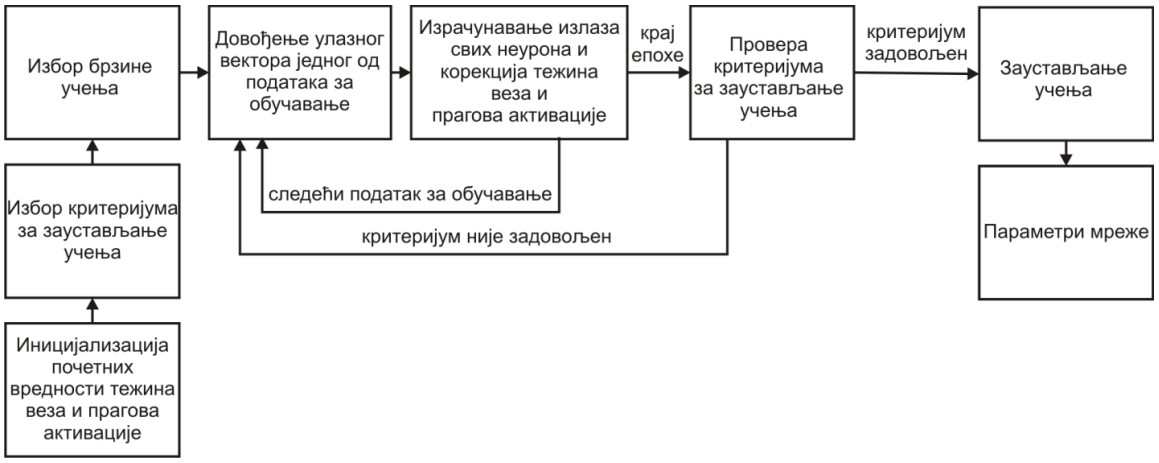
$b_{i(l)}$ - праг активације i-тог неурона у l-том слоју,

$v_{i(l)}$ - излаз i-тог неурона у l-том слоју,

f_l - активациона функција неурона у l-том слоју.

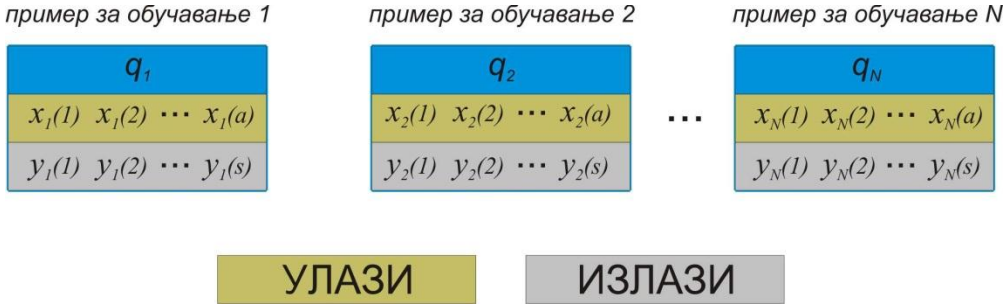
3.4.7.4 Алгоритам са пропагацијом грешке уназад

Поступак учења и адаптације параметара вишеслојног перцептрона приказаног на слици 3-17 заснива се на алгоритму са пропагацијом грешке уназад (енг. *Backpropagation algorithm*). Шематски приказ овог алгоритма је дат на слици 3-18.



Слика 3-18: Шематски приказ алгоритма са пропагацијом грешке уназад.

Овај алгоритам подразумева постојање скупа података за обучавање који су дефинисани својим улазним векторима и векторима жељених излаза (слика 3-19).



Слика 3-19: Скуп података за обучавање и тестирање вишеслојног перцептрона.

Први корак алгоритма са пропагацијом грешке уназад је иницијализација почетних вредности тежина веза и прагова активације $w_{i,j(l)}(1)$ и $b_{i(l)}(1)$. Ове вредности се бирају насумично у неком интервалу.

Следећи корак је избор критеријума за заустављање учења мреже. Један од критеријума за заустављање процеса учења је да средња квадратна грешка буде мања од унапред задате вредности или да промена ове грешке између две епохе буде довољно мала.

Након дефинисања брзине учења η, на улаз мреже се доводе улазни вектори скупа података за обучавање, један по један. За сваки улазни вектор из скупа података за обучавање врши се корекција тежина веза и прагова активације. Када се на улаз мреже доведе улазни вектор *k*-тог по реду елемента из скупа за обучавање врше се израчунавања дата једначинама (3.64)-(3.73).

Излаз *i*-тог неурона у *l*-том слоју на *k*-ти улаз је:

$$v_{i(l)}^{(k)} = f_l\left(g_{i(l)}^{(k)}\right) \tag{3.64}$$

где је:

$$g_{i(l)}^{(k)} = \sum_{j=1}^{z_{(l-1)}} w_{i,j(l)} v_{j(l-1)}^{(k)} + b_{i(l)} \tag{3.65}$$

Излаз *i*-тог неурона у *m*-том слоју је уједно и *i*-ти излаз мреже:

$$v_{i(m)}^{(k)} = f_m\left(g_{i(m)}^{(k)}\right) = \hat{y}_k(i) \tag{3.66}$$

где је:

$$g_{i(m)}^{(k)} = \sum_{j=1}^{z_{(m-1)}} w_{i,j(m)} v_{j(m-1)}^{(k)} + b_{i(m)} \tag{3.67}$$

Изрази (3.64)-(3.67) дефинишу излазе свих неурона мреже, тако да се може приступити корекцији тежина веза и прагова активације. Критеријумска функција која описује колико се стварни излаз мреже разликује од жељеног се израчунава на следећи начин:

$$E_k = \frac{1}{2}\sum_{i=1}^{s}\left(y_k(i) - \hat{y}_k(i)\right)^2 = \frac{1}{2}\sum_{i=1}^{s}\left(y_k(i) - v_{i(m)}^{(k)}\right)^2 \tag{3.68}$$

где је $y_k(i)$ тачна вредност i-тог излаза k-тог примера за обучавање, док је $\hat{y}_k(i)$ предвиђена вредност i-тог излаза k-тог примера за обучавање.

Задатак учења је прилагодити слободне параметре мреже тако да се излаз мреже у што мањој мери разликује од жељеног. У том циљу се врши минимизација функције E_k применом градијентне методе. Корекција тежина веза између неурона у (*m*-1)-ом слоју (последњи скривени слој) и *m*-том слоју (излазни слој) и прагова активације у *m*-том слоју обавља се итеративним поступком применом следећих израза:

$$w_{i,j(m)}(t+1) = w_{i,j(m)}(t) - \eta \frac{\partial E_k}{\partial w_{i,j(m)}} \tag{3.69}$$

$$b_{i(m)}(t+1) = b_{i(m)}(t) - \eta \frac{\partial E_k}{\partial b_{i(m)}} \tag{3.70}$$

Адаптација тежина веза и прагова активације у осталим слојевима се врши према једначинама (3.71)-(3.72).

$$w_{i,j(l)}(t+1) = w_{i,j(l)}(t) - \eta \frac{\partial E_k}{\partial w_{i,j(l)}} \tag{3.71}$$

$$b_{i(l)}(t+1) = b_{i(l)}(t) - \eta \frac{\partial E_k}{\partial b_{i(l)}} \tag{3.72}$$

Пошто се мрежи презентују сви елементи из скупа података за обучавање, тј. после завршене епохе, врши се провера критеријума за заустављање учења. Најчешће се рачуна средња квадратна грешка:

$$E_{av} = \frac{1}{N} \sum_{k=1}^{N} E_k \tag{3.73}$$

а затим се проверава да ли је она мања од унапред задате вредности. Ако јесте врши се заустављање процеса учења и мрежа је спремна да се пусти у експлоатацију. Ако израчуната средња квадратна грешка није мања од унапред задате вредности, мрежи се поново представљају улазни вектори из скупа података за обучавање (почиње нова епоха), врши се корекција слободних параметара мреже и након завршетка те нове епохе поново се проверава критеријум за заустављање учења.

Једначине (3.69)-(3.72) представљању основну верзију алгоритма са пропагацијом грешке уназад. Међутим, овакав итеративан поступак корекције параметара мреже често резултује спором конвергенцијом или заустављањем учења у локалном минимуму. У циљу решавања ових проблема развијене су бројне побољшане варијанте алгоритма са пропагацијом грешке уназад. Једна од модификација овог алгоритма се односи на увођење момента (α) у изразе за адаптацију тежинских коефицијената [40]. Ова константа узима у обзир промене вредности параметара мреже у претходној итерацији. Изрази за адаптацију тежина веза (једначине (3.69) и (3.71)) добијају следећи облик:

$$\begin{aligned} w_{i,j(l)}(t+1) &= w_{i,j(l)}(t) + \Delta w_{i,j(l)}(t+1) \\ \Delta w_{i,j(l)}(t+1) &= -\eta \frac{\partial E_k}{\partial w_{i,j(l)}} + \alpha \Delta w_{i,j(l)}(t) \end{aligned} \tag{3.74}$$

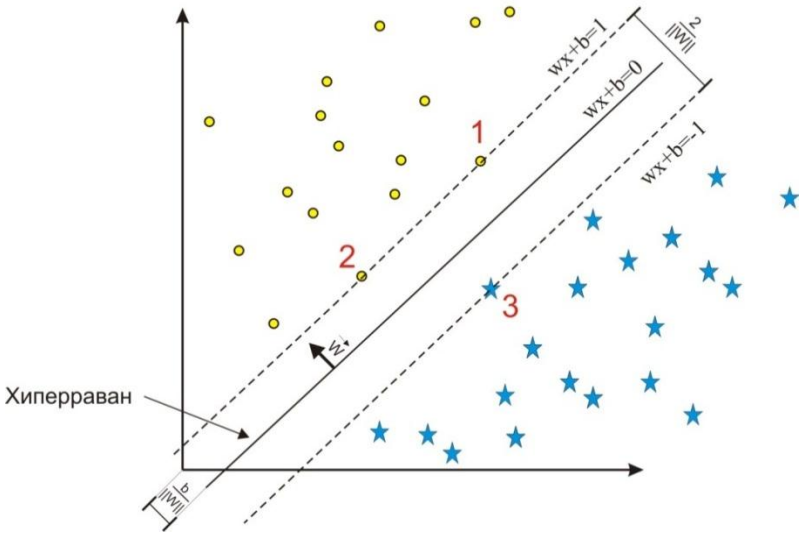
$$w_{i,j(m)}(t+1) = w_{i,j(m)}(t) + \Delta w_{i,j(m)}(t+1)$$
$$\Delta w_{i,j(m)}(t+1) = -\eta \frac{\partial E_k}{\partial w_{i,j(m)}} + \alpha \Delta w_{i,j(m)}(t) \quad (3.75)$$

Константа момента узима вредност из опсега $[0,1)$. У литератури су доступне и бројне друге модификације алгоритма са пропагацијом грешке уназад [41].

3.4.8 Метода потпорних вектора

Метода потпорних вектора (енг. *Support Vector Machine* - SVM) развијена од стране Владимира Вапника (енг. Vladimir Vapnik) [42], спада у групу најуспешнијих, како за проблеме класификације тако и за регресионе проблеме. За разлику од већине алгоритама, који теже минимизацији броја атрибута који се употребљавају за предвиђање, метода потпорних вектора користи све доступне атрибуте, чак и ако неки од њих нису од значаја. Овај алгоритам је погодан за решавање проблема са великим бројем примера који могу бити описани великим бројем атрибута.

У основи, овај алгоритам је дизајниран за решавање проблема припадности једној од две класе. Овај алгоритам креира хиперраван која раздваја примере који припадају различитим класама (слика 3-20). У случају да примере није могуће линеарно раздвојити, они се пресликавају у вишедимензионални простор F (енг. *Feature space*) у коме је могуће креирати хиперраван која успешно раздваја примере (кернелов трик). Након што се улазни вектор $\mathbf{x_k}$ тестног примера q_k преслика у простор F кернеловом функцијом K, у новом простору се одређује којој категорији нови вектор $K(\mathbf{x_k})$ припада. Оптимална хиперраван је она која максимизује маргину - растојање између примера најближих хиперравни који припадају различитим класама (потпорни вектори).



Слика 3-20: Оптимална хиперраван раздваја примере који припадају различитим класама.

Претпоставимо да имамо скуп примера у следећем облику:

$$D = \{(\mathbf{x_i}. y_i) \mid \mathbf{x_i} \in \mathbb{R}^a, y_i \in \{-1,1\}\}_{i=1}^{N} \quad (3.76)$$

Дакле, имамо укупно N примера који су описани a-димензионалним улазним векторима $\mathbf{x_i}$ и излазима y_i који могу бити -1 или 1 у зависности од тога да ли пример припада првој (C_1) или другој класи (C_2). Потребно је пронаћи оптималну хиперраван која успешно раздваја примере $y = -1$ од примера $y = 1$. Ова хиперраван може да се напише у облику:

$$\mathbf{w} \cdot \mathbf{x} - b = 0 \tag{3.77}$$

где $\cdot$ представља скаларни производ, а $\mathbf{w}$ вектор нормалан на хиперраван. Параметар $\frac{b}{||\mathbf{w}||}$ представља растојање хиперравни од координатног почетка у правцу вектора $\mathbf{w}$. У случају да су примери који припадају различитим класама линеарно раздвојиви, можемо одабрати две хиперравни тако да између њих нема примера (слика 3-20). Растојање између њих одговара маргини коју је потребно максимизовати. Ове две хиперравни су дате следећим једначинама:

$$\mathbf{w} \cdot \mathbf{x} - b = 1 \tag{3.78}$$

$$\mathbf{w} \cdot \mathbf{x} - b = -1 \tag{3.79}$$

Растојање између ове две хиперравни је $\frac{2}{||\mathbf{w}||}$. Дакле, потребно је минимизовати $\left|\left|\mathbf{w}\right|\right|$. Такође, потребно је онемогућити примере да се нађу између ових хиперравни, па се зато постављају следећа ограничења:

$$\mathbf{w} \cdot \mathbf{x} - b \geq 1, \text{ за примере } y = 1 \tag{3.80}$$

$$\mathbf{w} \cdot \mathbf{x} - b \leq -1, \text{ за примере } y = -1 \tag{3.81}$$

Ова ограничења могу бити написана као:

$$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1, \text{ за свако } 1 \leq i \leq N \tag{3.82}$$

Дакле, потребно је минизовати $\left|\left|\mathbf{w}\right|\right|$ односно $\frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2$ (минимуми ове две функције имају исто $\mathbf{w}$ и b) према претходно наведеним ограничењима. Овај оптимизациони проблем се решава помоћу лагранжових мултипликатора ($\alpha_i,\ i = 1, \dots, N$) употребом Лагранжове функције Λ која има следећи облик:

$$\Lambda(\mathbf{w}, b, \alpha_1, \dots, \alpha_N) = \frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2 - \sum_{i=1}^{N} \alpha_i [y_i(\mathbf{w} \cdot \mathbf{x_i} - b) - 1] \tag{3.83}$$

Парцијални изводи Лагранжове функције по примарним променљивима морају имати вредност 0 у циљу оптималности па се вектор $\mathbf{w}$ може представити као:

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x_i} \tag{3.84}$$

Само неколико α_i вредности ће бити веће од нуле. Њима одговарајући примери су заправо потпорни вектори, који леже на маргини и задовољавају:

$$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) = 1 \tag{3.85}$$

Применом (3.85) може се израчунати b:

$$\mathbf{w} \cdot \mathbf{x_i} - b = \frac{1}{y_i} = y_i \iff b = \mathbf{w} \cdot \mathbf{x_i} - y_i \tag{3.86}$$

где је $\mathbf{x_i}$ произвољан потпорни вектор. У пракси се b чешће израчунава усредњавањем вредности добијених употребом свих потпорних вектора:

$$b = \frac{1}{N_{sv}} \sum_{i=1}^{N_{sv}} (\mathbf{w} \cdot \mathbf{x_i} - y_i) \tag{3.87}$$

где је N_{sv} број потпорних вектора.

Након израчунавања $\mathbf{w}$ и b, излаз (припадност једној од две класе) за улазни вектор $\mathbf{x_k}$ се одређује на следећи начин:

$$\hat{c}_k = sign(\mathbf{w} \cdot \mathbf{x_k} - b) = sign\left(\sum_{i=1}^{n} \alpha_i y_i (\mathbf{x_i} \cdot \mathbf{x_k}) - b\right) \tag{3.88}$$

Заменом $\left|\left|\mathbf{w}\right|\right| = \mathbf{w} \cdot \mathbf{w}$ и једначина (3.84) и (3.86) у (3.83) добијамо дуалну форму Лагранжове функције:

$$\begin{aligned} \Lambda(\mathbf{w}, b, \alpha_1, \dots, \alpha_n) &= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x_i}^{\mathbf{T}} \mathbf{x_j} \\ &= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j}^{n} \alpha_i \alpha_j y_i y_j K\left(\mathbf{x_i}, \mathbf{x_j}\right) \end{aligned} \tag{3.89}$$

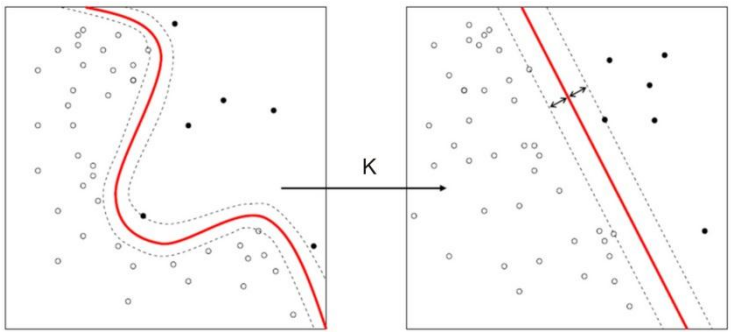
са следећим ограничењима:

$$\forall i \in 1 \dots n: \alpha_i \geq 0 \tag{3.90}$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{3.91}$$

У изразу (3.89) $K\left(\mathbf{x_i}, \mathbf{x_j}\right)$ представља кернелову функцију и у случају линеарно раздвојивих примера представља скаларни производ вектора $\mathbf{x_i}$ и $\mathbf{x_j}$.

Прва верзија алгоритма потпорних вектора развијена од стране Владимира Вапника 1963. године [42] била је намењена за решавање проблема линеарно раздвојивих примера (линерани класификатор). 1992. године овај алгоритам је надограђен тако да је омогућена и класификација линеарно нераздвојивих примера употребом кернеловог трика [43]. У овој новој верзији скаларни производ је замењен кернеловом функцијом која пресликава примере у простор F, па се онда у њему одређује оптимална хиперраван (јер су у овом простору примери линеарно раздвојиви - слика 3-21).



Слика 3-21: Кернелов трик.

Неке од уобичајних кернелових функција су:

- Полиномска (хомогена): $K(\mathbf{x_i}, \mathbf{x_j}) = (\mathbf{x_i} \cdot \mathbf{x_j})^d$
- Полиномска (нехомогена): $K(\mathbf{x_i}, \mathbf{x_j}) = (\mathbf{x_i} \cdot \mathbf{x_j} + 1)^d$
- Радијална (енг. *Radial Basis Function* - RBF): $K(\mathbf{x_i}, \mathbf{x_j}) = e^{-\gamma\left|\left|\mathbf{x_i}-\mathbf{x_j}\right|\right|^2}$
- Хиперболична тангентна: $K(\mathbf{x_i}, \mathbf{x_j}) = \tanh(a\mathbf{x_i} \cdot \mathbf{x_j} + c)$, за $a > 0, c < 0$

Дакле, у случају линеарно нераздвојивих примера користи се кернелов трик па једначина (3.88) постаје:

$$\hat{c}_k = sign(\mathbf{w} \cdot \mathbf{x} - b) = sign\left(\sum_{i=1}^{N} \alpha_i y_i K(\mathbf{x_i}, \mathbf{x}) - b\right) \tag{3.92}$$

1996. године развијена је верзија алгоритма за решавање регресионих проблема [44]. У случају регресионог проблема потребно је постојање скупа примера:

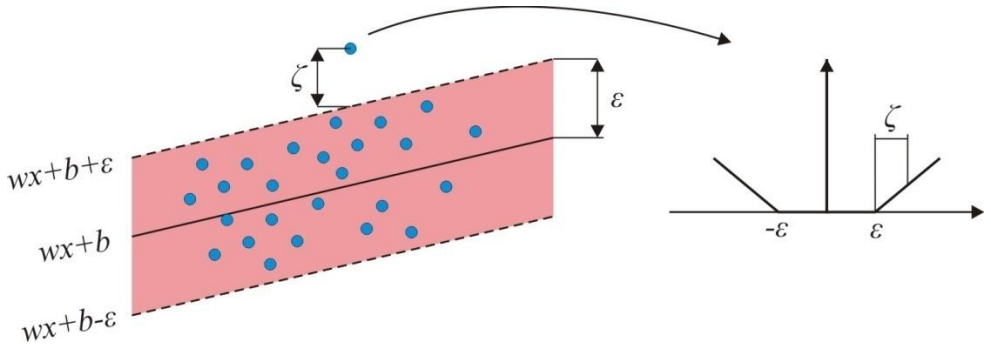
$$D = \{(\mathbf{x_i}. y_i) \mid \mathbf{x_i} \in \mathbb{R}^a, \}_{i=1}^N \tag{3.93}$$

где је $\mathbf{x_i}$ i-ти улазни вектор а y_i њему одговарајући излаз (континуална вредност).

За решавање регресионих проблема најпре је потребно дефинисати функцију грешке $\mathcal{L}(y, \hat{y})$, која показује колико SVM функција одступа од реалних података. Најчешће употребљавана функција грешке је Вапникова функција дефинисана као:

$$\mathcal{L}(y, \hat{y}) = \begin{cases} 0 & |y - \hat{y}| \leq \varepsilon \\ |y - \hat{y}| - \varepsilon & \text{у супротном} \end{cases} \tag{3.94}$$

где је $\varepsilon > 0$ толеранција грешке. Овај регресиони алгоритам не узима у обзир грешке мање од ε вредности (слика 3-22).



Слика 3-22: Вапникова функција грешке.

SVM регресиона функција је дефинисана као:

$$\hat{y} = \mathbf{w} \cdot \mathbf{x} + b \tag{3.95}$$

Како би одредили $\mathbf{w}$ и b потребно је решити следећи оптимизациони проблем:

$$\text{минимизовати} \quad \frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2 \tag{3.96}$$

$$\text{према} \quad \begin{cases} y_i - (\mathbf{w} \cdot \mathbf{x_i} + b) \leq \varepsilon \\ (\mathbf{w} \cdot \mathbf{x_i} + b) - y_i \leq \varepsilon \end{cases} \tag{3.97}$$

Претходно дефинисана ограничења се ослањају на претпоставку да постоји функција (3.95) која предвиђа излазе свих примера са прецизношћу ε. Међутим, у пракси ово често није могуће па се уводе променљиве ζ_i и ζ_i^* које дозвољавају одређену дозу грешке. Оптимизациони проблем сада постаје:

$$\text{минимизовати} \quad \frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2 + C\sum_{i=1}^{N}(\zeta_i + \zeta_i^*) \tag{3.98}$$

$$\text{према} \quad \begin{cases} y_i - (\mathbf{w} \cdot \mathbf{x_i} + b) \leq \varepsilon + \zeta_i \\ (\mathbf{w} \cdot \mathbf{x_i} + b) - y_i \leq \varepsilon + \zeta_i^* \\ \zeta_i, \zeta_i^* \geq 0 \end{cases} \tag{3.99}$$

Како би се решио претходни оптимизациони проблем потребно је конструисати Лагранжову функцију Λ која има следећи облик:

$$\begin{aligned} \Lambda(\mathbf{w}, b, \zeta_1, \ldots \zeta_n, \zeta_1^*, \ldots \zeta_n^*, \alpha_1, \ldots, \alpha_n, \lambda_1, \ldots, \lambda_n) \\ = \frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2 + C\sum_{i=1}^{N}(\zeta_i + \zeta_i^*) - \sum_{i=1}^{N}(\lambda_i\zeta_i + \lambda_i^*\zeta_i^*) \\ - \sum_{i=1}^{N}\alpha_i(\varepsilon + \zeta_i - y_i + (\mathbf{w} \cdot \mathbf{x_i} + b)) \\ - \sum_{i=1}^{N}\alpha_i^*(\varepsilon + \zeta_i^* + y_i - (\mathbf{w} \cdot \mathbf{x_i} + b)) \end{aligned} \tag{3.100}$$

где су $\alpha_i, \alpha_i^*, \lambda_i$ и λ_i^* Лагранжови мултипликатори који морају задовољити следећа ограничења:

$$\alpha_i, \alpha_i^*, \lambda_i, \lambda_i^* \geq 0 \tag{3.101}$$

Парцијални изводи Лагранжове функције по примарним променљивима $(w, b, \zeta_i, \zeta_i^*)$ морају имати вредност 0 у циљу оптималности:

$$\frac{\partial \Lambda}{\partial b} = \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) = 0 \tag{3.102}$$

$$\frac{\partial \Lambda}{\partial w} = \mathbf{w} - \sum_{i=1}^{N} (\alpha_i - \alpha_i^*)\mathbf{x_i} = 0 \tag{3.103}$$

$$\frac{\partial \Lambda}{\partial \zeta_i^{(*)}} = C - \alpha_i^{(*)} - \lambda_i^{(*)} = 0 \tag{3.104}$$

У претходној једначини $\alpha_i^{(*)}$ се односи на α_i и α_i^* (исто важи и за $\lambda_i^{(*)}$). Заменом (3.101), (3.102) и (3.103) у (3.100) добијамо дуални оптимизациони проблем:

$$\begin{aligned} &\text{максимизовати} \quad -\frac{1}{2}\sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)(\mathbf{x_i} \cdot \mathbf{x_j}) - \varepsilon \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) \\ &\text{према} \qquad \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0 \ \ and \ \ \alpha_i, \alpha_i^* \in [0, C] \end{aligned} \tag{3.105}$$

Величине $\mathbf{w}$ и b израчунавају на следећи начин:

$$\mathbf{w} = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*)\mathbf{x_i} \tag{3.106}$$

$$b = -\frac{1}{2}\left(\mathbf{w} \cdot (\mathbf{x_r} + \mathbf{x_s})\right) \tag{3.107}$$

У једначини (3.107) $\mathbf{x_r}$ и $\mathbf{x_s}$ су потпорни вектори (било који улазни вектори који имају ненулте вредности за α_i или α_i^*).

Коначно, за улазни вектор $\mathbf{x_k}$, излаз се израчунава помоћу следеће једначине:

$$\hat{y}_k = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*)(\mathbf{x_i} \cdot \mathbf{x}) + b \tag{3.108}$$

За нелинеарне проблеме користе се кернелове функције. Оптимизациони проблем се тада дефинише као:

$$\begin{aligned} &\text{максимизовати} \quad -\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)K(\mathbf{x_i}\cdot\mathbf{x_j})-\varepsilon\sum_{i=1}^{N}(\alpha_i+\alpha_i^*)+\sum_{i=1}^{N}y_i(\alpha_i-\alpha_i^*) \\ &\text{према} \quad \sum_{i=1}^{N}(\alpha_i-\alpha_i^*)=0 \;\; and \;\; \alpha_i,\alpha_i^*\in[0,C] \end{aligned} \tag{3.109}$$

па једначина (3.108) добија следећи облик:

$$\hat{y}_k=\sum_{i=1}^{N}(\alpha_i-\alpha_i^*)\cdot K(\mathbf{x_i},\mathbf{x})+b \tag{3.110}$$

3.5. Тестирање модела

3.5.1 Тестирање класификационих модела

Тестирање класификационог модела је израчунавање „мере“ која описује кориснику успешност посматраног модела. Постоје различите величине које описују успешност класификационих модела а неке од њих ће бити описане у овом поглављу.

Тачност, као мера тестирања класификационих модела се користи када се свакој погрешној класификацији придаје једнак значај. Тада је укупан број грешака на посматраном скупу примера добар показатељ тачности рада посматраног модела. Тачност представља однос исправно класификованих примера и укупног броја класификованих примера.

$$\text{Тачност}=\frac{\text{број исправно класификованих примера}}{\text{укупан број класификованих примера}}=\frac{TP+TN}{TP+TN+FP+FN} \tag{3.111}$$

Највећи број мера за тестирање класификационих модела се односи на класификационе проблеме са две класе. Ово не представља ограничење у погледу употребе тих мера с обизиром на чињеницу да се проблеми са више класа могу приказати у облику низа проблема са две класе. Табела 3-3 представља матрицу грешака (енг. *Confusion matrix*) која омогућава визуализацију грешака класификационог модела.

	Позитивни примери класе	Негативни примери класе
Позитивно предвиђање класе	Стварно позитивни (TP)	Лажно позитивни (FP)
Негативно предвиђање класе	Лажно негативни (FN)	Стварно негативни (TN)

Табела 3-3: Матрица грешака.

Из претходне табеле се види да су могућа четири исхода предвиђања. Стварно позитивни и стварно негативни исходи представљају исправну класификацију. Лажно позитивни и лажно негативни исходи представљају два могућа типа грешке. Лажно позитиван пример је негативан пример класе који је грешком класификован као позитиван. Обрнуто, лажно негативан пример је позитиван пример класе грешком класификован као негативан.

Сензитивност и *специфичност* спадају у мере за тестирање класификационих модела које разликују два поменута типа грешке. Сензитивност мери тачност у позитивним примерима, а специфичност у негативним.

$$\text{Сензитивност} = \frac{TP}{TP + FN} \tag{3.112}$$

$$\text{Специфичност} = \frac{TN}{TN + FP} \tag{3.113}$$

Одзив и *прецизност* су други пар мера за тестирање класификационих модела. Дефинисани су следећим изразима:

$$\text{Одзив} = \frac{TP}{TP + FN} \tag{3.114}$$

$$\text{Прецизност} = \frac{TP}{TP + FP} \tag{3.115}$$

Као и сензитивност, одзив представља тачност у позитивним примерима. Са друге стране прецизност је тачност у позитивном предвиђању циљне класе.

Позитивна и негативна предиктивна вредност је још један пар мера за тестирање класификационих модела.

$$\text{Позитивна предиктивна вредност} = \frac{TP}{TP + FP} \tag{3.116}$$

$$\text{Негативна предиктивна вредност} = \frac{TN}{TN + FN} \tag{3.117}$$

Понекад је успешност класификационог модела потребно изразити једним бројем, а не паром зависних мера. Једна од таквих мера је и Ф-мера (енг. *F-measure*).

$$\text{Ф} - \text{мера} = \frac{2 \cdot \text{одзив} \cdot \text{прецизност}}{\text{одзив} + \text{прецизност}} = \frac{2TP}{2TP + FP + FN} \tag{3.118}$$

3.5.2 Тестирање регресионих модела

Тестирање регресионих модела се најчешће врши израчунавањем неке од следећих грешака:

- *Средња квадратна грешка* (енг. *Mean Squared Error* - MSE):

$$MSE = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (y_i - \hat{y}_i)^2 \tag{3.119}$$

  где је N_{test} број примера за тестирање, y_i је стварни излаз i-тог примера, а $\hat{y}_i$ је предвиђена вредност регресионог модела за i-ти пример.

- *Корен средње квадратне грешке* (енг. *Root Mean Squared Error* - RMSE):

$$RMSE = \sqrt{\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (y_i - \hat{y}_i)^2} \tag{3.120}$$

- *Средња апсолутна грешка* (енг. *Mean Absolute Error* - MAE):

$$MAE = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} |y_i - \hat{y}_i| \tag{3.121}$$

- *Релативна средња квадратна грешка* (енг. *Relative Mean Squared Error* - RMSE):

$$RMSE = \frac{\sum_{i=1}^{N_{test}} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N_{test}} (y_i - \bar{y})^2} \tag{3.122}$$

где је $\bar{y}$ средња вредност излаза примера за тестирање:

$$\bar{y} = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} y_i \tag{3.123}$$

3.5.3 Тестирање на основу тестног скупа примера

Стварни квалитет модела за предвиђање (регресионог или класификационог) одређен је способношћу модела да исправно реагује када му се на улаз доводе нови примери, који нису били укључени у процес обучавања. Из тог разлога уобичајно је да се у поступку генерисања модела не користе сви доступни примери, већ се иницијални скуп примера дели на два дела:

- *Скуп примера за обучавање*, који се користи за обучавање модела,
- *Скуп примера за тестирање*, који се користи за накнадно тестирање модела.

При томе је веома важно да ова два скупа буду случајно одабрана и независна.

Подела иницијалног скупа на скуп за обучавање и скуп за тестирање омогућава да се тестирање модела врши на примерима који нису били коришћени процесу креирања модела, будући да се ради са новим, дотад невиђеним примерима. Највећа предност ове мотоде лежи у прорачунској незахтевности. Са друге стране недостатак је одбацивање великог броја примера приликом креирања модела.

3.5.4 Унакрсна валидација

Није редак случај да је за цео процес креирања интелигентног модела доступан само мали број примера. Ово је веома честа појава у медицини где се неке студије изводе на веома малом броју пацијената. У таквим ситуацијама је врло важно да се иницијални скуп примера што је могуће боље искористи.

Један од узрока могуће непрецизности методе тестирања модела на основу тестног скупа примера је проблем ослањања на једну, можда некарактеристичну партицију скупа примера за учење.

Основни начин избегавања оваквих аномалија је вишеструко понављање процеса тестирања на тестном скупу користећи различите, случајно одабране скупове за учење и тестирање, после чега се врши усредњавање грешака добијених на различитим тестним скуповима. Метода унакрсне валидације (енг. *Cross validation*) се заснива на овом принципу, уз својеврсну замену скупа података за обучавање и тестног скупа у свакој итерацији [45].

У поступку k-струке унакрсне валидације најпре се иницијални скуп примера подели на k међусобно различитих подскупова приближно исте величине. Сам поступак је итеративан с тим да се у једној итерацији $k-1$ подскупова користи као скуп за учење, а конструисани модел се тестира на преосталом подскупу који представља тестни скуп примера. Поступак се понавља k пута тако да је сваки подскуп бар једном у улози тестног скупа. Просечна грешка свих k итерација представља грешку унакрсном валидацијом (провером).

Приликом поделе иницијалног скупа примера у k подскупова често се поступак случајног одабира модификује са циљем осигуравања приближно једнаке заступљености класа у сваком од подскупова. Овај поступак се назива стратификација, а основна му је сврха побољшање репрезентативности сваког подскупа.

Од броја подскупова директно зависи рачунарска сложеност тестирања модела унакрсном валидацијом, будући да свака итерација укључује засебно конструисање и тестирање модела. Иако то није правило, у пракси се најчешће користи стратификована 10-струка унакрсна валидација.

Предности унакрсне валидације се огледају у чињеници да су сви доступни примери искоришћени за тестирање, а и конструкција модела се у свакој итерацији користи великом већином доступних примера.

3.5.5 Метода изостављања једног примера

Метода изостављања једног примера (енг. *Leave-One-Out Cross Validation* - LOOCV) је специјалан случај унакрсне валидације. Ако је са N означен укупан број иницијално доступних примера, изостављање једног примера је N-струка унакрсна валидација. Дакле, у свакој од N итерација ове методе модел се креира употребом $N-1$ примера, а тестира на преосталом примеру.

Највећа предност ове методе јесте максимална искоришћеност примера за обучавање. Са друге стране, недостатак је велика рачунарска сложеност. Из тог разлога је метода изостављања једног примера преферирана у случају мањег скупа примера, док за веће скупове примера може бити рачунарски прескупа.

3.6. Поузданост предвиђања

За дати модел за предвиђање, нека предвиђања могу бити поуздана а неке не. Тестирање модела за предвиђање нам даје увид у тачност целокупног модела, а не пружа никакву информацију од поузданости појединачних предвиђања. Мера поузданости појединачних предвиђања је од великог значаја у ситуацијама када погрешна одлука може „скупо“ коштати (медицина, берза итд). Примера ради, приликом медицинске дијагнозе, лекаре не интересује просечна прецизност интелигентног модела. Када се анализира одређени пацијент, лекарима је од великог значаја предвиђање, али и поузданост тог предвиђања како би знали колико се могу ослонити на интелигентни модел приликом успостављања дијагнозе.

Постоји различити агоритами за израчунавање поузданости предвиђања. У овом раду ће бити описане три мере поузданости предвиђања базиране на анализи осетљивости (SA_{var}, SA_{bias-s}, SA_{bias-a}), мера поузданости базирана на густини ($DENS$), две мере поузданости базиране на најближим суседима (CNK_s, CNK_a), и мера поузданости базирана на локалној унакрсној провери (LCV).

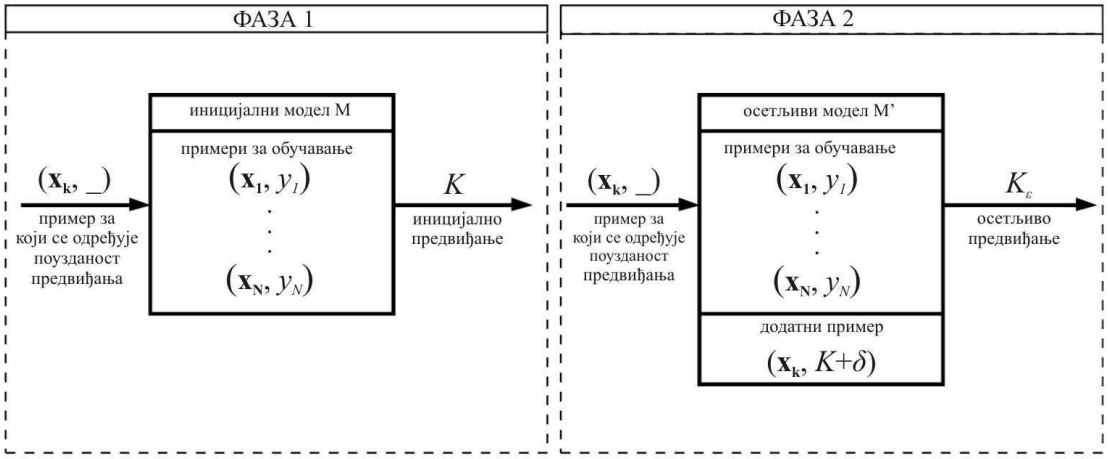
3.6.1 Поузданост предвиђања базирана на анализи осетљивости

Израчунавање поузданости предвиђања на основу анализе осетљивости увели су Боснић и Кононенко [4], [5]. Циљ обучавања модела за предвиђање је минимизација грешке над скупом примера за обучавање и тестирање. Додавањем или уклањањем једног примера из скупа примера за обучавање прави се минимална измена, па се очекује да промена у предвиђању излаза посматраног примера такође буде минимална. Велике промене у предвиђању излаза које се јављају услед минималних измена у подацима за обучавање могу представљати знак нестабилности генерисаног модела. Стога, величина промене излаза се може користити као мера нестабилности модела за посматрани пример. Минимална измена у подацима за обучавање се врши као што је описано у наставку текста.

Нека је $\mathbf{x_k}$ улаз а y_k излаз посматраног примера $q_k(\mathbf{x_k}, y_k)$. За израчунавање поузданости предвиђања на основу анализе осетљивости није неопходно познавање излаза посматраног примера. Стога се посматрани пример, за који је потребно израчунати поузданост предвиђања, може обележити као $q_k(\mathbf{x_k}, \_)$. *Иницијално предвиђање* K се добија пропуштањем примера кроз обучени модел $f_M(\mathbf{x_k}) = K$. Ово предвиђање носи назив иницијално предвиђање јер се добија на почетку анализе осетљивости употребом немодификованог скупа N примера. Након одређивања иницијалног предвиђања проширујемо скуп примера за обучавање додавањем примера $(\mathbf{x_k}, K + \delta)$, где δ означава малу промену (позитивну или негативну). Ако је интервал вредности излаза примера за обучавање ограничен са $[a, b]$, онда се δ обично дефинише помоћу параметра ε као $\delta = \varepsilon(b - a)$, где ε представља релативни удео интервала излаза.

