

УНИВЕРЗИТЕТ У БЕОГРАДУ

Милош Р. Симић

**ОПТИМИЗАЦИЈА АЛГОРИТАМСКИХ ТЕСТОВА  
НОРМАЛНОСТИ ЗАСНОВАНИХ НА  
НЕУРОНСКИМ МРЕЖАМА**

Докторска дисертација

Београд, 2024.

UNIVERSITY OF BELGRADE

Miloš R. Simić

**OPTIMIZATION OF ALGORITHMIC NORMALITY  
TESTS BASED ON NEURAL NETWORKS**

Doctoral Dissertation

Belgrade, 2024

Ментори:

др Зорица Станимировић, редовни професор  
Универзитет у Београду, Математички факултет

др Срђан Станковић, професор емеритус  
Универзитет у Београду, Електротехнички факултет

Чланови комисије:

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

др Вељко Јеремић, редовни професор  
Универзитет у Београду, Факултет организационих наука

др Стефан Мишковић, доцент  
Универзитет у Београду, Математички факултет

Датум одбране: \_\_\_\_\_

**Наслов докторске дисертације:** Оптимизација алгоритамских тестова нормалности заснованих на неуронским мрежама

**Сажетак:** У овој дисертацији разматран је алгоритамски приступ тестирању нормалности расподела. Предложена је методологија за конструкцију неуронских мрежа за испитивање нормалности, која се заснива на примени генетских алгоритама на проблем истовремене оптимизације свих параметара који утичу на перформансе мрежа. У њих спадају параметри генерисања података, репрезентације узорака, мрежне архитектуре и тренирања мреже. Извршена је евалуација метода машинског учења помоћу којих је могуће контролисати вероватноћу грешке бинарних класификатора на изабраној класи, а предложене су и три нове методе за контролу грешке. Применом методе која је у евалуацији остварила најбоље резултате, развијена су два омнибус теста нормалности на основу једносмерних и рекурентних неуронских мрежа, као и осам усмерених тестова нормалности за основне типове дистрибуција које не припадају фамилији нормалних. Искрпна евалуација на узорцима до 100 елемената, добијеним из разних расподела, показује предности мрежних тестова нормалности у односу на стандардне статистичке тестове. У готово свим случајевима, усмерени мрежни тестови имају већу моћ од усмерених статистичких тестова при сва три уобичајена прага статистичке значајности (1%, 5% и 10%). Омнибус тест изведен из класичне једносмерне мреже има ранг бољи од стандардних тестова при прагу значајности од 1%, док омнибус тест изведен из рекурентне мреже са дугорочном меморијом по моћи надмашује стандардне тестове на узорцима до 30 елемената из многих типова расподела које се у литератури користе за евалуацију тестова нормалности.

**Кључне речи:** неуронске мреже, тестови нормалности, генетски алгоритми, бинарна класификација, контрола вероватноће грешке

**Научна област:** Математика, Рачунарске науке

**Ужа научна област:** Вештачка интелигенција, Статистика, Оптимизација

**Title:** Optimization of Algorithmic Normality Tests Based on Neural Networks

**Abstract:** In this dissertation, we examine an algorithmic approach to normality testing. We propose a methodology based on a Genetic Algorithm for the simultaneous optimization of all parameters affecting the performance of a neural network for sample classification. Those are the parameters of data generation, sample representation, network architecture, and training. We evaluated state-of-the-art machine-learning methods for controlling the chosen class-conditional error of a binary classifier and proposed three new techniques. Using the best-performing method, we derived two omnibus normality tests from feedforward and recurrent neural networks and eight tests directed against basic non-normal distribution types. Rigorous evaluation using samples with up to 100 elements generated from various distributions shows the advantages of the network tests over the standard statistical normality tests. In almost all cases, directed network tests are more powerful than standard directed statistical tests of normality at the three most common significance thresholds (1%, 5%, and 10%). The feedforward omnibus network test outranks the statistical omnibus tests at the threshold equal to 1%. The recurrent omnibus network test is more powerful than the other tests on the samples, with up to 30 elements from many distributions typically used in the literature to evaluate normality tests.

**Keywords:** neural networks, normality testing, genetic algorithms, binary classification, error-rate control

**Scientific Field:** Mathematics, Computer Science

**Scientific Subfield:** Artificial Intelligence, Statistics, Optimization



# Садржај

<b>1</b>	<b>Увод</b>	<b>1</b>
<b>2</b>	<b>Тестирање нормалности</b>	<b>5</b>
2.1	Омнибус тестови . . . . .	6
2.1.1	Регресиони тестови . . . . .	6
2.1.2	Тестови момената . . . . .	7
2.1.3	Тестови емпиријских функција расподеле . . . . .	8
2.1.4	Тестови веродостојности . . . . .	10
2.2	Усмерени тестови . . . . .	11
2.2.1	Тестови искошености . . . . .	11
2.2.2	Тестови тежине репова . . . . .	11
2.3	Предности и мане статистичких тестова нормалности . . . . .	12
2.4	Испитивање нормалности неуронским мрежама . . . . .	13
2.4.1	Проблеми постојећих мрежа . . . . .	13
<b>3</b>	<b>Примена генетских алгоритама у развоју једносмерних неуронских мрежа за испитивање нормалности</b>	<b>17</b>
3.1	Генетски алгоритми . . . . .	18
3.2	Примена еволутивних метахеуристика у дизајнирању архитектура неуронских мрежа . . . . .	19
3.3	Структура јединки . . . . .	20
3.3.1	Метапараметри за генерисање података . . . . .	20
3.3.2	Метапараметри репрезентације . . . . .	22
3.3.3	Метапараметри мрежне архитектуре . . . . .	24
3.3.4	Метапараметри тренирања . . . . .	25
3.3.5	Врсте хромозома . . . . .	26
3.4	Генетски оператори . . . . .	26
3.4.1	Селекција . . . . .	26
3.4.2	Укрштање . . . . .	27
3.4.3	Мутација . . . . .	28
3.5	Функција прилагођености . . . . .	29
3.5.1	Тачност . . . . .	29
3.5.2	Нестабилност . . . . .	30
3.5.3	Дефективност облика . . . . .	30
3.5.4	Одређивање тежина . . . . .	31
3.6	Замена генерација . . . . .	31
3.7	Критеријуми заустављања . . . . .	33
3.8	Завршна евалуација и избор мреже . . . . .	33
3.9	Структура предложеног генетског алгоритама . . . . .	34

<b>4</b>	<b>Извођење статистичких тестова из бинарних класификатора</b>	<b>37</b>
4.1	Нотација и конвенције . . . . .	37
4.2	Конформална предикција . . . . .	38
4.3	Нејман-Пирсонова класификација . . . . .	39
4.4	Алгоритам UA . . . . .	40
4.5	Пермутационе $p$ -вредности . . . . .	41
4.6	Посредно тестирање . . . . .	42
4.7	Индекси типичности . . . . .	43
4.8	Извођење статистичких тестова из бинарних класификатора . . . . .	43
4.8.1	Интерпретација . . . . .	44
4.8.2	Равнотежа између FNR и FPR . . . . .	45
4.8.3	Естимација $p_c$ -вредности . . . . .	45
4.9	Контрола грешке посредним тестирањем . . . . .	46
4.9.1	Тестирање растојања до најближих суседа . . . . .	46
4.9.2	Тестирање тежинских растојања до најближих суседа . . . . .	47
4.9.3	Пример . . . . .	48
4.10	Евалуација . . . . .	49
4.10.1	Скупови за евалуацију . . . . .	49
4.10.2	Подешавање параметара . . . . .	49
4.10.3	Резултати и дискусија . . . . .	50
<b>5</b>	<b>Тестови нормалности засновани на једносмерним мрежама</b>	<b>55</b>
5.1	Усмерени мрежни тестови . . . . .	55
5.1.1	Конфигурације метапараметара . . . . .	55
5.1.2	Стабилност . . . . .	56
5.1.3	Извођење усмерених мрежних тестова нормалности . . . . .	58
5.1.4	Поређење са стандардним статистичким тестовима . . . . .	58
5.1.5	Дискусија . . . . .	59
5.2	Омнибус мрежни тестови . . . . .	60
5.2.1	Конфигурација метапараметара . . . . .	60
5.2.2	Стабилност . . . . .	60
5.2.3	Групе расподела за поређење . . . . .	61
5.2.4	Поређење са стандардним статистичким тестовима . . . . .	61
5.2.5	Дискусија . . . . .	64
<b>6</b>	<b>Омнибус тест нормалности заснован на дугорочној рекурентној мрежи типа LSTM</b>	<b>65</b>
6.1	Мреже типа LSTM . . . . .	65
6.2	Трансформација узорака . . . . .	67
6.3	Метапараметри . . . . .	67
6.4	Стабилност . . . . .	68
6.5	Евалуација . . . . .	69
6.6	Поређење са мрежним омнибус тестом NT . . . . .	73
6.7	Дискусија . . . . .	74
<b>7</b>	<b>Дискусија</b>	<b>75</b>
7.1	Тумачивост мрежних тестова . . . . .	75
7.2	Мреже као емпиријски индуковане статистике . . . . .	76
7.3	Величина скупа за тренирање . . . . .	76
7.4	Стабилност и монотоност тачности . . . . .	77
7.5	Контрола вероватноће грешке . . . . .	77