Након дефинисања параметра ε додајемо у скуп за обучавање пример $\big(\mathbf{x_k}, K + \varepsilon(b - a)\big)$. Употребом модификованог скупа за обучавање који сада садржи $N + 1$

примера креира се нови модел за предвиђање M'. Употребом модела за предвиђање M' врши се предвиђање излаза примера $q(\mathbf{x_k}, \_)$ и добија *осетљиво предвиђање* $f_{M'}(\mathbf{x_k}) = K_\varepsilon$. Описана процедура је приказана на слици 3-23.



Слика 3-23: Процес аланизе осетљивости. Фаза 1 – иницијално предвиђање K; Фаза 2 – осетљиво предвиђање K_ε.

Избором различитих вредности параметра $\varepsilon_i \in \{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_m\}$ итеративно се израчунава сет осетљивих предвиђања K_{ε_1}, $K_{-\varepsilon_1}$, K_{ε_2}, $K_{-\varepsilon_2}$,…, K_{ε_m}, $K_{-\varepsilon_m}$. У циљу одређивања стабилности модела израчунавају се три мере поузданости предвиђања (SA_{var}, SA_{bias-s}, SA_{bias-a}) употребом разлика $K_\varepsilon - K$. Слика 3-24 приказује како разлике $K_\varepsilon - K$ утичу на одређивање поузданости предвиђања.

Слика 3-24: Поузданост дефинисана као осетљивост предвиђања (три примера).

Дакле, за израчунавање поузданости предвиђања на бази анализе осетљивости се користе разлике између иницијалних и осетљивих предвиђања ($K_\varepsilon - K$). Ове разлике се комбинују на следећи начин како би се израчунале три мере поузданости предвиђања:

$$SA_{var} = \frac{\sum_{i=1}^{m}\left(K_{\varepsilon_i} - K_{-\varepsilon_i}\right)}{m} \tag{3.124}$$

$$SA_{bias-s} = \frac{\sum_{i=1}^{m}\left[\left(K_{\varepsilon_i} - K\right) + \left(K_{-\varepsilon_i} - K\right)\right]}{2m} \tag{3.125}$$

$$SA_{bias-a} = \frac{\sum_{i=1}^{m}\left[\left|K_{\varepsilon_i} - K\right| + \left|K_{-\varepsilon_i} - K\right|\right]}{2m} \tag{3.126}$$

где је m број различитих вредности параметра $\varepsilon_i \in \{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m\}$. Мање вредности SA_{var}, SA_{bias-s} и SA_{bias-a} указују на већу поузданост предвиђања.

3.6.2 Поузданост предвиђања базирана на густини

Традиционални поступак за одређивање поузданости предвиђања се базира на дистрибуцији примера за обучавање. У овом поглављу се користи термин потпростор који се односи на подскуп примера за обучавање који су повезани локално преко вредности атрибута. Ако се пореде два потпростора исте величине, потпростор који садржи већи број примера се означава као потпростор веће густине.

Одређивање поузданости на основу густине се заснива на претпоставци да је поузданост предвиђања већа за предвиђања у потпростору веће густине и обрнуто. Ово значи да верујемо предвиђањима на основу количине информација које су доступне за израчунавање предвиђања. Типичан пример за ово је стабло одлучивања где можемо веровати предвиђањима на основу броја примера који се налазе у спољашњим чворовима.

За одређивање густине потпростора се може користити Парзенов прозор (енг. *Parzen windows*) [46]. За пример $q_k(\mathbf{x_k}, y_k)$ густина (расподела вероватноће) се израчунава на следећи начин:

$$DENS(q_k) = \frac{1}{Nh^a}\sum_{i=1}^{N} K\left(\frac{\mathbf{x_k} - \mathbf{x_i}}{h}\right) \tag{3.127}$$

где је a број атрибута, N укупан број примера за обучавање, h дужина интервала, а K кернелова функција:

$$K(\mathbf{u}) = \left(\frac{1}{\sqrt{2\pi}}\right)^a e^{\left[-\frac{\|\mathbf{u}\|^2}{2}\right]} \tag{3.128}$$

где је $\|\mathbf{u}\|^2$ норма вектора $\mathbf{u}$.

3.6.3 Поузданост предвиђања базирана на најближим суседима

Нека је дат скуп N примера за обучавање $[q_1(\mathbf{x_1}, y_1), q_2(\mathbf{x_2}, y_2), \ldots, q_N(\mathbf{x_N}, y_N)]$. Како би се одредила поузданост предвиђања за неки нови пример $q_p(\mathbf{x_p}, y_p)$, најпре је потребно одредити иницијално предвиђање K пропуштањем примера кроз обучен модел $f_M(\mathbf{x_p}) = K$ (иницијални модел M је обучен употребом N примера). Овде се уводе две нове мере поузданости предвиђања које су базиране на најближим суседима посматраног примера $q_p(\mathbf{x_p}, \_)$ [5]:

$$CNK_s = \frac{\sum_{i=1}^{k} y_i}{k} - K \tag{3.129}$$

$$CNK_a = \left| \frac{\sum_{i=1}^{k} y_i}{k} - K \right| \tag{3.130}$$

где је k број најближих суседа, y_i је излаз i-тог најближег суседа (континуална вредност), а K иницијално предвиђање. За регресионе проблеме поузданост предвиђања је већа за мање вредности CNK_s и CNK_a.

Код класификационих проблема CNK се израчунава одузимањем просечног одступања предвиђене дистрибуције класа и дистрибуције класа најближих суседа од јединице. У овом случају, поузданост предвиђања је већа за веће вредности CNK.

$$CNK = 1 - \frac{\sum_{i=1}^{k} H(C_i, K)}{k} \tag{3.131}$$

где је K дистрибуција класа иницијалног предвиђања, C_i тривијална дистрибуција класа i-тог најближег суседа, а функција $H(C_i, K)$ је Хелингерово растојање (енг. *Hellinger distance*):

$$H(C_i, K) = \frac{1}{\sqrt{2}} \sqrt{\sum_{i=1}^{m} (\sqrt{C_i(i)} - \sqrt{K(i)})^2} \tag{3.132}$$

где је m број класа.

3.6.4 Поузданост предвиђања базирана на локалној унакрсној провери

Поузданост предвиђања се израчунава на основу грешака предвиђања добијених применом унакрсне провере изостављањем једног примера (LOOCV) над скупом најближих суседа посматраног примера $q_p(\mathbf{x_p}, y_p)$. Алгоритам за регресионе проблеме је приказан на слици 3-25.

1. Одређивање сета k најближих суседа: $(q_1(\mathbf{x_1}, y_1), q_2(\mathbf{x_2}, y_2), \ldots, q_k(\mathbf{x_k}, y_k))$.
2. Над овим сетом одрадити унакрсну проверу изостављањем једног примера (LOOCV) и израчунати предвиђања $\hat{y}_i$ и грешке предвиђања $E_i = |y_i - \hat{y}_i|$.
3. Израчунати поузданост предвиђања: $LCV(q_p) \frac{\sum_{i=1}^{k} D(q_i, q_p) \cdot E_i}{\sum_{i=1}^{k} D(q_i, q_p)}$,

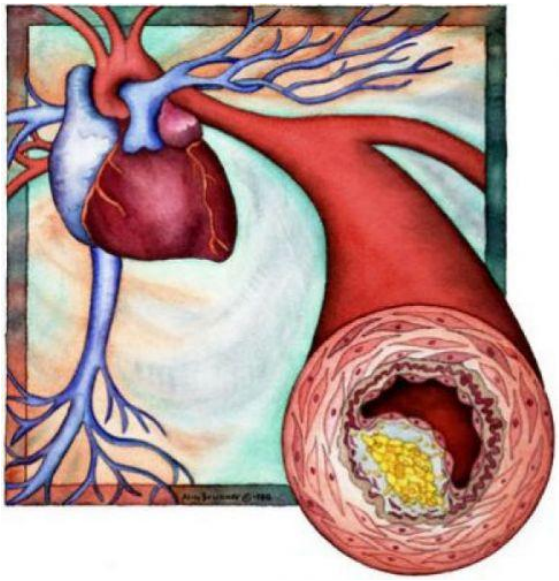
   где је $D(q_i, q_p)$ Еуклидско растојање између примера q_i и q_p.

Слика 3-25: LCV поузданост предвиђања - алгоритам за регресионе проблеме.

4. Повезивање података добијених из хемодинамичких симулација употребом техника истраживања података

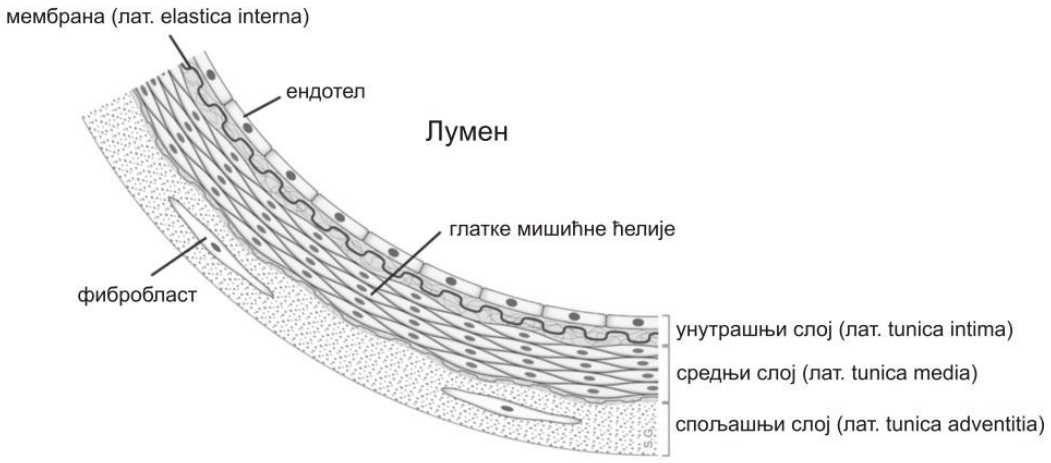
4.1. Смичући напон на зиду и атеросклероза

Атеросклероза је болест великих и малих мишићних артерија. Карактерише се дисфункцијом ендотела, васкулитисом и накупљањем липида, холестерола, калцијума и ћелијских елемената унутар зида крвног суда. Овај процес за последицу има формирање плака, васкуларно ремоделовање, акутну и хроничну опструкцију лумена крвног суда и смањену оксигенацију циљаних ткива. На слици 4-1 је приказано сужење артерије које је последица атеросклерозе.



Слика 4-1: Сужење артерије услед настанка плака.

Механизам атеросклерозе није у потпуности познат, али се почетак обично везује за повреду ендотела који представља део зида крвног суда (слика 4-2). Ендотел је биолошка структура одговорна за комуникацију између елемената крви и зида крвног суда. Ендотелне ћелије представљају баријеру продирању штетних агенаса у зид крвног суда. Здрав ендотел има антитромботичка својства и спречава адхезију ћелија крви (тромбоцита, еритроцита и леукоцита) за зид крвног суда. Међутим, у одређеним условима може доћи до повреде ендотелних ћелија које на оштећење одговарају структурним променама, секрецијом цитокина и експресијом адхезионих молекула. Поред тога, оштећене ендотелне ћелије секретују факторе који имају ефекат на диференцијацију и раст глатких мишићних ћелија.



Слика 4-2: Структура зида крвног суда.

Ендотелни фактори ослобођени у крвоток узрокују хемотаксију леукоцита из крви који мигрирају ка зиду крвног суда. Ендотелне ћелије индукују адхезију тромбоцита и леукоцита на својој површини експресијом специфичних адхезивних молекула (селектина, интегрина, супергенске фамилије имуноглобулина). Под дејством TNF-α (фактора туморске некрозе-α), IL-1 (интерлеукина-1) и ендотоксина повећава се експресија VCAM (енг. *Vascular Cell Adhesion Molecule*) и ICAM-1 (енг. *Intercellular Adhesion Molecule-1*), који припадају супергенској фамилији имуноглобулина, као и селектина ELAM-1 (енг. *Endothelium-Leucocyte Adhesion Molecule-1*).

Услед дисфункције ендотела долази до продирања липопротеина мале густине (енг. *Low-Density Lipoprotein* - LDL) унутар зида крвног суда. Након оксидације LDL-а долази до регрутовања моноцита који из крви прелазе у зид крвног суда. Моноцити се након уласка у зид крвног суда трансформишу у макрофаге који имају способност упијања оксидисаног LDL-а. Када упију довољну количину оксидисаног LDL-а макрофази прелазе у пенасте ћелије (енг. *Foam cells*).

У оваквим условима, ендотелне ћелије појачано синтетишу протеине ванћелијског матрикса (енг. *Extracellular matrix*) колаген IV, фибронектин и др., у циљу изолације промена и интими, формирајући тако фиброзну капу која проминира у лумен крвног суда. Овим би процес атеросклерозе био стабилизован, али макрофази секретују протеиназе које резлажу фибризну капу. На тај начин може доћи до пуцања атеросклерозног плака и других компликација.

И поред чињенице да је концентрација LDL-а у крви иста у свим артеријама, одређене регије, као што су места гранања и кривудаве регије, су подложније развоју атеросклерозе и настанку плака од осталих. Ово указује на то да локални хемодинамички фактори, као што је смичући напон на зиду артерија, играју значајну улогу у процесу позиционирања лезије. Испитивања су показала да сложен проток крви може изазвати митозу и умирање ћелија ендотела. Услед овога долази до промене пропустљивости тако да је омогућен пролазак великих молекула, као што су молекули LDL-а.

Силе које делују на зид крвног суда услед хемодинамике су (слика 4-3):

- Смичући напон на зиду (енг. *Wall Sear Sstress* - WSS)

$$\tau = \mu \frac{\partial u}{\partial y}\Big|_{y=0} \tag{4.1}$$

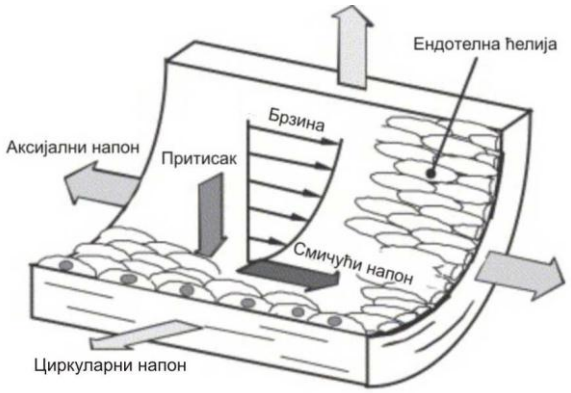
где је μ динамичка вискозност флуида, u тангенцијална брзина флуида, а y нормално растојање од зида.

- Циркуларни напон (енг. *Circumferential stress*)

$$\sigma_\theta = P \frac{r}{t} \tag{4.2}$$

где је P притисак, r унутрашњи полупречник крвног суда, а t дебљина зида.

- Притисак (енг. *Pressure*).



Слика 4-3: Силе које делују на зид крвног суда.

Интересантна је чињеница да је атеросклероза болест артерија, а да се никада не јавља у венама. Још увек се не може са сигурношћу објаснити овај феномен, али постоје бројне претпоставке о томе. Једна од њих тврди да атеросклероза напада само крвне судове високог притиска. Вене имају мањи пречник и мање су еластичне од артерија и спроводе крв под малим притиском. Артерије, са друге стране, су танкозидне и спроводе крв од срца. Оне захтевају висок притисак како би било омогућено спровођење крви.

Услед различитих протока и брзина струјања крви, артерије и вене су изложене различитим вредностима смичућег напона на зиду. У артеријама, нормална вредност смичућег напона на зиду варира између 10 и 70 dyn/cm². Код вена овај опсег варира између 1 и 6 dyn/cm².

Бројна литература показује да су зоне артерије са ниским вредностима смичућег напона на зиду склоне настанку и развоју атеросклерозе [47]-[49]. Такође, истраживања показују да су кривудаве регије и места гранања артерија, у којима је струјање често нестационарно и која су изложена просторним и временским варијацијама смичућег напона на зиду, веома погодна за настанак и развој атеросклерозе [49], [50].

Међутим, константна изложеност патогеним нивоима вредности смичућег напона на зиду може довести до промена на зиду артерија које могу довести до појаве атеросклеротичних лезија. Ове промене могу довести до пролиферације и миграције глатких мишићних ћелија, као и до адхезије и миграције леукоцита из крви [51]. Такође, изложеност веома високим вредностима смичућег напона на зиду може довести до дестабилизације већ постојећег плака [52].

Висока вредност смичућег напона може довести до смањења дебљине зида крвног суда. Ова појава доводи до повећања пречника крвног суда, а самим тим и до смањења вредности смичућег напона на зиду. Са друге стране, ниска вредност смичућег напона на зиду доводи до задебљања зида крвног суда. Ова појава доводи до смањења пречника крвног суда што за последицу има повећање вредности смичућег напона на зиду. Дуготрајно мењање нивоа смичућег напона, а самим тиме и структуре зида артерије може довести до настанка и развоја плака.

4.2. Предвиђање положаја и вредности највећег смичућег напона на зиду за модел каротидне бифуркације

4.2.1 Опис проблема

После срчаних болести и канцера, трећи најчешћи узрок смрти је шлог [2], [53]. Стеноза каротидне бифуркације (лат. *Bifurcus* - рачваст) може бити узрок шлога, доводећи до инфаркта емболизацијом или тромбозом на месту сужења. Тромбоза и емболизација су процеси који у великој мери зависе од локалне хемодинамике коју је могуће испитати компјутерским моделирањем (симулацијама). У овом раду је узета каротидна бифуркација у разматрање јер је она често место настанка болести [54], [55]. Бројна литература показује велики утицај смичућег напона на зиду на процес настанка и развоја атеросклерозе. Зоне артерија са ниским вредностима смичућег напона на зиду су погодне за настанак и развој атеросклерозе док изложеност веома високим вредностима смичућег напона на зиду може довести до дестабилизације већ постојећег плака [52], па је прорачун смичућег напона од великог значаја. Расподела смичућег напона на зиду може се израчунати компјутерским симулацијама базираним на нумеричким методама. Међутим, често је потребно резултате приказати у што краћем року, а компјутерске симулације некада могу трајати дуго. Алтернативно решење може бити базирано на техникама истраживања података. У овом раду је употребом техника истраживања података моделирана корелација између геометрије каротидне бифуркације са једне стране и вредности и положаја максималног смичућег напона на зиду артерије (енг. *Maximal Wall Shear Stress* - MWSS) са друге стране. Овом методологијом је могуће (са прихватљивом прецизношћу) предвиђати положај и вредност MWSS-а готово тренутно.

У овом раду су употребом техника истраживања података постигнути следећи резултати:

1) Аутоматско предвиђање максималне вредности смичућег напона на зиду за дату артерију, која се дефинише преко геометријских параметара. За решавање овог проблема пет различитих алгоритама је обучавано и тестирано.

2) Обезбеђено је објашњење кориснику о (I) знању које је модел стекао на основу базе података за учење и (II) разлогу сваког индивидуалног предвиђања. Овде је коришћена методологија за објашњење модела за предвиђање [6] како би се објаснило који геометријски фактори играју кључну улогу на повећање вредности MWSS-а.
3) Поред предвиђања вредности MWSS-а обезбеђена је и поузданост сваког индивидуалног предвиђања. Ово је посебно важно јер нам показује колико можемо веровати сваком индивидуалном предвиђању.
4) Аутоматско предвиђање позиције (координата) на каротидној бифуркацији где се MWSS појављује.

Резултати овог рада су публиковани у престижном међународном научном часопису [56].

4.2.2 Претходна истраживања

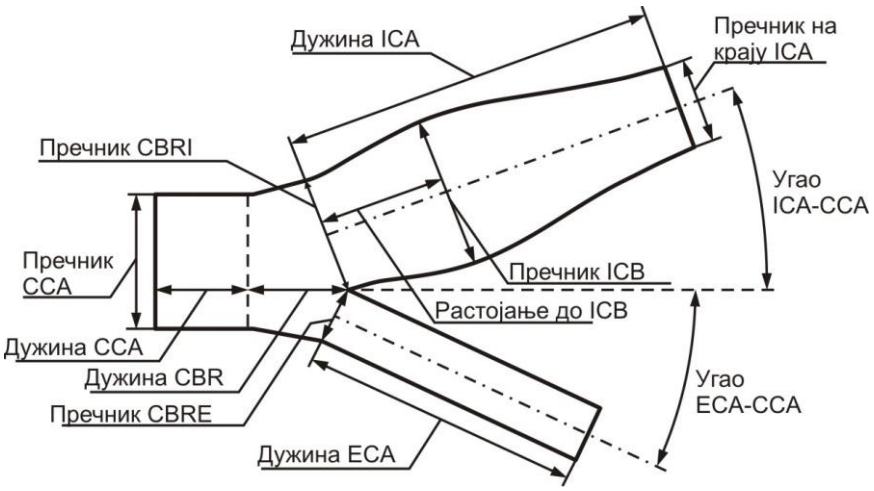
Повећање ризика од шлога може бити узроковано многим факторима: године, систолна и дијастолна хипертензија, дијабетес, пушење итд. Промене геометрије крвног суда у зони каротидне бифуркације утичу у великој мери на проток крви и могу довести до стенозе [57]. Показано је да се пречник крвног суда у зони каротидне бифуркације значајно мења са годинама [58]. Такође, геометрија каротиде се мења од пацијента до пацијента. Из поменутих разлога је тешко описати геометрију каротидне бифуркације употребом малог броја променљивих. Ипак, апроксимације геометрије су неопходне у ситуацији када је потребно резултате приказати у што краћем року. Основна идеја овог рада је креирање интелигентних модела који ће бити у стању да на основу одређеног броја геометријских параметара предвиде положај и вредност MWSS-а. На тај начин се постиже значајна уштеда времена у односу на компјутерске симулације.

Симулације пулзаторног струјања кроз каротидну артерију су извођене од стране Карла Перктолда (енг. Karl Perktold) [59]-[61]. Показано је да смичући напон на зиду игра веома значајну улогу у процесу настанка и прогресије атеросклерозе [63].

Вијаја Колахама (енг. Vijaya Kolachalama) је у свом раду користио Бајес-Гаусов процес (енг. *Bayesian Gaussian process*) како би пронашао везу између геометријских параметара и MWSS-а и како би пронашао геометрије које имају највећу и најмању вредност MWSS-а [64].

4.2.3 База података за модел каротидне бифуркације

Алгоритми техника истраживања података захтевају базу података на основу које ће „учити“ и касније бити тестирани. Упрошћен, геометријски параметризован модел каротидне бифуркације је приказан на слици 4-4. Геометрија каротидне бифуркације је дефинисана помоћу 12 геометријских параметара чије су средње вредности дате у табели 4-1. Ове средње вредности су усвојене на основу мерења из претходних истраживања [59]-[61].



Слика 4-4: Геометријски параметри каротидне бифуркације (преузето из [2]). Скраћенице: CCA - заједничка каротидна артерија (енг. *Common Carotid Artery*), CBR - регија каротидне бифуркације (енг. *Carotid Bifurcation Region*), CBRE - спољна регија каротидне бифуркације (енг. *Carotid Bifurcation Region External*), ECA - спољна каротидна артерија (енг. *External Carotid Artery*), ICA - унутрашња каротидна артерија (енг. *Internal Carotid Artery*), CBRI - унутрашња регија каротидне бифуркације (енг. *Carotid Bifurcation Region Internal*), ICB - унутрашњи каротидни булбус (енг. *Internal Carotid Bulbus*). Осе означавају координатни систем у коме се одређује положај MWSS-а.

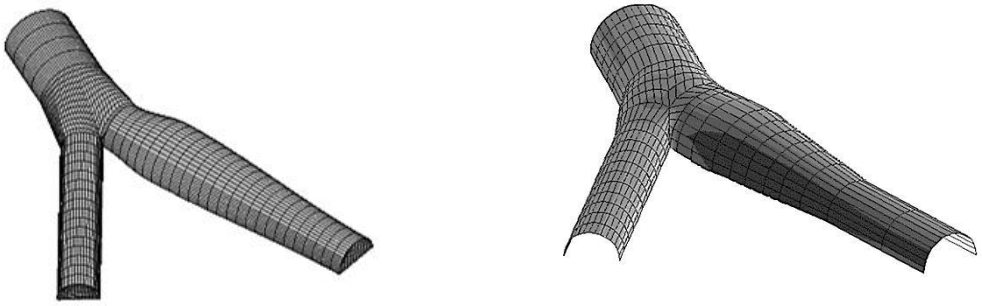
Геометријски параметри (слика 4-4) су коришћени за генерисање унутрашњег зида крвног суда, који представља просторно ограничење крвотока. Употребом ових параметара, креиран је тродимензионални модел коначних елемената (енг. *Finite Element (FE) model*) за домен крвотока (слика 4-5). Претпостављено је да бифуркација има раван симетрије, па је FE модел генерисан само за половину домена. Прорачун се ради само за ову половину, али резултати могу бити приказани за цео домен.

Геометријски параметар	**Средња вредност**	**Јединица**
Пречник CCA	6.2	[mm]
Дужина CCA	7.44	[mm]
Дужина CBR	7.316	[mm]
Пречник CBRI	4.9	[mm]
Пречник CBRE	3.658	[mm]
Угао ICA-CCA	25	[^{0}]
Угао ECA-CCA	25	[^{0}]
Дужина ICA	26.04	[mm]
Дужина ECA	18.6	[mm]
Растојање до ICB	5.39	[mm]
Пречник ICB	6.5	[mm]
Пречник на крају ICA	4.34	[mm]

Табела 4-1: Средње вредности геометријских параметара каротидне бифуркације.

FE модел каротидне бифуркације је креиран употребом тродимензионалних 8-чворних изопараметарских елемената (са израчунавањем брзине у свим чворовима и притиска на нивоу елемента). Постпроцесирање резултата прорачуна нам даје увид у локалну хемодинамику, као и утицај крви на зид крвног суда, као што је дистрибуција притиска и смичућег напона на зиду крвног суда.

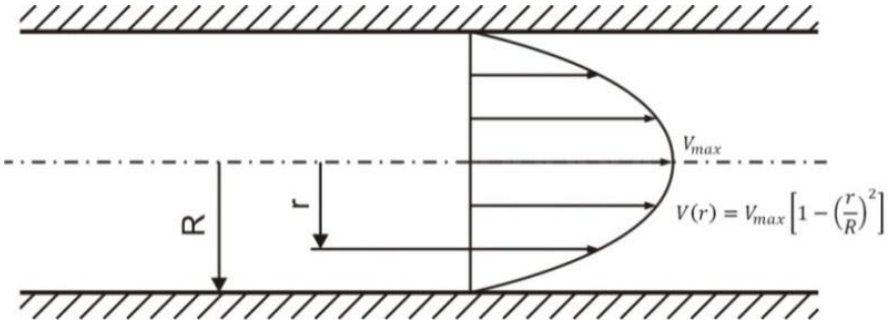
а) б)



Слика 4-5: FE модел каротидне бифуркације. а) домен флуида, б) домен солида.

За обучавање и тестирање интелигентних модела, креирано је 1886 различитих FE модела каротидне бифуркације. За све ове моделе су одрађене симулације стационарног струјања и израчунати су положај (у виду координата) и вредност максималног смичућег напона на зиду. На рачунару са INTEL CORE 2 DUO E7500 процесором на 2.93GHz и са 4GB рама, симулације трају између 10 и 25 минута (у зависности од броја итерација). Сви модели имају тачно 13649 чворова. Приликом креирања ових модела вредности геометријских параметара су вариране у опсегу $\pm 30\%$ у односу на средње вредности дате у табели 4-1. На овај начин је дозвољена већа варијабилност геометрије артерије у односу на Колахамин рад у коме је дозвољена варијација од $\pm 25\%$.

Параболоидни профил брзине је коришћен на улазном попречном пресеку заједничке каротидне артерије (слика 4-6). Максимална брзина (V_{max}) у овом попречном пресеку је 93.8 mm/s.

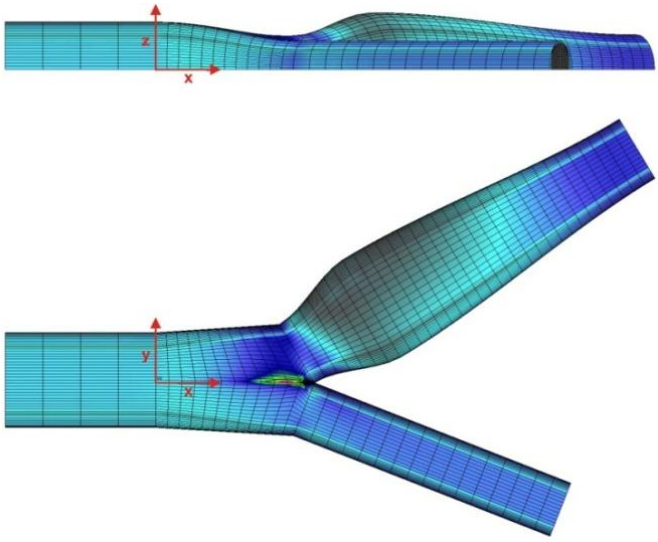


Слика 4-6: Параболоидни профил брзине (уздужни пресек).

Све компоненте брзине на фиксном зиду су постављене на нулу. Такође, компоненте брзине у равни симетрије у правцу управном на раван симетрије су постављене на нулу. Претпоставља се да је струјање стационарно, да је крв Њутнов нестишљив флуид и да су зидови крути. Густина крви је $\rho = 1.05[g/cm^3]$, а динамичка вискозност је $\mu = 0.0367[P]$. Сви модели каротидне бифуркације су креирани уношењем вредности геометријских параметара у софтвер CAD[2], а нумеричка израчунавања су извршена применом програма ПАК [62].

[2] CAD - Софтверски алат за генерисање мрежа, тродимензионалну визуализацију и анализу методом коначних елемената, развијен у истраживалко развојном центру за биоинжењеринг - БиоИРЦ.

Дакле, у циљу демонстрирања применљивости алгоритама техника истраживања података за моделирање везе између геометрије каротидне бифуркације са једне стране и вредности и положаја MWSS-а са друге стране креирана је база која садржи 1886 примера за обучавање. Ова база садржи 12 улаза (геометријских параметара) и 3 излаза (MWSS вредност, X и Y координату положаја MWSS-а). Предвиђање Z координате положаја MWSS-а није неопходно јер се она налази у пресеку нормале на раван симерије и зида модела (слика 4-7).



Слика 4-7: Координатни систем и расподела смичућег напона на зиду (пример бр. 3).

4.2.4 Избор и тестирање модела за предвиђање

Како би се демонстрирала применљивост алгоритама техника истраживања података за моделирање везе између геометрије каротидне бифуркације са једне стране и вредности и положаја MWSS-а са друге стране обучавани су и тестирани следећи модели за предвиђање:

1. Алгоритам k најближих суседа - KNN,
2. Линеарна регресија - LR,
3. Неуронска мрежа - вишеслојни перцептрон - MLP,
4. Алгоритам случајне шуме - RF,
5. Метода потпорних вектора - SVM.

Модели за предвиђање су тестирани применом методе изостављања једног примера (LOOCV) и израчунавањем њихове релативне средње квадратне грешке (RMSE). Ова грешка представља однос укупне квадратне грешке модела за предвиђање и укупне квадратне грешке неинтелигентног модела (модела који увек предвиђа средњу вредност излаза):

$$RMSE = \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \tag{4.3}$$

где је N укупан број примера (1886), y_i је стварни излаз i-тог примера, $\bar{y}$ је средња вредност излаза, а $\hat{y}_i$ је предвиђена вредност регресионог модела за i-ти пример.

Дакле, применом LOOCV, у свакој итерацији тестирања регресиони модел се обучава са 1885 примера, а тестира на једном изостављеном. Овај поступак се понавља 1886 пута, све док сви примери по једном не буду изостављени из скупа за обучавање. Вредност RMSE мања од 1.0 показује да је модел употребљив (има мању грешку од неинтелигентног модела) [32].

4.2.5 Предвиђање највеће вредности смичућег напона на зиду

4.2.5.1 Резултати тестирања модела за предвиђање

RMSE вредности за тестиране моделе су приказане у табели 4-2. Резултати тестирања показују да су сви модели употребљиви (RMSE<1), али да је најбољи резултат постигнут употребом неуронске мреже (RMSE=0.140).

Пошто је неуронска мрежа дала више него 4 пута бољи резултат од следећег најбољег модела (SVM, RMSE=0.612) у наредном делу ћемо се фокусирати на овај алгоритам. Употребом методологије за објашњење модела за предвиђање и индивидуалних предвиђања [6] биће показано који геометријски фактори играју кључну улогу у предвиђању вредности MWSS-а. Такође, биће приказана и поузданост индивидуалних предвиђања, што је посебно важно јер нам показује колико можемо веровати сваком индивидуалном предвиђању.

Модел	Опис модела	RMSE
KNN	Излаз се израчунава усредњавањем вредности излаза 19 најближих примера - суседа (k=19).	0.759
LR	Излаз се израчунава помоћу линеарне функције која се оптимизује употребом примера за обучавање.	0.748
MLP	Вишеслојни перцептрон са 5 неурона у једном скривеном слоју, биполарним сигмоидалним активационим функцијама у неуронима скривеног слоја и линеарном активационом функцијом у излазном слоју. Неуронска мрежа је обучена алгоритмом са пропагацијом грешке уназад. Критеријуми за заустављање учења су дефинисани као 2000 епоха учења и да промена средње квадратне грешке између 2 епохе буде мања од 10^{-6}.	0.140
RF	Излаз се израчунава усредњавањем вредности предвиђања 500 стабала ($N_{trees} = 500$). Број атрибута који се насумично бира за сваки чвор је 4 ($m_{try} = \frac{a}{3} = 4$).	0.628
SVM	Коришћен је кернелов трик како би се примери пресликали у простор F (енг. *Feature space*). Употребљена је радијална функција (енг. *Radial Basis Function* - RBF) са параметрима: $\gamma = 1/a$ (a - број атрибута), $C = 1$ и употребом SMO алгоритма оптимизације [65].	0.612

Табела 4-2: Опис и RMSE модела за предвиђање MWSS-а.

4.2.5.2 Објашњење модела за предвиђање

У циљу анализе најпрецизнијег модела за предвиђање (неуронска мрежа) и разјашњавања значајности појединачних атрибута у поступку предвиђања MWSS-а, примењена је методологија за објашњење модела за предвиђање (енг. *Model explanation methodology*) [6]. Ова методологија је независна од самог модела и омогућава разумевање логике модела чак и у ситуацијама када модел сам по себи није репрезентативан (као нпр. неуронска мрежа). Разумевање понашања модела је од великог значаја јер пружа експертима увид у вредности параметара које могу довести до развоја болести.

Примена методологије за објашњење модела за предвиђање резултује квантитативним описом утицаја атрибута и њихових индивидуалних вредности на излаз (MWSS). Другим речима, ова методологија показује колико у просеку на излаз утиче свака вредност посматраног атрибута. За израчунавање утицаја (значаја) i-те вредности j-тог атрибута ($\psi_{i,j}$) користи се алгоритам приказан на слици 4-8.

$\psi_{i,j} = 0$ // $\psi_{i,j}$ – глобални значај j-те вредности i-тог атрибута

За $r = 1$ до k // k – параметар који задаје корисник

случајним избором селектовати пример R

креирати нови пример:

$R_1 \leftarrow$ поставити i-ти атрибут на вредност j, остале атрибуте узети из R

$\psi_{i,j} = \psi_{i,j} + f(R) - f(R_1)$ // $f(R)$, $f(R_1)$ – излази модела за предвиђање за // улазне векторе примера R и R_1

Kraj { r }

$$\psi_{i,j} = \frac{\psi_{i,j}}{k}$$

Слика 4-8: Објашњење модела за предвиђање: алгоритам за израчунавање глобалног значаја j-те вредности i-тог атрибута $\psi_{i,j}$.

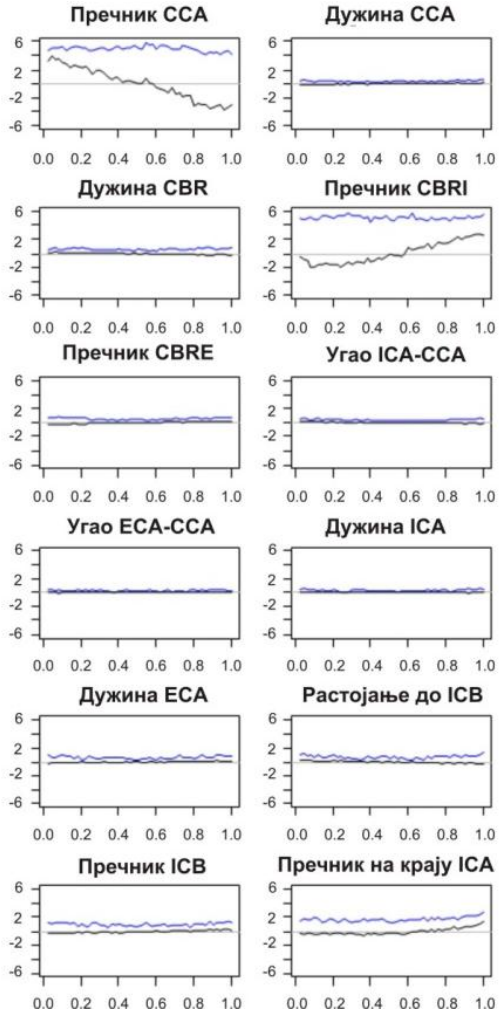
Применом алгоритма са слике 4-8 израчунато је колико у просеку различите вредности појединачних атрибута утичу на предвиђање излаза неуронском мрежом. На овај начин је креирано 12 дијаграма (за сваки геометријски параметар по један), који су приказани на слици 4-9.

На слици 4-9 доприноси различитих атрибута су приказани на различитим дијаграмима. Допринос различитих вредности атрибута (црна линија на дијаграмима) је дат у облику интензитета и знака. Позитиван допринос показује да та вредност атрибута повећава вредност предвиђеног MWSS-а. Аналогно томе, негативан допринос показује да та вредност атрибута смањује вредност предвиђеног MWSS-а. Коначно, нулти допринос указује да посматрана вредност атрибута у просеку нема утицаја на предвиђање излаза. Као додатак доприносима појединих вредности атрибута, на дијаграмима је приказана и стандардна девијација појединачних доприноса (плава

линија). Што је стандардна девијација већа (што је плава линија више) то је атрибут значајнији [6], [66].

Посматрањем дијаграма на слици 4-9, може се закључити да атрибути *Пречник CCA* и *Пречник CBRI* у највећој мери утичу на предвиђање излаза (високе плаве криве). Значајан допринос се може приметити и на дијаграмима атрибута *Пречник ICB* и *Пречник на крају ICA*. Остали атрибути имају безначајан утицај на предвиђање MWSS-а.

Дакле од укупно пет атрибута који представљају пречнике појединих делова каротидне бифуркације, четири су одабрана као значајна методологијом за објашњење модела за предвиђање. Ово је логично јер на вредност смичућег напона на зиду у великој мери утиче пречник крвног суда кроз који крв протиче.



Слика 4-9: Дијаграми глобалног значаја атрибута коришћених за предвиђање вредности MWSS-а. Хоризонтална оса сваког дијаграма представља вредности појединачних атрибута (вредности су скалиране на опсег [0,1]). Вертикална оса означава просечан допринос различитих вредности атрибута на предвиђање MWSS-а. Црном линијом је представљен допринос, а плавом стандардна девијација доприноса.

4.2.5.3 Објашњење индивидуалних предвиђања

Као додатак објашњењу комплетног модела, слична методологија може бити примењена за објашњење индивидуалних предвиђања [67]. Методологија објашњења

индивидуалних предвиђања додељује свакој вредности атрибута допринос, који може бити позитиван, негативан или нула. Позитиван допринос показује да вредност атрибута утиче на повећање вредности предвиђања MWSS-а, негативан допринос показује да вредност атрибута утиче на смањење вредности предвиђања MWSS-а, и нулти допринос показује да вредност атрибута нема утицаја на предвиђање MWSS-а. Алгоритам за израчунавање доприноса вредности атрибута у процесу предвиђања MWSS-а дат је на слици 4-10.

$\varphi_q(i) = 0$ // $\varphi_q(i)$ - значај вредности i-тог атрибута примера q за модел f

За $r = 1$ до k // k - параметар који задаје корисник

случајним избором селектовати пример R

креирати нове примере q_1 и q_2 чији су улазни вектори:

$$\mathbf{x_1} = \{x_q(1), x_q(2), \dots, x_q(i-1), x_q(i), x_R(i+1), \dots, x_R(a)\}$$

$$\mathbf{x_2} = \{x_q(1), x_q(2), \dots, x_q(i-1), x_R(i), x_R(i+1), \dots, x_R(a)\}$$

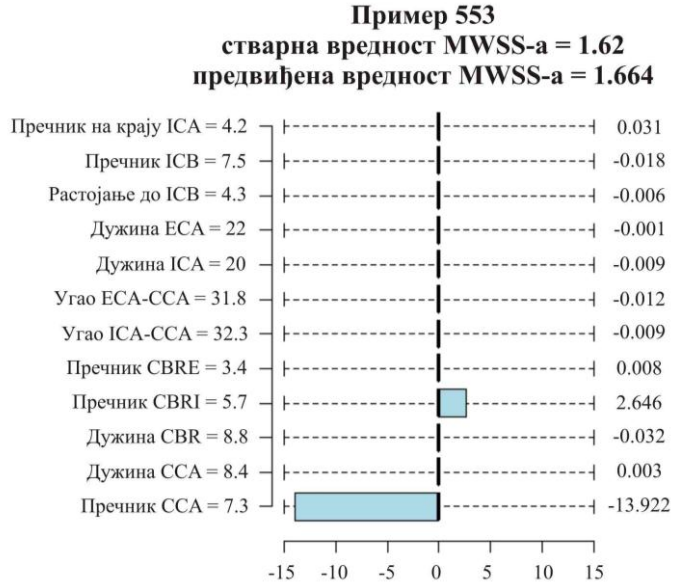
$\varphi_q(i) = \varphi_q(i) + f(\mathbf{x_1}) - f(\mathbf{x_2})$ // $f(\mathbf{x_1}), f(\mathbf{x_2})$ - излази модела за предвиђање // за улазе $\mathbf{x_1}$ и $\mathbf{x_2}$

Kraj { r }

$$\varphi_q(i) = \frac{\varphi_q(i)}{k}$$

Слика 4-10: Објашњење индивидуалних предвиђања: алгоритам за израчунавање значаја вредности i-тог атрибута примера q за модел f.

На слици 4-11, графички је приказано објашњење индивидуалног предвиђања за пример #553. На овој слици се јасно види колико вредности атрибута овог примера утичу на предвиђање MWSS-а неуронском мрежом.



Слика 4-11: Визуализација доприноса вредности атрибута у предвиђању вредности MWSS-а за пример #553. На левој страни су имена атрибута и њихове вредности за пример #553. Десно се налазе доприноси (значаји) вредности сваког атрибута у процесу предвиђања MWSS-а неуронском мрежом.

Посматрајући вредности доприноса (на десној страни слике) можемо лако закључити да атрибути *Пречник CCA* и *Пречник CBRI* највише утичу на предвиђену вредност MWSS-а (1.664). Вредност атрибута *Пречник CBRI* (5.7) утицала је на повећање вредности предвиђеног MWSS-а. Са друге стране, вредност атрибута *Пречник CCA* (7.3) утицала је на смањење вредности предвиђеног MWSS-а. Доприноси осталих атрибута су безначајни, што је у складу са објашњењем комплетног модела за предвиђање (слика 4-9).

Објашњење индивидуалних предвиђања је од великог значаја јер корисницима пружа јасан увид у разлоге предвиђене вредности излаза.

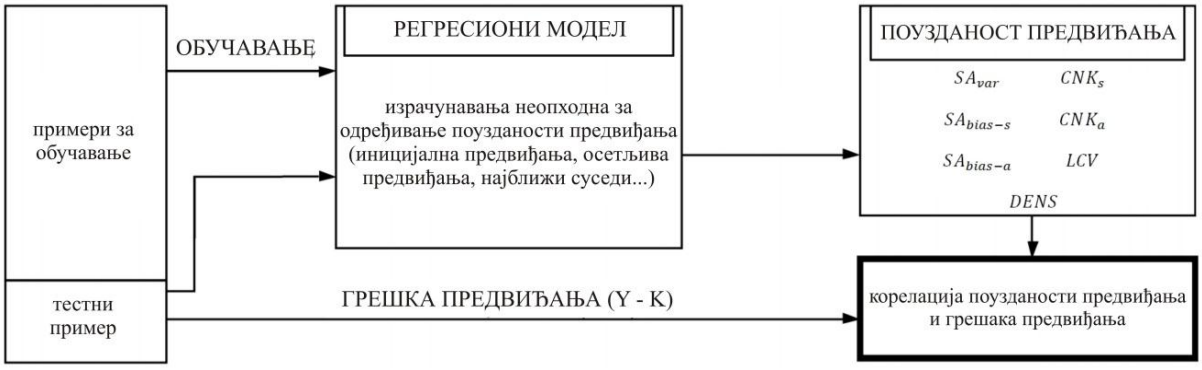
4.2.5.4 Поузданост предвиђања

До сада је израчуната прецизност неуронске мреже (RMSE), дато објашњење модела за предвиђање и дато објашњење индивидуалних предвиђања. Овде се апликација проширује тако што се поред предвиђања неуронске мреже додаје и информација о поузданости предвиђања.

У поглављу 3.6 је описано седам различитих мера поузднсти предвиђања. Како би се проверила могућност употребе мера поузданости предвиђања за проблем предвиђања MWSS-а коришћена је неуронска мрежа као најпрецизнији модел. Поузданости предвиђања и грешке предвиђања су израчунате методом изостављања једног примера – LOOCV (слика 4-12). Како би се тестирала могућност употребе мера поузданости предвиђања израчунате су вредности Пирсоновог коефицијента корелације између мера поузданости предвиђања и грешака предвиђања. Статистичка значајност Пирсонових коефицијената корлације је израчуната на следећи начин:

$$t = \frac{C}{\sqrt{\frac{1 - C^2}{N - 2}}} \tag{4.4}$$

где је C Пирсонов коефицијент корлације, а N укупан број примера. Израчунавањем t вредности и броја степени слободе $\mathrm{df} = \mathrm{N} - 2$ можемо одредити статистичку значајност коефицијента корелације.



Слика 4-12: Поступак одређивања поузданости предвиђања и њихова корелација са грешкама предвиђања (за случај LOOCV).

Вредности коефицијената корелације за 7 мера поузданости предвиђања као и њихова статистичка значајност су приказане у табели 4-3.

Поузданост предвиђања	Коефицијент корелације	Ниво значајности
SA_{bias-s}	0.203	<0.001
SA_{var}	0.144	<0.001
$DENS$	0.140	<0.001
SA_{bias-a}	0.130	<0.001
LCV	0.129	<0.001
CNK_s	0.126	<0.001
CNK_a	0.107	<0.001

Табела 4-3: Корелациони коефицијенти поузданости предвиђања и њихов ниво значајности.

Резултати приказани у табли 4-3 показују да се апликација предвиђања MWSS-а може успешно проширити додавањем поузданости предвиђања. На овај начин ће корисницима поред тражене вредности MWSS-а бити пружена и поузданост предвиђања чиме ће добити информацију о поузданости тј. у којој мери се могу ослонити на предвиђену вредност.

4.2.6 Предвиђање положаја највеће вредности смичућег напона на зиду у артерији

У циљу одређивања положаја MWSS-а на артерији, креирани су различити модели за предвиђање његових координата. За предвиђање координата (X и Y) коришћени су исти алгоритми као и за предвиђање вредности MWSS-а. Такође, користе се исти улазни геометријски параметари (слика 4-4). Координате положаја MWSS-а примера за обучавање су добијене из истих симулација заједно са вредностима MWSS-а. С обзиром на чињеницу да вредности X и Y координата у потпуности одређују вредност Z координате (Z координата се налази у пресеку нормале на раван симерије (X-Y раван) и зида модела - слика 4-7), модели за предвиђање предвиђају само координате X и Y.

Табела 4-4 приказује RMSE вредности 5 различитих модела за предвиђање координата X и Y. Резултати тестирања показују да су сви модели употребљиви (RMSE<1). Најбољи резултат је, као и у случају предвиђања вредности MWSS-а, постигнут употребом неуронске мреже.

Модел	Опис модела	RMSE за X	RMSE за Y
KNN	Излаз се израчунава усредњавањем вредности излаза 13 најближих примера-суседа (k=13) за предвиђање координате X и 19 најближих примера-суседа (k=19) за предвиђање координате Y. Коришћен је KNN са тежинским коефицијентима (3.37).	0.150	0.209
LR	Излаз се израчунава помоћу линеарне функције која се оптимизује употребом примера за обучавање.	0.044	0.032
MLP	Вишеслојни перцептрон са 5 неурона у једном скривеном слоју, биполарним сигмоидалним активационим функцијама у неуронима скривеног слоја и линеарном активационом функцијом у излазном слоју. Неуронска мрежа је обучена алгоритмом са пропагацијом грешке уназад. Критеријуми за заустављање учења су дефинисани као 2000 епоха учења и да промена средње квадратне грешке између две епохе буде мања од 10^{-6}.	0.029	0.018
RF	Излаз се израчунава усредњавањем вредности предвиђања 500 стабала ($N_{trees} = 500$). Број атрибута који се насумично бира за сваки чвор је 4 ($m_{try} = \frac{a}{3} = 4$).	0.046	0.066
SVM	Коришћен је кернелов трик како би се примери пресликали у вишедимензионални простор F. Употребљена је радијална функција (енг. *Radial Basis Function* - RBF) са параметрима: $\gamma = 1/a$ (a – број атрибута), $C = 1$ и употребом SMO алгоритма оптимизације [65].	0.047	0.025

Табела 4-4: Опис и RMSE модела за предвиђање координата X и Y.

4.2.7 Оптимизација постигнутих резултата конструисањем нових атрибута

Геометрија каротидне бифуркације је дефинисана помоћу 12 геометријских параметара приказаних на слици 4-4. Употребом ових параметара успешно је могуће предвиђати вредност и положај MWSS-а (табеле 4-2 и 4-4). Међутим, на вредност и положај MWSS-а могу у великој мери утицати односи пречника појединих сегмената бифуркације. Наиме, у случају смањења пречника по дужини артерије долази до повећања вредности MWSS-а и обрнуто. Из тог разлога је креирано пет нових атрибута, који представљају односе пречника појединих сегмената бифуркације:

- (Пречник CCA) / (Пречник CBRI)
- (Пречник CCA) / (Пречник CBRE)
- (Пречник ICB) / (Пречник CBRI)
- (Пречник ICB) / (Пречник на крају ICA)
- (Пречник CBRI) / (Пречник CBRE)

Након креирања пет нових атрибута, поново су трениране неуронске мреже за предвиђање вредности и положаја MWSS-а. Вишеслојни перцептрон сада има 17 неурона у улазном слоју (12 старих + 5 нових атрибута), 5 неурона у једном скривеном слоју, биполарне сигмоидалне активационе функције у неуронима скривеног слоја и линеарну активациону функцију у излазном слоју. Неуронска мрежа је обучена алгоритмом са пропагацијом уназад. Критеријуми за заустављање учења су дефинисани као 2000 епоха учења и да промена средње квадратне грешке између 2 епохе буде мања од 10^{-6}. Резултати тестирања показују да је увођењем пет нових атрибута постигнут напредак у односу на претходне резултате (мање вредности RMSE). У табели 4-5 су приказани упоредни резултати тестирања неуронске мреже употребом основног и проширеног скупа атрибута.

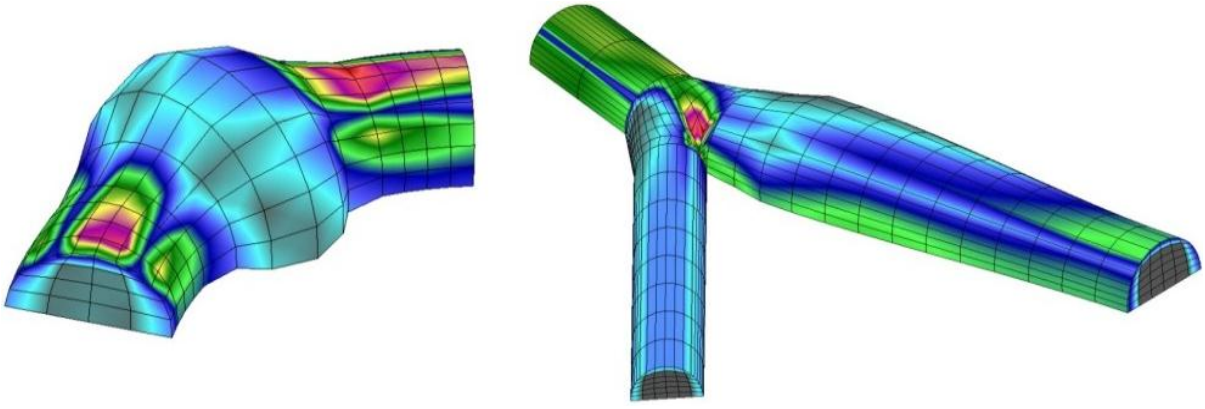
Проблем	RMSE основног скупа атрибута	RMSE проширеног скупа атрибута
Предвиђање MWSS-а	0.140	0.097
Предвиђање координате X	0.029	0.028
Предвиђање координате Y	0.018	0.016

Табела 4-5: RMSE неуронске мреже употребом основног и проширеног скупа атрибута.

4.3. Предвиђање комплетне расподеле смичућег напона на зиду за моделе анеуризме и каротидне бифуркације

4.3.1 Опис проблема

У поглављу 4.2 је описан поступак предвиђања вредности и положаја највећег смичућег напона на зиду (MWSS) за модел каротидне бифуркације. Овде се одлази корак даље и праве се модели који ће бити у стању да предвиђају комплетну расподелу смичућег напона на зиду, и то за моделе каротидне бифуркације и анеуризме (слика 4-13).



Слика 4-13: Расподела смичућег напона на зиду за модел анеуризме (лево) и модел каротидне бифуркације (десно).

Пуцање анеуризме може довести до озбиљног крварења, других компликација па и смрти. Показано је да се раст анеуризме јавља у зонама ниског смичућег напона [68].

Одређивање расподеле смичућег напона на зиду је од великог значаја како би се вршила процена ризика настанка и евентуалног каснијег пуцања анеуризме.

Главни мотив овог рада је чињеница да предвиђање расподеле смичућег напона на зиду употребом техника истраживања података доводи до значајне уштеде времена у односу на компјутерске симулације. У овом поглављу је описан поступак предвиђања комплетне расподеле смичућег напона на зиду за моделе анеуризме и каротидне бифуркације на основу њихове геометрије и густине, вискозности и брзине крви која кроз њих протиче. Резултати овог рада су презентовани на међународној научној конференцији у Бостону 2011. године [69].

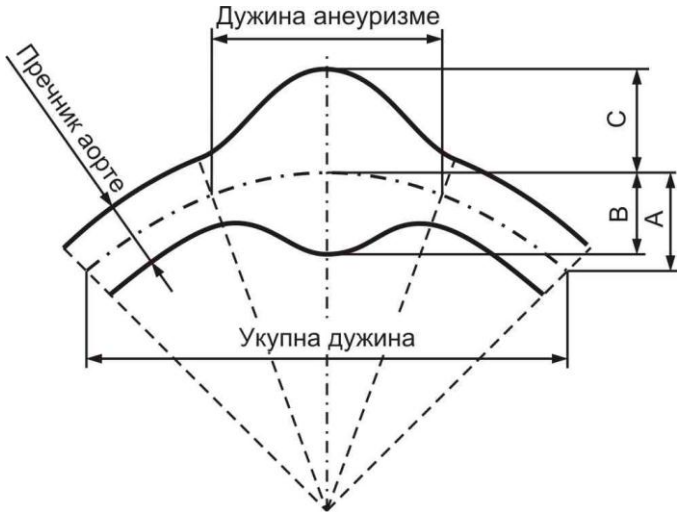
4.3.2 База података за моделе анеуризме и каротидне бифуркације

У циљу демонстрирања способности техника истраживања података за моделирање везе између геометријских параметара, густине крви, динамичке вискозности и брзине струјања са једне стране и расподеле смичућег напона са друге стране, креиране су две базе података (по једна за анеуризму и каротидну бифуркацију) које садрже резултате 4779 симулација. Дакле, креирано је по 4779 различитих модела анеуризме и каротидне бифуркације насумичном варијацијом улазних параметара у опсегу $\pm 30\%$ у односу на средње вредности дате у табелама 4-6 и 4-7.

Улазни параметар	**Средња вредност**	**Јединица**
Пречник CCA	6.2	[mm]
Дужина CCA	7.44	[mm]
Дужина CBR	7.316	[mm]
Пречник CBRI	4.9	[mm]
Пречник CBRE	3.658	[mm]
Угао ICA-CCA	25	[^{0}]
Угао ECA-CCA	25	[^{0}]
Дужина ICA	26.04	[mm]
Дужина ECA	18.6	[mm]
Растојање до ICB	5.39	[mm]
Пречник ICB	6.5	[mm]
Пречник на крају ICA	4.34	[mm]
Густина	0.00105	[gr/mm]
Динамичка вискозност	0.00367	[Pa·s]
Брзина	233	[mm/s]

Табела 4-6: Средње вредности улазних параметара каротидне бифуркације.

Геометријски параметри који дефинишу модел анеуризме приказани су на слици 4-14 (геометријски параметри каротидне бифуркације су дати на слици 4-4).



Слика 4-14: Геометријски параметри модела анеуризме (преузето из [2]). „Укупна дужина“ је параметар који дефинише дужину пројекције генерисаног модела анеуризме, „A“ је висина лука централне линије, „Пречник аорте“ је абдоминални пречник аорте, „B“ је радијус између централне линије и унутрашњег зида анеуризме, „C“ је радијус између централне линије и спољашњег зида анеуризме, „Дужина анеуризме“ је параметар који дефинише дужину анеуризме по централној линији.

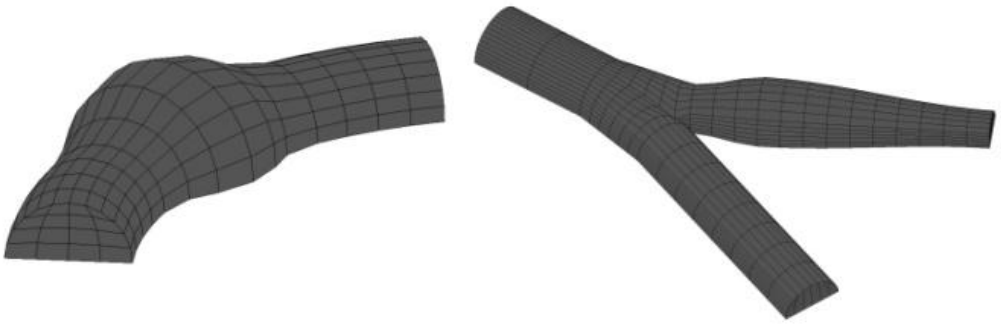
Улазни параметар	**Средња вредност**	**Јединица**
Укупна дужина	100	[mm]
Дужина анеуризме	40	[mm]
A	30	[mm]
Пречник аорте	20	[mm]
C	20	[mm]
B	20	[mm]
Густина	0.00105	[gr/mm]
Динамичка вискозност	0.00367	[Pa·s]
Брзина	28.13	[mm/s]

Табела 4-7: Средње вредности улазних параметара анеуризме.

FE модели анеуризме и каротидне бифуркације који се овде користе садрже мали број чворова. Разлог томе је, са једне стране чињеница да 9558 симулација (4779 анеуризми и 4779 каротидних бифуркација) захтева много времена у случају великог броја чворова по моделу. Са друге стране, проблем који решавамо је проблем предвиђања великог броја излаза (енг. *Multi-target prediction problem*) па зато обучавамо и тестирамо по један модел за предвиђање (неуронску мрежу) за сваки чвор на омотачу, што изискује много времена. Сврха овог рада је показивање могућности предвиђања расподеле смичућег напона на зиду за моделе анеуризме и каротидне бифуркације. Међутим, за евентуалну експлоатацију би било неопходно креирати финије моделе који садрже већи број чворова.

FE модел анеуризме се састоји од 375 чворова од којих 195 леже на омотачу. Са друге стране FE модел каротидне бифуркације садржи 1854 чворова од којих 642 леже на омотачу. Употребом компјутерских симулација израчунате су вредности смичућег напона у свим чворовима омотача за свих 4779 различитих геометрија (и за анеуризму и за каротидну бифуркацију), и резултати су сачувани у јединствене базе које се даље

користе за обучавање и тестирање модела за предвиђање. FE модели анеуризме и каротидне бифуркације су приказани на слици 4-15.



Слика 4-15: FE модели анеуризме (лево) и каротидне бифуркације (десно).

4.3.3 Модели за предвиђање

За предвиђање расподеле смичућег напона на зиду за моделе анеуризме и каротидне бифуркације обучавани су и тестирани следећи модели за предвиђање:

1. Алгоритам k најближих суседа - KNN,
2. Неуронска мрежа - вишеслојни перцептрон (обучен алгоритмом са пропагацијом грешке уназад са моментом и варијабилном брзином учења) - MLP.

Алгоритам KNN је коришћен тако што се пронађе k најближих суседа, а затим се вредности одговарајућих чворова усредње како би се израчунао излаз:

$$\hat{y}_j(p) = \frac{1}{k}\sum_{i=1}^{k} y_i(p) \tag{4.5}$$

где је $y_i(p)$ вредност смичућег напона p-тог чвора i-тог најближег суседа, $\hat{y}_j(p)$ је предвиђена вредност смичућег напона p-тог чвора за тестни пример j, а k је број најближих суседа.

Традиционални облик алгоритма са пропагацијом грешке уназад (3.71)-(3.72) има проблема са локалним минимумима и са спором конвергенцијом. Како би се превазишли ови проблеми развијене су бројне варијације овог алгоритма [41]. За предвиђање расподеле смичућег напона на зиду за моделе анеуризме и каротидне бифуркације коришћен је вишеслојни перцептрон обучен алгоритмом са пропагацијом грешке уназад са моментом и варијабилном брзином учења. Адаптација тежинских коефицијената се врши на следећи начин (исто важи и за прагове активација $b_{i(l)}$):

$$\begin{aligned} w_{i,j(l)}(t+1) &= w_{i,j(l)}(t) + \Delta w_{i,j(l)}(t+1) \\ \Delta w_{i,j(l)}(t+1) &= \eta\alpha\frac{\partial E_k}{\partial w_{i,j(l)}} + \alpha\Delta w_{i,j(l)}(t) \end{aligned} \tag{4.6}$$

где је $w_{i,j(l)}$ тежина везе између i-тог неурона у l-том слоју и j-тог неурона у (l-1)-ом слоју, η је брзина учења, α је константа момента, E_k је критеријумска функција која

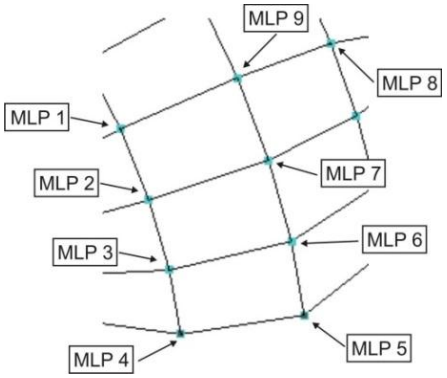
описује колико се стварни излаз мреже разликује од жељеног (3.68), а $\Delta w_{i,j(l)}(t)$ је претходна промена (из претходне итерације) тежине везе $w_{i,j(l)}$.

Брзина учења у једначини (4.6) је варијабилна. У свакој епохи, ако се критеријумска функција смањује ка циљу, брзина учења се повећава за фактор η_{inc}. Ако се критеријумска функција повећава више него што max_{inc} фактор дозвољава, брзина учења се смањује према фактору η_{dec}, а промена која је довела до повећања критеријумске функције се одбацује. Уобичајне вредности параметара α, η_{inc}, η_{dec} и max_{inc} су приказане у табели 4-8.

$\boldsymbol{\alpha}$	$\boldsymbol{\eta_{inc}}$	$\boldsymbol{\eta_{dec}}$	$\boldsymbol{max_{inc}}$
0.9	1.05	0.7	1.04

Табела 4-8: Вредности параметара алгоритма са поропагацијом грешке уназад са моментом и варијабилном брзином учења.

Проблем који решавамо је проблем предвиђања великог броја излаза (енг. *Multi-target prediction problem*). У случају анеуризме потребно је предвиђати вредности смичућег напона на зиду за 195 чворова (који леже на омотачу). У случају каротидне бифуркације потребно је предвиђати вредности смичућег напона на зиду за 642 површинска чвора. То значи да би било потребно креирати неуронску мрежу са 195 излаза у случају анеуризме, односно 642 излаза у случају каротидне бифуркације. Уместо тога, креира се по једна неуронска мрежа за сваки чвор који лежи на омотачу (слика 4-16). Свака од ових неуронских мрежа се засебно обучава и врши предвиђање вредности смичућег напона само за посматрани чвор.



Слика 4-16: Чворови на омотачу и њима одговарајуће неуронске мреже.

4.3.4 Резултати тестирања модела за предвиђање

У овом раду је креирано 4779 модела анеуризме и каротидне бифуркације насумичном варијацијом улазних параметара у опсегу $\pm$30% у односу на средње вредности дате у табелама 4-6 и 4-7. 70% ових примера (3346 примера) је коришћено за обучавање, а преосталих 30% (1433 примера) за тестирање. Модели за предвиђање су тестирани израчунавањем релативне средње квадратне грешке (RMSE).

Најпре се израчунавају квадратне грешке свих тестних примера:

$$SE_i = \sum_{j=1}^{N_{surf}} \left(y_i(j) - \hat{y}_i(j)\right)^2 \tag{4.7}$$

где је SE_i квадратна грешка i-тог тестног примера, N_{surf} број чворова на омотачу, $y_i(j)$ стварна вредност смичућег напона j-тог чвора i-тог тестног примера, а $\hat{y}_i(j)$ предвиђена вредност смичућег напона j-тог чвора i-тог тестног примера.

На исти начин се израчунавају квадратне грешке неинтелигентног модела (модела који увек предвиђа средњу вредност излаза):

$$\overline{SE_i} = \sum_{j=1}^{N_{surf}} \left(y_i(j) - \bar{y}(j)\right)^2 \tag{4.8}$$

где је $\bar{y}(j)$ средња вредност смичућег напона на зиду за j-ти чвор свих примера за обучавање:

$$\bar{y}(j) = \frac{1}{N_{train}} \sum_{i=1}^{N_{train}} y_i(j) \tag{4.9}$$

где је N_{train} број примера за обучавање (3346).

Коначно RMSE се израчунава на следећи начин:

$$RMSE = \frac{\sum_{i=1}^{N_{test}} SE_i}{\sum_{i=1}^{N_{test}} \overline{SE_i}} \tag{4.10}$$

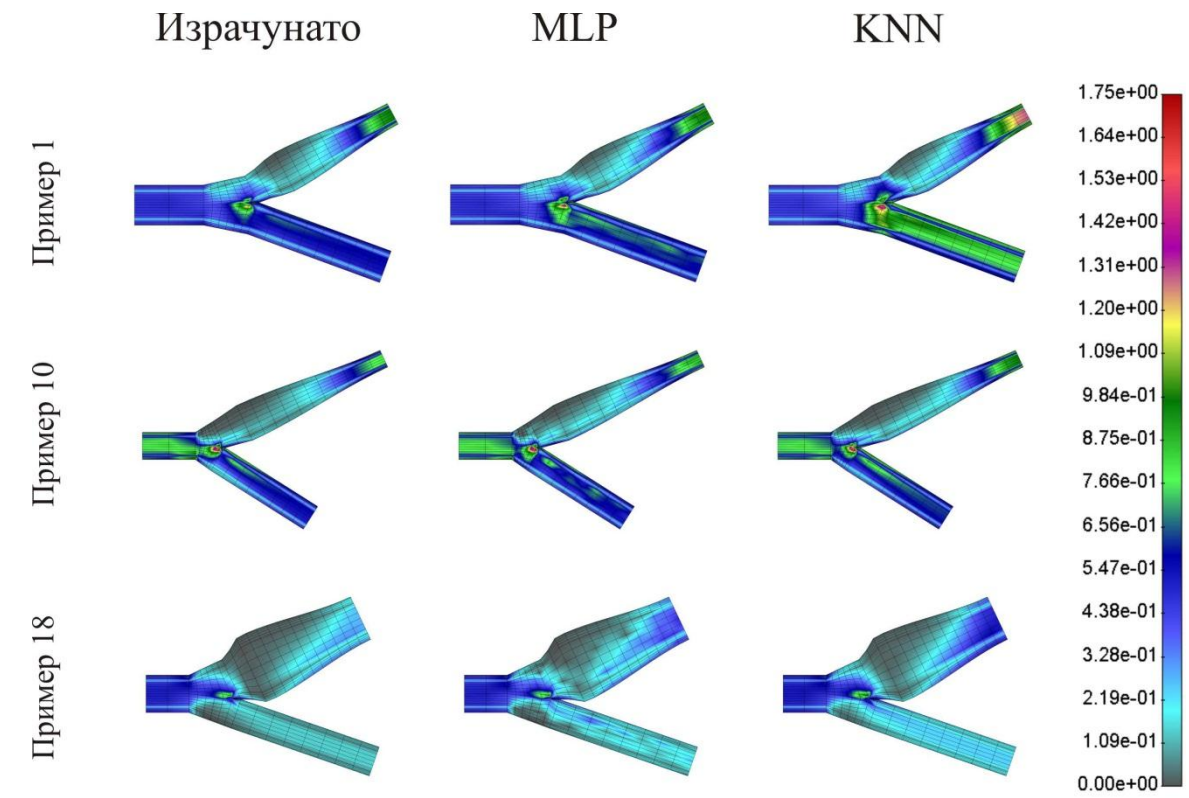
где је N_{test} број примера за тестирање (1433).

Резултати тестирања (RMSE вредности) KNN и MLP алгоритма су приказане у табели 4-9.

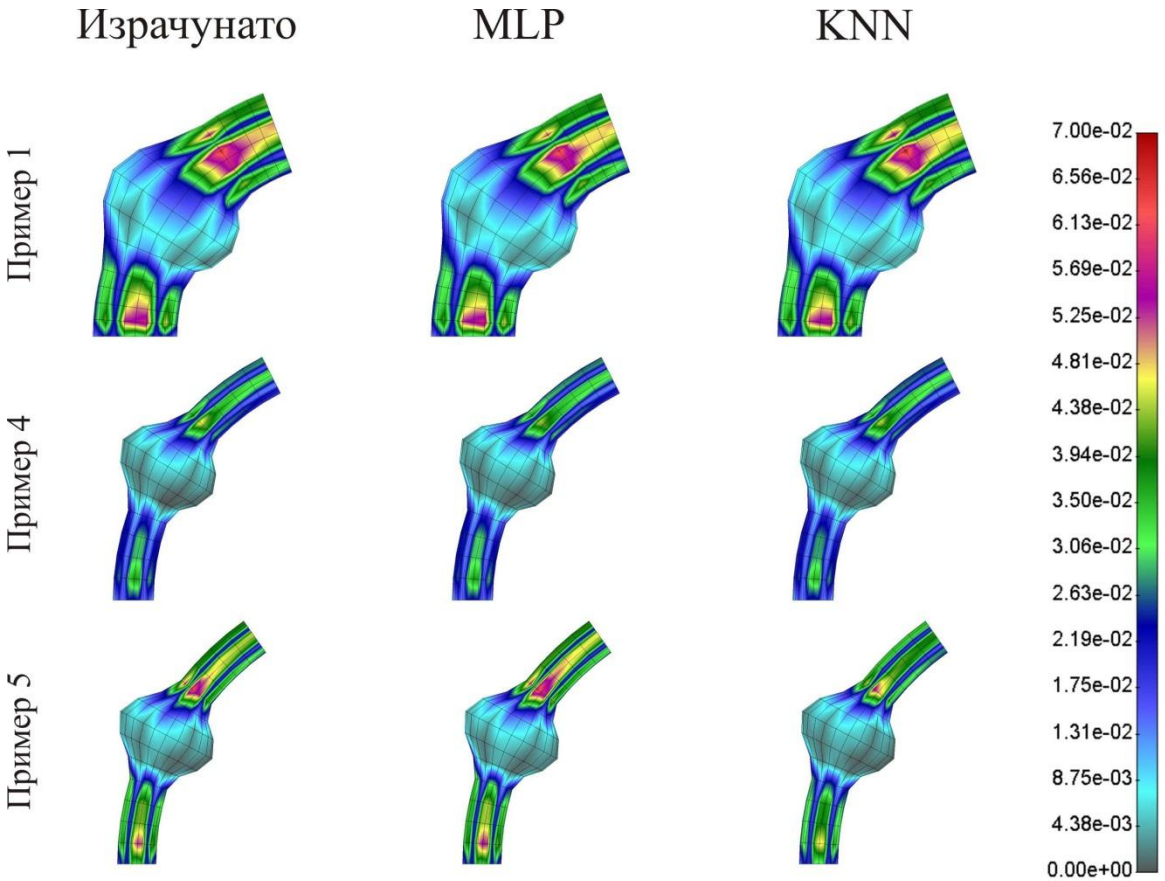
Модел	**RMSE анеуризма**	**RMSE каротидна бифуркација**
KNN	0.1008	0.2416
MLP	0.0351	0.0305

Табела 4-9: RMSE вредности добијене тестирањем KNN и MLP алгоритма за предвиђање расподеле симчућег напона на зиду за моделе анеуризме (средња колона) и каротидне бифуркације (последња колона).

Посматрајући табелу 4-9 може се закључити да су оба модела (KNN и MLP) показала висок потенцијал за предвиђање расподеле смичућег напона на зиду и за анеуризму и за каротидну бифуркацију. Исто се може закључити и посматрањем слика 4-17 (за каротидну бифуркацију) и 4-18 (за анеуризму) на којима је дат упоредни приказ израчунате и предвиђене расподеле смичућег напона за три случајно одабрана тестна примера.



Слика 4-17: Упоредни приказ израчунате, предвиђене MLP алгоритмом и предвиђене KNN алгоритмом расподеле смичућег напона за модел каротидне бифуркације. Скала је у [Pa].



Слика 4-18: Упоредни приказ израчунате, предвиђене MLP алгоритмом и предвиђене KNN алгоритмом расподеле смичућег напона за модел анеуризме. Скала је у [Pa].

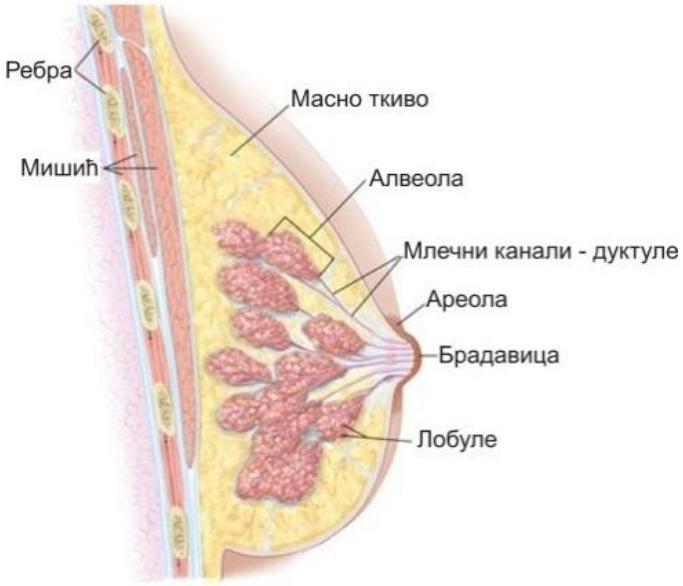
Посматрањем слика 4-17 и 4-18 (упоређивањем колона) може се закључити да су KNN и MLP алгоритми способни да са великом прецизношћу предвиђају расподелу смичућег напона на зиду за моделе анауризме и каротидне бифуркације. MLP алгоритам је пружио знатно већу прецизност (мања RMSE) у односу на KNN. Овим радом је показано да се алгоритми техника истраживања података могу користити за решавање описаног проблема. Међутим, треба напоменути да би за евентуалну експлоатацију ових алгоритама било ипак неопходно да се креирају финији FE модели који имају гушћу мрежу тј. већи број чворова.

5. Детекција канцера дојке на дигитализованим мамографима употребом техника истраживања података

5.1. Опис проблема

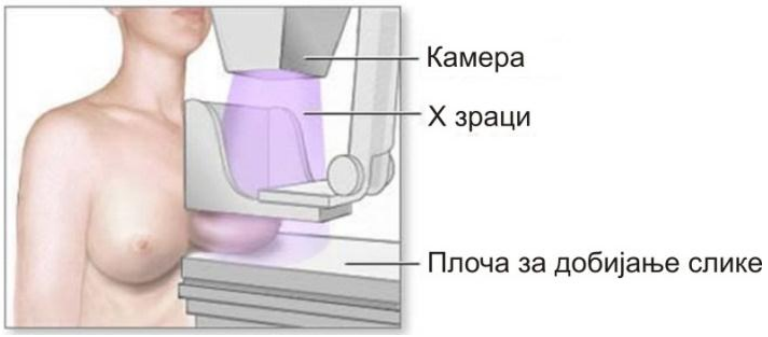
Једна од најсмртоноснијих болести данашњице је рак. Рак дојке је најчећи облик рака код жена. Откривање болести у раној фази драстично повећава шансе за излечење, па је у клиничкој пракси од великог значаја употреба алата за дијагностификовање болести. Мамографија се показала као ефикасан поступак за откривање канцера дојке па је уведена широм света са циљем откривања канцера у што ранијој фази.

Анатомија дојке је веома комплексна (слика 5-1). Свака дојка садржи између 15 и 25 појединачних жлезда (алвеола) које имају мање одсечке - лобуле. Свака од њих заједно са везивним и масним ткивом изграђује по један режањ. Режњеви су међусобно подељени гушћим везивним ткивом, а сваком режњу груди припада по један главни одводни канал (дуктула) који завршава левкастим проширењем на брадавици. Брадавица је окружена тамније пигментисаним подручјем коже које се назива ареола.



Слика 5-1: Анатомија дојке.

Мамографија је неинвазивна метода рендгенског прегледа дојки код жена, а у веома ретким случајевима и код мушкараца. Снимање се ради специјалним рендген апаратом - мамографом, којим се виде промене на меким ткивима. Лекар (радиолог) уз помоћ ове методе може да утврди садржај и разлику између оболелог и здравог ткива. Овом методом се могу открити и микрокалцификације које су често први показатељи тумора дојки. Поступак мамографског снимања је приказан на слици 5-2.