7.6	Аутоматизација нумеричких експеримената . . . . .	78
7.7	Изазови . . . . .	78
<b>8</b>	<b>Закључак</b>	<b>79</b>



# Глава 1

## Увод

Нормалност расподеле представља један од најчешћих услова за примену стандардних статистичких тестова и алата. Постоји више разлога за постављање оваквог услова у статистичким анализама у различитим областима:

- Неадекватност модела за дати процес огледа се у резидуалима јасно изражене структуре, која показује да грешке модела нису насумичне. Законитост коју оне прате одговара компонентама процеса које модел није обухватио. Од свих дистрибуција са коначним очекивањем и коначном варијансом, фамилија нормалних расподела одликује се највећим степеном ентропије [1]. Сходно томе, нормалност резидуала представља показатељ да су грешке насумичне и да из модела није остављена ниједна статистички значајна компонента.
- Потом, централна гранична теорема гарантује да аритметичка средина независних и идентично дистрибуираних случајних променљивих тежи нормалној расподели. Стога, претпоставка нормалности олакшава извођење расподела многих статистика и омогућава аналитичко израчунавање њихових функција расподеле, момената и других нумеричких својстава, без примене рачунски захтевних метода.
- Такође, разни природни процеси прате нормалну расподелу или се у пракси показало да их нормални модели довољно добро апроксимирају, па је статистика бројне тестове за анализу података развила полазећи од претпоставке да ће анализирани узорци бити нормални.

Међутим, емпиријске студије су показале да стандардни тестови за испитивање нормалности имају малу моћ на малим узорцима (до сто елемената) [2, 3]. То значи да је њиховом применом тешко утврдити да расподела из које узорци потичу није нормална. Са циљем да тај недостатак превазиђу, развијене су неуронске мреже за тестирање нормалности и у евалуацији показале компетитивност са класичним статистичким тестовима [4–6]. У предности неуронског приступа спадају већа моћ на одређеним класама не-нормалних расподела и тумачив излаз. Резултат њихове примене је процена вероватноће да је тестирани узорак нормалан, а не, као што је случај са класичним статистичким тестовима нормалности,  $p$ -вредност, чије је тумачење знатно теже и често доводи до логичких грешака у закључивању [7–10].

Ипак, испитивање нормалности путем неуронских мрежа има извесне недостатке. Прво, тачност неуронских мрежа зависи од структуре и величине скупа за тренирање. Иако је могуће генерисати произвољан број узорака из разних нормалних и не-нормалних расподела, у литератури је остало отворено питање какав тај скуп треба да буде да би се тренирањем на њему добила мрежа оптималних перформанси. За разлику

од других класификационих проблема на које се примењују алгоритми машинског учења, у овом случају је могуће генерисати узорке који ће бити искоришћени за тренирање. То овај проблем чини изазовним пошто захтева оптимизацију не само неуронске мреже, већ и података за њено тренирање. При томе, тачност те мреже не треба да зависи од карактеристика расподела уврштених у скуп за тренирање, што је случај са постојећим мрежама. Друга мана неуронских мрежа је што се њима не може контролисати вероватноћа погрешне класификације нормалних расподела (тзв. вероватноћа грешке I врсте), а то је у директној супротности са начином на који се користе статистички тестови. На крају, стандардне мреже развијене су за објекте из коначних простора са унапред дефинисаним редоследом и значењем димензија. Са друге стране, узорци су скупови или колекције, што значи да могу садржати произвољан број елемената у ма којем редоследу. Овај проблем је до сада решаван пресликавањем узорака у простор фиксних и унапред задатих димензија које представљају узорачке дескриптивне статистике или статистике одабраних тестова нормалности и њихове  $p$ -вредности. Међутим, није утврђено која је репрезентација оптимална. У вези са тим је и питање оптималне архитектуре мреже, као и избора алгоритма за тренирање и подешавања његових параметара.

Циљ ове дисертације је отклањање наведених недостатака, па су формулисане следеће хипотезе од којих се пошло у истраживању:

- ( $\mathcal{H}_1$ ) Може се формирати скуп података на којем је могуће тренирати неуронску мрежу тако да она по перформансама надмаши стандардне тестове нормалности.
- ( $\mathcal{H}_2$ ) Може се развити неуронска мрежа која нулту хипотезу нормалности испитује наспрам алтернативне хипотезе да задати узорак потиче из тачно одређене расподеле или фамилије расподела, а не из скупа свих расподела које нису нормалне.
- ( $\mathcal{H}_3$ ) Могуће је оптимизовати не само параметре неуронске мреже, већ и начин репрезентације узорака и структуру података за тренирање.
- ( $\mathcal{H}_4$ ) Применом метода контроле грешке, развијеним у оквиру машинског учења, могуће је задати горњу границу вероватноће да се нормална расподела не класификује као таква, чиме би се омогућило коришћење неуронске мреже као да је у питању стандардан статистички тест. При томе, услов је да се горња граница може накнадно мењати тако да не захтева поновно тренирање мреже.

Остатак дисертације организован је на следећи начин. У глави 2 описано је тестирање нормалности у класичној статистици и представљено је неколико стандардних тестова нормалности који се најчешће користе у пракси. Представљене су и мреже које су до сада предложене за испитивање нормалности. Избор репрезентације узорака, архитектуре мреже, конструкција скупа за тренирање, као и избор алгоритма за тренирање и подешавање његових параметара формулисани су као један оптимизациони проблем. Глава 3 приказује примену генетских алгоритама на симултану оптимизацију свих параметара који утичу на перформансе мреже. Мрежа добијена том методологијом представља бинарни класификатор, а не статистички тест, зато што јој се не може контролисати вероватноћа да погрешно класификује нормалан узорак. У глави 4 представљене су методе машинског учења којима се бинарном класификатору након тренирања може контролисати вероватноћа грешке на једној. У истој глави представљени су резултати опсежне евалуације тих метода на скуповима података различите величине и структуре. У глави 5 представљени су резултати евалуације једносмерних неуронских мрежа добијених методологијом из главе 3. Након провере стабилности, из тих мрежа су најефикаснијом методом из главе 4 изведени мрежни тестови нормалности. Извршено је

њихово поређење са стандардним статистичким тестовима и дате су смернице за примену тих тестова у пракси. Поред омнибус мрежног теста, изведено је осам усмерених тестова за испитивање нормалности. У глави 6 представљена је примена генетских алгоритама у конструкцији рекурентних мрежа за класификацију узорака. Показано је како се применом оваквих архитектура може избећи дефинисање репрезентационих простора пре тренирања и омогућити развој стабилних мрежа које на улазу могу да прихвате узорак ма којег обима. Изведен је један омнибус мрежни тест нормалности. Дискусија о доказаности полазних хипотеза  $\mathcal{H}_1$ – $\mathcal{H}_4$  и оствареним научним доприносима налази се у глави 7, док глава 8 садржи закључак.

У дисертацији је, осим уколико није другачије наглашено, коришћена уобичајена математичка нотација.



# Глава 2

## Тестирање нормалности

Нека је  $X$  случајна променљива која моделује расподелу чију нормалност тестирамо. Нека је  $\{X_i\}_{i=1}^n$  прост случајан узорак из те расподеле и нека променљиве  $X_1, X_2, \dots, X_n$  прате исту расподелу као  $X$  (у ознаци  $X_i \sim X$ ,  $i = 1, 2, \dots, n$ ). Нека је узорак  $\{x_i\}_{i=1}^n$  реализација променљивих  $\{X_i\}_{i=1}^n$ .

Статистички тестови нормалности на основу узорка  $\{x_i\}_{i=1}^n$  испитују да ли променљива  $X$  прати нормалну расподелу. Нулта хипотеза  $H_0$  ових тестова може се формулисати на два начина. Прва формулација назива се општа хипотеза о нормалности:

$$H_0 : (\exists \mu \in \mathbb{R}, \exists \sigma^2 > 0) \quad X \sim \mathcal{N}(\mu, \sigma^2). \quad (2.1)$$

а друга као специфична хипотеза о нормалности за унапред задате параметре  $\mu$  и  $\sigma$ :

$$H_0 : X \sim \mathcal{N}(\mu, \sigma^2). \quad (2.2)$$

У пракси се знатно чешће користе тестови са општом нултом хипотезом.

Алтернативна хипотеза  $H_a$  такође се може дефинисати на два начина. Ако претпоставимо да променљива  $X$  прати или нормалну расподелу или тачно одређену не-нормалну дистрибуцију  $\Psi$ , у питању је усмерена алтернативна хипотеза:

$$H_a : X \sim \Psi. \quad (2.3)$$

При томе,  $\Psi$  може бити и фамилија не-нормалних расподела. Ако таква претпоставка није оправдана, алтернативна хипотеза не наводи ниједну не-нормалну расподелу, већ представља негацију опште хипотезе о нормалности и назива се неусмерена алтернативна хипотеза:

$$H_a : (\forall \mu, \sigma^2 \in \mathbb{R}) X \not\sim \mathcal{N}(\mu, \sigma^2). \quad (2.4)$$

Тестови са неусмереном алтернативном хипотезом називају се неусмерени или омнибус тестови нормалности, а усмерени тестови су они са усмереном алтернативном хипотезом.