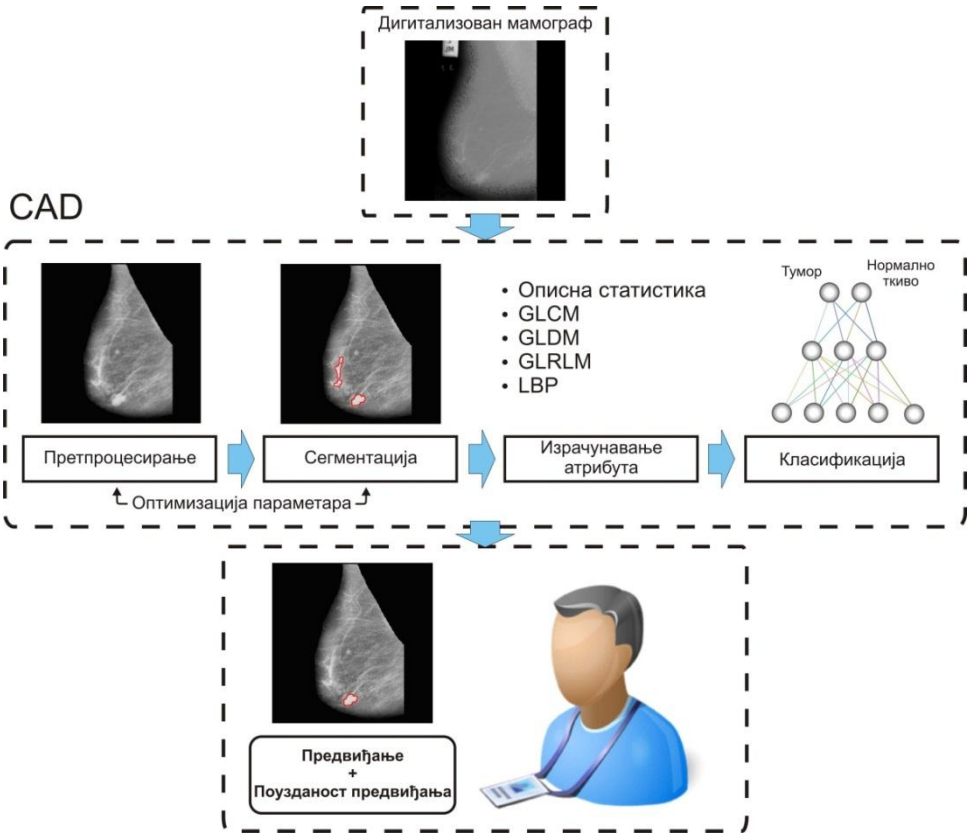


Слика 5-2: Мамографски преглед.

Преглед мамографа је веома тежак задатак чак и за искусне радиологе. Процена лекара зависи од обуке, искуства и субјективног критеријума. Развој компјутерски помогнуте дијагнозе (енг. *Computer Aided Diagnosis* - CAD) је од великог значаја јер може смањити број грешака лекара приликом мамографског снимања дојки. На овај начин се могу смањити и број лажно позитивних (дијагностификован је непостојећи канцер) и лажно негативних (постојећи канцер није дијагностификован) случајева. Софтвер може указати на неки сумњив знак или помоћи у класификовању тумора у малигне или бенигне. Поступак компјутерски помогнуте дијагнозе користи дигитализоване мамографе и састоји се од четири модула (слика 5-3):

1. Претпроцесирање. У фази претпроцесирања се врши филтрирање и побољшање квалитета мамографа (нпр. побољшање контраста).
2. Сегментација. У циљу одређивања релевантних карактеристика мамографа није потребно користити целу слику. Након претпроцесирања потребно је издвојити сумњиве регије (потенцијалне туморе). Овај поступак укључује уклањање пекторалног мишића, уклањање свих објеката који се налазе изван дојке и сегментацију сумњивих регија дојке.
3. Израчунавање атрибута. У оквиру ове фазе свака сумњива регија (потенцијални тумор) се описује нумеричким вредностима (атрибутима) које се израчунавају применом различитих методологија.
4. Класификација. У овој фази се свака сумњива регија, издвојена у фази сегментације и описана у фази израчунавања атрибута, класификује у једну од две категорије: тумор или нормално ткиво. За класификацију сумњивих регија се могу користити различити алгоритми техника истраживања података као нпр. стабла одлучивања, метода потпорних вектора, неуронске мреже, логистичка регресија итд.

Дакле, циљ овог истраживања је развој система компјутерски помогнуте дијагнозе, базираног на техникама истраживања података, који ће бити у стању да са великом прецизношћу региструје присутност тумора и његову позицију на мамографима. Фазе претпроцесирања, сегментације и израчунавања атрибута су реализоване у програмском пакету MATLAB, док је фаза класификације реализована у софтверу WEKA.



Слика 5-3: Фазе за детекцију тумора на дигитализованим мамографима.

Резултати овог истраживања су презентовани на међународним научним конференцијама у Патрасу [70] и Ханији [71] 2013. године.

5.2. Претходна истраживања у области

Преглед мамографа је веома тежак задатак чак и за искусне радиологе. Сходно томе, последњих година је уложен велики напор у циљу развоја компјутерски помогнуте дијагнозе за аутоматску детекцију тумора. Комбинација система компјутерски помогнуте дијагнозе и знања радиолога може значајно повећати проценат успешности успостављања дијагнозе. Сензитивност детекције без CAD-а је око 80% а са CAD-ом око 90% [71].

Ченг и други [73] су у оквиру свог истраживања описали различите приступе за аутоматску детекцију и класификацију тумора (малигни или бенигни) поредећи њихове предности и мане. У оквиру овог истраживања су приказани различити алгоритми за претпроцесирање и сегментацију. Дат је опис различих група атрибута за описивање сумњивих регија (атрибути облика, атрибути интензитета и атрибути текстуре). Коначно, приказани су и различити алгоритми техника истраживања података који се могу користити за решавање проблема детекције и класификације тумора на мамографима (стабла одлучивања, неуронске мреже итд.).

Уен и други [74] су приказали употребу атрибута текстуре за класификацију тумора. У оквиру овог истраживања сегментација је извршена употребом алгоритама за детекцију маса на основу ивица (енг. *Edge based mass detection algorithm*) и постигнута је сензитивност 94.5% (11.3 лажно позитивних по слици). За класификацију

је коришћена неуронска мрежа - вишеслојни перцептрон и постигнути резултати укључују површину испод ROC криве (AUC) 0.815, сензитивност 85% и 3.47 лажно позитивних по слици (енг. *False Positive per image* - FPpi).

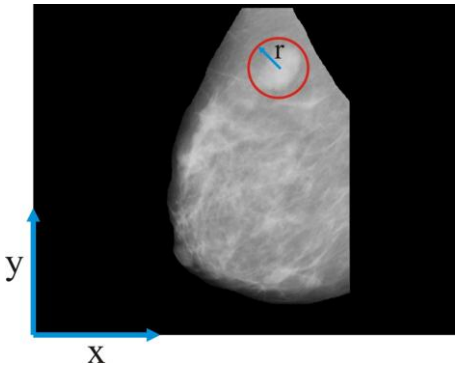
Насер и Мохамед [75] су у оквиру свог истраживања приказали употребу напредних методологија за обраду слика у циљу успешне сегментације маса на мамографима. У оквиру овог рада приказана је употреба алгоритма адаптивне еквализације са лимитираним контрастом за побољшање контраста и алгоритма 2D адаптивног медијан филитрирања за уклањање шума. Приказани резултати укључују сензитивност 92.3% и 2.75 лажно позитивних по слици. Тестирање је извршено употребом 36 мамографа из mini-MIAS базе (која се користи и у оквиру нашег истражива).

5.3. База дигитализованих мамографа

Циљ овог истраживања је развој CAD система који ће бити у стању да са великом прецизношћу детектује присутност тумора и његову позицију на мамографима. Овај систем се састоји од четири главне фазе: претпроцесирања, сегментације, израчунавања атрибута и класификације. Како би се тестирала успешност предложеног CAD система, неопходно је постојање базе података. У оквиру овог истраживања коришћена је mini-MIAS база (доступна преко веба) [76].

Mini-MIAS база је креирана од стране истраживачке организације из Велике Британије (енг. *Mammographic Image Analysis Society* - MIAS). База садржи 322 дигитализована мамографа за 161 пацијенткињу (за сваку пацијенткињу постоји снимак леве и десне дојке). За сваки мамограф познате су информације о типу околног ткива дојке (масно, масно-жлездано или густо-жлездано), типу присутне абнормалности (микрокалцификације, јасно дефинисане масе, асиметрија итд.), врсти тумора (бенигни или малигни) као и локацији и величини тумора (координате x, y и полупречник r). Информације о мамографима су доступне у облику који је приказан на слици 5-4.

Редни број мамографа	Тип околног ткива дојке	Тип присутне абнормалности	Врста тумора	x	y	r
mdb015	Масно-жлездано	јасно дефинисана	бенигни	595	864	68



Слика 5-4: Мамограф 015 - mini-MIAS база.

Мамографи mini-MIAS базе имају резолуцију 1024×1024 пиксела величине 200×200 микрона. Све слике су сачињене у медиолатералној пројекцији (МЛО). Укупно 322 мамографа је подељено у четири групе: нормалне мамографске слике (209), бенигне мамографске слике (61), малигне мамографске слике (51) и мамографске слике са бенигним и малигним тумором (1). Структура mini-MIAS базе је приказана у табели 5-1.

Група мамографа	Укупан број мамографа	Укупан број абнормалности (тумора)
нормални (без тумора)	209	-
бенигни	61	66
малигни	51	53
бенигни и малигни	1	2

Табела 5-1: Стуктура mini-MIAS базе података.

На мамографима се микрокалцификације виде као веома мале (величине 0.1-1 mm) светле површине које се јављају обично у групама. Детекција микрокалцификација није једноставна у великом броју случајева, али постоје успешне методе за њихову сегментацију [77], [78]. Са друге стране, абнормалности другог типа су веће и обично нису груписане, па је неопходна употреба другачијих алгоритама за њихову сегментацију. У оквиру ове студије биће одрађена сегментација и детекција свих абнормалности изузев микрокалцификација. Ово смањује mini-MIAS базу података са 322 слике на 298 слика (избачене су и четири слике са непотпуним подацима о положају и величини тумора). У оквиру ове редуковане mini-MIAS базе 92 тумора је присутно на 89 мамографа). Структура ове редуковане mini-MIAS базе је приказана у табели 5-2.

Група мамографа	Укупан број мамографа	Укупан број абнормалности (тумора)
нормални (без тумора)	209	-
бенигни	50	52
малигни	38	38
бенигни и малигни	1	2

Табела 5-2: Стуктура редуковане mini-MIAS базе података (без непотпуних података и без микрокалцификација).

5.4. Претпроцесирање слика

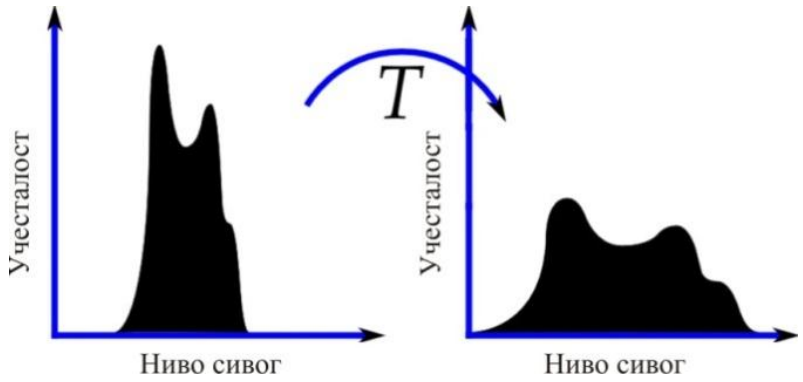
Претпроцесирање је веома важна фаза у оквиру CAD-а и има за циљ побољшање квалитета дигитализованог мамографа. Главне идеје претпроцесирања су истицање маса у односу на околно ткиво и побољшање текстуре маса. У оквиру овог истраживања претпроцесирање мамографа обухвата две фазе:

1. побољшање контраста чиме се повећава контраст између маса и околног ткива и
2. филтрирање чиме се постиже уклањање шумова насталих приликом дигитализације слике.

Поступци побољшања контраста и филтрирања су описани детаљно у поглављима 5.4.1 и 5.4.2.

5.4.1 Побољшање контраста

Дигитализовани мамографи су често лошег контраста па је у циљу успешне сегментације потребно исти побољшати. У оквиру овог истраживања, за побољшање контраста је употребљен алгоритам адаптивне еквализације хистограма са лимитираним контрастом (енг. *Contrast-Limited Adaptive Histogram Equalization* - CLAHE). CLAHE алгоритам је заправо побољшана верзија алгоритма адаптивне еквализације хистограма (енг. *Adaptive Histogram Equalization* - AHE). Алгоритам адаптивне еквализације хистограма врши еквализацију хистограма засебно за појединачне регионе слике и на тај начин врши редистрибуцију нивоа сивог. Еквализација хистограма представља метод побољшања контраста слике употребом хистограма вредности нивоа сивог слике (слика 5-5).



Слика 5-5: Поступак еквализације хистограма.

Да би се извршила еквализација слике најпре је потребно израчунати функцију кумулативне расподеле вероватноће (енг. *Cumulative Distribution Function* - CDF*)*:

$$CDF(i) = \sum_{j=1}^{i} n_i \tag{5.1}$$

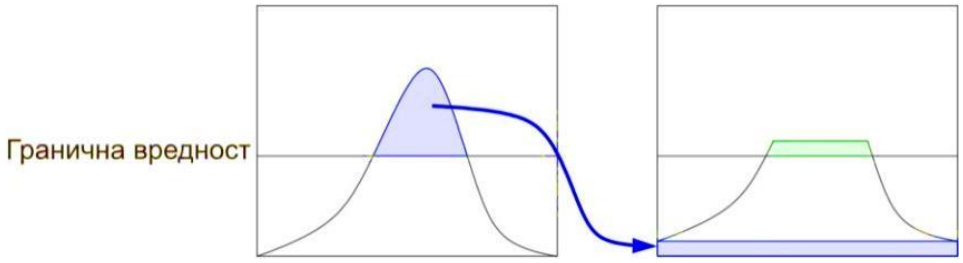
где је n_i укупан број пиксела вредности i.

После израчунавања CDF функције врши се еквализација хистограма тако што се вредност сваког пиксела израчуна помоћу функције трансформације:

$$h(v) = round\left(\frac{CDF(v) - CDF_{min}}{(M \times N) - CDF_{min}} \times (L-1)\right) \tag{5.2}$$

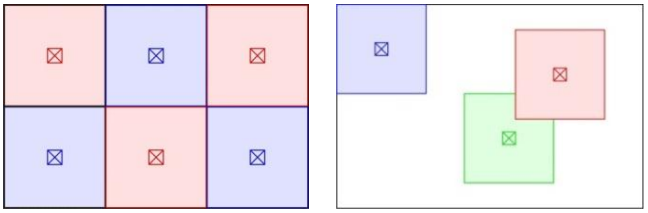
где је v вредност пиксела, $CDF(v)$ је вредност CDF функције за пиксел вредности v, CDF_{min} је минимална ненулта вредност CDF функције, M и N дефинишу величину слике, L је број нивоа сивог излазне слике, а $round$ је функција која врши заокруживање на најближу целобројну вредност.

Када слика садржи прилично хомогене регије пиксела, хистограм слике садржи изражен пик, па ће еквализација мапирати узак опсег вредности пиксела у цео опсег вредности пиксела. Ово доводи до тога да АНЕ алгоритам прекомерно појачава малу количину шума присутну унутар хомогене регије пиксела слике. Овај проблем је решен увођењем CLAHE алгоритма који ограничава повећање контраста. Ово се постиже одсецањем врха хистограма према унапред дефинисаној граничној вредности (слика 5-6). После одсецања врха хистограма врши се његова еквализација на исти начин као и код АНЕ алгоритма.



Слика 5-6: Одсецање горњег дела хистограма у циљу ограничавања повећања контраста - CLAHE.

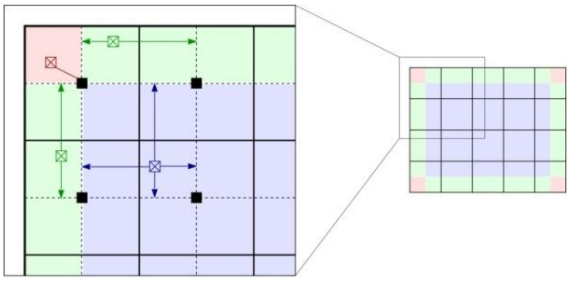
Побољшање контраста слике засебном еквализацијом хистограма различитих делова слике доводи до вештачки креираних „граница“ (слика 5-7а). Са друге стране, трансформација појединачних вредности пиксела еквализацијом хистограма вредности пиксела његовог суседства је превише прорачунски захтевна (слика 5-7б).



а) б)

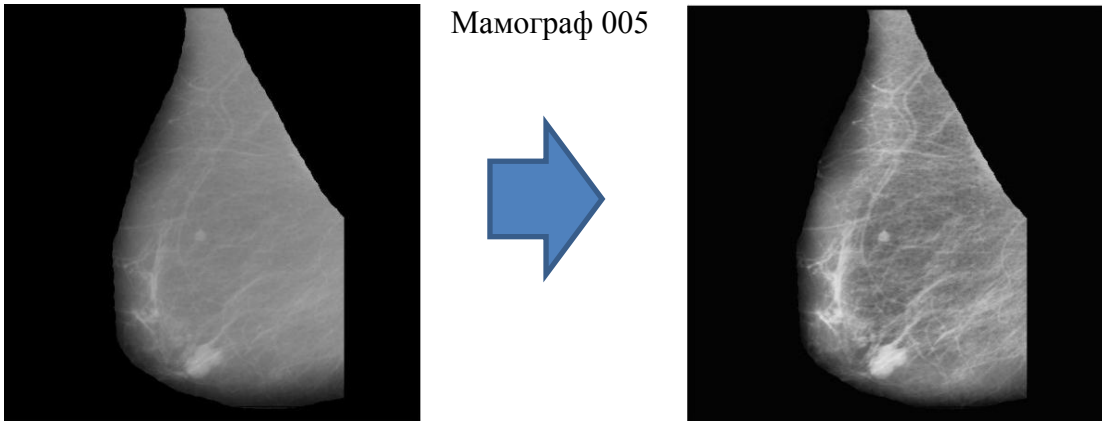
Слика 5-7: а) Трансформација вредности пиксела засебних регија еквализацијом хистограма регије, б) трансформација појединачних вредности пиксела еквализацијом хистограма његовог суседства.

Решење претходно поменутог проблема лежи у интерполацији. Наиме, слика се најпре дели на засебне регије једнаких величина (слика 5-8 десно). Затим се хистограм, функција кумулативне расподеле вероватноће и функција трансформације израчунавају за сваку регију. Функције трансформације су одговарајуће само за централне пикселе регија (црни квадратићи на слици 5-8 лево). Вредности осталих пиксела се израчунавају применом интерполације употребом функција трансформације најближих централних пиксела. Вредности пиксела који се налазе унутар површине дефинисане помоћу четири централна пиксела се одређују билинеарном интерполацијом (плави пиксел на слици 5-8 лево). Вредности пиксела који се налазе близу неке од ивица слике се одређују линеарном интерполацијом (зелени пиксели на слици 5-8 лево). Коначно, вредности пиксела који леже у близини неког од углова слике се одређују применом функције трансформације најближег централног пиксела (црвени пиксел на слици 5-8 лево).



Слика 5-8: Поступак интерполације вредности пиксела применом функција трансформације најближих централних пиксела.

У оквиру ове студије побољшање контраста мамографа је извршено применом CLAHE алгоритма помоћу функције adapthisteq програмског пакета MATLAB. На слици 5-9 је приказан мамограф 005 пре и после побољшања контраста.

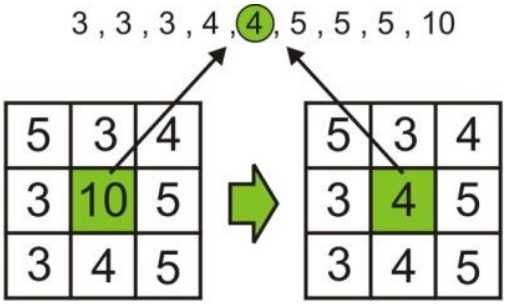


Слика 5-9: Побољшање контраста применом CLAHE алгоритма.

5.4.2 Филтрирање

Након побољшања контраста потребно је извршити и филтрирање слике како би се уклонили шумови настали приликом дигитализације слике (нпр. изоловани пиксели или линије на слици). У оквиру овог истраживања, за филтрирање се користи алгоритам медијан филтрирања.

Медијан филитрирање (енг. *Median filtering*) се обавља заменом вредности пиксела улазне слике са медијаном локалног суседства тог пиксела. Локално суседство има најчешће непаран број елемената (пиксела) и димензије 3×3 ($D = 3$), 5×5 ($D = 5$) или 7×7 ($D = 7$) елемената. Пример употребе медијан филтрирања над локалним суседством 3×3 је приказан на слици 5-10.



Слика 5-10: Пример медијан филтрирања за локално суседство 3×3 ($D = 3$).

Поступак одређивања медијана своди се на сортирање вредности пиксела у оквиру локалног суседства у растући или опадајући низ, и замену вредности текућег пиксела са централним пикселом у сортираном низу. У случају да је број пиксела у оквиру локалног суседства непаран (најчешће је тако) централни пиксел сортираног низа постоји. У супротном, ако је број пиксела у оквиру локалног суседства паран, вредност медијана се одређује као средња вредност два централна пиксела.

Оптимална величина локалног суседства (вредност D) је одређена посебно за различите типове околног ткива оптимизацијом у циљу максимизације сензитивности процеса сегментације (детаљи процеса оптимизације параметра D заједно са осталим параметрима сегментације су дати у поглављу 5.6). Оптималне вредности параметра D за три различита типа околног ткива дојке су приказане у табели 5-3.

Тип околног ткива дојке	**Величина локалног суседства - D**
Масно (енг. *Fatty*)	3
Масно-жлездано (енг. *Fatty-glandular*)	3
Густо-жлездано (енг. *Dense-glandular*)	9

Табела 5-3: Оптималне вредности величине локалног суседства (D) за медијан филтрирање.

У оквиру ове студије медијан филитрирање мамографа је извршено применом функције medfilt2 програмског пакета MATLAB.

5.5. Сегментација слика

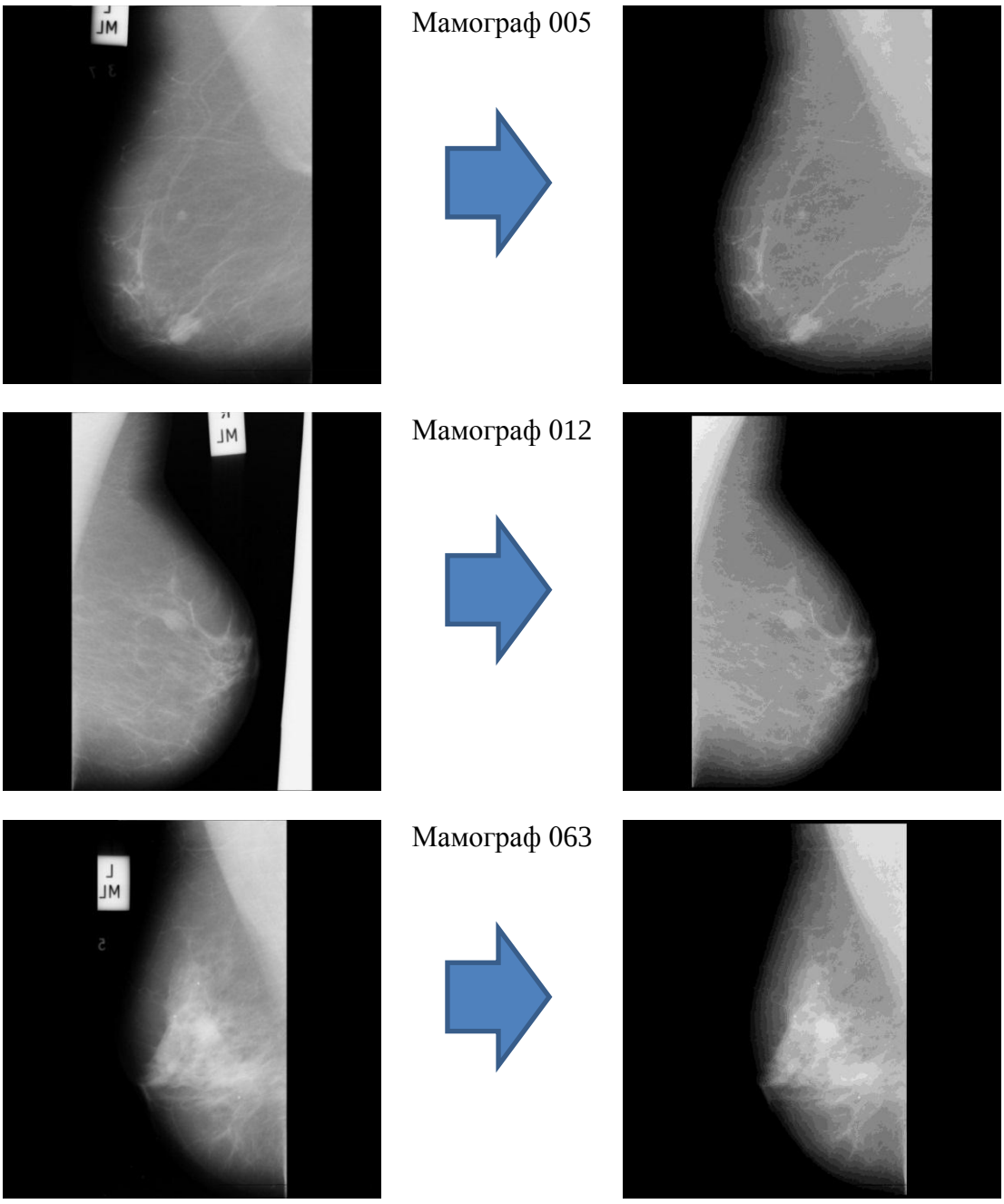
Циљ сегментације је издвајање сумњивих регија (потенцијалних тумора) на мамографима. Ове сумњиве регије су непреклапајуће површине које се у фази класификације класификују у једну од две категорије: тумор или нормално ткиво. Сумњиве регије су обично светлије од околног ткива, имају скоро униформну густину и нејасне границе [79]. Успешна сегментација треба да издвоји висок проценат тумора чак и по цену великог броја лажно позитивних. Лажно позитивне сумњиве регије се уклањају у фази класификације. У оквиру овог истраживања сегментација обухвата два корака:

1. уклањање позадине и пекторалног мишића и
2. детекција сумњивих регија.

Алгоритми за уклањање позадине и пекторалног мишића и детекцију сумњивих регија су описани детаљно у поглављима 5.5.1 и 5.5.2.

5.5.1 Уклањање позадине и пекторалног мишића

Први корак сегментације је уклањање свих објеката који се налазе изван граница дојке. Ово се постиже једноставним алгоритмом који проналази највећу површину повезаних пиксела са интензитетом већим од нуле, а затим свим осталим пикселима додељује вредност 0 (црна боја). На овај начин се уклањају сви изоловани објекти (нпр. ознаке на мамографима) који се налазе изван граница дојке. Ова идеја је илустрована на слици 5-11.

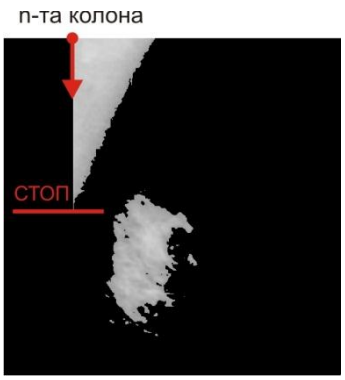


Слика 5-11: Поступак уклањања позадине (свих елемената изван граница дојке). Лево - оригинални мамографи; десно - мамографи са уклоњеном позадином.

Осим позадине, са мамографа је потребно уклонити и пекторални мишић. Пекторални мишић је веома важно уклонити јер је он обично светле боје на мамографу па може правити проблем приликом детекције а затим и класификације сумњивих регија (потенцијалних тумора). Пекторални мишић и централни део дојке су обично веће густине (светлије површине) у односу на остатак дојке па се сходно томе могу издвојити применом оператора прага (енг. *Thresholding*) над вредностима пиксела оригиналног мамографа (слика 5-13а):

$$I_{thres}(i,j) = \begin{cases} I(i,j) & I(i,j) \geq 155 \\ 0 & I(i,j) < 155 \end{cases} \tag{5.3}$$

На слици 5-13б је приказано изоловано централно ткиво дојке са пекторалним мишићем. Следећи корак је издвајање централног дела дојке. Ово се постиже применом веома једноставног алгоритма приказаног на слици 5-12.



Алгоритам за издвајање централног ткива дојке:

Корак 1: Кренути од првог реда и *n*-те колоне; *n* је колона првог ненултог пиксела у првом реду.

Корак 2: Скенирати пикселе са лева на десно.

Корак 3: Ако пиксел има ненулту вредност заменити га са нулом, преместити се на следећи пиксел у истом реду и поновити корак 3.

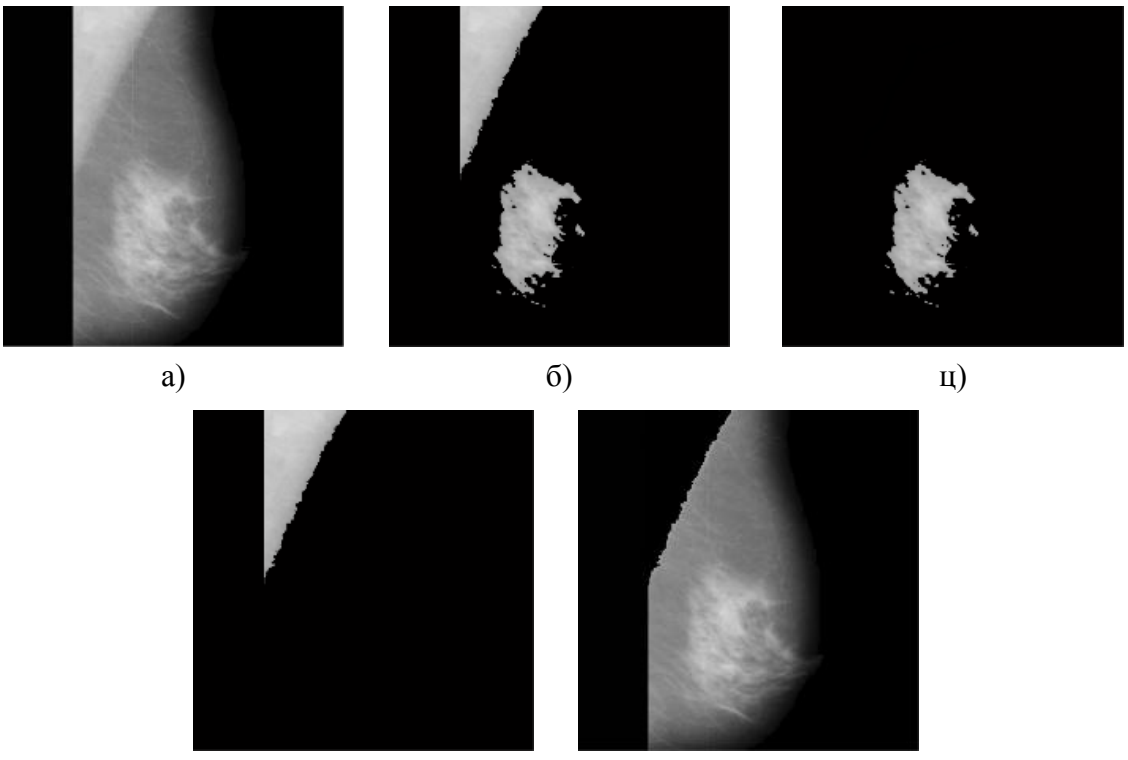
Ако пиксел има нулту вредност, прећи на *n*-ту колону следећег реда и поновити корак 3.

Прекинути процедуру када се исцрпе сви редови.

Слика 5-12: Алгоритам за издвајање централног ткива дојке.

Изоловано централно ткиво дојке је приказано на слици 5-13ц. Одузимањем слике 5-13ц од слике 5-13б добија се слика изолованог пекторалног мишића (слика 5-13д). Коначно, одузимањем слике 5-13д од слике 5-13а добија се приказ дојке са уклоњеним пекторалним мишићем (слика 5-13е). Слика 5-13е представља улаз у следећу фазу сегментације која издваја сумњиве регије унутар дојке (поглавље 5.5.2).

Алгоритми за уклањање позадине и пекторалног мишића су реализовани у програмском пакету MATLAB.



а) б) ц)

д) е)

Слика 5-13: а) Оригинални мамограф са уклоњеном позадином - I, б) оригинални мамограф након *thresholding* оператора - I_{thres}, ц) издвојен централни део дојке, д) издвојен пекторални мишић, е) мамограф са уклоњеним пекторалним мишићем.

5.5.2 Детекција сумњивих регија

У овој фази сегментације врши се детекција сумњивих регија њиховим издвајањем од остатка дојке. Веома је важно постићи високу сензитивност тј. успешно издвојити велики проценат тумора у оквиру ове фазе па чак и по цену великог броја лажно позитивних. Лажно позитивне сумњиве регије се уклањају у последњој фази CAD-а - фази класификације. Улаз у ову фазу је претпроцесиран мамограф (побољшан контраст, филтриран) са уклоњеном позадином и пекторалним мишићем (слика 5-13е).

У оквиру овог истраживања за издвајање сумњивих регија коришћена је функција прага (енг. *Threshold function*):

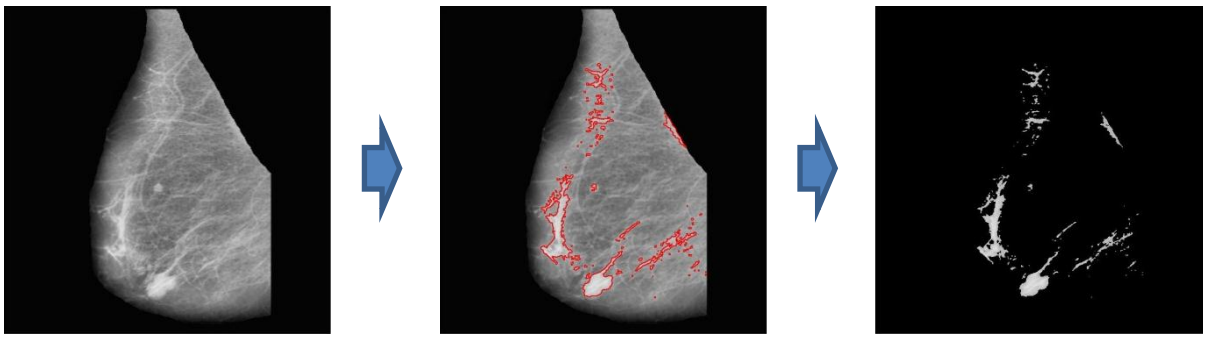
$$Thres = a_1 I_{mean} + a_2 I_{max} + a_3 I_{stdev} + a_4 \tag{5.4}$$

где су I_{mean}, I_{max}, I_{stdev} средња вредност, максимална вредност и стандардна девијација пиксела мамографа. Параметри функције a_1, a_2, a_3 и a_4 су одређени оптимизацијом (поглавље 5.6) са циљем максимизације сензитивности поступка сегментације (броја успешно сегментираних маса). Након израчунавања вредности функције прага за посматрани мамограф сумњиве регије се издвајају на следећи начин:

$$I_{thres}(i,j) = \begin{cases} I(i,j) & I(i,j) \geq Thres \\ 0 & I(i,j) < Thres \end{cases} \tag{5.5}$$

где је $thres$ вредност функције прага (5.4) за посматрани мамограф I. Поступак издвајања сумњивих регија применом функције прага илустрован је на слици 5-14.

Мамограф 005

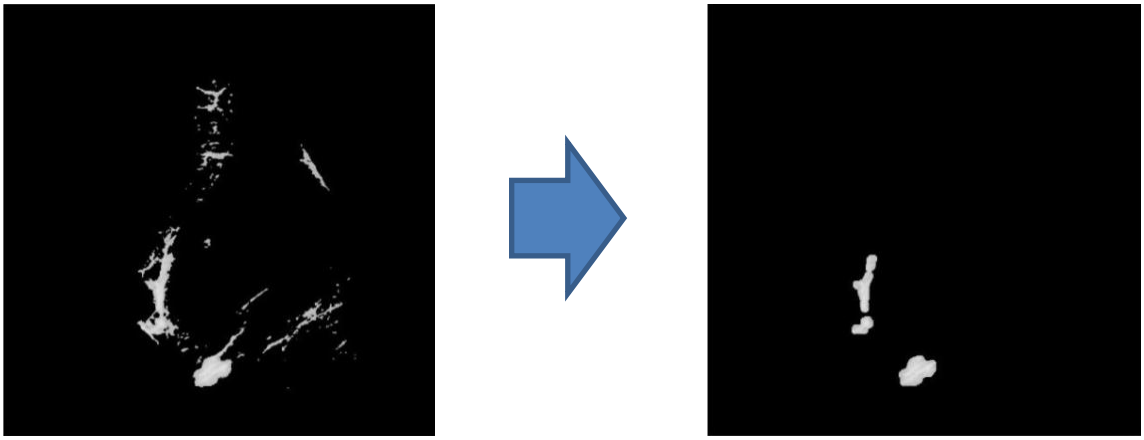


Слика 5-14: Поступак издвајања сумњивих регија применом функције прага (лево - улазни мамограф I, десно – мамограф после примене оператора прага I_{thres}).

После примене оператора прага потребно је уклонити мале објекте који сигурно нису масе и на тај начин драстично смањити број сумњивих регија. Ово се постиже применом операције морфолошког отварања (енг. *Morphological opening*) употребом кружног структурног елемента B чија је величина одређена поступком оптимизације (поглавље 5.6). Морфолошко отварање је поступак који ублажава контуру објекта, укида танке везе између делова објекта и елиминише мале оштре делове објекта. Пример операције морфолошког отварања кружним структурним елементом над троуглом приказан је на слици 5-15.

Слика 5-15: Поступак морфолошког отварања.

Операција морфолошког отварања је реализована у програмском пакету MATLAB применом функције imopen. На слици 5-16 приказан је мамограф 005 после примене операције морфолошког отварања. Може се приметити да је применом морфолошког отварања уклоњен огроман број малих сумњивих регија које не припадају масама.



Слика 5-16: Мамограф 005 после примене операције морфолошког отварања.

5.6. Оптимизација параметара претпроцесирања и сегментације

Параметри функције прага a_1, a_2, a_3 и a_4, полупречник кружног структурног елемента морфолошког отварања R_B и величина локалног суседства медијан филтрирања (вредност D) су независно одређени за три типа околног ткива дојке (масно, масно-жлездано или густо-жлездано).