Временом је развијено више омнибус и усмерених тестова нормалности [11]. У одељцима 2.1 и 2.2 представљени су класични статистички тестови нормалности релевантни за ову дисертацију. Изабрани су они који се најчешће користе у пракси и истраживању и за које се у симулационим студијама показало да имају највећу моћ. У одељку 2.3 наведене су предности и мане класичних тестова нормалности, а у одељку 2.4 представљене су неуронске мреже развијене за испитивање нормалности, као и отворена питања у вези са њиховом конструкцијом и применом у пракси, чије је решавање предмет ове дисертације.

## 2.1 Омнибус тестови

### 2.1.1 Регресиони тестови

Регресиони тестови добили су назив по томе што статистику изводе полазећи од релација, односно модела, у којима параметри нормалне расподеле ( $\mu$  и  $\sigma$ ) имају улогу регресионих коефицијената.

#### Шапиро–Вилков тест

Шапиро–Вилков тест (SW) [12] заснива се на следећој релацији:

$$(\forall \mu \in \mathbb{R}, \sigma^2 > 0) \quad \mathcal{N}(\mu, \sigma^2) = \mu + \sigma \mathcal{N}(0, 1). \quad (2.5)$$

Нека је  $Z_{(1)} \leq Z_{(2)} \leq \dots \leq Z_{(n)}$  варијациони низ простог случајног узорка из стандардне нормалне расподеле  $\mathcal{N}(0, 1)$ . Нека је  $\mathbf{z} = [Z_{(i)}]_{i=1}^n$  његова репрезентација у векторском облику,  $\mathbf{b} = [b_1, b_2, \dots, b_n]^T$  вектор очекиваних вредности променљивих  $\{Z_{(i)}\}_{i=1}^n$  ( $b_i = \mathbb{E}[Z_{(i)}]$ ,  $i = 1, 2, \dots, n$ ), а  $\mathbf{V} = [v_{i,j}]_{i,j=1}^n$  ( $v_{i,j} = \text{Cov}[Z_{(i)}, Z_{(j)}]$ ;  $i, j = 1, 2, \dots, n$ ) матрица њихових коваријанси. Нека је  $X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(n)}$  варијациони низ простог случајног узорка  $\{X_i\}_{i=1}^n$ . Ако узорак потиче из произвољне нормалне расподеле  $\mathcal{N}(\mu, \sigma^2)$  и уколико је матрица  $\mathbf{V}$  инвертибилна, а вектор  $\mathbf{b}$  различит од нула-вектора, из релације (2.5) следи да је најбољи центрирани линеарни естиматор девијације  $\sigma$  дат следећом формулом [12]:

$$\hat{\sigma} = \frac{\mathbf{b}^T \mathbf{V}^{-1} \mathbf{z}}{\mathbf{b}^T \mathbf{V}^{-1} \mathbf{b}}. \quad (2.6)$$

Одговарајући естиматор варијансе добија се квадрирањем естиматора девијације. Статистика  $W$  Шапиро–Вилковог теста представља нормализовани количник тако оцењене варијансе и класичног естиматора вредности  $(n-1)\sigma^2$ :

$$S^2 = \sum_{i=1}^n (X_i - \bar{X})^2, \quad \text{где је } \bar{X} = \frac{1}{n} \sum_{i=1}^n X_i. \quad (2.7)$$

Статистика  $W$  дата је следећом формулом:

$$W = \frac{R^4 \hat{\sigma}^2}{C^2 S^2} = \frac{(\sum_{i=1}^n a_i X_i)^2}{\sum_{i=1}^n (X_i - \bar{X})^2}, \quad (2.8)$$

где вредности  $R^4$  и  $C^2$  служе за нормализацију статистике и дефинисане су на следећи начин:

$$\begin{aligned} R^2 &= \mathbf{b}^T \mathbf{V}^{-1} \mathbf{b} \\ C^2 &= \mathbf{b}^T \mathbf{V}^{-1} \mathbf{V}^{-1} \mathbf{b}, \end{aligned} \quad (2.9)$$

а коефицијенти  $a_i$  ( $i = 1, 2, \dots, n$ ) рачунају се формулом (2.10):

$$\mathbf{a} = [a_1, a_2, \dots, a_n]^T = \frac{\mathbf{b}^T \mathbf{V}^{-1}}{(\mathbf{b}^T \mathbf{V}^{-1} \mathbf{V}^{-1} \mathbf{b})^{1/2}}. \quad (2.10)$$

Вредности статистике  $W$  припадају одсечку  $[0, 1]$ , при чему су више вредности карактеристичне за узорке из нормалних расподела. Овај тест има високу моћ на бројним класама не-нормалних расподела, због чега се у литератури и пракси често препоручује његова примена (нпр, видети радове [13, 14]).

## Шапиро–Францијин тест

Шапиро–Францијин тест (SF) [15] представља рачунско упрошћење Шапиро–Вилковог теста. Оно се састоји у томе да се променљиве  $\{X_{(i)}\}_{i=1}^n$  посматрају као независне у великим узорцима, на основу чега се матрица  $\mathbf{V}^{-1}$  може заменити јединичном матрицом  $\mathbf{I}_n$ . Одатле следи да се коефицијенти  $a_i$  могу брзо израчунати на следећи начин:

$$\mathbf{a}' = [a'_1, a'_2, \dots, a'_n]^T = \frac{\mathbf{b}}{(\mathbf{b}^T \mathbf{b})^{1/2}}. \quad (2.11)$$

Шапиро–Францијин тест је од Шапиро–Вилковог моћнији на класи симетричних алтернативних расподела.

## Чен–Шапиров тест

Чен–Шапиров тест (CS) [16] полази од следећег регресионог модела:

$$X_{(i)} = \mu + \sigma b_i + \varepsilon_i \quad (i = 1, 2, \dots, n), \quad (2.12)$$

где случајне променљиве  $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n$  моделују случајну грешку модела  $\mu + \sigma b_i$ , а вредност  $b_i = \mathbb{E}[Z_{(i)}]$  ( $i = 1, 2, \dots, n$ ) дефинисана је исто као код Шапиро–Вилковог теста. Одатле добијамо:

$$X_{(i+1)} - X_{(i)} = \sigma \cdot (b_{i+1} - b_i) + \varepsilon_{i+1} - \varepsilon_i \quad (2.13)$$

$$\frac{X_{(i+1)} - X_{(i)}}{b_{i+1} - b_i} = \sigma + \frac{\varepsilon_{i+1} - \varepsilon_i}{b_{i+1} - b_i}. \quad (2.14)$$

Минимизацијом вредности  $\sum_{i=1}^{n-1} (\varepsilon_{i+1} - \varepsilon_i)^2 / (b_{i+1} - b_i)^2$  добија се следећи центрирани естиматор девијације  $\sigma$ :

$$\hat{\sigma}_{CS} = \frac{1}{n-1} \sum_{i=1}^{n-1} \frac{X_{(i+1)} - X_{(i)}}{b_{i+1} - b_i}. \quad (2.15)$$

Класичан естиматор девијације је  $S/\sqrt{n-1}$ , где је променљива  $S$  дефинисана формулом (2.7). Статистика Чен–Шапировог теста представља њихов модификовани количник:

$$HQ = \frac{1}{\sqrt{n-1}S} \sum_{i=1}^{n-1} \frac{X_{(i+1)} - X_{(i)}}{b_{i+1} - b_i}. \quad (2.16)$$

У дефиницији статистике  $HQ$ , очекивања  $b_1, b_2, \dots, b_n$  апроксимирана су као у Вајсберг–Бингамовом тесту [17]:

$$b_i \approx \Phi^{-1} \left( \frac{i - 3/8}{n + 1/4} \right) \quad (i = 1, 2, \dots, n), \quad (2.17)$$

где је  $\Phi$  функција расподеле стандардне нормалне расподеле  $\mathcal{N}(0, 1)$ .

### 2.1.2 Тестови момената

Статистике тестова момената заснивају се на естиматорима трећег и четвртог момента. Трећи момент ( $\sqrt{\beta_1} = \frac{\mathbb{E}[(X - \mathbb{E}[X])^3]}{(\text{Var}[X])^{3/2}}$ ) квантификује искошеност расподеле, док четврти момент ( $\beta_2 = \frac{\mathbb{E}[(X - \mathbb{E}[X])^4]}{(\text{Var}[X])^2}$ ) показује релативну концентрацију расподеле у реповима у односу на централни део дистрибуције. Трећи момент назваћемо коефицијент искошености, а четврти момент зваћемо коефицијент тежине репова.