Оптимизација параметара је извршена применом Nelder-Mead алгоритма оптимизације (функција fminsearch у MATLAB-у). Функција циља је дефинисана као:

$$ERROR = \sum_{i=1}^{N} \left(1 - \frac{A_i \cap T_i}{A_i \cup T_i}\right) \tag{5.6}$$

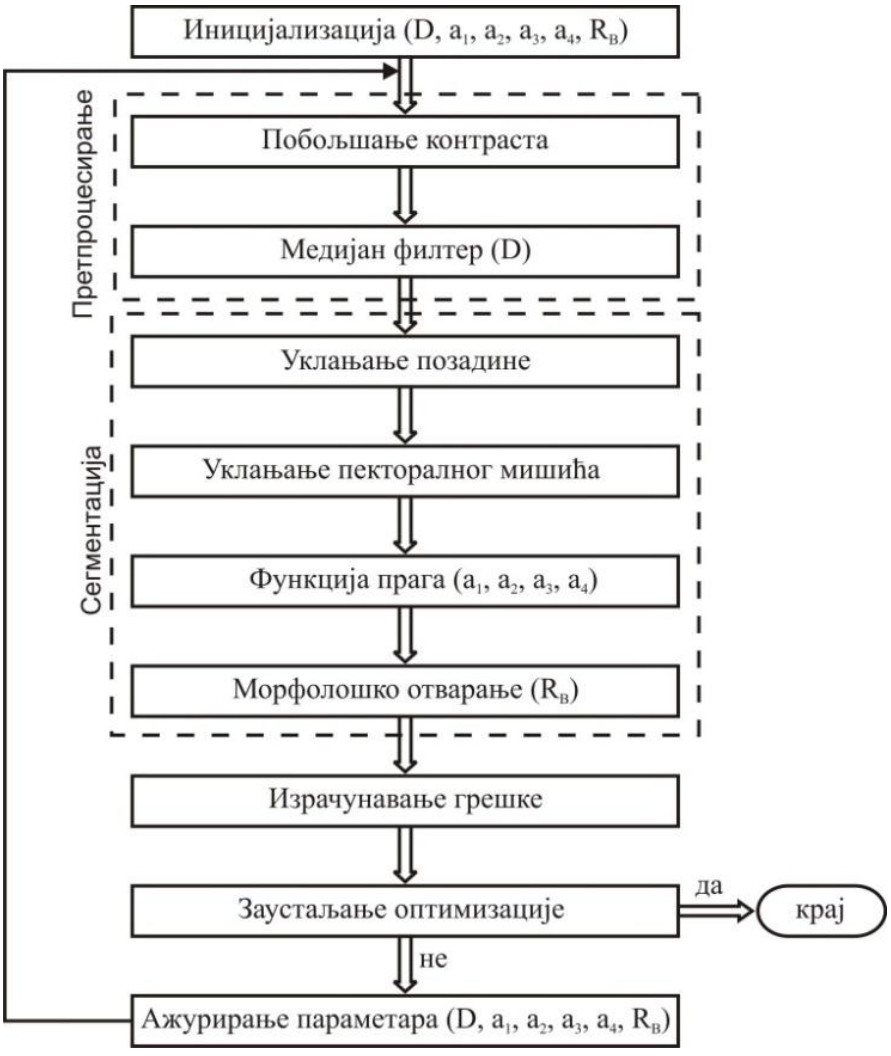
где је N укупан број тумора на свим мамографима, T_i површина i-тог тумора дефинисана позицијом центра и полупречником круга (ове површине су означене од стране лекара), A_i површина добијена сегментацијом (сумњива регија) чији пресек са површином T_i није празан скуп, $A_i \cap T_i$ је површина која представља пресек површи A_i

и T_i, а $A_i \cup T_i$ је површина која представља унију површи A_i и T_i. Резултати оптимизације су приказани у табели 5-4.

Тип околног ткива дојке	a_1	a_2	a_3	a_4	**Полупречник кружног структурног елемента морфолошког отварања - R_B**	**Величина локалног суседства медијан филтера - D**
Масно	0.22	0.56	0.46	1.54	10	3
Масно-жлездано	0.39	0.34	0.29	31.7	12	3
Густо-жлездано	0.30	0.55	0.33	0.52	15	9

Табела 5-4: Оптималне вредности параметара функције прага a_1, a_2, a_3 и a_4, полупречника кружног структурног елемента морфолошког отварања R_B и величине локалног суседства медијан филтрирања D.

Поступак оптимизације је приказан на слици 5-17.



Слика 5-17: Поступак оптимизације параметара претпроцесирања и сегментације.

Предложеним алгоритмима за сегметацију мамографа успешно је сегментирано 91.3% тумора (84 од 92) са 4.14 лажно позитивних по мамографу (укупно 1233 лажно позитивних). Најбољи резултати сегментације су постигнути код групе мамографа са масним типом околног ткива дојке 97.22%, нешто лошији резултати су постигнути код групе мамографа са масно-жлезданим типом околног ткива дојке 93.33%, а најлошији

резултати су постигнути код групе мамографа са густо-жлезданим типом околног ткива дојке 80.77%. Резултати сегментације су приказани у табели 5-5.

Тип околног ткива дојке	Укупан број мамографа	Укупан број абнормалности (тумора)	Укупан број детектованих тумора
Масно	99	36	35 (97.22%)
Масно-жлездано	97	30	28 (93.33%)
Густо-жлездано	102	26	21 (80.77%)
Укупно	298	92	84 (91.30%)

Табела 5-5: Резултати сегментације.

У табели 5-6 приказано је поређење постигнутих резултата сегментације са другим ауторима [80]-[84].

Методе сегментације	Сензитивност	Број лажно позитивних
[80]	<60	>8.50
[81]	<65	>9.00
[82]	<90	>12.00
[83]	<72	>10.50
[84]	0.910	4.77
Предложен метод	0.913	4.14

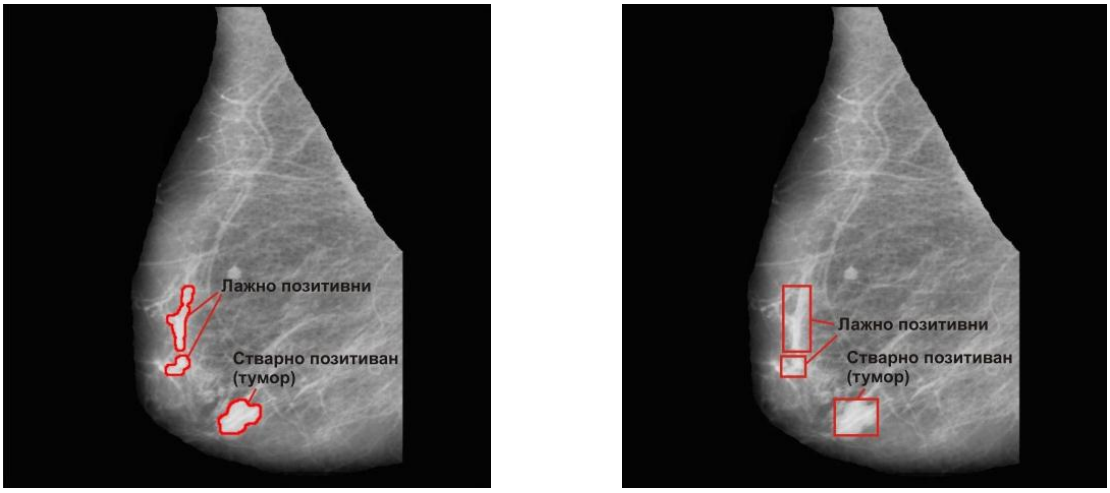
Табела 5-6: Поређење резултата сегментације са другим ауторима.

5.7. Израчунавање атрибута

Израчунавање атрибута је веома важна фаза у поступку детекције канцера дојке. У оквиру ове фазе свака сумњива регија (потенцијални тумор) се описује нумеричким вредностима (атрибутима) које се израчунавају применом различитих методологија. У оквиру овог рада израчунато је укупно 106 атрибута и то применом следећих метофологија:

- описна статистика - 11 атрибута,
- статистика здруженог појављивања нивоа сивог - 20 атрибута,
- статистика разлике интензитета - 5 атрибута,
- статистика појављивања низа пиксела - 11 атрибута и
- метода локалних бинарних шаблона - 59 атрибута.

За израчунавање атрибута описне статистике (11 атрибута) користе се пиксели унутар сумњиве регије (слика 5-18а). Са друге стране, за израчунавање свих осталих атрибута (95 атрибута) користи се правоугана површина настала постављањем правоугаоника који свим својим страницама додирује површину сумњиве регије (слика 5-18б).



а) б)

Слика 5-18: Дефинисање сумњивих регија за израчунавање атрибута, а) атрибути описне статистике, б) атрибути текстуре и атрибути локалних бинарних шаблона.

Сет од 106 атрибута је израчунат за сваку сумњиву регију применом програмског пакета MATLAB, а затим употребљен као улаз за различите класификационе алгоритме чији је задатак да сумњиву регију класификују у једну од две класе: тумор или нормално ткиво.

5.7.1 Атрибути описне статистике

Описна статистика (енг. *Descriptive statistics*) је грана статистике која се бави предочавањем и описивањем главних карактеристика сакупљених података. Ову групу атрибута је најједноставније израчунати. Ако је вектор вредности пиксела унутар сумњиве регије I, а број пиксела унутар сумњиве регије N, за сваку сумњиву регију изачунате су вредности следећих 11 атрибута описне статистике ($f_1 - f_{11}$):

- Минимум (енг. *Minimum*) – минимална вредност пиксела (вредност најтамнијег пиксела) сумњиве регије:

$$f_1 = \min I(i), \quad i = 1, \dots, N \tag{5.7}$$

- Максимум (енг. *Maximum*) – максимална вредност пиксела (вредност најсветлијег пиксела) сумњиве регије:

$$f_2 = \max I(i), \quad i = 1, \dots, N \tag{5.8}$$

- Средња вредност (енг. *Mean*) – просечна вредност свих пиксела унутар сумњиве регије:

$$f_3 = \frac{1}{N}\sum_{i=1}^{N} I(i) \tag{5.9}$$

- Стандардна девијација (енг. *Standard deviation*) – стандардна девијација вредности пиксела сумњиве регије:

$$f_4 = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(I(i) - f_3)^2} \tag{5.10}$$

- Медијан (енг. *Median*) – вредност средњег члана сортираног низа пиксела сумњиве регије:

$$f_5 = \begin{cases} I_{sorted}\left(\frac{N+1}{2}\right) & \text{ако је } N \text{ непаран број} \\ \frac{I_{sorted}\left(\frac{N}{2}\right) + I_{sorted}\left(\frac{N+2}{2}\right)}{2} & \text{ако је } N \text{ паран број} \end{cases} \tag{5.11}$$

  где је I_{sorted} вектор I сортиран у растућем редоследу.

- Интервал варијације (енг. *Range*) – разлика вредности најсветлијег (максимум) и најтамнијег (минимум) пиксела унутар сумњиве регије:

$$f_6 = f_2 - f_1 \tag{5.12}$$

- Најчешћа вредност (енг. *The commonest value*) – најчешћа вредност пиксела унутар сумњиве регије:

$$f_7 = \underset{n\in\{0,\ldots,255\}}{\operatorname{argmax}} \left\{\sum_{i=1}^{N}\delta(I(i), n)\right\} \tag{5.13}$$

  где је $\{0, \ldots, 255\}$ сет могућих нијанси сивог, а δ функција:

$$\delta(a, b) = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases} \tag{5.14}$$

- Поткресана средња вредност 50% (енг. *Trimmed mean*) – средња вредност низа пиксела који се добија када се уклони 50% најтамнијих пиксела сумњиве регије:

$$f_8 = \begin{cases} \frac{1}{(N+1)/2}\sum_{i=(N+1)/2}^{N} I_{sorted}(i) & \text{ако је } N \text{ непаран број} \\ \frac{1}{N/2}\sum_{i=(N+2)/2}^{N} I_{sorted}(i) & \text{ако је } N \text{ паран број} \end{cases} \tag{5.15}$$

- Коефицијент асиметричности (енг. *Skewness*) – параметар који показује да ли је дистрибуција вредности пиксела унутар сумњиве регије асиметрична (негативна асиметрија означава већу фреквенцију натросечних вредности и

обрнуто, позитивна асиметрија означава већу фреквенцију исподпросечних вредности):

$$f_9 = \frac{\sum_{i=1}^{N}(I(i) - f_3)^3}{(N-1)f_4^3} \tag{5.16}$$

- Коефицијент спљоштености (енг. *Kurtosis*) – параметар који пружа информацију о томе у којој мери су вредности пиксела концентрисане око аритметичке средине (спљоштене дистрибуције имају негативну а зашиљене позитивну вредност овог коефицијента):

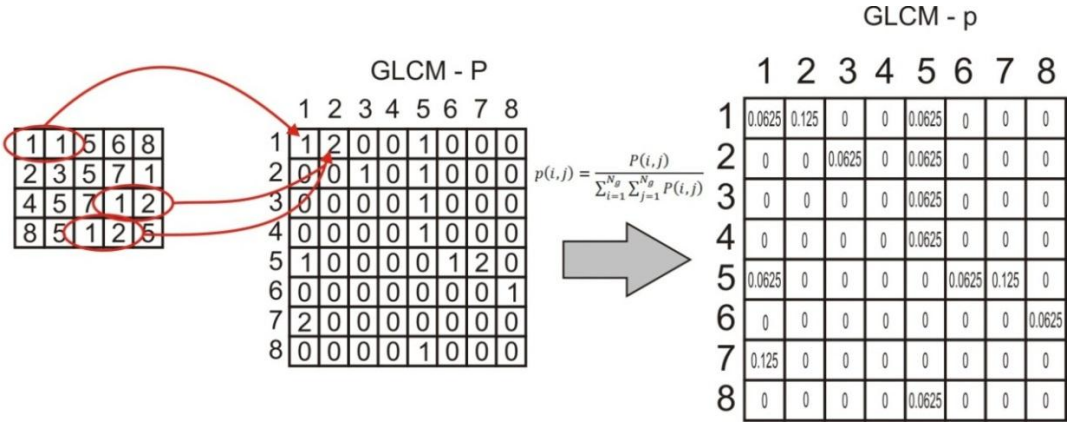
$$f_{10} = \frac{\sum_{i=1}^{N}(I(i) - f_3)^4}{(N-1)f_4^4} \tag{5.17}$$

- Величина (енг. *Size*) – број пиксела унутар сумњиве регије:

$$f_{11} = N \tag{5.18}$$

5.7.2 GLCM атрибути - статистика здруженог појављивања нивоа сивог

Статистика здруженог појављивања нивоа сивог (енг. *Gray level co-occurrence*) је једна од најчешће коришћених метода за описивање текстуре мамографа. Ова методологија представља статистичку методу која испитује просторне релације између пиксела. Статистика здруженог појављивања нивоа сивог карактерише текстуру слике тако што израчунава учесталости појављивања пара пиксела са одређеним вредностима и са дефинисаним међусобним просторним односом на слици, креирајући на тај начин GLCM матрицу (енг. *Gray Level Co-occurrence Matrix* - GLCM) на основу које се израчунавају различите статистичке величине. GLCM матрица се израчунава дефинисањем угла θ $(0^\circ, 45^\circ, 90^\circ, 135^\circ)$ и растојања d међу пикселима. Сваки елемент GLCM матрице $p(i,j)$ представља вероватноћу појављивања пара пиксела са интензитетима i и j и са дефинисаним међусобним просторним односом (d,θ) на посматраној слици. Пример креирања GLCM матрице за растојање $d = 1$, угао $\theta = 0^\circ$ и број нивоа сивог N_g =8 је приказан на слици 5-19.



Слика 5-19: Пример креирања GLCM матрице ($d = 1$, $\theta = 0^\circ$, N_g =8).

У оквиру ове студије GLCM матрица је израчуната за четири угла 0°, 45°, 90° и 135° и за растојање $d = 1$. Извршено је усредњавање како би се правац учинио инваријантним. У циљу умањења утицаја шума на израчунавање GLCM атрибута, број нивоа сивог је умањен на 8 ($N_g = 8$) пре израчунавања GLCM матрице. Укупно 20 GLCM атрибута [85]-[87] је израчунато употребом GLCM матрице. Пре израчунавања GLCM атрибута потребно је израчунати:

$$R = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P(i,j) \tag{5.19}$$

$$p(i,j) = \frac{P(i,j)}{R} \tag{5.20}$$

$$p_x(i) = \sum_{j=1}^{N_g} p(i,j) \tag{5.21}$$

$$p_y(i) = \sum_{i=1}^{N_g} p(i,j) \tag{5.22}$$

$$p_{x+y}(k) = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j)\,, \quad i + j = k \; and \; k = 2,3, \dots, 2N_g \tag{5.23}$$

$$p_{x-y}(k) = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j), \quad |i - j| = k \; and \; k = 0,1, \dots, N_g - 1 \tag{5.24}$$

$$HXY = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j)\big(\log\big(p(i,j)\big)\big) \tag{5.25}$$

$$HXY1 = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j)\big(\log\{p_x(i)p_y(j)\}\big) \tag{5.26}$$

$$HXY2 = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_x(i)p_y(j)\big(\log\{p_x(i)p_y(i)\}\big) \tag{5.27}$$

$$\mu_x = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} ip(i,j) \tag{5.28}$$

$$\mu_y = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} jp(i,j) \tag{5.29}$$

$$\sigma_x = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(i-\mu_x)^2 p(i,j) \tag{5.30}$$

$$\sigma_y = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}\left(j-\mu_y\right)^2 p(i,j) \tag{5.31}$$

Употребом претходно дефинисаних израза (5.19)-(5.31), за сваку сумњиву регију израчунате су вредности следећих 20 GLCM атрибута ($f_{12} - f_{31}$):

- Аутокорелација (енг. *Autocorrelation*):

$$f_{12} = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(ij)p(i,j) \tag{5.32}$$

- Контраст (енг. *Contrast*):

$$f_{13} = \sum_{n=0}^{N_g-1} n^2 \left\{\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)\,, |i-j| = n\right\} \tag{5.33}$$

- Корелација (енг. *Correlation*):

$$f_{14} = \frac{\sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(ij)p(i,j) - \mu_x\mu_y}{\sigma_x\sigma_y} \tag{5.34}$$

- IV централни момент (енг. *Cluster prominence*):

$$f_{15} = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}\left(i+j-\mu_x-\mu_y\right)^4 p(i,j) \tag{5.35}$$

- III централни момент (енг. *Cluster shade*):

$$f_{16} = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}\left(i+j-\mu_x-\mu_y\right)^3 p(i,j) \tag{5.36}$$

- Различитост (енг. *Dissimilarity*):

$$f_{17} = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}|i-j|p(i,j) \tag{5.37}$$

- Енергија (енг. *Energy*):

$$f_{18} = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}\{p(i,j)\}^2 \tag{5.38}$$

- Ентропија (енг. *Entropy*):

$$f_{19} = -\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)\log\big(p(i,j)\big) \tag{5.39}$$

- Хомогеност (енг. *Homogeneity*, *inverse difference moment*):

$$f_{20} = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} \frac{p(i,j)}{1+(i-j)^2} \tag{5.40}$$

- Максимална вероватноћа (енг. *Maximum probability*):

$$f_{21} = \max p(i,j) \tag{5.41}$$

- Варијанса (енг. *Variance*):

$$f_{22} = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} (i-\mu)^2 p(i,j) \tag{5.42}$$

где је μ просечна вредност матрице p.

- Просек суме (енг. *Sum average*):

$$f_{23} = \sum_{i=2}^{2N_g} i p_{x+y}(i) \tag{5.43}$$

- Варијанса суме (енг. *Sum variance*):

$$f_{24} = \sum_{i=2}^{2N_g} (i-f_{25})^2 p_{x+y}(i) \tag{5.44}$$

- Ентропија суме (енг. *Sum entropy*):

$$f_{25} = -\sum_{i=2}^{2N_g} p_{x+y}(i)\log\{p_{x+y}(i)\} \tag{5.45}$$

- Варијанса разлике (енг. *Difference variance*):

$$f_{26} = \sum_{i=0}^{N_g-1} i^2 p_{x-y}(i) \tag{5.46}$$

- Ентропија разлике (енг. *Difference entropy*):

$$f_{27} = -\sum_{i=0}^{N_g-1} p_{x-y}(i)\log\{p_{x-y}(i)\} \tag{5.47}$$

- Информацијска мера корелације 1 (енг. *Information measure of correlation 1*):

$$f_{28} = \frac{HXY - HXY1}{\max\{HX, HY\}} \tag{5.48}$$

- Информацијска мера корелације 2 (енг. *Information measure of correlation 2*):

$$f_{29} = (1 - exp[-2.0(HXY2 - HXY)])^{\frac{1}{2}} \tag{5.49}$$

- Нормализована инверзна разлика (енг. *Inverse difference normalized*):

$$f_{30} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{p(i,j)}{1 + |i-j|/N_g^2} \tag{5.50}$$

- Нормализована хомогеност (енг. *Inverse difference moment normalized*):

$$f_{31} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{p(i,j)}{1 + (i-j)^2/N_g^2} \tag{5.51}$$

5.7.3 GLDM атрибути - статистика разлике интензитета

Као локално обележје се може користити и апсолутна вредност разлике интензитета сивог посаматраног пиксела и пиксела на растојању $\delta = (\Delta x, \Delta y)$.

$$I_\delta(x,y) = |I(x,y) - I(x + \Delta x, y + \Delta y)| \tag{5.52}$$

За дат вектор $\delta = (\Delta x, \Delta y)$ израчунава се GLDM матрица (енг. *Gray Level Difference Matrix* - GLDM) употребом једначине (5.52), а затим се израчунава вектор расподеле вероватоће (димензије N_g, где је N_g број нивоа сивог):

$$D(i,\delta) = \frac{N(I_\delta = i)}{M} \tag{5.53}$$

где је $N(I_\delta = i)$ број појављивања пиксела са интензитетом i унутар GLDM матрице, а M укупан број израчунатих разлика за померај δ. Оваква статистика се назива статистика разлике интензитета (енг. *Gray level difference*) [88].

У оквиру ове студије израчунато је 5 атрибута ($f_{32} - f_{36}$) употребом функције расподеле вероватноће (5.53). Функција раподеле вероватноће је израчуната за четири главна правца ($\delta = (0,1)$, $\delta = (-1,1)$, $\delta = (-1,0)$ и $\delta = (-1,-1)$) и извршено је усредњавање како би се правац учинио инваријантним. Број нивоа сивог је умањен на 8 ($N_g = 8$) пре израчунавања GLDM матрице.

- Контраст (енг. *Contrast*):

$$f_{32} = \sum_{i=0}^{N_g - 1} i^2 \cdot D(i,\delta) \tag{5.54}$$

- Други момент (енг. *Angular second moment*):

$$f_{33} = \sum_{i=0}^{N_g-1} [D(i,\delta)]^2 \tag{5.55}$$

- Ентропија (енг. *Entropy*):

$$f_{34} = \sum_{i=0}^{N_g-1} D(i,\delta)\log\big(D(i,\delta)\big) \tag{5.56}$$

- Средња вредност (енг. *Mean*):

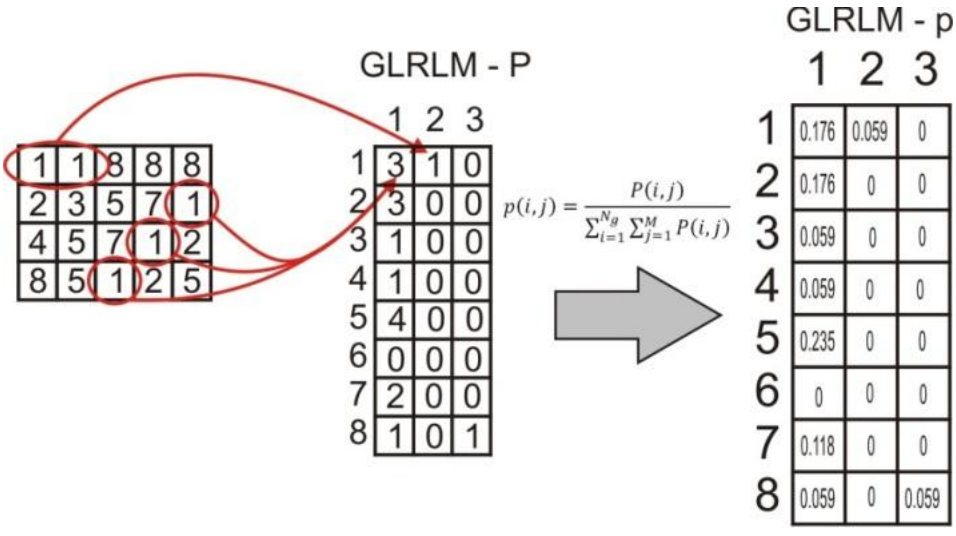
$$f_{35} = \sum_{i=0}^{N_g-1} i \cdot D(i,\delta) \tag{5.57}$$

- Хомогеност (енг. *Homogeneity, inverse difference moment*):

$$f_{36} = \sum_{i=0}^{N_g-1} \frac{D(i,\delta)}{i^2+1} \tag{5.58}$$

5.7.4 GLRLM атрибути - статистика појављивања низа пиксела

Ако се као локално обележје узме број појављивања низа пиксела истог интензитета сивог, добија се метод познат као статистика појављивања низа пиксела (енг. *Gray level run length*) [89]. Суштина овог метода је садржана у GLRLM матрици (енг. *Gray Level Run Length Matrix* - GLRLM) чији сваки елемент $p(i,j)$ представља вероватноћу да се дуж одабраног правца, дефинисаног углом θ, појави непрекидан низ од j пиксела са интензитетом i. На слици 5-20 је приказан пример креирања GLRLM матрице за $d = 1$, $\theta = 0^\circ$ и N_g =8.



Слика 5-20: Пример креирања GLRLM матрице ($d = 1$, $\theta = 0^\circ$, N_g =8).

У оквиру ове студије GLRLM матрица је израчуната за четири угла 0°, 45°, 90° и 135° и за растојање $d = 1$. Извршено је усредњавање како би се правац учинио инваријантним. Број нивоа сивог је умањен на 8 ($N_g = 8$) пре израчунавања GLRLM матрице. Укупно 11 GLRLM атрибута ($f_{37} - f_{47}$) је израчунато употребом GLRLM матрице. У следећим изразима M је највећа дужина низа пиксела истог интензитета сивог, n_r је укупан број низова а n_p број пиксела на слици.

- Истакнутост кратких низова (енг. *Short run emphasis*):

$$f_{37} = \frac{1}{n_r} \sum_{i=1}^{N_g} \sum_{j=1}^{M} \frac{p(i,j)}{j^2} \tag{5.59}$$

- Истакнутост дугих низова (енг. *Long run emphasis*):

$$f_{38} = \frac{1}{n_r} \sum_{i=1}^{N_g} \sum_{j=1}^{M} p(i,j) \cdot j^2 \tag{5.60}$$

- Неуниформност нивоа сивог (енг. *Gray-level non-uniformity*):

$$f_{39} = \frac{1}{n_r} \sum_{i=1}^{N_g} \left(\sum_{j=1}^{M} p(i,j) \right)^2 \tag{5.61}$$

- Неуниформност дужине низова (енг. *Run length non-uniformity*):

$$f_{40} = \frac{1}{n_r} \sum_{j=1}^{M} \left(\sum_{i=1}^{N_g} p(i,j) \right)^2 \tag{5.62}$$

- Неунифомност текстуре (енг. *Run percentage*):

$$f_{41} = \frac{n_r}{n_p} \tag{5.63}$$

- Истакнутост низова ниског нивоа сивог (енг. *Low gray-level run emphasis*):

$$f_{42} = \frac{1}{n_r} \sum_{i=1}^{N_g} \sum_{j=1}^{M} \frac{p(i,j)}{i^2} \tag{5.64}$$

- Истакнутост низова високог нивоа сивог (енг. *High gray-level run emphasis*):

$$f_{43} = \frac{1}{n_r} \sum_{i=1}^{N_g} \sum_{j=1}^{M} p(i,j) \cdot i^2 \tag{5.65}$$

- Истакнутост кратких низова ниског нивоа сивог (енг. *Short run low gray-level emphasis*):

$$f_{44} = \frac{1}{n_r}\sum_{i=1}^{N_g}\sum_{j=1}^{M}\frac{p(i,j)}{i^2 \cdot j^2} \tag{5.66}$$

- Истакнутост кратких низова високог нивоа сивог (енг. *Short run high gray-level emphasis*):

$$f_{45} = \frac{1}{n_r}\sum_{i=1}^{N_g}\sum_{j=1}^{M}\frac{p(i,j) \cdot i^2}{j^2} \tag{5.67}$$

- Истакнутост дугих низова ниског нивоа сивог (енг. *Long run low gray-level emphasis*):

$$f_{46} = \frac{1}{n_r}\sum_{i=1}^{N_g}\sum_{j=1}^{M}\frac{p(i,j) \cdot j^2}{i^2} \tag{5.68}$$

- Истакнутост дугих низова високог нивоа сивог (енг. *Long run high gray-level emphasis*):

$$f_{47} = \frac{1}{n_r}\sum_{i=1}^{N_g}\sum_{j=1}^{M}p(i,j) \cdot i^2 \cdot j^2 \tag{5.69}$$

5.7.5 LBP атрибути – хистограм локалних бинарних шаблона

Метода локалних бинарних шаблона (енг. *Local Binary Pattern* - LBP) [90], [91] је једноставна али врло ефикасна метода која сваком пикселу додељује бинарни број који се добија упоређивањем вредности суседа са вредношћу посматраног пиксела. Наиме, око посматраног пиксела се формира R x R суседа (укупно P суседа), а затим се вредност сваког суседа пореди са вредношћу централног пиксела. У случају да је вредност суседног пиксела већа или једнака од вредности централног пиксела на бинарни број се додаје 1, а у случају да је вредност централног пиксела већа од вредности суседног пиксела на бинарни број се додаје 0. Пример израчунавања LBP вредности је дат на слици 5-21.

За сваки пиксел c слике I, LBP вредност је израчуната на следећи начин:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_P - g_c)2^P \tag{5.70}$$

где је P број суседа који се узимају у обзир а s функција дефинисана на следећи начин:

$$s(g_P - g_c) = \begin{cases} 1 & (g_P - g_c) \geq 0 \\ 0 & (g_P - g_c) < 0 \end{cases} \tag{5.71}$$

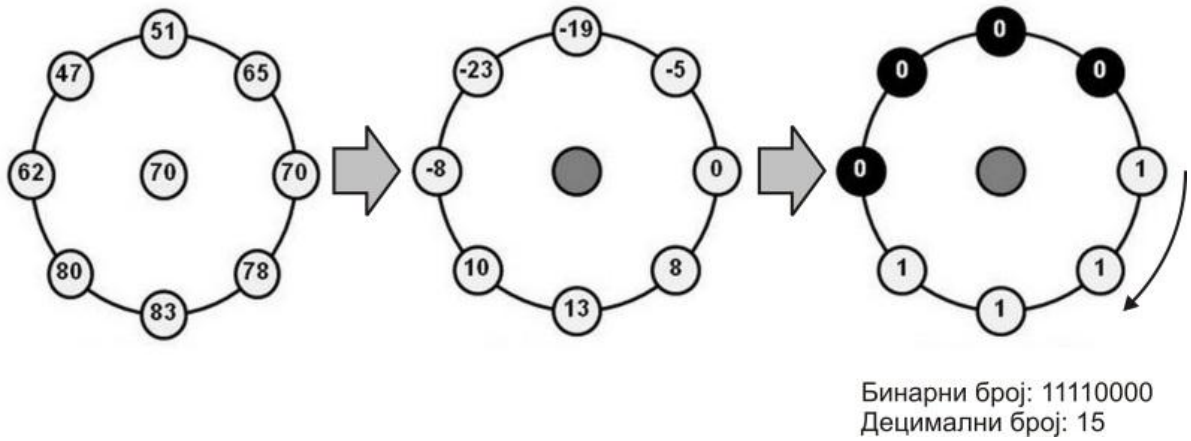
Хистограм 2^P различитих LBP вредности може бити коришћен као сет атрибута за обучававање класификационих модела. За LBP слику I_{LBP} израчунава се функција расподеле вероватноће:

$$D(i, R) = \frac{N(I_{LBP} = i)}{M} \tag{5.72}$$

где је $N(I_{LBP} = i)$ број појављивања вредности i унутар LBP слике I_{LBP}, а M укупан број израчунатих LBP вредности за задату величину суседства R.

Проблем код LBP методе може бити велики број атрибута који се добија (у случају да је $R = 3$ постоји $2^8 = 256$ атрибута). У циљу смамења броја атрибута уведен је појам униформних шаблона. LBP вредност је униформна ако њен бинарни вектор садржи највише две транзиције из 0 у 1 или обрнуто. Ојала је у својим истраживањима показао да је 90% LBP вредности униформно за $R = 3$ и $P = 8$, а 70% LBP вредности униформно за $R = 5$ и $P = 16$ [91].

У оквиру ове студије коришћено је 3 x 3 окружење суседа ($R = 3$ и $P = 8$) и коришћене су само униформне LBP вредности (све неуниформне вредности су сврстане унутар једног атрибута). На овај начин израчунато је 59 LBP атрибута f_{48} - f_{106}.



Слика 5-21: Пример израчунавања LBP вредности ($R = 3$, $P = 8$).

5.8. Класификациони модели, селекција атрибута и тестирање

Последња фаза CAD система предложеног у оквиру овог истраживања је класификација. Циљ класификације је успешно раздвајање лажно позитивних сумњивих регија и стварно позитивних маса издвојених у фази сегментације. Свака сумњива регија је у фази израчунавања атрибута описана са 106 нумеричких атрибута. Успешан класификациони алгоритам би требао у највећем броју случајева тачно класификовати сумњиву регију на основу израчунатих атрибута. Сумњиве регије се класификују у једну од две класе: тумор или нормално ткиво. За решавање овог

класификационог проблема у оквиру овог истраживања су обучавани и тестирани следећи класификациони алгоритми применом софтверског пакета WEKA:

1. Наивни Бајесов класификатор - NB,
2. Логистичка регресија - LOGR,
3. Метода потпорних вектора - SVM,
4. Алгоритам k најближих суседа - KNN,
5. Стабла одлучивања - DT,
6. Неуронска мрежа - вишеслојни перцептрон - MLP,
7. Алгоритам случајне шуме - RF.

Класификациони модели су тестирани употребом 10-струке унакрсне валидације. У циљу поређења резултата добијених тестирањем различитих класификационих алгоритама израчунавају се тачност, сензитивност и специфичност.

$$AC = \frac{TP + TN}{TP + FP + TN + FN} \tag{5.73}$$

$$SENS = \frac{TP}{TP + FN} \tag{5.74}$$

$$SPEC = \frac{TN}{TN + FP} \tag{5.75}$$

Додатно, у циљу поређења резултата различитих класификатора израчунава се и површина испод ROC (енг. *Receiver Operating Characteristics*) криве - AUC (енг. *Area Under ROC Curve*). ROC крива је графички приказ сензитивности (на ординати) и специфичности одузете од 1 (на апсциси) за различите вредности граничног скора бинарног класификатора. Идеалан класификатор има AUC вредност 1 када ROC крива садржи горњи леви угао (тачка у којој се спајају максималне вредности сензитивности и специфичности). Табела 5-7 приказује оцене класификатора према AUC вредностима.

Вредност AUC	**Оцена класификатора**
0.9 - 1.0	Одличан
0.8 - 0.9	Добар
0.7 - 0.8	Фер
0.6 - 0.7	Слаб
0.5 - 0.6	Лош

Табела 5-7: Оцена класификатора према AUC вредностима.

У оквиру ове студије, сегментацијом је креирана база која садржи 84 позитивна примера (сумњиве регије које су тумори) и 1233 негативна примера (сумњиве регије које су нормално ткиво). Свега нешто преко 6% примера припада класи позитивних па

је ова база података неуједначена (енг. *Imbalanced dataset*). Велика неуједначеност базе података може довести до лоше прецизности неких класификатора [92]. Проблем неуједначених база података је уобичајан код реалних проблема, као нпр. код анализе слика [93], класификације текста [94], у медицини [95], [96] итд. Када број примера већинске класе пуно надмашује број примера мањинске класе, неки класификатори теже игнорисању мањинске класе. Тачност као мера успешности класификације међутим, не узима ово у обзир. Проблем неуједначености базе података може се решавати применом различитих алгоритама. У оквиру овог истраживања употребљен је SMOTE (енг. *Synthetic Minority Oversampling Technique*) алгоритам [11]. Уместо уклањања примера већинске класе или додавања дупликата примера мањинске класе, SMOTE алгоритам генерише синтетичке примере употребом постојећих примера мањинске класе. Сваки синтетички пример је креиран употребом једног од примера мањинске класе и једног од његових најближих суседа, интерполацијом случајним тежинским вектором (који садржи вредност између 0 и 1). У свакој итерацији 10-струке унакрсне валидације, база података за обучавање се „балансира“ употребом SMOTE алгоритма.

Седам различитих класификатора је најпре тестирано употребом свих 106 атрибута. После тога, у циљу побољшања резултата, извршена је селекција 25 најзначајнијих атрибута комбинацијом mRMR, Relief и алгоритма селекције атрибута према информацијском добитку. Оцена атрибута се израчунава према следећем изразу:

$$RANK(i) = (a - rank_{mRMR}(i)) + \left(a - rank_{relief}(i)\right) + (a - rank_{IG}(i)) \qquad (5.76)$$

где је a број атрибута (106), $RANK(i)$ је укупна оцена i-тог атрибута, а $rank_{mRMR}(i)$, $rank_{relief}$ и $rank_{IG}(i)$ су редни бројеви i-тог атрибута унутар сортираног низа атрибута алгоритмима mRMR, Relief и селекцијом према информацијском добитку. Класификациони алгоритми су након тога тестирани употребом редукованог сета атрибута.

5.9. Резултати

5.9.1 Резултати тестирања класификационих модела

У оквиру ове студије, сегментацијом је креирана база која садржи 84 позитивна и 1233 негативна примера. Свака од ових сумњивих регија је описана са укупно 106 атрибута који су израчунати у фази израчунавања атрибута (поглавље 5.7). Укупно седам различитих класификационих алгоритама је тестирано са циљем успешног раздвајања позитивних (тумор) од негативних (нормално ткиво) примера. У циљу поређења резултата различитих класификационих алгоритама израчунати су тачност, сензитивност, специфичност и површина испод ROC криве - AUC. Постигнути резултати (добијени тестирањем 10-струком унакрсном валидацијом) заједно са детаљним описима класификационих модела приказани су у табели 5-8.

Опис модела	AC	$SENS$	$SPEC$	AUC
NB - Излаз се одређује на основу Бајесове теореме и тренинг података. Атибути су дикретизовани применом алгоритма дискретизације засноване на ентропији. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (8 пута је увећан број примера мањинске класе).	0.798	0.726	0.804	0.830
LOGR - алгоритам израчунава вероватноће припадности појединачним класама помоћу сигмоидалне функције и бира највећу вероватноћу као излаз. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (15 пута је увећан број примера мањинске класе).	0.839	0.833	0.839	0.892
SVM - Употребљена је хомогена полиномска кернелова функција првог степена како би се примери пресликали у простор F. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (15 пута је увећан број примера мањинске класе).	0.848	0.833	0.849	0.841
KNN - Излаз се одређује на основу 5 најближих суседа (k=5). У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (15 пута је увећан број примера мањинске класе).	0.832	0.655	0.845	0.814
DT - C4.5 алгоритам који у процесу избора атрибута користи информацијски добитак нормализован ентропијом атрибута (енг. *Gain ratio*). Скраћивање стабла је постигнуто применом алгоритма *reduced error pruning*. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (15 пута је увећан број примера мањинске класе).	0.877	0.667	0.892	0.835
RF - Излаз је доминантна класа међу предвиђањима 10 стабала ($N_{trees} = 10$). Број атрибута који се насумично бира за сваки чвор је 10 ($m_{try} = 10$). У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (20 пута је увећан број примера мањинске класе).	0.911	0.655	0.929	0.901
MLP - Вишеслојни перцептрон са 15 неурона у једном скривеном слоју, униполарним сигмоидалним активационим функцијама у свим неуронима. Неуронска мрежа је обучена алгоритмом са пропагацијом грешке уназад са моментом. Брзина учења и константа момента имају вредности 0.2. Критеријум за заустављање учења је дефинисан у виду 500 епоха учења. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (20 пута је увећан број примера мањинске класе).	0.878	0.726	0.889	0.902

Табела 5-8: Опис и резултати тестирања (тачност (AC), сензитивност ($SENS$), специфичност ($SPEC$) и површина испод ROC криве (AUC)) класификатора добијени употребом 106 атрибута.

Посматрајући табелу 5-8 може се приметити да је највећа тачност (0.911) постигнута применом алгоритма случајне шуме. Међутим сензитивност овог алгоритма је недовољних 0.655. Као релевантну меру за поређење класификатора користићемо површину испод ROC криве - AUC. Највећу AUC вредност постигла је

неуронска мрежа вишеслојни перцептрон - 0.902. Овај алгоритам је постигао сензитивност 0.726 и специфичност 0.889.