## Жарк–Бераов тест (JB)

За сваку нормалну расподелу важе следеће релације:

$$\sqrt{\beta_1} = 0 \quad \text{и} \quad \beta_2 = 3. \quad (2.18)$$

Првобитна формулација статистике Жарк–Бераовог теста (JB) показује у којој мери се узорачке оцене коефицијената искошености и тежине репова,  $\widehat{B}_1$  и  $\widehat{B}_2$ , разликују од својих теоретских вредности код нормалних расподела [18]:

$$J = \frac{(\widehat{B}_1)^2}{6/n} + \frac{(\widehat{B}_2 - 3)^2}{24/n}. \quad (2.19)$$

при чему су  $6/n$  и  $24/n$  асимптотске варијансе естиматора  $\widehat{B}_1$  и  $\widehat{B}_2$  за узорак величине  $n$  [19].

Како естиматор  $\widehat{B}_2$  споро конвергира ка теоретској вредности, прва формулација овог теста није имала високу моћ на малим узорцима. Статистика  $J$  је касније модификована тако да користи очекивања и варијансе естиматора  $\widehat{B}_1$  и  $\widehat{B}_2$  за дато  $n$  [20]:

$$J = n \left( \frac{(\widehat{B}_1)^2}{c_1} + \frac{(\widehat{B}_2 - c_2)^2}{c_3} \right), \quad (2.20)$$

где је:

$$\begin{aligned} c_1 &= \frac{6(n-2)}{(n+1)(n+3)} \\ c_2 &= \frac{3(n-1)}{n+1} \\ c_3 &= \frac{24n(n-2)(n-3)}{(n+1)^2(n+3)(n+5)}. \end{aligned}$$

То је омогућило њену примену и на узорке са малим бројем елемената.

## Д’Агостино–Пирсонов тест (DP)

Попут Жарк–Бераовог, и Д’Агостино–Пирсонов тест (DP) [21, 22] комбинује узорачке естиматоре искошености и тежине репова. Његова статистика дефинисана је на следећи начин:

$$K^2 = Z(\widehat{B}_1)^2 + Z(\widehat{B}_2)^2, \quad (2.21)$$

где су  $Z(\widehat{B}_1)^2$  и  $Z(\widehat{B}_2)^2$  трансформације естиматора  $\widehat{B}_1$  и  $\widehat{B}_2$  које прате нормалну расподелу [22] (видети формуле (2.33) и (2.43) у одељцима 2.2.1 и 2.2.2). Статистика  $K^2$  приближно прати  $\chi^2$  расподелу са два степена слободe [22].

### 2.1.3 Тестови емпиријских функција расподеле

Тестови у овој групи пореде емпиријску функцију расподеле ( $\widehat{F}_n$ ) процењену на основу узорка  $\{x_i\}_{i=1}^n$  са теоретском функцијом расподеле ( $F$ ) која карактерише циљну дистрибуцију за коју се тест врши.

## Колмогоров–Смирновљев тест (KS)

Статистика Колмогоров–Смирновљевог теста (KS) [23] дефинисана је на следећи начин:

$$D = \sup_x \left| \widehat{F}_n(x) - F(x) \right|. \quad (2.22)$$

У питању је тест којим се може тестирати припадност узорка ма којој дистрибуцији чији је аналитички облик функције расподеле познат. То значи да за тестирање нормалности, примена овог теста захтева специфичну нулту хипотезу у облику (2.2), са унапред задатим параметрима  $\mu$  и  $\sigma$ . Међутим, у пракси је знатно чешће прикладнија општа нулта хипотеза о нормалности (у облику (2.1)), што ограничава примену теста KS.

## Лилифорсов тест (LF)

Лилифорсов тест нормалности (LF) [24, 25] представља корекцију Колмогоров–Смирновљевог теста. Корекција се састоји у томе да се као параметри  $\mu$  и  $\sigma$  (који одређују специфичну нулту хипотезу у Колмогоров–Смирновљевом тесту) искористе аритметичка средина узорка ( $\bar{x}$ ) и узорачка стандардна девијација ( $\text{sd}(x)$ ):

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad \text{sd}(x) = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}.$$

То значи да је Лилифорсов тест за узорак  $x = \{x_i\}_{i=1}^n$  једнак Колмогоров–Смирновљевом тесту који у нултој хипотези наводи нормалну расподелу  $\mathcal{N}(\bar{x}, \text{sd}(x)^2)$  [26].

Лилифорсов тест изводи се на следећи начин. Узорак  $x = \{x_i\}_{i=1}^n$  прво се стандардизује трансформацијом која би, у случају да важи специфична нулта хипотеза о нормалности са параметрима  $\mu = \bar{x}$  и  $\sigma^2 = \text{sd}(x)^2$ , требало да га преведе у узорак  $z = \{z_i\}_{i=1}^n$  из стандардне нормалне расподеле  $\mathcal{N}(0, 1)$ :

$$z_i = \frac{x_i - \bar{x}}{\text{sd}(x)} \quad (i = 1, 2, \dots, n). \quad (2.23)$$

Приликом израчунавања статистике Лилифорсовог теста користимо емпиријску функцију расподеле  $\widehat{F}_n$  добијену из узорка  $z$  и  $\Phi$ , теоретску функцију расподеле стандардне нормалне дистрибуције  $\mathcal{N}(0, 1)$ :

$$D = \max_{i=1}^n \left| \widehat{F}_n(z_i) - \Phi(z_i) \right|. \quad (2.24)$$

Као и у Колмогоров–Смирновљевом тесту, више вредности статистике  $D$  представљају јачи доказ да узорак не потиче из нормалне расподеле.

## Андерсон–Дарлинггов тест (AD)

За разлику од Лилифорсовог и Колмогоров–Смирновљевог теста, чија је статистика дефинисана као супремум разлике између емпиријске и теоретске функције расподеле, статистика Андерсон–Дарлингговог теста (AD) [27] представља очекивање тежинске разлике између њих.

Нека је  $\psi : [0, 1] \rightarrow (0, \infty)$  тежинска функција. Статистика Андерсон–Дарлингговог теста који тежине дефинише функцијом  $\psi$  дата је следећом формулом:

$$A = n \int_{-\infty}^{+\infty} \left[ F(x) - \widehat{F}_n(x) \right]^2 \psi(F(x)) \, dF(x). \quad (2.25)$$

Улога тежине  $\psi$  јесте да тест учини осетљивим на оне делове расподеле на којима је одступање  $\hat{F}_n$  од  $F$  јачи показатељ да узорак не потиче из циљне расподеле. Приликом тестирања нормалности, тежинска функција дефинише се тако да тест буде осетљив на репове расподеле [26, 27]:

$$\psi(t) = [t(1-t)]^{-1}. \quad (2.26)$$

Тада статистику (2.25) рачунамо на следећи начин [11, 28]:

$$A = -n - \frac{1}{n} \sum_{i=1}^n [2i-1] [\ln \Phi(z_{(i)}) + \ln(1 - \Phi(z_{(n-i+1)}))], \quad (2.27)$$

где је  $z_{(1)} \leq z_{(2)} \leq \dots \leq z_{(n)}$  сортирана пермутација стандардизованог узорка  $z = \{z_i\}_{i=1}^n$  добијеног као у формули (2.23). Веће вредности статистике  $A$  показатељ су да тестирани узорак не потиче из нормалне расподеле.

### Крамер – фон Мизесов тест (СVM)

Крамер – фон Мизесов тест (СМ) [29–32] представља специјалан случај Андерсон–Дарлинговог теста када су све тежине јединичне ( $\psi(\cdot) = 1$ ) [26]. Сходно томе, статистика СVM теста је:

$$C = n \int_{-\infty}^{+\infty} [F(x) - \hat{F}(x)]^2 dF(x). \quad (2.28)$$

Овај тест има мању моћ од Андерсон–Дарлинговог теста зато што не користи функцију  $\psi(\cdot)$  да урачуна посебна својства нормалне расподеле [26].

### 2.1.4 Тестови веродостојности

У овом одељку биће представљена два Жанг–Воова теста [33] заснована на количнику веродостојности: ZC и ZA. Као и тестови емпиријске функције расподеле, и ови тестови могу испитати припадност узорка ма којој дистрибуцији. Они ће прво бити представљени за општи случај циљне расподеле, а потом ће бити показано како се могу применити у тестирању нормалности.

Статистике ова два теста имају следећи општи облик [34, 35]:

$$Z = \int_{-\infty}^{\infty} Z_t dw(t) \quad (2.29)$$

где је  $w(\cdot)$  тежинска функција. Приликом дефинисања тестова ZC и ZA, за  $Z_t$  се узима следећа статистика количника веродостојности [33]:

$$G_t^2 = 2n \left\{ \hat{F}_n(t) \ln \frac{\hat{F}_n(t)}{F(t)} + [1 - \hat{F}_n(t)] \ln \frac{1 - \hat{F}_n(t)}{1 - F(t)} \right\}. \quad (2.30)$$

Дефинишћући  $\hat{F}_n(X_{(i)}) = (i - 1/2)/n$ , статистике тестова ZC и ZA добијамо тако што  $dw(t)$  заменимо изразима  $F(t)^{-1}[1 - F(t)]^{-1} dF(t)$  и  $\hat{F}_n(t)^{-1}[1 - \hat{F}_n(t)]^{-1} d\hat{F}_n(t)$ :

$$ZC = \sum_{i=1}^n \left[ \ln \frac{F(X_{(i)})^{-1} - 1}{(n - 1/2)(i - 3/4) - 1} \right]^2 \quad (2.31)$$

$$ZA = - \sum_{i=1}^n \left[ \frac{\ln F(X_{(i)})}{n - i + 1/2} + \frac{\ln[1 - F(X_{(i)})]}{i - 1/2} \right]. \quad (2.32)$$

Приликом тестирања нормалности, користи се теоретска функција расподеле која одговара нормалној расподели чије је очекивање  $\bar{X}$ , а варијанса  $\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$  [34]. Тако дефинисаним статистикама  $ZC$  и  $ZA$  није могуће аналитички одредити функцију расподеле [33], па се приликом употребе тих тестова у пракси користе емпиријски добијене апроксимације.