У циљу побољшања класификационих резултата извршена је селекција 25 најзначајнијих атрибута комбинацијом mRMR, Relief и алгоритма селекције атрибута према информацијском добитку (5.76). У табели 5-9 приказано је 25 селектованих атрибута.

Редни број атрибута	Група атрибута	Назив атрибута
f1	описна статистика	Минимум
f11	описна статистика	Величина
f13	GLCM	Контраст
f14	GLCM	Корелација
f17	GLCM	Различитост
f19	GLCM	Ентропија
f20	GLCM	Хомогеност
f26	GLCM	Варијанса разлике
f27	GLCM	Ентропија разлике
f28	GLCM	Информацијска мера корелације 1
f29	GLCM	Информацијска мера корелације 2
f30	GLCM	Нормализована инверзна разлика
f31	GLCM	Нормализована хомогеност
f33	GLDM	Други момент
f34	GLDM	Ентропија
f36	GLDM	Хомогеност
f37	GLRLM	Истакнутост кратких низова
f38	GLRLM	Истакнутост дугих низова
f39	GLRLM	Неуниформност нивоа сивог
f41	GLRLM	Неунифомност текстуре
f45	GLRLM	Истакнутост кратких низова високог нивоа сивог
f47	GLRLM	Истакнутост дугих низова високог нивоа сивог
f57	LBP	LBP 10
f85	LBP	LBP 38
f106	LBP	LBP 59 (неуниформни шаблони)

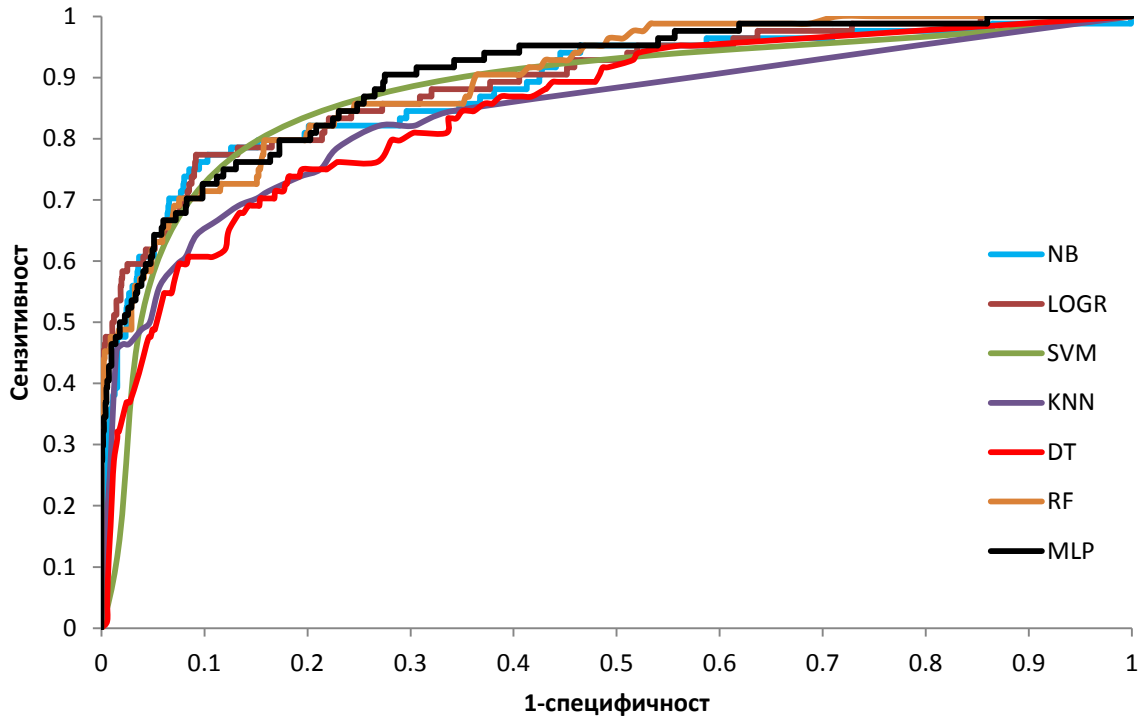
Табела 5-9: Листа 25 селектованих атрибута.

Седам класификационих алгоритама је поново обучавано и тестирано, али сада применом редукованог сета атрибута. Резултати тестирања 10-струком унаксрсном валидацијом као и детаљан опис алгоритама приказани су у табели 5-10.

Опис модела	AC	$SENS$	$SPEC$	AUC
NB - Излаз се одређује на основу Бајесове теореме и тренинг података. Атибути су дикретизовани применом алгоритма дискретизације засноване на ентропији. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (8 пута је увећан број примера мањинске класе).	0.844	0.798	0.848	0.885
LOGR - алгоритам израчунава вероватноће припадности појединачним класама помоћу сигмоидалне функције и бира највећу вероватноћу као излаз. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (18 пута је увећан број примера мањинске класе).	0.898	0.774	0.907	0.894
SVM - Употребљена је хомогена полиномска кернелова функција првог степена како би се примери пресликали у простор F. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (15 пута је увећан број примера мањинске класе).	0.845	0.798	0.848	0.823
KNN - Излаз се одређује на основу 7 најближих суседа (k=7). У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (15 пута је увећан број примера мањинске класе).	0.873	0.667	0.887	0.842
DT - C4.5 алгоритам који у процесу избора атрибута користи информацијски добитак нормализован ентропијом атрибута (енг. *Gain ratio*). Скраћивање стабла је постигнуто применом алгоритма *reduced error pruning*. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (15 пута је увећан број примера мањинске класе).	0.841	0.690	0.851	0.848
RF - Излаз је доминантна класа међу предвиђањима 500 стабала ($N_{trees} = 500$). Број атрибута који се насумично бира за сваки чвор је 5 ($m_{try} = 5$). У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (20 пута је увећан број примера мањинске класе).	0.914	0.679	0.930	0.898
MLP - Вишеслојни перцептрон са 6 неурона у једном скривеном слоју, униполарним сигмоидалним активационим функцијама у свим неуронима. Неуронска мрежа је обучена алгоритмом са пропагацијом грешке уназад са моментом. Брзина учења и константа момента имају вредности 0.3 односно 0.2. Критеријум за заустављање учења је дефинисан у виду 5000 епоха учења. У свакој итерацији 10-струке унакрсне валидације база података је балансирана применом алгоритма SMOTE (20 пута је увећан број примера мањинске класе).	0.874	0.774	0.881	0.902

Табела 5-10: Опис и резултати тестирања (тачност, сензитивност, специфичност и површина испод ROC криве – AUC) класификатора добијени употребом 25 селектованих атрибута.

Поређењем резултата из табела 5-8 и 5-10 можемо закључити да је 4 од 7 класификационих алгоритама показало побољшане резултате према AUC. Према резултатима из табеле 5-10 највећу тачност постигао је алгоритам случајне шуме (0.914), али ово није релевантно јер је база података високо неуједначена. Као и у случају комплетног сета атрибута за тренирање, најбољи резултат је постигнут неуронском мрежом вишеслојни перцептрон. Применом редукованог сета атрибута овај алгоритам је задржао највећу вредност AUC - 0.902. Постигнути разултати укључују сензитивност 0.774 (побољшано са претходних 0.726) и специфичност 0.881 (0.49 лажно позитивних по слици). ROC криве тестираних класификационих алгритама су приказане на слици 5-22.



Слика 5-22: ROC криве тестираних класификационих алгритама тренираних применом редукованог сета атрибута који садржи 25 атрибута.

5.9.2 Поузданост предвиђања

Као додатак класификацији, од великог значаја би била информација о томе колико је она поуздана тј. колико се лекари могу ослонити на њу. За ову сврху тестирана су три различита алгоритма поузданости предвиђања:

1. поузданост предвиђања базирана на најближим суседима - CNK,
2. поузданост предвиђања базирана на локалној унакрсној провери - LCV и
3. поузданост предвиђања базирана на густини - DENS.

У свакој итерацији 10-струке унакрсне валидације израчунате су CNK, LCV и DENS вредности као и класификациона грешка (Хелингерово растојање) изостављених примера. После завршене 10-стуке унакрсне валидације израчунат је Пирсонов коефицијент корелације између мера поузданости (CNK, LCV и DENS) и грешке.

Вредности коефицијената корелације за три тестиране мере поузданости предвиђања као и њихова статистичка значајност су приказане у табели 5-11.

Поузданост предвиђања	**Коефицијент корелације**	**Ниво значајности**
CNK	0.5714	<0.001
LCV	0.2873	<0.001
$DENS$	0.0374	Није статистички значајно

Табела 5-11: Коефицијенти корелације поузданости предвиђања и њихов ниво значајности.

Посматрањем табеле 5-11 може се закључити да су CNK и LCV мере поузданости предвиђања погодне за имплементацију унутар CAD система за детекцију тумора на дигитализованим мамографима.

6. Оптимизација математичких модела за симулацију настанка и развоја плака

6.1. Математички модели за симулацију атеросклерозе – студија на људима

У оквиру овог поглавља ће бити приказана два математичка модела за симулацију настанка и развоја плака код људи:

- модел базиран на обичним диференцијалним једначинама који описује еволуцију плака у времену (поглавље 6.1.2) и
- модел базиран на парцијалним диференцијалним једначинама који описује еволуцију плака у простору и времену (поглавље 6.1.3).

Оба модела имају своје параметре које је потребно одредити поступком оптимизације како би у што већој мери одговарали доступним експерименталним подацима (поглавље 6.1.1).

6.1.1 Експериментални подаци

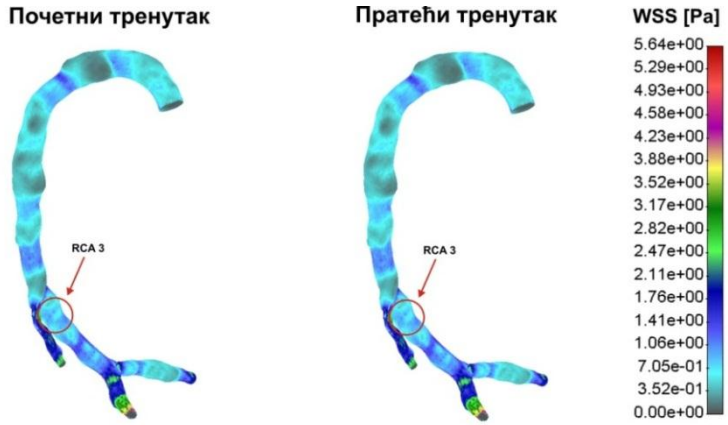
Експериментални подаци, који су неопходни за оптимизацију и валидацију математичких модела за симулацију атеросклерозе, обезбеђени су од CNR института за физиологију из Пизе (енг. *CNR Clinical Physiology Institute* - IFC*),* клиничког партнера на европском оквирном пројекту ARTreat [3]. Експериментални подаци садрже информације за три групе пацијената:

- пацијенти са новоформираним плаковима (3 пацијента – 4 плака),
- пацијенти са плаковима који су имали прогресију (3 пацијента – 6 плакова),
- контролни пацијенти који су имали плакове који нису имали даљу прогресију (4 пацијента – 7 плакова).

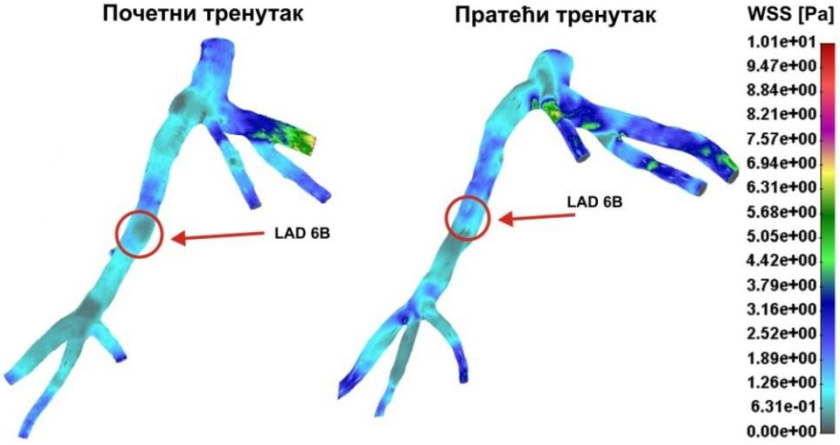
Информације о пацијентима су доступне у два временска тренутка: почетни T0 (енг. *Baseline*) и пратећи T1 (енг. *Follow up*) у периоду од 861-1596 дана након почетног тренутка у зависности од пацијента. Експериментални подаци садрже информације о концентрацијама следећих елемената у крви:

- интерћелијски адхезиони молекул (енг. *Intercellular Adhesion Molecule* - ICAM),
- васкуларни ћелијски адхезиони молекул (енг. *Vascular Cell Adhesion Molecule* - VCAM),
- липопротеин мале густине (енг. *Low Density Lipoprotein* - LDL),
- холестерол,
- липопротеин велике густине (енг. *High Density Lipoprotein* - HDL) и
- E-селектин.

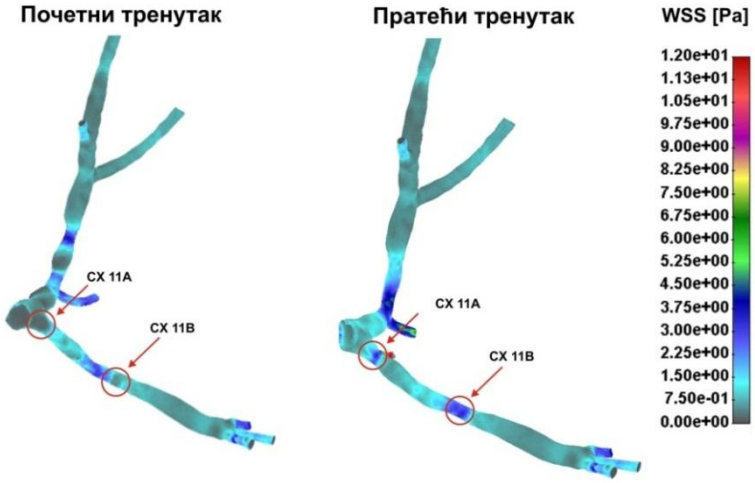
Поменути експериментални подаци су проширени вредностима смичућег напона на зиду који је израчунат за почетни и пратећи тренутак за све пацијенте употребом програма ПАК [62] (слике 6-1 - 6-10). Тродимензионална реконструкција коронарних артерија за оба временска тренутка је извршена на основу CT слика употребом DICOM[3] софтвера.



Слика 6-1: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 13.

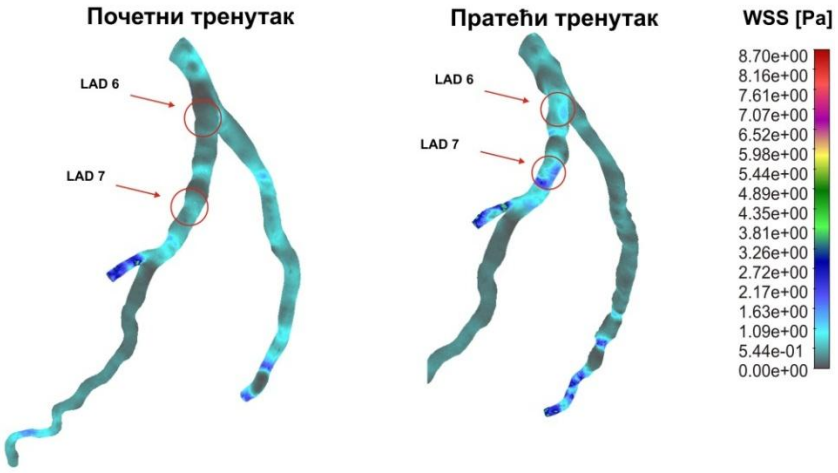


Слика 6-2: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 16.

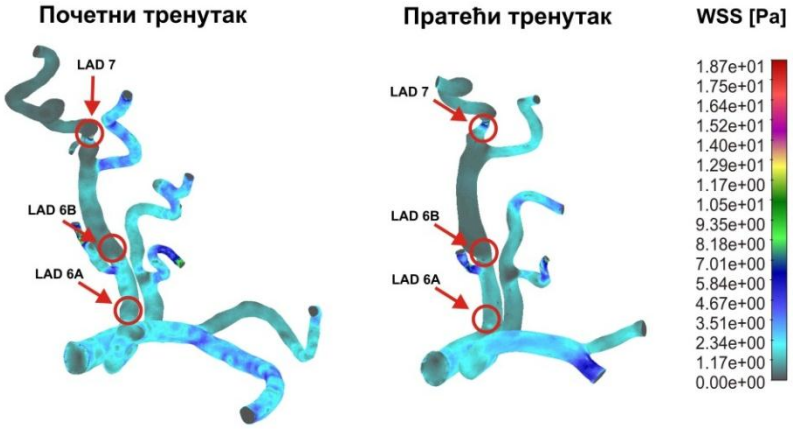


Слика 6-3: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 24.

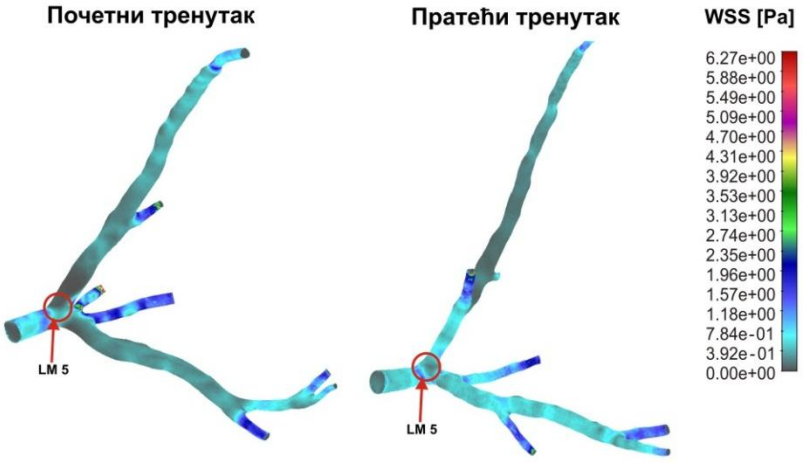
[3] Софтвер за тродимензионалну реконструкцију медицинских слика развијен у оквиру истраживачко развојсног центра за биоинжењеринг – БиоИРЦ у Крагујевцу.



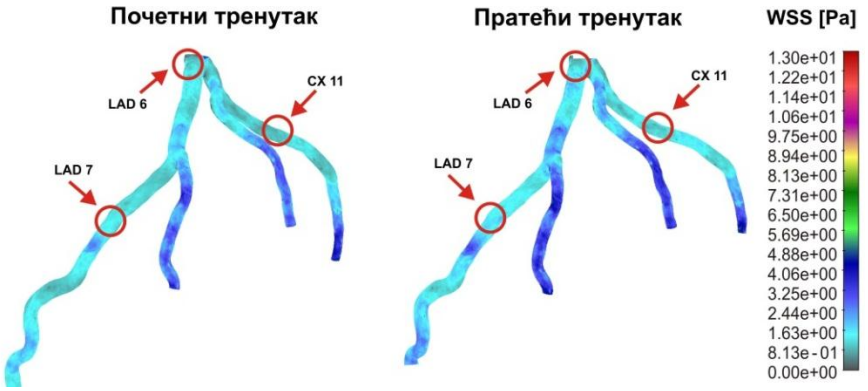
Слика 6-4: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 17.



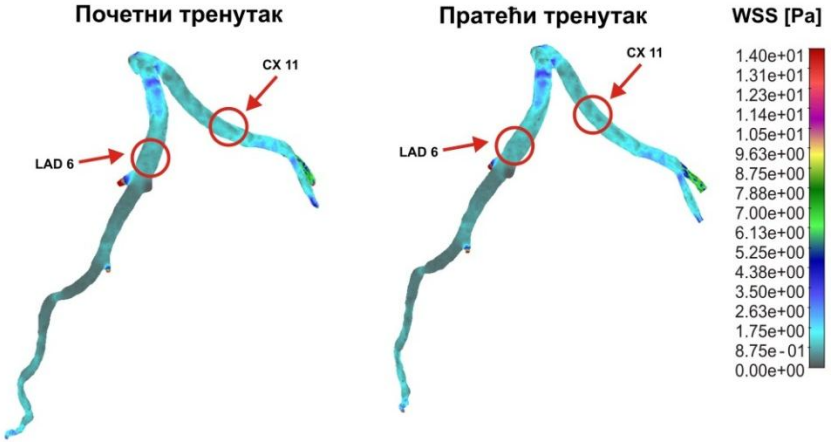
Слика 6-5: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 25.



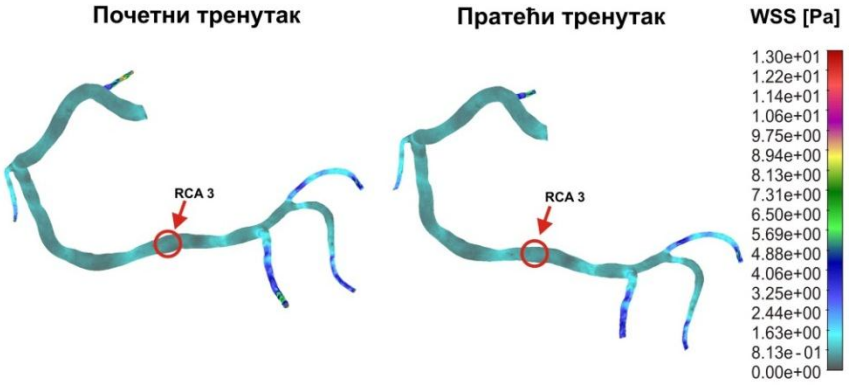
Слика 6-6: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 39.



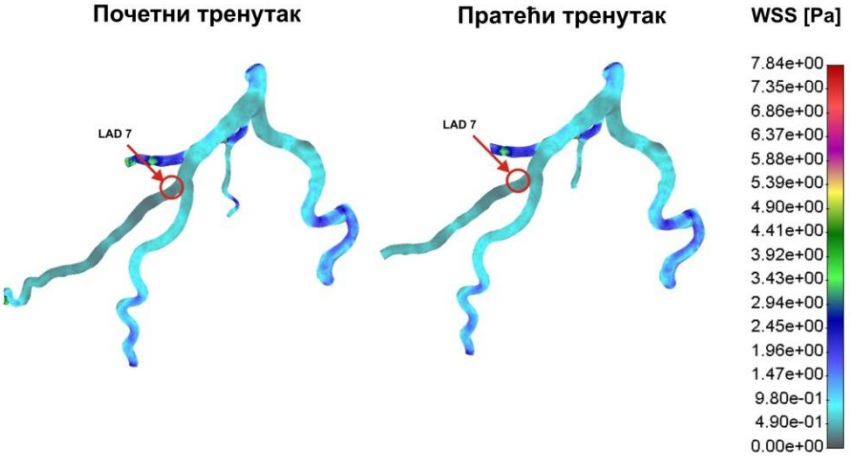
Слика 6-7: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 27.



Слика 6-8: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 28.



Слика 6-9: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 29.



Слика 6-10: Расподела смичућег напона на зиду (WSS) у почетном и пратећем тренутку за пацијента 30.

Експериментални подаци који садрже концентрације појединих елемената у крви (за почетни тренутак T0) и информације о величини плака (проценту сужења), проширени вредностима смичућег напона на зиду у почетном (WSS_{T0}) и пратећем тренутку (WSS_{T1}), су сумирани у табли 6-1. Вредности смичућег напона у овој табели представљају просечне вредности смичућег напона на зиду у пресецима у којима се развио плак.

Група	Ознака пацијента	Праћени тренутак [дани]	ICAM [ng/mL]	VCAM [ng/mL]	LDL [mg/dL]	Холестерол/HDL	E-селектин [ng/mL]	Ознака плака	WSS_{T0} [Pa]	WSS_{T1} [Pa]	Величина плака T0 (% сужења)	Величина плака T1 (% сужења)
Нови плакови	Пацијент 13	1078	401.7	682.3	148.4	7.07	72.2	RCA 3	0.256	0.485	0	16.88
	Пацијент 16	1036	255.0	529.5	60.0	4.56	31.3	LAD 6B	0.335	1.488	0	17.08
	Пацијент 24	1162	202.7	440.3	175.8	5.67	24.0	CX 11A	0.440	1.705	0	28.94
								CX 11B	0.355	1.635	0	29.79
Плакови са прогресијом	Пацијент 17	1015	314.6	651.5	221.0	6.66	19.1	LAD 6	0.585	1.079	0.07	1.46
								LAD 7	0.300	0.600	0.52	13.75
	Пацијент 25	1414	181.5	455.2	191.6	5.24	52.8	LAD 6A	1.290	1.662	0.89	2.27
								LAD 6B	1.999	2.129	1.03	4.51
								LAD 7	1.139	1.199	0.31	2.17
	Пацијент 39	861	194.4	498.7	64.0	2.22	46.1	LM 5	0.285	0.355	2.74	29.08
Контролни	Пацијент 27	1519	83.9	426.7	67.8	6.84	27.2	LAD 6	0.680	0.900	0.12	1.10
								LAD 7	0.351	1.223	0.88	1.57
								CX 11	0.622	0.822	0.05	0.47
	Пацијент 28	1190	133.2	450.2	135.0	3.91	26.5	LAD 6	0.989	1.305	0.08	0.43
								CX 11	1.096	1.146	0.19	0.93
	Пацијент 29	1596	212.6	489.5	161.0	7.18	44.7	RCA 3	1.079	1.125	0.19	1.42
	Пацијент 30	1071	171.9	546.6	96.0	2.75	35.7	LAD 7	0.493	0.839	0.16	0.26

Табела 6-1: Експериментални подаци за три групе пацијената.

6.1.2 Симулација атеросклерозе употребом система обичних дифиренцијалних једначина

6.1.2.1 Опис модела

У овом поглављу ће бити описан математички модел за моделирање раста плака у времену који је базиран на обичној диференцијалној једначини (енг. *Ordinary Differential Equation* - ODE) и који је развијен у оквиру европског оквирног пројекта ARTreat [3]. Нека је Y величина плака (изражена као проценат сужења лумена) и нека је са t означено време, математичи модел се може описати следећом једначином:

$$\dot{Y}_i(t) = \left(\frac{f\left(ICAM^i(t_0), LDL^i(t_0), \frac{CHOLESTEROL^i(t_0)}{HDL^i(t_0)}, ESEL^i(t_0), VCAM^i(t_0) \right)}{wss(t)} \right) \cdot I \qquad (6.1)$$

где су $ICAM^i(t_0)$, $LDL^i(t_0)$, $CHOLESTEROL^i(t_0)/HDL^i(t_0)$, $ESEL^i(t_0)$ и $VCAM^i(t_0)$ концентрације одговарајућих елемената у крви i-тог пацијента у тренутку t_0, $wss(t)$ је просечна вредност смичућег напона на зиду у одређеном пресеку (пресеку у коме се развио плак) у тренутку t, а функције f и I су описане једначинама (6.2)-(6.3):

$$f = a_0 \cdot ICAM^i(t_0) + a_1 \cdot LDL^i(t_0) + a_2 \cdot \frac{CHOLESTEROL^i(t_0)}{HDL^i(t_0)} + a_3 \cdot ESEL^i(t_0) + a_4 \cdot VCAM^i(t_0) + a_5 \qquad (6.2)$$

$$I = \begin{cases} 1 & Q(t) > a_6 \text{ и } f > 0 \\ 0 & \text{у супротном} \end{cases} \qquad (6.3)$$

где је $Q(t)$ количина LDL-а које је до тренутка t „ушла“ унутар зида крвног суда:

$$Q(t) = \int_0^t C_{intima}(s)ds \qquad (6.4)$$

$C_{intima}(s)$ је количина LDL-а која у тренутку s „улази“ унутар зида крвног суда:

$$C_{intima}(s) = P(s)\frac{CHOLESTEROL(t_0)}{HDL(t_0)}LDL(t_0) \qquad (6.5)$$

где је $P(s)$ пермеабилност зида крвног суда моделирана као:

$$P(s) = \left(a_7 \cdot log\left(1 + \frac{a_8}{wss(s) + a_9}\right)\right) \qquad (6.6)$$

а $wss(s)$ је вредност смичућег напона у тренутку s која се израчунава интерполацијом на основу вредности смичућег напона у почетном (T0) и пратећем тренутку (T1).

Према ODE моделу описаном једначинама (6.1)-(6.6), раст плака почиње када количина LDL-а која је „ушла“ унутар зида крвног суда пређе неки праг дефинисан параметром a_6 (под условом да функција f има позитивну вредност). Коефицијенти a_0, a_1,..., a_9 се одређују оптимизацијом према доступним експерименталним подацима о настанку и развоју плака код пацијената (поглавље 6.1.1).

6.1.2.2 Оптимизација модела

Оптимизација математичког модела описаног једначинама (6.1)-(6.6) се врши према експерименталним подацима доступним за 10 пацијената (17 плакова).

Коефицијенти a_0, a_1,..., a_9 ODE модела су одређени применом Nelder-Mead алгоритма оптимизације [16] према доступним подацима о настанку и развоју плака,

чија се величина изражава у облику процента сужења лумена. Коефицијенти су одређени минимизацијом следеће функције:

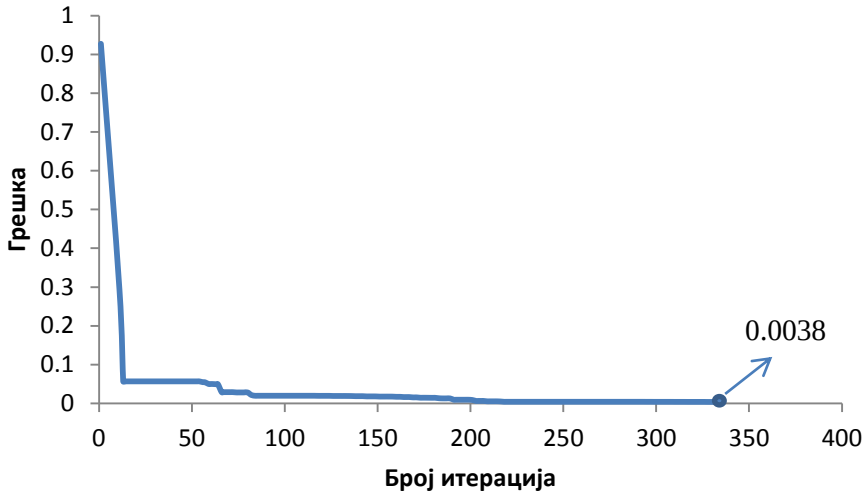
$$ERROR = \sum_{i=1}^{N} \left(Y_i - \hat{Y}_i\right)^2 \tag{6.7}$$

где је N укупан број плакова (17), Y_i је измерена величина плака у пратећем тренутку, а $\hat{Y}_i$ је величина плака у пратећем тренутку предвиђена ODE моделом. Оптимизоване вредности коефицијената a_0, a_1,..., a_9 су приказане у табели 6-2.

Коефицијент	**Оптимизована вредност**
$\boldsymbol{a_0}$	$6.33 \cdot e^{-3}$
$\boldsymbol{a_1}$	$5.40 \cdot e^{-5}$
$\boldsymbol{a_2}$	$-6.94 \cdot e^{-2}$
$\boldsymbol{a_3}$	$-6.63 \cdot e^{-3}$
$\boldsymbol{a_4}$	$-4.73 \cdot e^{-3}$
$\boldsymbol{a_5}$	1.74
$\boldsymbol{a_6}$	$6.38 \cdot e^{-13}$
$\boldsymbol{a_7}$	$4.89 \cdot e^{-7}$
$\boldsymbol{a_8}$	$1.49 \cdot e^{-8}$
$\boldsymbol{a_9}$	$-2.01 \cdot e^{-4}$

Табела 6-2: Оптималне вредности параметара ODE модела за симулацију раста плака (6.1)-(6.6).

Постигнута минимална вредност функције циља износи 0.0038, а сам поступак минимизације функције је приказан на слици 6-11.



Слика 6-11: Поступак минимизације функције циља (6.7).

Упоредни приказ експерименталних и предвиђених вредности величине плака у пратећем временском тренутку дат је на слици 6-12.

Слика 6-12: Упоредни приказ експерименталних и предвиђених вредности величине плака за три групе пацијената.

Поређењем експерименталних и предвиђених вредности величине плака (слика 6-12), за све три групе пацијената, може се закључити да је ODE модел успешно оптимизован према доступним експерименталним подацима.

6.1.3 Симулација атеросклерозе употребом система парцијалних дифиренцијалних једначина

6.1.3.1 Опис модела

У овом поглављу ће бити приказан тродимензионални модел за симулацију настанка и развоја плака који је базиран на законима континуума [97] и развијен у оквиру европског оквирног пројекта ARTreat [3]. Нумерички модел, базиран на парцијалним диференцијалним једначинама (енг. *Partial Differential Equation* - PDE), развијен је на бази Сановог рада [98] и уграђен је у програмски пакет ПАК.

Водеће једначине за моделирање струјања су Навије-Стоксове једначине (6.8) и једначина континуитета (6.9):

$$-\mu\nabla^2 u_l + \rho(u_l \cdot \nabla)u_l + \nabla p_l = 0 \tag{6.8}$$

$$\nabla u_l = 0 \tag{6.9}$$

где је ρ густина флуида (0.00105 [gr/mm]), μ је динамичка вискозност (0.00367 [Pa·s]), u_l је брзина флуида у лумену и p_l је притисак.

Гранични услови су дефинисани задавањем параболоидног профила брзине и концентрације LDL-а на улазном попречном пресеку артерија. Трансфер масе у лумену се моделира конвективно-дифузионом једначином и спрегнут је са једначинама за моделирање струјања:

$$\nabla \cdot (-D_l \nabla c_l + c_l u_l) = 0 \tag{6.10}$$

где су c_l и D_l ($5 \cdot e^{-6}$ [mm^2/s], [99]) концентрација и коефицијент дифузије посматране супстанце у лумену.

Проток кроз зид артерије се моделира Дарсијевим законом (6.11) и једначином континуитета (6.12):

$$u_w - \nabla(p_w) = 0 \tag{6.11}$$

$$\nabla u_w = 0 \tag{6.12}$$

где је u_w брзина струјања кроз зид артерије и p_w притисак у зиду артерије.

Трансфер масе у зиду артерије се моделира конвективно-дифузионо-реакционом једначином:

$$\nabla \cdot (-D_w \nabla c_w + k c_w u_w) = r_w c_w \tag{6.13}$$

где су c_w и D_w концентрација и коефицијент дифузије посматране супстанце у зиду, k је коефицијент заостајања супстанце (енг. *Solute lag coefficient*) и r_w је коефицијент потрошње посматране супстанце (енг. *Consumption rate constant*).

Лумен и зид артерије су повезани Кедем-Качалски једначинама [99], [101] које дају изразе за флуксеве кроз ендотел:

$$J_v = L_p(\Delta p - \sigma_d \Delta \pi) \tag{6.14}$$

$$J_s = P\Delta c + (1 - \sigma_f) J_v \bar{c} \tag{6.15}$$

где је L_p хидрауличка кондуктивност, J_v је запремински флукс, J_s је флукс посматране супстанце (растворка), Δp је пад притиска кроз ендотел, $\Delta \pi$ је разлика онкотског притиска између површина ендотела (овај члан се занемарује), σ_d је осмотски коефицијент рефлексије, P је пермеабилност ендотела, Δc је разлика концентрације посматране супстанце између површина ендотела, σ_f је коефицијент рефлексије растворка и $\bar{c}$ је средња логаритамска концентрација растворка у ендотелу:

$$\bar{c} = \frac{c_l - c_r}{\ln c_l - \ln c_r} \tag{6.16}$$

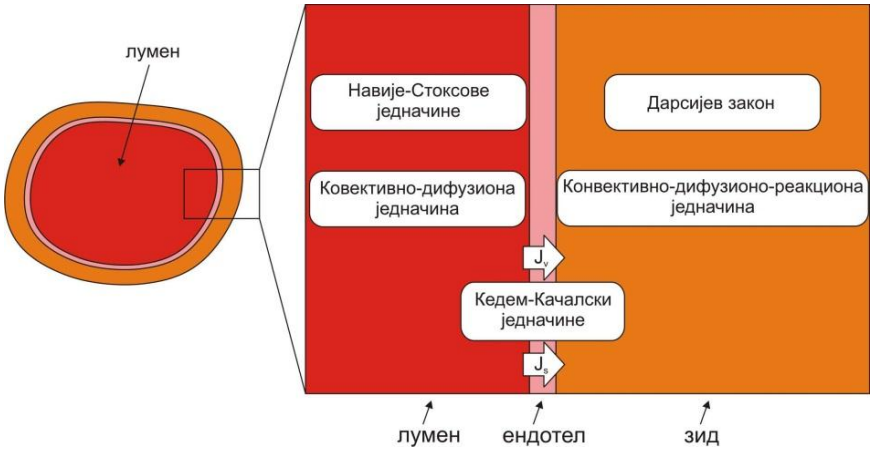
где су c_l и c_r концентрације супстанце са различитих страна ендотела.

Пермеабилност ендотела се моделира на следећи начин:

$$P = \frac{a_1 \cdot \sigma + a_2}{a_3 \cdot \tau + a_4} \tag{6.17}$$

где су a_1, a_2, a_3 и a_4 коефицијенти које је потребно одредити поступком оптимизације, τ је смичући напон на зиду, а σ је напон у солиду.

Шематски приказ једначина (6.8)-(6.15) је приказан на слици 6-13.



Слика 6-13: Шематски приказ једначина за моделирање лумена, ендотела и зида артерије.

Упални процес атеросклерозе се моделира помоћу три додатне парцијалне диференцијалне једначине:

$$\partial_t O = d_1 \Delta O - k_1 O M \tag{6.18}$$

$$\partial_t M + div(p_M v_w M) = d_2 \Delta M - k_1 O M + \frac{S}{1+S} \tag{6.19}$$

$$\partial_t S = d_3 \Delta S - \lambda S + k_1 O M + \gamma (O - O^{thr}) \tag{6.20}$$

где је O концентрација оксидисаног LDL-а унутар интиме (c_w у једначини (6.13)), M и S су концентрације макрофага и цитокина унутар интиме, d_1, d_2 и d_3 су одговарајући дифузиони коефицијенти, p_M је коефицијент померања плака, λ је коефицијент деградације цитокина, γ је коефицијент детекције LDL-а, k_1 је коефицијент раста плака, O^{thr} је праг активације цитокина и v_w је брзина раста плака која је дефинисана као:

$$div(v_w) = k_1 O M \tag{6.21}$$

Полазећи од почетног временског тренутка (T0) применом програма ПАК врши се симулација струјања крви и развоја атеросклерозе до пратећег временског тренутка (T1) решавањем PDE модела описаног у овом поглављу.

6.1.3.2 Оптимизација модела

PDE модел, описан једначинама (6.8)-(6.21), описује еволуцију плака (концентрације LDL-а, макрофага и цитокина) у простору и времену. Оптимизација овог модела се врши према доступним експерименталним подацима за сваког од десет пацијената понаособ. Циљ оптимизације је одређивање вредности коефицијената: k, r_w, L_p, σ_f, d_1, d_2, d_3, k_1, λ, γ, O^{thr}, p_M, a_1, a_2, a_3 и a_4 које доводе до минималних одступања између геометрије предвиђене PDE моделом и реалне геометрије у пратећем тренутку. Ова одступања се израчунавају за сваког пацијента у десет међусобно једнако удаљених пресека. Оптимизација је извршена применом хибридног генетског алгоритма који саджи две фазе:

1. проналажење зоне глобалног минимума применом генетског алгоритма [24] (25 генерација, величина популације 50) и
2. проналажење минимума применом Nelder-Mead алгоритма оптимизације [16].

Функција циља је дефинисана као:

$$ERROR = \sum_{i=1}^{N} \left(Y_i - \hat{Y}_i\right)^2 \tag{6.22}$$

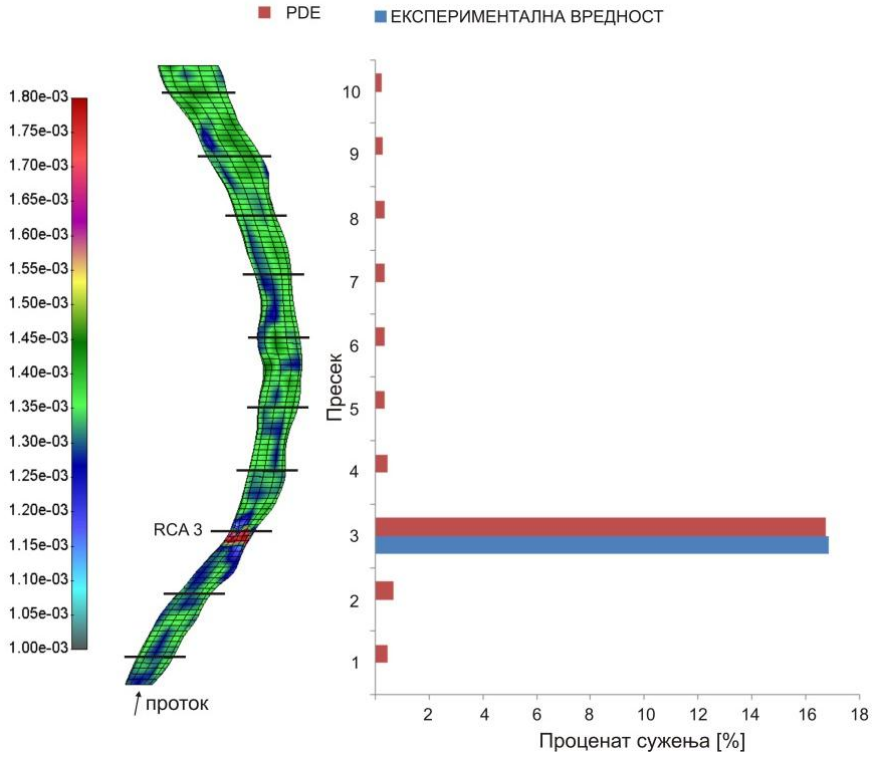
где је N укупан број пресека у којима се израчунава одступање (10), Y_i је експериментална вредност величине плака (проценат сужења) у пратећем тренутку и $\hat{Y}_i$ је величина плака у пратећем тренутку предвиђена PDE математичким моделом.

Оптимизоване вредности параметара PDE модела за сваког пацијента и одговарајуће просечне вредности су приказане у табели 6-3.

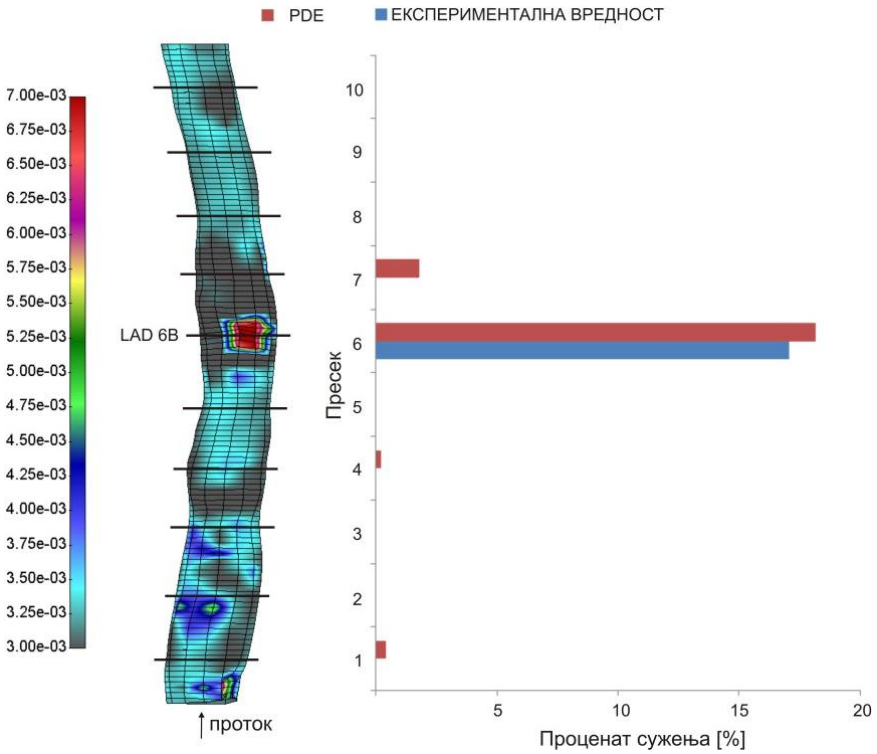
	Пацијент 13	Пацијент 16	Пацијент 24	Пацијент 17	Пацијент 25	Пацијент 39	Пацијент 27	Пацијент 28	Пацијент 29	Пацијент 30	Просечна вредностт
k	0.79	0.71	0.33	0.71	0.55	0.53	0.61	0.61	0.98	0.60	0.64
r_w	$-8.0\cdot e^{-4}$	$-7.6\cdot e^{-4}$	$-2.4\cdot e^{-4}$	$-7.6\cdot e^{-4}$	$-7.9\cdot e^{-4}$	$-1.2\cdot e^{-4}$	$-6.5\cdot e^{-4}$	$-6.9\cdot e^{-4}$	$-2.8\cdot e^{-4}$	$-6.4\cdot e^{-4}$	$-5.7\cdot e^{-4}$
L_p	$1.1\cdot e^{-9}$	$2.7\cdot e^{-9}$	$5.0\cdot e^{-9}$	$2.7\cdot e^{-9}$	$5.0\cdot e^{-9}$	$1.4\cdot e^{-9}$	$2.6\cdot e^{-9}$	$2.6\cdot e^{-9}$	$1.1\cdot e^{-9}$	$2.7\cdot e^{-9}$	$2.7\cdot e^{-9}$
σ_f	0.94	0.93	0.78	0.94	0.61	0.92	0.94	0.93	0.98	0.95	0.89
p_M	0.16	0.8	$4.1\cdot e^{-2}$	$4.6\cdot e^{-2}$	$1.2\cdot e^{-2}$	4.07	$2.5\cdot e^{-3}$	$3.3\cdot e^{-3}$	$5.6\cdot e^{-4}$	$2.8\cdot e^{-3}$	0.51
d_1	$3.4\cdot e^{-6}$	$5.0\cdot e^{-6}$	$4.9\cdot e^{-6}$	$5.0\cdot e^{-6}$	$5.0\cdot e^{-6}$	$5.0\cdot e^{-6}$	$4.9\cdot e^{-6}$	$3.5\cdot e^{-6}$	$4.0\cdot e^{-6}$	$4.5\cdot e^{-6}$	$4.5\cdot e^{-6}$
d_2	$1.8\cdot e^{-9}$	$2.0\cdot e^{-9}$	$1.2\cdot e^{-9}$	$1.9\cdot e^{-9}$	$9.1\cdot e^{-10}$	$1.9\cdot e^{-9}$	$1.6\cdot e^{-9}$	$1.7\cdot e^{-9}$	$2.4\cdot e^{-9}$	$1.5\cdot e^{-9}$	$1.7\cdot e^{-9}$
d_3	$4.5\cdot e^{-11}$	$1.0\cdot e^{-10}$	$9.2\cdot e^{-11}$	$1.0\cdot e^{-10}$	$1.0\cdot e^{-10}$	$1.0\cdot e^{-11}$	$9.9\cdot e^{-11}$	$9.8\cdot e^{-11}$	$2.2\cdot e^{-12}$	$9.8\cdot e^{-11}$	$7.5\cdot e^{-11}$
k_1	$4.5\cdot e^{-6}$	$1.0\cdot e^{-5}$	$1.9\cdot e^{-5}$	$9.6\cdot e^{-8}$	$5.4\cdot e^{-6}$	$6.2\cdot e^{-7}$	$6.3\cdot e^{-8}$	$9.9\cdot e^{-8}$	$1.6\cdot e^{-6}$	$8.2\cdot e^{-8}$	$4.2\cdot e^{-6}$
λ	$3.9\cdot e^{-2}$	0.38	0.70	0.67	$8.7\cdot e^{-3}$	0.44	0.69	0.67	0.47	0.68	0.48
γ	$1.2\cdot e^{-5}$	$5.5\cdot e^{-6}$	$4.8\cdot e^{-3}$	$5.5\cdot e^{-6}$	$9.8\cdot e^{-7}$	$5.8\cdot e^{-7}$	$5.1\cdot e^{-6}$	$5.1\cdot e^{-6}$	$1.8\cdot e^{-7}$	$5.2\cdot e^{-6}$	$4.9\cdot e^{-4}$
O^{thr}	$1.3\cdot e^{-2}$	0.49	0.69	0.49	$3.7\cdot e^{-3}$	$3.9\cdot e^{-3}$	0.26	0.41	0.23	$4.3\cdot e^{-2}$	0.26
a_1	$2.5\cdot e^{-10}$	$3.1\cdot e^{-10}$	$2.2\cdot e^{-12}$	$3.1\cdot e^{-8}$	$3.7\cdot e^{-10}$	$9.7\cdot e^{-11}$	$3.0\cdot e^{-7}$	$3.1\cdot e^{-9}$	$2.7\cdot e^{-9}$	$3.1\cdot e^{-9}$	$4.4\cdot e^{-9}$
a_2	1.04	0.85	0.56	2.00	-0.12	1.25	-0.63	-56.6	30.5	-60.7	-8.4
a_3	$-8.2\cdot e^{9}$	$-5.5\cdot e^{10}$	$2.0\cdot e^{6}$	$5.5\cdot e^{10}$	$-2.1\cdot e^{7}$	$2.4\cdot e^{8}$	$5.6\cdot e^{11}$	$5.6\cdot e^{9}$	$1.6\cdot e^{9}$	$5.5\cdot e^{9}$	$5.7\cdot e^{10}$
a_4	$4.0\cdot e^{9}$	$1.2\cdot e^{10}$	$2.2\cdot e^{8}$	$1.2\cdot e^{10}$	$5.7\cdot e^{8}$	$1.9\cdot e^{8}$	$1.2\cdot e^{12}$	$1.2\cdot e^{10}$	$1.3\cdot e^{10}$	$1.2\cdot e^{10}$	$1.3\cdot e^{11}$

Табела 6-3: Оптимизоване вредности параметара PDE модела.

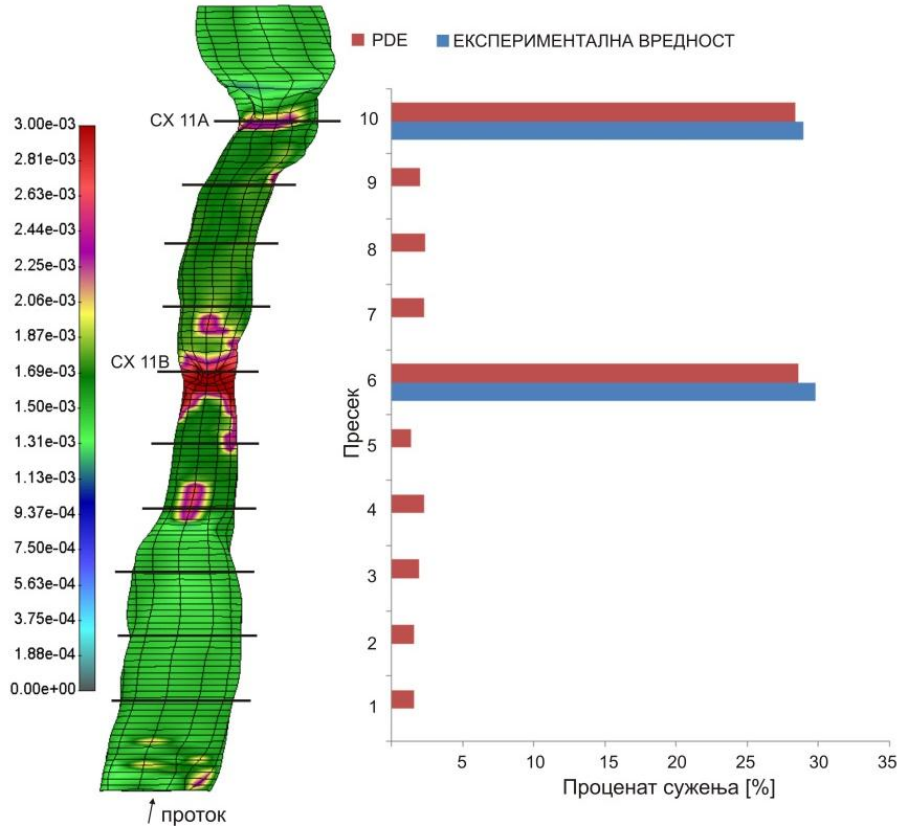
На сликама 6-14 – 6-23 приказано је поређење реалне геометрије и геометрије добијене симулацијом у пратећем тренутку за 10 пацијената.



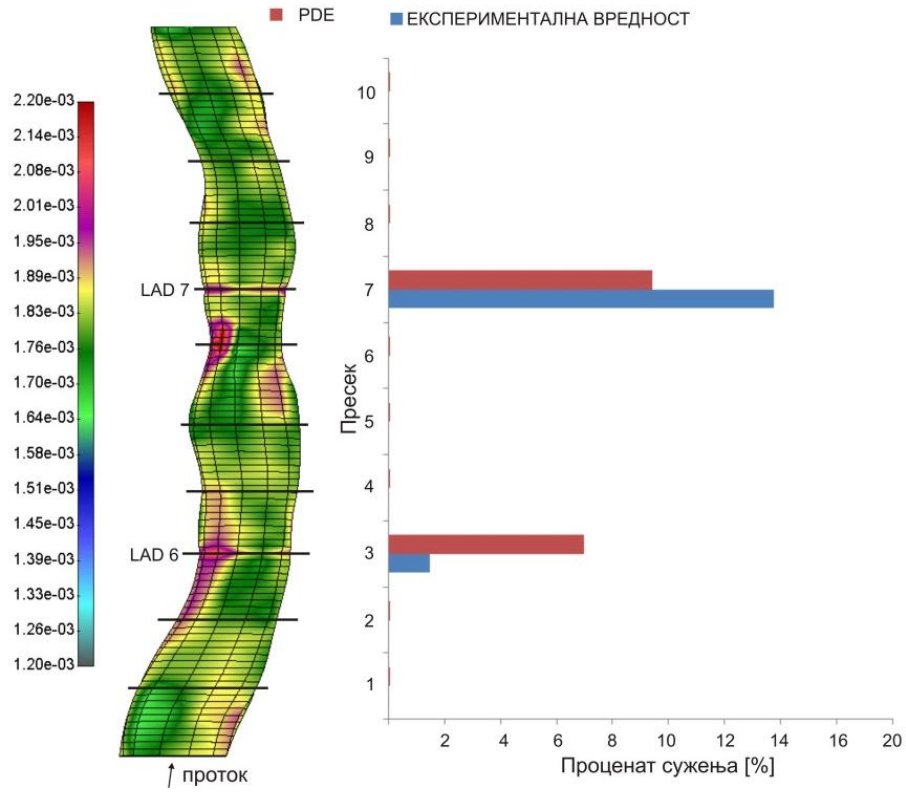
Слика 6-14: Резултати оптимизације за пацијента 13 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1).



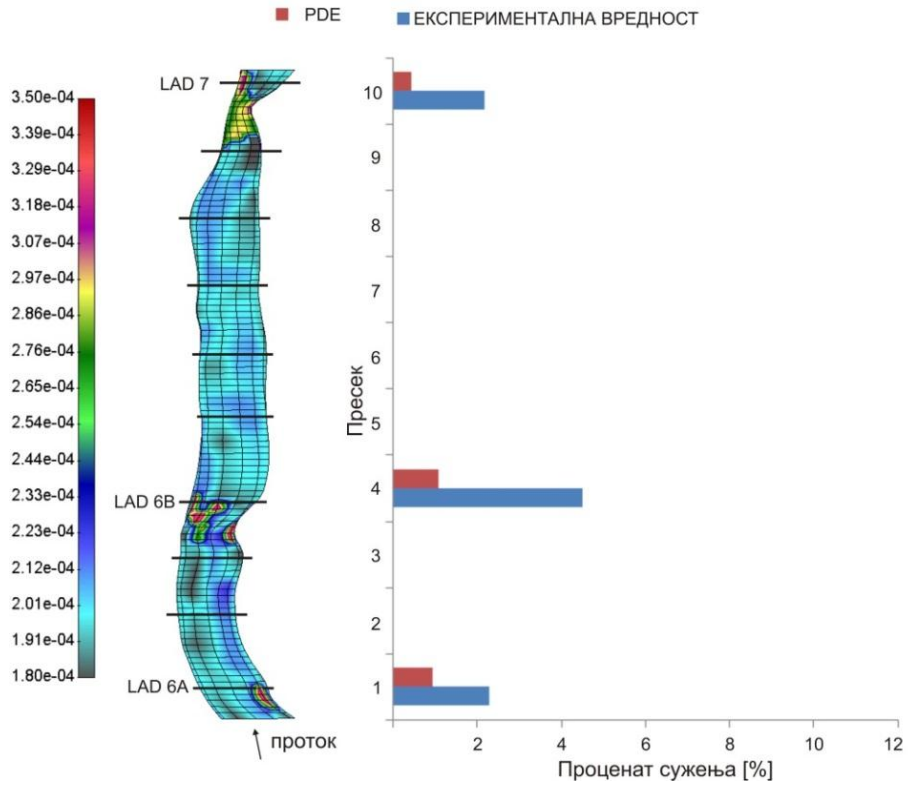
Слика 6-15: Резултати оптимизације за пацијента 16 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1).



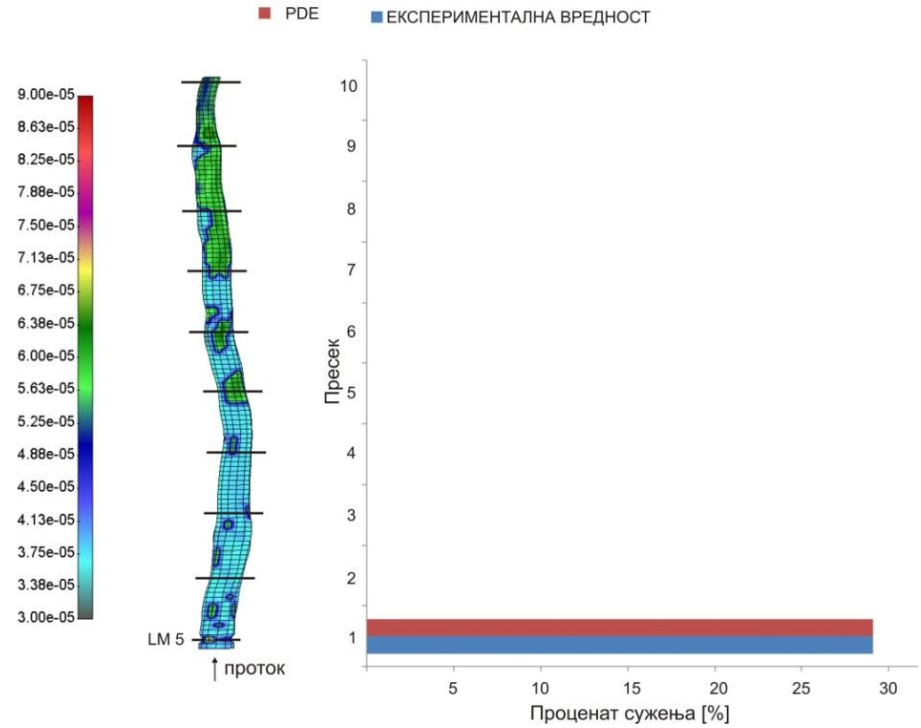
Слика 6-16: Резултати оптимизације за пацијента 24 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1).



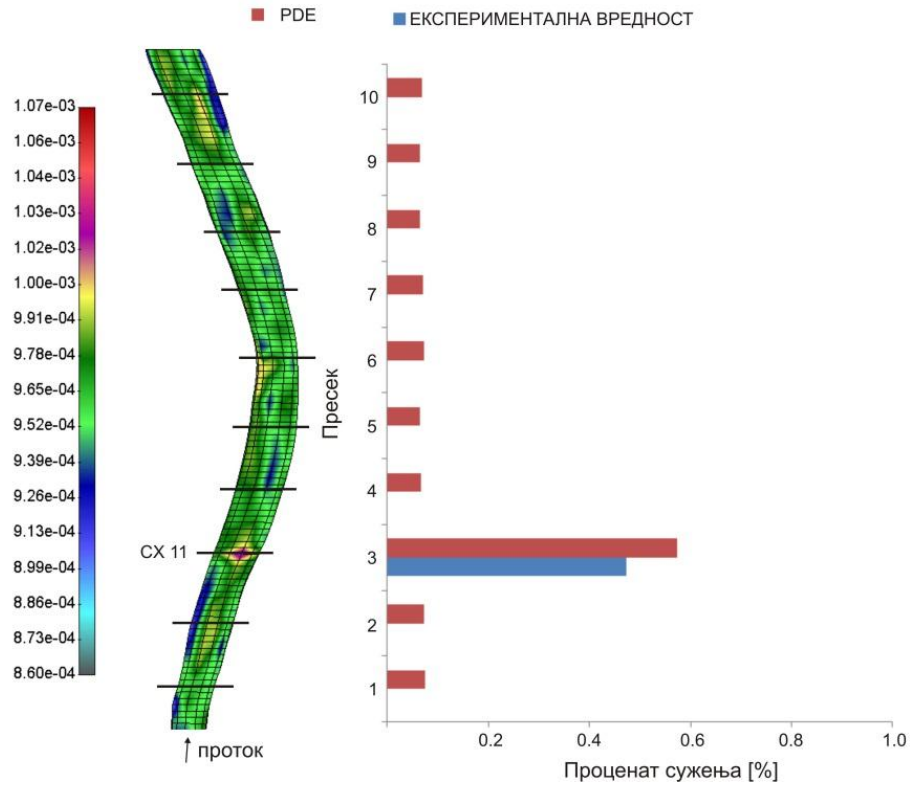
Слика 6-17: Резултати оптимизације за пацијента 17 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1).



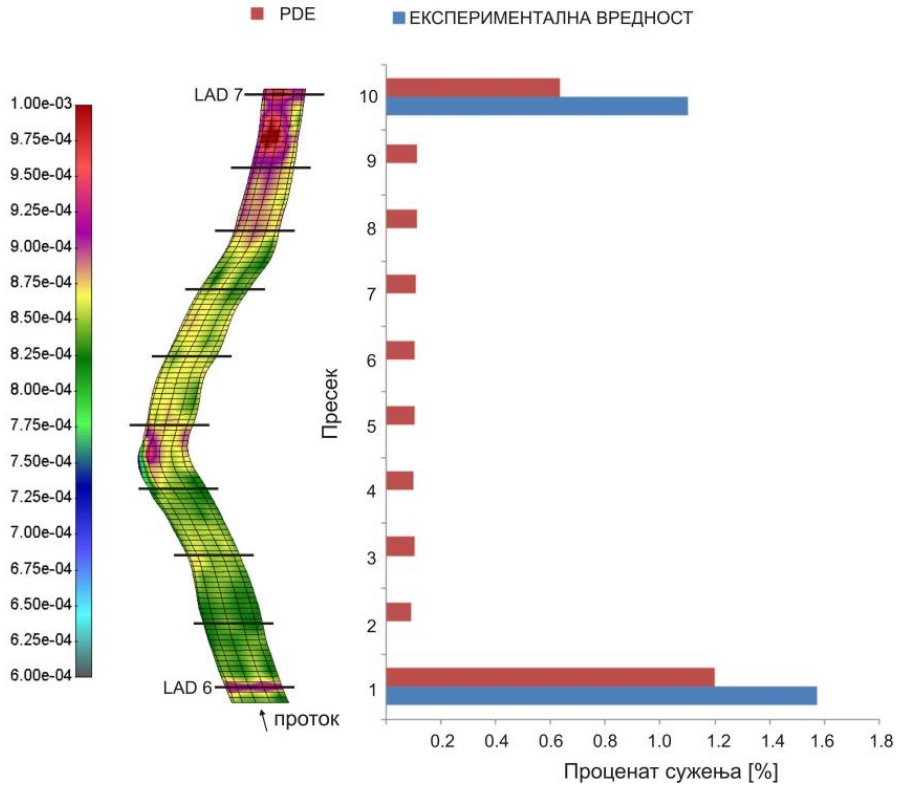
Слика 6-18: Резултати оптимизације за пацијента 25 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1).



Слика 6-19: Резултати оптимизације за пацијента 39 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1).

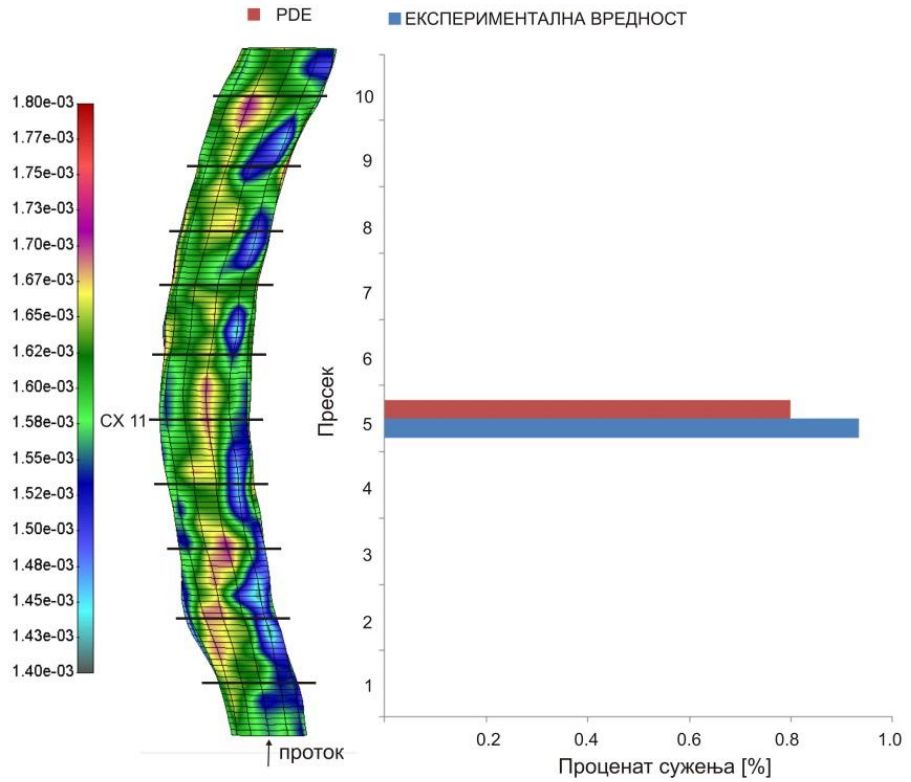


а)

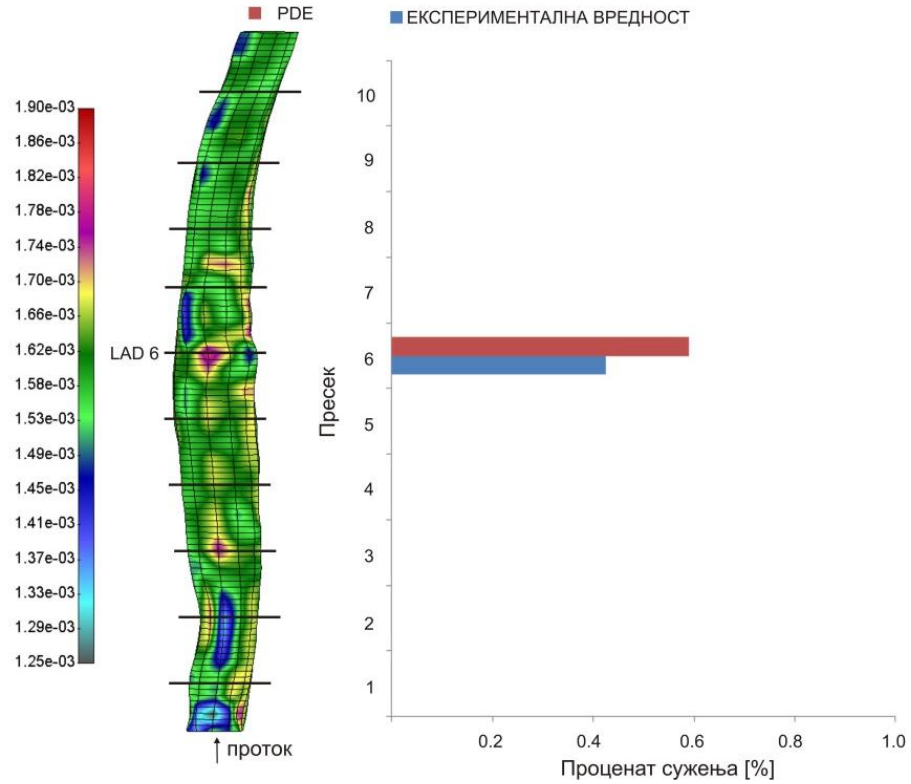


б)

Слика 6-20: Резултати оптимизације за пацијента 27 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1), а) циркумфлексна артерија, б) лева десцендентна артерија.

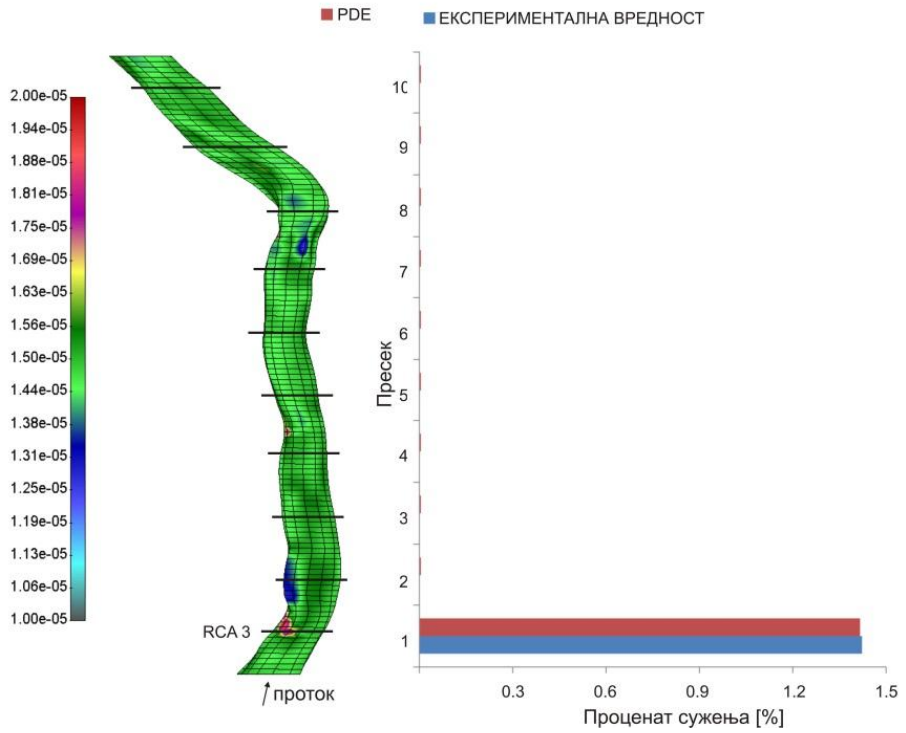


а)

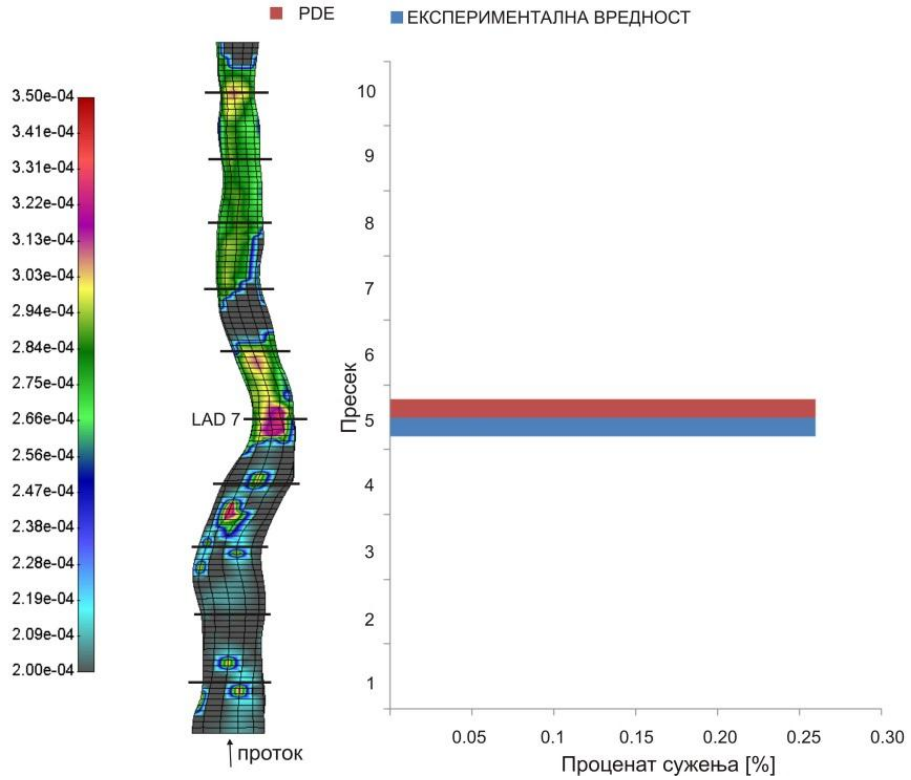


б)

Слика 6-21: Резултати оптимизације за пацијента 28 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1), а) циркумфлексна артерија, б) лева десцендентна артерија.



Слика 6-22: Резултати оптимизације за пацијента 29 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1).



Слика 6-23: Резултати оптимизације за пацијента 30 - одступање реалне геометрије у односу на геометрију предвиђену PDE моделом и расподела концентрације макрофага [g/mm^3] у пратећем тренутку (T1).

У циљу верификације PDE модела, оптимизоване вредности параметара (приказане у табели 6-3) су упоређене са вредностима из литературе (за доступне параметре). Упоредни приказ вредности параметара је дат у табели 6-4.

Параметар	**PDE модел (средња вредност за свих 10 пацијената)**	**Вредност из литературе**
k	0.642	0.686 [102]
r_w [1/s]	$-5.7\cdot e^{-4}$	$-1.40\cdot e^{-4}$ [103] $-2.76\cdot e^{-4}$ [102] $-6.05\cdot e^{-4}$ [98]
L_p [mm/(s·Pa)]	$2.69\cdot e^{-9}$	$3.00\cdot e^{-9}$ [104]
σ_f	0.892	0.99 [105]
d_1 [mm^2/s]	$4.52\cdot e^{-6}$	$3.50\cdot e^{-6}$ [98] $8.00\cdot e^{-7}$ [106]
d_2 [mm^2/s]	$1.72\cdot e^{-9}$	$1.00\cdot e^{-9}$ [107]
d_3 [mm^2/s]	$7.45\cdot e^{-11}$	Занемарљиво мали [108]

Табела 6-4: Поређење оптимизованих вредности параметара PDE модела са вредностима из литературе.

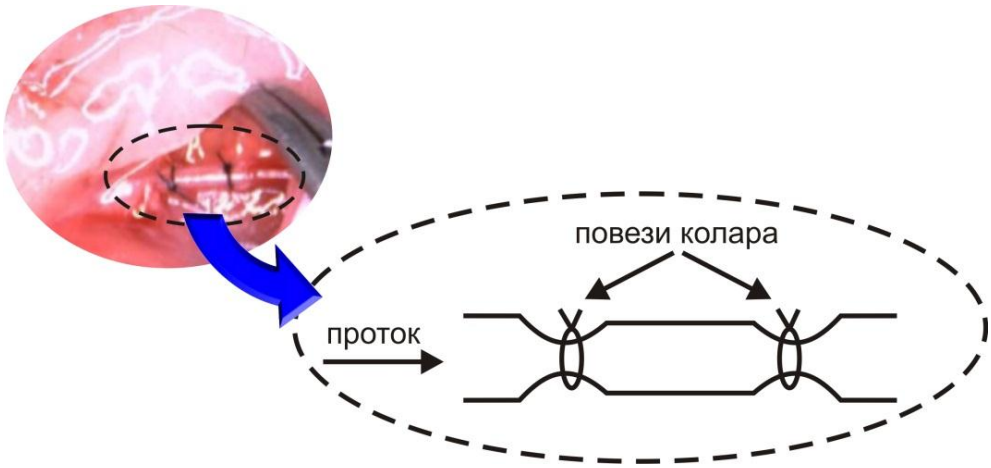
Посматрањем табеле 6-4 можемо закључити да оптимизоване вредности параметара k, r_w, L_p, σ_f, d_1, d_2 и d_3 имају исти ред величина као и вредности пронађене у литератури [98],[102]-[108]. Такође, посматрањем слика 6-14 – 6-23 можемо закључити да позиције и величине плакова предвиђене PDE моделом у највећем броју случајева одговарају експерименталним вредностима. Значајнија одступања се могу приметити једино код пацијената 17 и 25.

6.2. Оптимизација регресионих модела за предвиђање раста плака – студија на зечевима

У оквиру овог рада четири регресиона модела (вишеструка линеарна регресија, полиномска регресија, факторијска регресија и метода одзивне површине) су тестирана за моделирање прогресије плака као функције времена, резултата анализе крви (концентрација холестерола, HDL-а, LDL-а и триглицерида) и смичућег напона на зиду каротидне артерије. Експериментални подаци су добијени извођењем експеримената над зечевима на Кембриџ Универзитету у оквиру европског оквирног пројекта ARTreat, док је расподела смичућег напона на зиду израчуната симулацијама методом коначних елемената. Регресиони модели су оптимизовани тако да у што већој мери одговарају експерименталним резултатима прогресије плака добијеним хистолошком анализом. Резултати овог рада су презентовани на међународној научној конференцији у Косу 2013. године [109].

6.2.1 Експериментални подаци

У циљу добијања података прогресије атеросклерозе у времену 13 зечева је подвргнуто експерименту на Кембриџ Универзитету у оквиру европског оквирног пројекта ARTreat. Анализом крви су најпре добијени подаци о концентрацији холестерола, HDL-а, LDL-а и триглицерида. У циљу симулације стенозе 13 зечева је подвргнуто постављању колара око каротидне артерије (слика 6-24). Након тога зечеви су подвргнути атерогеној дијети и жртвовани након 8, 12 или 16 недеља. Хистололошком анализом је утврђена величина плака у зони проксимално од колара и на тај начин је одређена величина плака код жртвованих зечева у три различита временска тренутка.