## 2.2 Усмерени тестови

У овом одељку биће представљени усмерени тестови нормалности у односу на две групе алтернативних расподела: асиметричне (искошене) дистрибуције и расподеле које се од нормалних разликују по тежини репова.

### 2.2.1 Тестови искошености

Тестови искошености (ST) [22] имају општу хипотезу о нормалности, док им се алтернативна хипотеза може формулисати на један од следећа три начина:

- $\sqrt{\beta_1} \neq 0$ : расподела  $X$  је искошена.
- $\sqrt{\beta_1} < 0$ : расподела  $X$  је искошена улево.
- $\sqrt{\beta_1} > 0$ : расподела  $X$  је искошена удесно.

У сва три случаја, алтернативна хипотеза говори да расподела коју тестирамо није симетрична. У овој дисертацији, користимо ознаке  $ST_{\neq 0}$ ,  $ST_{< 0}$  и  $ST_{> 0}$  да наведемо коју алтернативну хипотезу користи тест искошености.

Приликом извођења овог теста у пракси, као статистика се не користи узорачка оцена коефицијента искошености ( $\hat{B}_1$ ), већ њена трансформација  $Z(\hat{B}_1)$  која приближно прати нормалну расподелу на нормалним узорцима [22]. Та трансформација применљива је само код узорака са више од осам елемената и дефинисана је на следећи начин [22]:

$$Z(\hat{B}_1) = \delta \ln (Y/U + ((Y/U)^2 + 1)^{1/2}), \quad (2.33)$$

где је:

$$U = (2/(W^2 - 1))^{1/2} \quad (2.34)$$

$$\delta = 1/\sqrt{\ln W} \quad (2.35)$$

$$W^2 = -1 + (2(V - 1))^{1/2} \quad (2.36)$$

$$V = \frac{3(n^2 + 27n - 70)(n + 1)(n + 3)}{(n - 2)(n + 5)(n + 7)(n + 9)} \quad (2.37)$$

$$Y = \hat{B}_1 \left( \frac{(n + 1)(n + 3)}{6(n - 2)} \right)^{1/2}. \quad (2.38)$$

### 2.2.2 Тестови тежине репова

Тестови тежине репова (КТ, енгл. *kurtosis tests*) [22] имају општу нулту хипотезу о нормалности, а усмерену алтернативну хипотезу да су репови расподеле чији узорак испитујемо другачији од репова нормалних расподела. Као и код тестова искошености, и код тестова тежине репова разликујемо три формулације алтернативне хипотезе:

- $\beta_2 \neq 3$ : расподела  $X$  је платикуртична (са реповима лакшим од нормалне) или лептокуртична (са реповима тежим од нормалне).
- $\beta_2 < 3$ : расподела  $X$  је платикуртична.
- $\beta_2 > 3$ : расподела  $X$  је лептокуртична.

Тестове са наведене три алтернативне хипотезе означаваћемо као  $\text{КТ}_{\neq 3}$ ,  $\text{КТ}_{< 3}$  и  $\text{КТ}_{> 3}$ .

У пракси се уместо узорачке оцене коефицијента тежине репова ( $\widehat{B}_2$ ) користи нормална трансформација  $Z(\widehat{B}_2)$  [36], формулисана у раду [37]. Нека је  $\tilde{B}_2$  стандардизовани коефицијент који се добија следећом формулом:

$$\tilde{B}_2 = \frac{\widehat{B}_2 - \mathbb{E}[\widehat{B}_2]}{\sqrt{\text{Var}[\widehat{B}_2]}}, \quad (2.39)$$

где је  $\mathbb{E}[\widehat{B}_2]$  очекивање а  $\text{Var}[\widehat{B}_2]$  варијанса коефицијента  $\widehat{B}_2$  за узорак од  $n$  елемената:

$$\mathbb{E}[\widehat{B}_2] = \frac{3(n-1)}{n+1}, \quad \text{Var}[\widehat{B}_2] = \frac{24n(n-2)(n-3)}{(n+1)^2(n+3)(n+5)}. \quad (2.40)$$

Нека је  $U$  трећи стандардизовани момент коефицијента  $\widehat{B}_2$  за узорак од  $n$  елемената:

$$U = \frac{6(n^2 - 5n + 2)}{(n+7)(n+9)} \sqrt{\frac{6(n+3)(n+5)}{n(n-2)(n-3)}} \quad (2.41)$$

и нека је  $A$  функција момента  $U$ , дефинисана формулом (2.42):

$$A = 6 + \frac{8}{U} \left[ \frac{2}{U} + \sqrt{1 + \frac{4}{U^2}} \right]. \quad (2.42)$$

Тада, нормалну трансформацију  $Z(\widehat{B}_2)$  добијамо на следећи начин [22]:

$$Z(\widehat{B}_2) = \frac{1 - \frac{2}{9A} - \left( \frac{1 - \frac{2}{A}}{1 + \widehat{B}_2 \sqrt{\frac{2}{A-4}}} \right)^{1/3}}{\sqrt{\frac{2}{9A}}}. \quad (2.43)$$

Трансформацију (2.43) можемо применити уколико узорак има бар 20 елемената, и тада променљива  $Z(\widehat{B}_2)$  приближно прати нормалну расподелу.

## 2.3 Предности и мане статистичких тестова нормалности

Класични статистички тестови нормалности представљају стандардан алат којим се проверава да ли узорак потиче из нормалне расподеле. Они се фокусирају на тачно одређено својство нормалних дистрибуција и испитују у којој мери је оно присутно у датом узорку. Расподеле њихових статистика углавном се могу извести аналитички, што омогућава њихово брзо израчунавање.

Међутим, ови тестови имају две мане. Прва мана је ниска моћ на малим узорцима (до 100 елемената), што је потврђено у бројним симулационим студијама. Друга мана је што

се, као и други фреквентистички тестови, и ови тестови спроводе рачунањем  $p$ -вредности реализованих статистика и њиховим поређењем са праговима статистичке значајности (који обично износе 1% или 5%). У пракси се показало да се појам  $p$ -вредности често на погрешан начин разуме и користи [7, 9, 10]. Такође, различити тестови могу дати различите резултате за исти узорак, што крајње кориснике тестова оставља у недоумици којим резултатима веровати и који тест употребити.

## 2.4 Испитивање нормалности неуронским мрежама

Наведене мане статистичких тестова представљале су главну мотивацију за развој тестова нормалности заснованих на вештачким неуронским мрежама. Како су се мреже показале као успешни класификатори у многим проблемима машинског учења, од њих се очекивала висока тачност и приликом тестирања нормалности. Поред тога, мреже на излазу враћају процењену вероватноћу да је тестирани узорак нормалан. Таква вероватноћа је знатно разумљивија од  $p$ -вредности. Такође, претпоставка у овом приступу јесте и да ће неуронске мреже успети да сагледају узорке из различитих (и комплементарних) перспектива и да ће обучавањем на скуповима са нормалним и не-нормалним узорцима успети да науче класификациона правила тачнија од појединачних статистичких тестова нормалности.

Прва таква мрежа развијена је у раду [4]. Она је тренирана на скупу од 70 нормалних и 50 не-нормалних узорака са 30 елемената, који су прво пресликани у векторе  $p$ -вредности 16 статистика. У раду није наведено које су све статистике коришћене, као ни како су одабране. По речима аутора, те статистике су „изабране интуитивно, а не научном методом” [4].

У раду [5] добијена је мрежа тренирањем на десетинама хиљада узорака генерисаних из нормалних расподела и разних не-нормалних дистрибуција у Џонсоновом систему расподела [38]. Сваки узорак је прво пресликан у вектор који садржи величину узорка и статистике Лин-Мудхолкаровог теста [39] и Шапиро-Вилковог [12] теста, две статистике Васичековог ентропијског теста [1], као и узорачке оцене искошености и тежине репова. Мрежа садржи један скривени слој са шест неурона, а активациона функција није наведена.

Мрежа из рада [6] састоји се од пет скривених слојева са по 100 неурона. Обучавана је на 13050 нормалних и исто толико не-нормалних узорака из различитих не-нормалних расподела у Пирсоновом систему дистрибуција [40–43]. Пре примене мреже, сваки узорак се пресликава у вектор који садржи величину узорка, његову аритметичку средину, минимум, максимум, медијану, стандардну девијацију и средњу апсолутну девијацију, као и стандардизоване вигинтиле [6].