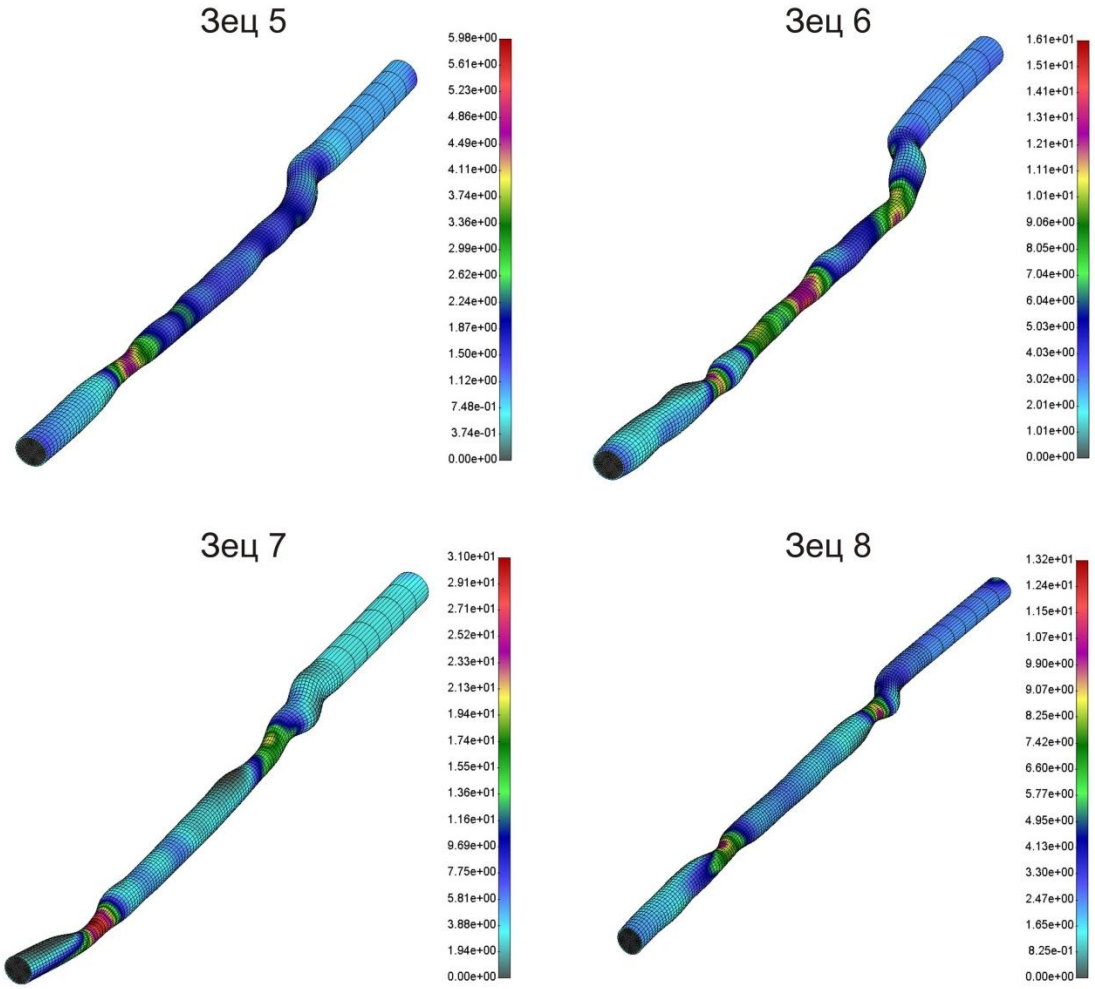


Слика 6-24: Симулација атеросклерозе применом колара.

6.2.2 FEM симулације струјања крви кроз каротидне артерије

Након постављања колара извршена је тродимензионална реконструкција каротидне артерије за сваког од 13 зечева на основу IVUS снимака. Употребом програмског пакета ПАК извршене су симулације струјања крви и израчунате су расподеле смичућег напона на зиду артерија (слика 6-25). Приликом симулација зид се третира као да је потпуно крут. Сви чворови на зиду и на улазном попречном пресеку су потпуно ограничени. Брзине су задате у свим чворовима на улазном попречном пресеку. Струјање је ламинарно, а флуид се третира као вискозан и нестишљив. Водеће једначине за моделирање струјања су Навије-Стоксове једначине (6.8) и једначина континуитета (6.9).

Посматрањем резултата може се закључити да све артерије имају у зони повеза колара сужење где је највећа вредност смичућег напона на зиду. Такође, зона између почетне и крајње тачке колара је више сужена него зоне дистално и проксимално од колара. Ово директно утиче на расподелу смичућег напона на зиду доводећи до тога да је најнижа вредност смичућег напона на зиду непосредно пре и после колара што ове зоне чини најризичнијим за настанак плака. У оквиру ове студије је посматрана зона непосредно пре колара (проксимално) и израчуната је средња вредност смичућег напона на зиду у пресеку који одговара хистолошкој анализи која се врши након жртвовања зечева и која пружа информацију о прогресији плака.



Слика 6-25: Расподела смичућег напона на зиду за зечеве 5, 6, 7 и 8.

6.2.3 Регресиони модели за предвиђање раста плака

За моделирање прогресије плака као функције смичућег напона на зиду, анализе крви (холестерол, HDL, LDL и триглицериди) и времена t употребљени су следећи регресиони модели:

- Вишеструка линеарна регресија (енг. *Multiple regression*):

$$plaque = a_0 + \sum_{i=1}^{6} a_i \cdot INPUT_i \tag{6.23}$$

- Полиномска регресија другог реда (енг. *Second order polynomial regression*):

$$plaque = a_0 + \sum_{i=1}^{6} \left(a_i \cdot INPUT_i + b_i \cdot {INPUT_i}^2\right) \tag{6.24}$$

- Факторијска регресија (енг. *Factorial regression*):

$$plaque = a_0 + \sum_{i=1}^{6} a_i \cdot INPUT_i + \sum_{i=1}^{6}\sum_{j=1}^{6}\left(c_{i,j} \cdot INPUT_i \cdot INPUT_j\right);\ \forall i \leq j : c_{i,j} = 0 \tag{6.25}$$

- Метода одзивне површине (енг. *Response surface regression*):

$$plaque = a_0 + \sum_{i=1}^{6} a_i \cdot INPUT_i + \sum_{i=1}^{6} b_i \cdot {INPUT_i}^2 + \sum_{i=1}^{6}\sum_{j=1}^{6}\left(c_{i,j} \cdot INPUT_i \cdot INPUT_j\right);\ \forall i \le j : c_{i,j} = 0 \quad (6.26)$$

где су a_i, b_i и $c_{i,j}$ $(i = 1, \dots, 6,\ j = 1, \dots, 6)$ коефицијенти који се одређују поступком оптимизације. Вектор $INPUT$ је:

$$INPUT = \begin{bmatrix} t \\ WSS \\ Cholesterol \\ HDL \\ LDL \\ Triglycerides \end{bmatrix}$$

где је t време, WSS средња вредност смичућег напона на зиду у пресеку непосредно испред колара (проксимално), а $Cholesterol$, HDL, LDL и $Triglycerides$ концентрације измерене у крви на почетку експеримента.

Параметри регресионих модела a_i, b_i и $c_{i,j}$ су одређени Nelder-Mead оптимизацијом употребом података приказаних у табели 6-5. Као мерило величине плака употребљен је однос $\frac{A_{plaque\_endothelial}}{A_{wall}}$ (слика 6-26) где је $A_{plaque\_endothelial}$ површина плака унутар ендотела, и A_{wall} површина зида (интима, медија и ендотел).

Ознака зеца	**Време [недеља]**	**WSS [Pa]**	**Холестерол [mmol/L]**	**HDL [mmol/L]**	**LDL [mmol/L]**	**Триглицериди [mmol/L]**	$\frac{A_{endothelial}}{A_{intima}+A_{media}}$ **[%]**
3	16	2.28	8.1	0.98	6.6	1.1	8.635
4	16	3.20	21.8	1.47	17.8	5.6	3.995
5	16	2.14	21.5	0.90	20	1.3	12.05
6	16	5.65	17.5	0.91	7	1.1	2.87
7	16	5.44	20.2	1.10	17.7	2.9	3.52
8	16	3.64	17.2	0.80	16	0.8	11.56
9	16	2.58	14.9	0.78	13.9	0.5	7.49
14	12	3.39	8.4	0.91	7.3	0.5	5.63
15	12	3.30	12	1.32	10.3	0.8	7.4
16	12	5.79	34.6	0.70	32.4	3.2	14.195
17	8	2.09	6.9	0.70	6	0.4	7.96
18	8	2.04	4.5	1.13	3.1	0.5	7.51
19	8	2.47	7.5	0.87	6.4	0.5	7.62

Табела 6-5: Подаци за оптимизацију регресионих модела.

Слика 6-26: Хистолошки пресек зеца 16 са означеним плаком унутар ендотела (зелена боја) и границом зида артерије (плава боја).

6.2.4 Резултати

Параметри регресионих модела (a_i, b_i и $c_{i,j}$, $i = 1, \dots, 6$, $j = 1, \dots, 6$) су одређени Nelder-Mead оптимизацијом са циљем што прецизнијег предвиђања раста плака на основу смичућег напона на зиду, анализе крви (холестерол, HDL, LDL и триглицериди) и времена t. Регресиони модели су тестирани применом методе изостављања једног примера (LOOCV) и израчунавањем релативне средње квадратне грешке (RMSE):

$$RMSE = \frac{(p_1 - t_1)^2 + \cdots + (p_n - t_n)^2}{(t_1 - \bar{t})^2 + \cdots + (t_n - \bar{t})^2} \tag{6.27}$$

где је n број зечева, p_i је предвиђена вредност величине плака i-тог зеца, t_i је стварна вредност величине плака i-тог зеца, а $\bar{t}$ је средња вредност величине плака израчуната као:

$$\bar{t} = \frac{1}{n}\sum_{i=1}^{n} t_i \tag{6.28}$$

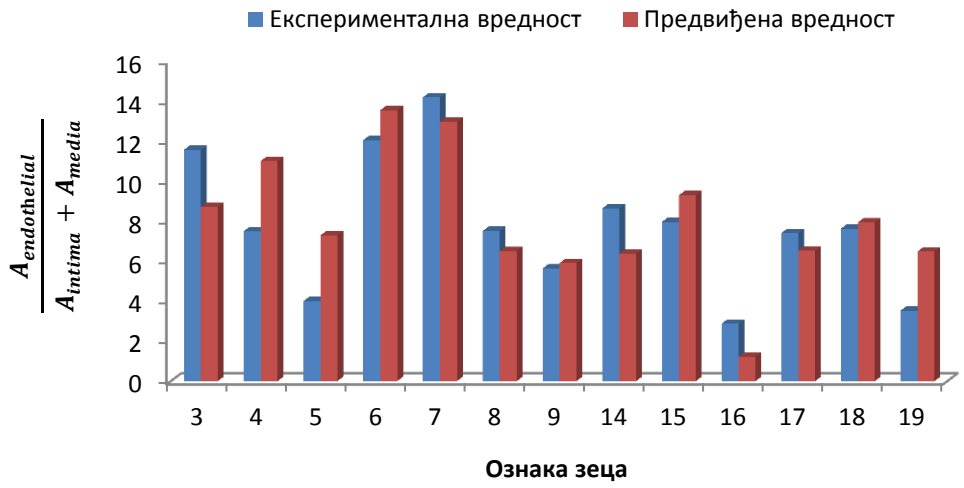
Резултати тестирања (релативна средња квадратна грешка и коефицијент корелације) четири предложена регресиона модела су приказани у табели 6-6.

Регресиони модел	RMSE	Коефицијент корелације - C
Вишеструка линеарна регресија	0.801	0.573
Полиномска регресија другог реда	0.408	0.792
Факторијска регресија	0.480	0.772
Метода одзивне површине	0.562	0.711

Табела 6-6: Резултати тестирања регресионих модела за предвиђање раста плака.

Резултати тестирања показују да су сви модели употребљиви (RMSE<1), али да је најбољи резултат постигнут оптимизацијом полиномске регресије другог реда (RMSE=0.480, $C = 0.792$). Упоредни приказ експерименталних (добијених из

хистологије) и предвиђених (полиномском регресијом другог реда) вредности величине плака је дат на слици 6-27.



Слика 6-27: Поређење експерименталних и предвиђених (полиномском регресијом другог реда) вредности величине плака.

7. Закључна разматрања

У оквиру ове дисертације, применом алгоритама оптимизације и техника истраживања података, решавана су три различита проблема из области биомедицинског инжењеринга. Постигнути циљеви као и смернице за даљи развој за сваки од решаваних проблема ће детаљно бити описани у овом поглављу.

7.1. Постигнути циљеви

Повезивање података добијених из хемодинамичких симулација употребом техника истраживања података.

- Доказана је могућност употребе алгоритама техника истраживања података за аутоматско предвиђање максималне вредности смичућег напона на зиду за дату каротидну бифуркацију која се дефинише преко геометријских параметара.
- Применом методологије за објашњење модела за предвиђање обезбеђено је објашњење кориснику о знању које је модел стекао на основу базе података за учење.
- Применом методологије за објашњење индивидуалних предвиђања обезбеђено је објашњење кориснику о разлогу сваког индивидуалног предвиђања.
- Обезбеђена је мера поузданости сваког индивидуалног предвиђања вредности MWSS-а.
- Доказана је могућност употребе алгоритама техника истраживања података за аутоматско предвиђање позиције (координата) на каротидној бифуркацији где се MWSS појављује.
- Доказана је могућност употребе алгоритама техника истраживања података за предвиђање комплетне расподеле смичућег напона на зиду за моделе каротидне бифуркације и анеуризме.

Детекција канцера дојке на дигитализованим мамографима употребом техника истраживања података.

- Оптимизацијом предложене регресионе функције за детекцију сумњивих регија и осталих параметара претпроцесирања и сегментације постигнут је напредак у односу на резултате приказане у литератури.
- Доказана је могућност употребе алгоритама техника истраживања података за успешно раздвајање лажно позитивних сумњивих регија и стварно позитивних маса издвојених у фази сегментације CAD система.
- Доказана је могућност имплементације мера поузданости предвиђања (CNK и LCV) унутар CAD система за детекцију тумора на дигитализованим мамографима.

Оптимизација математичких модела за симулацију настанка и развоја плака.

- Успешно је оптимизован PDE модел за симулацију раста плака у простору и времену. Оптимизација је извршена на основу доступних експерименталних података за коронарне артерије десет пацијената.
- Успешно је оптимизован ODE модел за симулацију раста плака у времену. Оптимизација је извршена на основу доступних експерименталних података за коронарне артерије десет пацијената.
- Приказана је могућност употребе четири регресиона модела (вишеструка линеарна регресија, полиномска регресија, факторијска регресија, метода одзивне површине) за моделирање прогресије плака као функције времена, резултата анализе крви (концентрација холестерола, HDL-а, LDL-а и триглицерида) и смичућег напона на зиду каротидне артерије. Регресиони модели су оптимизовани према доступним експерименталним подацима за каротидне артерије 13 зечева.

7.2. Смернице за даља истраживања

Повезивање података добијених из хемодинамичких симулација употребом техника истраживања података.

Овом дисертацијом је доказана могућност употребе техника истраживања података у области хемодинамичких симулација. Решавани проблеми су се односили на предвиђање смичућег напона на зиду за моделе анеуризме и каротидне бифуркације на основу геометријских параметара.

Даљи ток истраживања би могао бити усмерен на тестирање алгоритама техника истраживања података за аутоматско предвиђање других величина од интереса као нпр. поља брзина или расподеле притисака. Потребна су тестирања и на другим моделима (нпр. аорта) као и на моделима са великим бројем чворова (за реалне пацијенте се примењују модели са веома густом мрежом).

Детекција канцера дојке на дигитализованим мамографима употребом техника истраживања података.

У оквиру ове дисертације развијен је CAD систем за детекцију тумора на дигитализованим мамографима на бази постојећих алгоритама за сегментацију и претпроцесирање. Унапређење је постигнуто увођењем регресионе функције за сегментацију потенцијалних тумора, оптимизацијом параметара претпроцесирања и сегментације и имплементацијом поузданости предвиђања.

Даљи развој CAD система за детекцију тумора на дигитализованим мамографима може бити усмерен на разликовање бенигних и малигних случајева. Потребно је извршити и додатна тестирања CAD система на другим доступним базама (нпр. DDSM[4], BCDR[5]). Коначно, када се установи да је предложени CAD систем спреман за

[4] http://marathon.csee.usf.edu/Mammography/Database.html
[5] http://bcdr.inegi.up.pt/

употребу потребно је креирати финалну апликацију са све корисничким интерфејсом коју би лекари у клиничким центрима могли користити као помоћ приликом прегледа.

Оптимизација математичких модела за симулацију настанка и развоја плака.

Поглавље 6 ове дисертације се односи на оптимизацију математичких модела за симулацију настанка и развоја плака. Оптимизовани су ODE и PDE модели према доступним експерименталним подацима за коронарне артерије 10 пацијената. Оба модела су показала висок потенцијал за решавање проблема симулације настанка и развоја плака. На тај начин је показана могућност употребе оваквих модела у клиничкој пракси. Овакви модели могу пружити увид у даљи развој болести и на тај начин помоћи лекарима приликом избора терапије. Додатно, употребом доступних експерименталних података за зечеве, приказана је и могућност употребе једноставних регресионих модела за моделирање прогресије плака као функције времена, резултата анализе крви и смичућег напона на зиду каротидне артерије

Приказани ODE модел је једноставан за решавање и његова оптимизација не захтева много времена. Недостатак овог модела је што не описује еволуцију плака у простору, већ само у времену. Са друге стране, PDE модел описује еволуцију плака у простору и времену и веома је сложен. За његово решавање је потребно користити неку од нумеричких метода (нпр. метод коначних елемената) које су често веома прорачунски захтевне па оптимизација оваквог модела захтева много времена. Будући развој може бити усмерен на паралелизацију процеса решавања и оптимизације PDE модела. Такође, неопходно је извршити тестирања и валидацију модела на додатним експерименталним подацима (по могућству експерименталним подацима који садрже више временских тачака).

Литература

[1] Mathers C.D., Boerma T., Fat D.M., *Global and regional causes of death*, British medical bulletin, Vol. 92, pp. 7-32, 2009.

[2] Kojic M., Filipovic N., Stojanovic B., Kojic N., *Computer Modeling in Bioengineering*, John Wiley & Sons Inc., 2008.

[3] *Multi-level patient-specific artery and atherogenesis model for outcome prediction, decision support treatment, and virtual hand-on training-ARTreat*, FP7-224297 for Large-scale Integrating Project (IP), september 2008 - january 2013.

[4] Bosnic Z., Kononenko I., *Estimation of individual prediction reliability using local sensitivity analysis*, Applied Intelligence, Vol. 29, No. 3, pp. 187-203, 2007.

[5] Bosnic Z., Kononenko I., *Comparison of approaches for estimating reliability of individual regression predictions*, Data & Knowledge Engineering, Vol. 67, No. 3, pp. 504-516, 2008.

[6] Štrumbelj E., Kononenko I., *An efficient explanation of individual classifications using game theory*, Journal of Machine Learning Research, Vol. 11, pp. 1-18, 2010.

[7] Peng H., Long F., Ding C., *Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 8, pp. 1226-1238, 2005.

[8] Ding C., Peng H., *Minimum redundancy feature selection from microarray gene expression data*, Journal of Bioinformatics and Computational Biology, Vol. 3, No. 2, pp. 185-205, 2005.

[9] Kira K., Rendell L., *A Practical Approach to Feature Selection*, Proc. Ninth Int'l Conf. Machine Learning, pp. 249-256, 1992.

[10] Kononenko I., *Estimating Attributes: Analysis and Extensions of Relief*, Proc. Seventh European Conf. Machine Learning, pp. 171-182, 1994.

[11] Chawla N.V., Bowyer K.W., Hall L.O., Kegelmeyer W.P., *SMOTE: Synthetic minority over-sampling technique*, Journal of Artificial Intelligence Research, Vol. 16, No. 1, pp. 321-357, 2002.

[12] Dantzig G.B., *Linear Programming and Extensions*, Princenton University Press, Princeton, New Yersey, 1963.

[13] Kuhn H.W., Tucker A.W., *Nonlinear Programming*, Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, Berkeley, California, 1951.

[14] Bellman R.E., *Dynamic Programming*, Princenton University Press, Princeton, New Yersey, 1957.

[15] Marjanovic N., *Optimizacija zupčastih prenosnika snage*, Univerzitet u Kragujevcu, Kragujevac, 2007.

[16] Nelder J.A., Mead R., *A simplex method for function minimization*, Computer Journal, Vol. 7, No. 4, pp. 308-313, 1965.

[17] Petrić J., *Nelinearno programiranje*, Univerzitet u Beogradu, Beograd, 1979.

[18] Press W.H., Flannery B.P., Tenkolsky S.A., Vetterling W.T., *Numerical Recipes in C - The Art of Scientific Computing*, Second Edition, Cambridge, Cambridge University Press, 1992.

[19] Ellen F., *Global optimization of lennard-jones atomic clusters*, Technical report, McMaster University, February 2002.

[20] *Global Optimization Toolbox User's Guide*, The MathWorks, Inc, http://www.mathworks.com/help/pdf_doc/gads/gads_tb.pdf.

[21] Holland J.H., *Adaptation in Natural and Artifcial Systems*, University of Michigan Press, Ann Arbor, 1975.

[22] De Jong K.A, *Analysis of the behavior of a class of genetic adaptive systems*, Ph.D. Dissertation, University of Michigan, Ann Arbor, 1975.

[23] Goldberg D.E, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Reading, MA: Addison-Wesley, 1989.

[24] Haupt R.L., Haupt S.E., *Practical Genetic Algorithms*, Second edition, John Wiley & Sons Inc., 2004.

[25] Wright A.H., *Genetic algorithms for real parameter optimization*, edited by G.J.E. Rawlins, Foundations of Genetic Algorithms, Morgan Kaufmann, pp. 205–218, 1991.

[26] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolution Programs*, 2nd ed., New York, Springer-Verlag, 1994.

[27] Alba E., Tomassini M., *Parallelism and evolutionary algorithms*, IEEE transactions on evolutionary computation, Vol. 6, No. 5, pp. 443-462, 2002.

[28] Nowostawski M., Poli R., *Parallel genetic algorithm taxonomy*, Proceedings of Third International Conference on Knowledge-based Intelligent Information Engineering Systems KES'99, Adelaide, South Australia, 1999.

[29] Watson J.P., *A performance assessment of modern niching methods for parameter optimization problems*, Proceedings of the Genetic and Evolutionary Computation Conference GECCO-1999, Orlando, Florida, USA, 1999.

[30] Gordon V.S., Whitley D., *Serial and parallel genetic algorithms as function optimizers*, Proceedings of the Fifth International Conference on Genetic Algorithms and their Application, San Mateo, California, USA, pp. 177-183, 1993.

[31] Holte, R.C., *Very Simple Classification, Rules Perform Well on Most Commonly Used Datasets*, Machine Learning, Vol. 11, pp. 63-90, 1993.

[32] Kerber, R., *ChiMerge: Discretization of Numeric Attributes*, X National Conf. on Artificial Intelligence American Association (AAAI92), USA, pp. 123-128, 1992.

[33] Kononenko I., Kukar M., *Machine learning and data mining*, Horwood Publishing Chichester, UK, 2007.

[34] Hall M.A., *Correlation-Based Feature Selection for Machine Learning*, PhD thesis, Dept. of Computer Science, Univ. of Waikato, Hamilton, New Zealand, 1999.

[35] Hall M.A., *Correlation-Based Feature Selection for Discrete and Numeric Class Machine Learning*, Proc. 17th Int'l Conf. Machine Learning (ICML2000), 2000.

[36] Aha D., Kibler D., *Instance-based learning algorithms*, Machine Learning, Vol. 6, No. 1, pp. 37-66, 1991.

[37] Fix E., Hodges J.L., *Discriminatory analysis, nonparametric discrimination: Consistency properties*, Technical Report 4, USAF School of Aviation Medicine, Randolph Field, Texas, 1951.

[38] Quinlan J.R., *Induction of Decision Trees*, Machine Learning, Vol. 1, No. 1, pp. 81-106, 1986.

[39] Breiman L., *Bagging predictors*, Machine Learning, Vol. 24, pp. 123-140, 1996.

[40] Rumelhart D.E., Hinton G.E., Williams, R.J., *Learning internal representations by error propagation*. In Rumelhart D.E. et al., eds., Parallel Distributed Processing, MIT Press, Cambridge, MA, pp. 318-362, 1986.

[41] Hudson M.B, Hagan M.T., Demuth H.B., *Neural Network Toolbox User's Guide*, The MathWorks, Inc,
http://www.mathworks.com/help/pdf_doc/nnet/nnet_ug.pdf.

[42] Vapnik V., Lerner A., *Pattern recognition using generalized portrait method*, Automation and Remote Control, Vol. 24, pp. 774-780, 1963.

[43] Boser B.E., Guyon I.M., Vapnik V.N., *A training algorithm for optimal margin classifiers*, 5th Annual ACM Workshop on COLT, pp. 144–152, ACM Press, Pittsburgh, PA, 1992.

[44] Drucker H., Burges C.J.C., Kaufman L., Smola A., Vapnik V., *Support vector regression machines*, In Mozer M., Jordan M., Petsche T., editors, Advances in Neural Information Processing Systems, Vol. 9, pp. 155-161, Cambridge, MA, MIT Press, 1997.

[45] Kohavi R., *A study of cross-validation and bootstrap for accuracy estimation and model selection*, in Proceedings of the 14 th International Joint Conference on Artificial Intelligence, Morgan Kaufmann, San Francisco, 1995.

[46] Alpaydin E., *Introduction to machine learning*, MIT Press, Cambridge, 2004.

[47] Malek A.M., Alper S.L., Izumo S, *Hemodynamic shear stress and its role in atherosclerosis*, Journal of the American Medical Association - JAMA, Vol. 282, No. 21, pp. 2035-2042, 1999.

[48] Koskinas K.C., Chatzizisis Y.S., Baker A.B., Edelman E.R., Stone P.H., Feldman C.L., *The role of low endothelial shear stress in the conversion of atherosclerotic lesions from stable to unstable plaque*, Current Opinion in Cardiology, Vol. 24, No. 6, pp. 580-590, 2009.

[49] Akram M.S, André J.D., *Wall Shear Stress and Early Atherosclerosis*, American Journal of Roentgenology - AJR, Vol. 174, No. 6, pp. 1657-1665, 2000.

[50] Haga M., Yamashita A., Paszkowiak J., Sumpio B.E., Dardik A., *Oscillatory shear stress increases smooth muscle cell proliferation and akt phosphorylation*, Journal of Vascular Surgery, Vol. 37, No. 6, pp. 1277-1284, 2003.

[51] Kroll M.H., Hellums J.D., Guo Z., Durante W., Razdan K., Hrbolich J.K., Schafer A.I., *Protein kinase C is activated in platelets subjected to pathological shear stres*, Journal of Biological Chemistry, Vol. 268, No. 5, pp. 3520-3524, 1993.

[52] Dolan J.M., Kolega J., Meng H., *High Wall Shear Stress and Spatial Gradients in Vascular Pathology: A Review*, Annals of Biomedical Engineering, 2012.

[53] Barnes M.E., Miyasaka Y., Seward J.B., Gersh B., Rosales A.G., Bailey K.R., Petty G.W., Wiebers D.O., Tsang T.S.M., *Left Atrial Volume in the Prediction of First*

Ischemic Stroke in an Elderly Cohort Without Atrial Fibrillation, Mayo Clinic Proceedings, Vol. 79, No. 8, pp. 1008-1014, 2004.

[54] Bharadvaj B.K., Mabon R.F., Giddens D.P., *Steady flow in a model of the human carotid bifurcation, Part I - Flow visualization*, Journal of Biomechanics, Vol. 15, No. 5, pp. 349-362, 1982.

[55] Lorthois S., Lagree P.Y., Marc-Vergnes J.P., Cassot F., *Maximal wall shear stress in arterial stenoses: Application to the internal carotid arteries*, ASME Journal of Biomechanical Engineering, Vol. 122, No. 6, pp. 661-666, 2000.

[56] Bosnic Z., Vracar P., Radovic M.D., Devedžic G., Filipovic N.D., Kononenko I., *Mining data from hemodynamic simulations for generating prediction and explanation models*, IEEE Trans Inf Technol Biomed., Vol. 16, No. 2, pp. 248-254, 2012.

[57] Schulz U.G.R., Rothwell P.M., *Sex differences in carotid bifurcation anatomy and the distribution of atherosclerotic plaque*, Vol. 32, No. 7, pp. 1525-1531, 2001.

[58] Peterson R.E., Livingston K.E., Escobar A., *Development and distribution of gross atherosclerotic lesions at cervical carotid bifurcation*, Neurology, Vol. 10, pp. 955-959, I960.

[59] Perktold K., Hilbert D., *Numerical simulation of pulsatile flow in a carotid bifurcation model*, Journal of Biomedical Engineering, Vol. 8, No. 3, pp. 193-199, 1986.

[60] Perktold K., Florian H., Hilbert D., *Analysis of pulsatile blood flow: A carotid siphon model*, Journal of Biomedical Engineering, Vol. 9, No. 1, pp. 46-53, 1987.

[61] Perktold K., Peter R.O., Resch M., Langs G., *Pulsatile non-Newtonian blood flow in three-dimensional carotid bifurcation models: A numerical study of flow phenomena under different bifur-cation angles*, Journal of Biomedical Engineering, Vol. 13, No. 6, pp. 507-515, 1991.

[62] Kojic M., Filipovic N., Slavkovic R., Zivkovic M., Grujovic N., PAK-FS - Finite Element Program for Fluid Flow and Fluid-Solid Interaction, University of Kragujevac and R&D Center for Bioengineering, Kragujevac, Serbia, 1998.

[63] Glagov S., Zarins C., Giddens D.P., Ku D.N., *Hemodynamics and atherosclerosis Insights and perspectives gained from studies of human arteries*, Archives of Pathology & Laboratory Medicine, Vol. 112, No 10, pp. 1018-1031, 1988.

[64] Kolachalama V.B., Bressloff N.W., Nair P.B., *Mining data from hemodynamic simulations via Bayesian emulation*, BioMedical Engineering OnLine, 2007.

[65] Shevade S.K., Keerthi S.S., Bhattacharyya C., Murthy K.R.K., *Improvements to the SMO Algorithm for SVM Regression*, IEEE Transactions on Neural Networks, Vol. 11, No. 5, pp. 1188-1193, 2000.

[66] Pevec D., Štrumbelj E., Kononenko I., *Evaluating Reliability of Single Classifications of Neural Networks*, Adaptive and Natural Computing Algorithms, 10th International Conference, ICANNGA 2011, Part I, Ljubljana, Slovenia, pp. 22-30, 2011.

[67] Štrumbelj E., Kononenko I., *A General Method for Visualizing and Explaining Black-Box Regression Models*, Adaptive and Natural Computing Algorithms, 10th International Conference, ICANNGA 2011, Part II, Ljubljana, Slovenia, Proceedings, pp. 21-30, 2011.

[68] Boussel L., Rayz V., McCulloch C., Martin A., Acevedo-Bolton G., Lawton M., Higashida R., Smith W.S., Young W., Saloner D., *Aneurysm growth occurs at region of low wall shear stress: Patientspecific correlation of hemodynamics and growth in a longitudinal study*, Stroke, Vol. 39, No. 11, pp. 2997-3002, 2008.

[69] Radovic M., Petrovic D., Filipovic N., *Mining Data from CFD Simulation for Aneurysm and Carotid Bifurcation Models*, 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society - EMBC'11, Boston, MA USA, Sept., 2011.

[70] Radovic M., Djokovic M., Peulic A., Filipovic N., *Application of Data Mining Techniques for Mammogram Classification*, 19th Congress of the European Society of Biomechanics (ESB2013), Patras, Greece, August 28-28, 2013.

[71] Radovic M., Djokovic M., Peulic A., Filipovic N., *Application of Data Mining Algorithms for Mammogram Classification*, 13 th IEEE International Conference on BioInformatics and BioEngineering, Chania, Greece, 10-13 November 2013.

[72] Doi K., *Computer-aided diagnosis: potential usefulness in diagnostic radiology and telemedicine*, in: Proceedings of National Forum 95, pp. 9-13, 1996.

[73] Cheng H.D., Shi X.J., Min R., Hu L.M., Cai X.P., Du H.N., *Approaches for automated detection and classification of masses in mammograms*, Pattern Recognition, Vol. 39, pp. 646-668, 2006.

[74] Nguyen V.D., Nguyen D.T., Nguyen T.D., Pham V.T., *An Automated Method to Segment and Classify Masses in Mammograms*, International Journal of Electrical and Computer Engineering, 2009.

[75] Nasseer M.B., Mohammed M.H., *Segmentation of Breast Masses in Digital Mammograms Using Adaptive Median Filtering and Texture Analysis*, International Journal of Recent Technology and Engineering (IJRTE), Vol. 2, No.1, pp. 39-43, 2013.

[76] Suckling J., Parker J., Dance D., *The Mammographic Image Analysis Society Digital Mammogram Database Exerpta Medica*, International Congress Series 1069, pp. 375-378, 1994.

[77] Stojic T., Reljin I., Reljin B., *Adaptation of multifractal analysis to segmentation of microcalcifications in digital mammograms*, Physica A, Vol. 367, pp. 494-508, 2006.

[78] Reljin B., Milosevic Z., Stojic T., Reljin I., *Computer aided system for segmentation and visualization of microcalcifications in digital mammograms*, Folia Histochemica et Cytobiologica, Vol. 47, No. 3, pp. 525-532, 2009.

[79] Lai S.M., Li X., Biscof W.F., *On techniques for detecting circumscribed masses in mammograms*, IEEE Trans. Med. Imaging, Vol. l8, No. 4, pp. 377-386, 1989.

[80] Eltonsy N.H., Tourassi G.D., Elmaghraby A.S., *A concentric morphology model for the detection of masses in mammography*. IEEE Trans Med Imaging, Vol 26, pp. 880-889, 2007.

[81] Kom G., Tiedeu A., Kom M., *Automated detection of masses in mammograms by local adaptive thresholding*, Comput Biol Med., Vol. 37, pp. 37-48, 2007.

[82] Petrick N., Chan H.P., Sahiner B., Wei D., *An adaptive density-weighted contrast enhancement filter for mammographic breast mass detection*, IEEE Trans Med Imaging, Vol. 15, pp. 59-67, 1996.

[83] Oliver A., Freixenet J., Martí J., Pérez E., Pont J., Denton E.R., *A review of automatic mass detection and segmentation in mammographic images*, Med Image Anal., Vol. 14, pp. 87-110, 2010.

[84] Domingues I., Kozegar E., Minaei B., Soryani M., *Assessment of a novel mass detection algorithm in mammograms*, Journal of Cancer Research and Therapeutics, Vol. 9, No. 4, pp. 592-600, 2013.

[85] Haralick R.M., Shanmugam K., Dinstein I., *Textural features for image classification*, IEEE Transactions on systems, man and cybernetics, Vol. 3, No. 6, pp. 610-621, 1973.

[86] Soh L.K., Tsatsoulis C., *Texture Analysis of SAR Sea Ice Imagery Using Gray Level Co-Occurrence Matrices*, IEEE Transactions on geoscience and remote sensing, Vol. 37, No. 2, pp. 780-795, 1999.

[87] Clausi D.A., *An analysis of co-occurrence texture statistics as a function of grey level quantization*, Canadian Journal of Remote Sensing, Vol. 28, No. 1, pp. 45-62, 2002.

[88] Weszka J.S., Dyer C.R., Rosenfeld A., *A comparative study of texture measures for terrain classification*, IEEE Trans. Syst., Man, Cybern., Vol. SMC-6, pp. 269-285, Apr. 1976.

[89] Galloway R.M.M., *Texture analysis using gray level run lengths*, Comput. Graphic. Image Processing, Vol. 4, pp. 172-179, 1975.

[90] Ojala T., Pietikäinen M., Harwood D., *A Comparative Study of Texture Measures with Classification Based on Feature Distributions*, Pattern Recognition, Vol 19, No.3, pp. 51-59, 1996.

[91] Ojala T., Pietikäinen M., Mäenpää T., *Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns*, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, No.7, pp. 971-987, 2002.

[92] Sun Y., Kamel M.S., Wong A.K.C., Wang Y., *Cost-sensitive boosting for classication of imbalanced data*, Pattern Recognition, Vol. 40, pp. 3358-3378, 2007.

[93] Kubat M., Holte R., Matwin S., *Machine learning for the detection of oil spills in satellite radar images*, Machine Learning, Vol. 30, No. 2-3, pp. 195-215, 1998.

[94] Zheng Z., Wu X., Srihari R., *Feature selection for text categorization on imbalanced data*, SIGKDD Explorations Newsletter, Vol. 6, No. 1, pp. 80-89, 2004.

[95] Namee M.B., Cunningham P., Byrne S., Corrigan O., *The problem of bias in training data in regression problems in medical decision support*, Artificial Intelligence in Medicine, Vol. 24, No. 1, 2002.

[96] Cohen G., Hilario M., Sax H., Hugonnet S., Geissbuhler A., *Learning from imbalanced data in surveillance of nosocomial infection*, Artificial Intelligence in Medicine, Vol. 37, No. 1, pp. 7-18, 2006.

[97] N. Filipovic, Z. Teng, M. Radovic, I. Saveljic, D. Fotiadis, O. Parodi, *Computer simulation of three-dimensional plaque formation and progression in the carotid artery*, Medical and Biological Engineering and Computing, Vol. 51, No. 6, pp. 607-616, 2013.

[98] Sun N., Wood N.B., Hughes A.D., Thom S.A.M., Xu X.Y., *Effects of transmural pressure and wall shear stress on LDL accumulation in the arterial wall: a numerical study using a multilayered model*, Am. J. Physiol. Heart. Circ. Physiol. Vol. 292, pp. 3148-3157, 2007.

[99] Curry F.E., *Mechanics and thermodynamics of transcapillary exchange. Handbook of physiology*. The cardiovascular system. Microcirculation. Bethesda, MD: American Physiological Society, 1984.

[100] Kedem O., Katchalsky A., *Thermodynamic analysis of the permeability of biological membranes to non-electrolytes*, Biochim. Biophys. Acta, Vol. 27, pp. 229-246, 1958.

[101] Kedem O., Katchalsky A., *A physical interpretation of the phenomenological coefficients of membrane permeability*, J. Gen. Physiol., Vol. 45, pp. 143-179, 1961.

[102] Prosi M., *Computer Simulation von Massetransportvorgangen in Arterien*, Ph.D. thesis, Technische Universitat Graz, 2003.

[103] Zunino P., *Mathematical and Numerical Modeling of Mass Transfer in the Vascular System*, Ph.D. thesis, Ecole Polytechnique Federale de Lausanne, 2002.

[104] Sun N., Wood N.B., Hughes A.D., Thom S.A.M., Xu X.Y., *Fluid-Wall Modelling of Mass Transfer in an Axisymmetric Stenosis: Effects of Shear-Dependent Transport Properties*, Annals of Biomedical Engineering, Vol. 34, No. 7, pp. 1119-1128, 2006.

[105] Karner G., Perktold K., Zehentner H.P.. *Computational modeling of macromolecule transport in the arterial wall*, Comput. Methods. Biomech. Biomed. Engin., Vol. 4., pp. 491-504, 2001.

[106] Prosi M., Zunino P., Perktold K., Quarteroni A.. *Mathematical and numerical models for transfer of low-density lipoproteins through the arterial walls: a new methodology for the model set up with applications to the study of disturbed lumenal flow*, J. Biomech. Vol. 38, pp. 903-917, 2005.

[107] Grajdeanu P.B., Schugart R.C., Friedman A., Valentine C., Agarwal A.K., Rovin B.H., *A mathematical model of venous neointimal hyperplasia formation*. Theor. Biol. Med. Model. Vol. 5, No. 2, 2008.

[108] Cilla M., Pena E., Martinez M.A.. *Mathematical modelling of atheroma plaque formation and development in coronary arteries*, J. R. Soc. Interface, Vol. 11, 2014.

[109] Radovic M., Milosevic Z., Nikolic D., Saveljic I., Obradovic M., Petrovic D., Zdravkovic N., Teng Z., Bird J., Filipovic N., *Modeling and Correlation of Plaque Size with Histological and Blood Analysis Data for Animal Rabbit Experiments*, The 3rd South-East European Conference on Computational Mechanics, Kos Island, Greece, June 12-14, 2013.