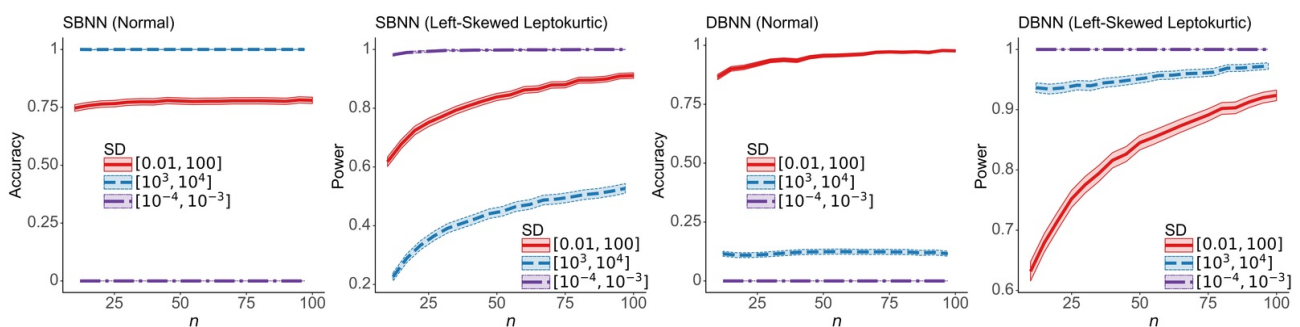
### 2.4.1 Проблеми постојећих мрежа

Иако су се мреже [5] и [6], применљиве на узорке различитих величина, показале компетитивним са класичним тестовима нормалности, одређена питања везана за њихову конструкцију и примену остала су отворена.

- Узорци су колекције, што значи да могу садржати произвољан број елемената у било ком редоследу. Међутим, код класичних (једносмерних) неуронских мрежа, улазни објекти морају бити вектори са тачно одређеним редоследом и значењем димензија. Стога је, да би се мреже примениле у тестирању нормалности, неопходно да се узорци прво пресликају у векторе одређеног коначнодимензионалног простора, који се назива репрезентациони простор или простор репрезентације узорака.

У литератури [4–6] су за димензије таквог простора коришћене различите дескриптивне статистике, као и статистике и  $p$ -вредности неких тестова нормалности, али још увек није пронађена оптимална структура репрезентационих простора.

- Архитектуру класичних једносмерних мрежа чине дубина (број скривених слојева), ширина (број неурона у сваком скривеном слоју) и активациона функција. Оптимална мрежна архитектура за испитивање нормалности још увек није позната.
- На перформансе мреже утиче и скуп података за тренирање. Будући да можемо генерисати онолико узорака колико нам рачунски ресурси допуштају, отворено је питање какву структуру и величину скуп за тренирање треба да има да би резултујућа мрежа показала оптималне перформансе. С једне стране, претпоставка је да већи скупови омогућавају конструкцију тачнијих мрежа, али они истовремено успоравају процес тренирања.
- На перформансе мреже и брзину тренирања утичу још и алгоритам одабран за њено тренирање, као и његови параметри. Оптималан избор алгоритма и оптимална подешавања његових параметара, који омогућавају брзо обучавање мреже високе тачности, још увек нису одређени.
- Методологија којом су мреже [5, 6] развијене не проверава да ли оне задржавају исти степен тачности на узорцима који су добијени из расподела знатно другачије варијабилности него узорци на којима су мреже трениране. Ту нестабилност мрежа показује слика 2.1. На њој видимо да се тачност мрежа из радова [5, 6] знатно мења приликом евалуације на узорцима чија стандардна девијација одступа од девијација узорака искоришћених за тренирање. Услед те нестабилности у односу на варијансу, примена ових мрежа у пракси не може се препоручити.
- Мрежама добијеним стандардним поступком конструкције није могуће подешавати вероватноћу погрешне класификације објеката једне класе, што је важан захтев за статистичко тестирање нормалности, где тестовима контролишемо вероватноћу погрешне класификације нормалних узорака.



**Слика 2.1:** Утицај стандардне девијације (SD) на тачност мрежа SBNN [5] и DBNN [6]. Криве тачности представљене су уз нормално апроксимиране интервале поверења са поверењем од 99,9%.

На основу наведених отворених питања, закључујемо да је за развој стабилне, тачне и брзе мреже неопходно оптимизовати све параметере који утичу на њен рад. У њих спадају:

- параметри генерисања података,
- параметри репрезентације,

- параметри архитектуре и
- параметри тренирања.

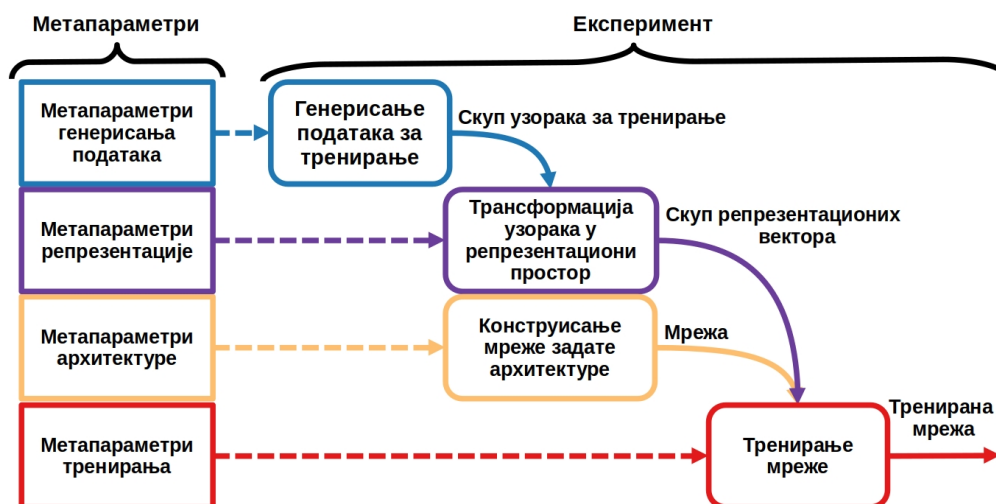
Методологија којом је то постигнуто формулисана је у глави 3, а методе за контролу грешке представљене су и евалуиране у глави 4.



## Глава 3

# Примена генетских алгоритама у развоју једносмерних неуронских мрежа за испитивање нормалности

Као што је наведено у одељку 2.4.1, параметри који утичу на перформансе мрежа класификују се у четири категорије, од којих свака дефинише једну фазу нумеричког експеримента у којем се развија мрежа за тестирање нормалности. Те параметре ћемо у наставку називати метапараметри да бисмо их разликовали од мрежних тежина које се у литератури обично називају параметри (модела). На слици 3.1 приказан је ток експеримента у четири фазе: генерисање података за тренирање, трансформација тих података у одговарајуће елементе репрезентационог простора, одређивање мрежне архитектуре и тренирање тако дефинисане мреже на трансформисаним подацима. Свака фаза изводи се у складу са подешавањем метапараметара из одговарајуће групе.



Слика 3.1: Четири фазе експеримента у којем се развија неуронска мрежа за тестирање нормалности расподела.

У овој глави изложена је методологија за симултану оптимизацију свих метапараметара, заснована на генетским алгоритмима. При томе, разматране су архитектуре које одговарају класичним једносмерним неуронским мрежама (енгл. *feedforward neural networks*) будући да су оне најједноставније за имплементацију и да, на основу резултата у литератури, представљају добру основу за развој мрежних тестова нормалности.

## 3.1 Генетски алгоритми

Генетски алгоритми (енгл. *Genetic Algorithms*, GA) представљају једну од првих класа еволутивних метахеуристика – општих оптимизационих алгоритама заснованих на принципима еволуције [44, 45]. Општи концепт GA представљен је алгоритмом 3.1.

---

**Алгоритам 3.1:** Општи облик генетског алгоритма

---

**Улаз:** проблем, услови заустављања, параметри.

**Израз:** најбоље пронађено решење

1  $P^{(0)} \leftarrow$  генерисати почетну популацију.

2 Израчунати прилагођеност сваке јединке у  $P^{(0)}$ .

3  $\mathbf{x}^{(naj)} \leftarrow$  наћи најбољу јединку у  $P^{(0)}$ .

4  $u \leftarrow 0$

5 **док** нису испуњени услови за крај :

6  $R^{(u)} \leftarrow$  изабрати парове родитеља из  $P^{(u)}$ .

7  $D^{(u)} \leftarrow$  укрстити парове из  $R^{(u)}$ .

8  $MD^{(u)} \leftarrow$  применити оператор мутације на јединке у  $D^{(u)}$ .

9 Израчунати прилагођеност сваке јединке у  $D^{(u)}$ .

10  $\mathbf{x}^{(naj)} \leftarrow$  наћи најбољу јединку у  $MD^{(u)} \cup \{\mathbf{x}^{(naj)}\}$ .

11  $P^{(u+1)} \leftarrow$  саставити нову популацију од јединки у  $P^{(u)}$  и  $MD^{(u)}$ .

12  $u \leftarrow u + 1$

13 **Резултат:** генетски код најбоље пронађене јединке  $\mathbf{x}^{(naj)}$

---

GA почиње генерисањем почетне популације јединки,  $P^{(0)}$ , од којих свака представља једно решење задатог проблема. Решење које јединка представља посматра се као њен генетски код (хромозом), а сваки елемент тог решења као један ген. У наставку, јединке ћемо означавати подебљаним малим латиничним словима, индексираним где је то неопходно ( $\mathbf{x}, \mathbf{z}, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_i, \dots$ ), а генетски код јединке  $\mathbf{x}$  означаваћемо као  $\mathcal{G}[\mathbf{x}]$ .

Као мерило квалитета генетског кода јединки користи се функција прилагођености  $\varphi$ . Она изражава прилагођеност јединке условима средине, тј. задатом проблему, која се кроз еволуцију повећава. Прилагођеност јединке  $\mathbf{x}$ ,  $\varphi(\mathbf{x})$ , може се дефинисати тако да буде једнака вредности циљне функције за генетски код јединке  $\mathbf{x}$ , тј.  $\varphi(\mathbf{x}) = f(\mathcal{G}[\mathbf{x}])$ . Уколико је циљна функција  $f$  рачунски или временски захтевна, као прилагођеност се може користити нека друга функција која је у корелацији са циљном, а чије се вредности рачунају лакше и брже. У наставку ћемо сматрати да веће вредности функције прилагођености одговарају бољим вредностима функције циља (мањим у случају минимизације, већим у случају максимизације).

На почетку се генерише почетна популација јединки  $P^{(0)}$ . Јединке у њој се евалуирају и проналази најбоља међу њима, која се копира и чува као  $\mathbf{x}^{(naj)}$ . Потом се у циклусима (итерацијама) симулира процес еволуције кроз дејство генетских оператора (селекција, укрштање и мутација) који се узастопно примењују уз евалуацију јединки и замену генерација до задовољења задатог критеријума заустављања. Нека је  $u \geq 1$ . Наведени кораци изводе се на следећи начин у  $u$ -том циклусу.

(а) **Селекција.** У овом кораку, из тренутне популације јединки  $P^{(u)}$  бирају се оне које ће учествовати у стварању нове популације. Формирају се парови јединки родитеља који ће се у наредном кораку укрстити. Тако се добија група парова јединки родитеља  $R^{(u)}$ .

(б) **Укрштање.** Парови родитеља укрштају се тако што се њихови генетски кодови

комбинују. Као резултат укрштања добија се група јединки потомака  $D^{(u)}$ , које су свој генетски код делом наследиле од једног, а делом од другог родитеља.

- (в) **Мутација.** На потомке добијене укрштањем родитеља примењује се оператор мутације који с малом вероватноћом врши мале промене у њиховим генима, уносећи тако разноврсност у генетски материјал популације.
- (г) **Евалуација.** За сваку јединку потомка  $\mathbf{x} \in D^{(u)}$  рачуна се њена прилагођеност  $\varphi(\mathbf{x})$  и у случају да је  $\varphi(\mathbf{x}) > \varphi(\mathbf{x}^{(naj)})$ , генетски код јединке  $\mathbf{x}^{(naj)}$  мења се генетским кодом јединке  $\mathbf{x}$ .
- (д) **Замена генерација.** Нова популација  $P^{(u+1)}$ , на којој ће се извршити кораци наредног циклуса, саставља се од јединки потомака у  $D^{(u)}$  и јединки у популацији  $P^{(u)}$  са којом је отпочео тренутни циклус.

Описани кораци чине један циклус (итерацију) ГА. Напреднији концепти ГА користе и друге кораке као што су, између осталог, миграција и убацавање других јединки.

Алгоритам на излазу даје генетски код најбоље јединке коју је пронашао током свог рада ( $\mathbf{x}^{(naj)}$ ). Сви описани кораци могу се имплементирати на различите начине, па се у литератури и пракси може наћи више варијанти ГА.

Теоретска својства генетских алгоритама и услови неопходни за њихову конвергенцију ка оптималном решењу могу се наћи у радовима [46–49]. За скорије састављен преглед научних и индустријских примена ГА могу се погледати радови [46, 50–52].

## 3.2 Примена еволутивних метахеуристика у дизајнирању архитектура неуронских мрежа

Поред решавања бројних комбинаторних и континуалних оптимизационих проблема, генетски алгоритми и еволутивне метахеуристике све се чешће користе у потрази за оптималном архитектуром неуронских мрежа у машинском учењу [53–59]. Одређивање оптималне архитектуре представља захтеван задатак чак и за експерте, који се често ослањају на интуицију и стручно знање приликом дефинисања мрежних архитектура, не вршећи систематичну и ригорозну евалуацију различитих архитектура. Са друге стране, алгоритми претраге попут ГА могу систематично да претраже велик простор архитектура у потрази за архитектуром најбољих перформанси.

У постојећим приступима потраге за оптималном архитектуром заснованим на еволутивним метахеуристикима, могу се уочити две групе метода [59]: еволутивна претрага за мрежном архитектуром (енгл. *evolutionary neural architecture search*, ENAS) и неуроеволуција. Технике које припадају групи ENAS служе се еволутивним алгоритмима искључиво за проналажење оптималне архитектуре, док се тренирање мреже са задатом архитектуром обавља стандардним алгоритмима тренирања. Са друге стране, неуроеволутивне методе користе еволутивне метахеуристике и за проналажење оптималне архитектуре, и за тренирање резултујуће мреже.

Међутим, постојеће методе нису адекватне за развој мрежа за тестирање нормалности пошто се у свим неуроеволутивним и ENAS техникама претпоставља доступност довољно великог и квалитетног скупа података за тренирање. За разлику од те класичне претпоставке у проблемима машинског учења, у проблему који је предмет ове дисертације неопходно је конструисати и скуп за тренирање мрежа.

### 3.3 Структура јединки

Однос метапараметара и фаза нумеричког експеримента представљен је на слици 3.1. Како нам је циљ да их истовремено оптимизујемо, у предложеној имплементацији ГА репрезентација решења је таква да сваком метапараметру одговара један ген у генетском коду јединке (хромозому).

У наставку ће бити описане четири наведене групе метапараметара, као и начин на који су метапараметри представљени у хромозомима.

#### 3.3.1 Метапараметри за генерисање података

Метапараметри за генерисање података имају улогу при одређивању структуре скупа за тренирање. Наиме, избор не-нормалних расподела из којих ће потицати не-нормални узорци утиче на то које ће разлике у односу на нормалне узорке мрежа моћи да научи током тренирања, као и на то да ли ће мрежа испитивати нормалност као усмерен или омнибус тест. Уколико се не-нормални узорци генеришу из тачно одређене алтернативне расподеле или фамилије расподела, мрежа ће се понашати као усмерен тест. У супротном, уколико класу не-нормалних узорака у скупу за тренирање чине узорци добијени из различитих и разноврсних не-нормалних расподела, добијена мрежа функционисаће као омнибус тест нормалности.

У овој дисертацији искоришћен је Пирсонов систем расподела за избор не-нормалних дистрибуција на основу прва четири момента. Пирсон је идентификовао дванаест типова расподела [40–43]. Густина  $g(x)$  сваког појединачног типа представља решење следећег система диференцијалних једначина [60]:

$$\frac{g'(x)}{g(x)} = \frac{x - a}{b_0 + b_1x + b_2x^2}, \quad (3.1)$$

чији су параметри дефинисани преко другог, трећег и четвртог централног момента припадајућих расподела ( $m_2, m_3, m_4$ ;  $m_k = \mathbb{E}[(X - \mathbb{E}[X])^k]$ ,  $k = 2, 3, 4, \dots$ ):

$$a = b_1 = -\frac{m_3(m_4 + 3m_2^2)}{C} \quad (3.2)$$

$$b_2 = -\frac{m_2(4m_2m_4 - 3m_3^2)}{C} \quad (3.3)$$

$$b_3 = -\frac{2m_2m_4 - 3m_3^2 - 6m_2^3}{C} \quad (3.4)$$

$$C = 10m_4m_2 - 12m_3^2 - 18m_2^3. \quad (3.5)$$

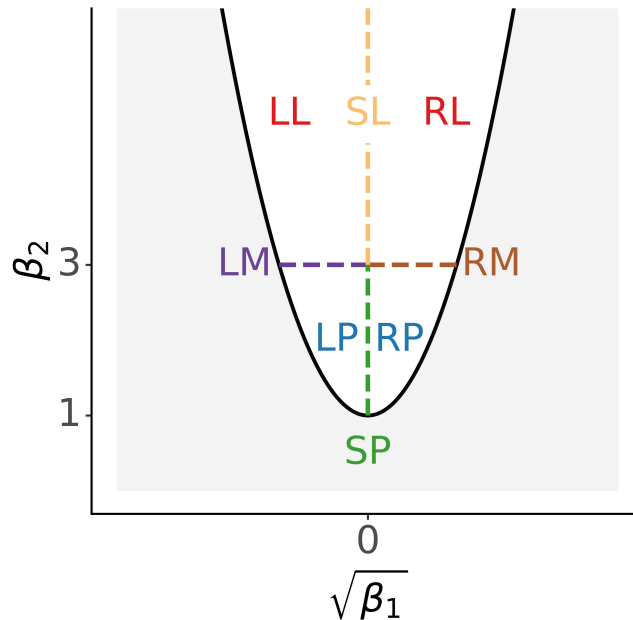
За задато очекивање, стандардну девијацију, коефицијент искошености  $\sqrt{\beta_1} = m_3/(m_2^{3/2})$  и коефицијент тежине репова  $\beta_2 = m_4/m_2^2$ , могуће је израчунати параметре  $a, b_0, b_1, b_2$  и генерисати узорке из одговарајуће расподеле. Како се нормална расподела карактерише коефицијентом искошености  $\sqrt{\beta_1} = 0$  и коефицијентом тежине репова  $\beta_2 = 3$ , разликујемо осам типова не-нормалних расподела у односу на вредности коефицијената  $\sqrt{\beta_1}$  и  $\beta_2$ . Поменутих осам типова не-нормалних расподела наведени су у табели 3.1 и илустровани на слици 3.2.

Како знак коефицијента  $\sqrt{\beta_1}$  одређује искошеност, расподеле позитивног коефицијента  $\sqrt{\beta_1}$  искошене су удесно, расподеле негативног коефицијента улево, док су симетричне оне расподеле код којих је он једнак нули. Репови дистрибуција чији је коефицијент  $\beta_2 = 3$  носе исту тежину као репови нормалне расподеле. Такве дистрибуције називају се мезокуртичне. Расподеле код којих је  $\beta_2 > 3$  имају теже репове од нормалне расподеле и зову се лептокуртичне, док код платикуртичних расподела важи  $\beta_2 < 3$  и код

њих су репови лакши него репови нормалних расподела [61]. Симетричне мезокуртичне расподеле нису уврштене у основне не-нормалне типове због тога што се у Пирсоновом систему расподела оне не разликују од нормалних.

**Табела 3.1:** Типови не-нормалних расподела у односу на опсег коефицијента искошености ( $\sqrt{\beta_1}$ ) и коефицијента тежине репова ( $\beta_2$ )

Код	Тип	$\sqrt{\beta_1}$	$\beta_2$
LP	Улево искошене платикуртичне	$(-\sqrt{\beta_2 - 1}, 0)$	$(1, 3)$
LM	Улево искошене мезокуртичне	$(-\sqrt{2}, 0)$	3
LL	Улево искошене лептокуртичне	$(-\sqrt{\beta_2 - 1}, 0)$	$(3, \infty)$
SP	Симетричне платикуртичне	0	$[1, 3)$
SL	Симетричне лептокуртичне	0	$(3, \infty)$
RP	Удесно искошене платикуртичне	$(-\sqrt{\beta_2 - 1}, 0)$	$(1, 3)$
RM	Удесно искошене мезокуртичне	$(0, \sqrt{2})$	3
RL	Удесно искошене лептокуртичне	$(-\sqrt{\beta_2 - 1}, 0)$	$(3, \infty)$



**Слика 3.2:** Графички приказ основних типова не-нормалних расподела у односу на искошеност и тежину репова.

Приметимо да су коефицијенти  $\sqrt{\beta_1}$  и  $\beta_2$  међусобно зависни јер за њих важи:

$$\beta_2 \geq \beta_1 + 1. \quad (3.6)$$

Не-нормални узорци генерисани су само за комбинације ових коефицијената које испуњавају услов (3.6).

Описани поступак производи скуп података који се могу користити за тренирање мрежа за испитивање нормалности. Поступак генерисања контролишу следећи метапараметри:

- $\beta_{2,\max}$  – максимални коефицијент тежине репова. Вредност овог метапараметра дефинише допуштене интервале коефицијената  $\sqrt{\beta_1}$  и  $\beta_2$  у складу са изабраним типом (или типовима) не-нормалних расподела и спецификацијом у табели 3.1

- $M$  – број не-нормалних расподела које ће бити насумично одабране у Пирсоновом систему за сваки не-нормални тип у алтернативној хипотези. Свака не-нормална расподела одређена је насумичним избором очекивања  $\mu$ , стандардне девијације  $\sigma$ , као и коефицијената  $\sqrt{\beta_1}$  и  $\beta_2$  из интервала који одговарају датом не-нормалном типу и вредности  $\beta_{2,\max}$ . При развоју усмерених мрежа, парови ових коефицијената бирају се само из региона  $(\sqrt{\beta_1}, \beta_2)$ -равни који одговара не-нормалном типу на који се мрежа усмерава. У случају омнибус мрежа, коефицијенти се бирају из целе равни тако да свих осам основних не-нормалних типова буде једнако заступљено. Број нормалних расподела једнак је броју не-нормалних расподела.
- $L$  - број узорака који ће за сваки задати обим  $n$  бити генерисани из сваке не-нормалне расподеле одабране за састављање скупа за тренирање.
- $PD$  – индикатор који одређује да ли ће сваки генерисан пар  $(\mu, \sigma)$  бити искоришћен и за дефинисање једне нормалне расподеле  $\mathcal{N}(\mu, \sigma^2)$ , и за дефинисање једне не-нормалне дистрибуције у сваком типу наведеном у алтернативној хипотези. Нека је  $D_{\mathcal{N}}$  скуп парова  $\{(\mu_k, \sigma_k)\}_{k=1}$  који дефинишу нормалне расподеле, а  $SK_{-\mathcal{N}}$  скуп парова  $\{(\sqrt{\beta_{1,k}}, \beta_{2,k})\}_{k=1}$  за дати тип не-нормалних расподела. У случају да је  $PD = \top$ , не-нормалне расподеле одређене су уређеним четворкама  $\{(\mu_k, \sigma_k, \sqrt{\beta_{1,k}}, \beta_{2,k})\}_{k=1}^M$ . У супротном, очекивање и девијација не-нормалних узорака бирају се насумично из унапред фиксираних интервала.

Следећи метапараметри за генерисање података постављени су на фиксне вредности и нису оптимизовани генетским алгоритмом:

- Интервал из којег се насумично бира очекивање  $\mu$ :  $[-100, 100]$ .
- Интервал из којег се насумично бира девијација  $\sigma$ :  $[10^{-2}, 10^2]$ .
- Обим узорака. У овом експерименту су за тренирање генерисани узорци обима  $n = 10, 20, \dots, 100$ .

Кроз формулацију функције прилагођености (в. одељак 3.5) и стандардизацију узорака (в. одељак 3.3.2) обезбеђено да перформансе добијених мрежа не зависе од ових фиксираних метапараметара.

Сваки скуп за тренирање је балансиран и садржи исти број нормалних као и не-нормалних узорака. Као и у раду [6], усвојена је конвенција да нормални узорци чине позитивну класу (с знаком 1), а не-нормални узорци негативну класу (с знаком 0).

### 3.3.2 Метапараметри репрезентације

Метапараметри репрезентације одређују димензије простора у који се узорци пре-сликавају пре примене неуронске мреже. Свака димензија представља атрибут репрезентације. Фиксирање димензија решава проблем неограниченог обима узорака као и проблем редоследа димензија.

#### Стандардизација

Под стандардизацијом се у овој дисертацији подразумева рачунски поступак који узорак  $x = \{x_i\}_{i=1}^n$  преводи у форму  $z = \{z_i\}_{i=1}^n$  такву да се на основу  $z_1, z_2, \dots, z_n$  не могу израчунати средина  $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$  и девијација  $\text{sd}(x) = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$ . Нека је  $\text{med}(x)$  медијана, а  $\text{mad}(x)$  просечна апсолутна разлика елемената узорка  $x$  од аритметичке средине  $\bar{x}$ . У овој дисертацији разматрамо четири стандардизациона поступка

који улазни узорак  $x$  преводи у његову стандардизовану форму  $z$ , где се елементи  $z_i$  ( $i = 1, 2, \dots, n$ ) рачунају на следећи начин:

$$z_i = s_1(x_i) = (x_i - \bar{x})/\text{sd}(x) \quad (3.7)$$

$$z_i = s_2(x_i) = (x_i - \text{med}(x))/\text{mad}(x) \quad (3.8)$$

$$z_i = s_3(x_i) = (x_i - \min(x))/(\max(x) - \min(x)) \quad (3.9)$$

$$z_i = s_4(x_i) = -3 + 6 \cdot (x_i - \min(x))/(\max(x) - \min(x)). \quad (3.10)$$

Мотивација за стандардизациони поступак (3.10) јесте што он преводи узорке у опсег  $[-3, 3]$ , који садржи скоро целокупну стандардну нормалну расподелу  $\mathcal{N}(0, 1)$ .

Атрибути репрезентација разматраних у овој дисертацији могу се сврстати у две групе: предстандардизационе атрибуте, који се одређују пре, и постстандардизационе, који се рачунају после стандардизације.

### Предстандардизациони атрибути

У предстандардизационе атрибуте спадају  $p$ -вредности стандардних статистичких тестова нормалности насумично одабраних из скупа који садржи омнибус тестове SW, SF, DP, ZC и ZA, као и усмерене тестове  $ST_{<0}$ ,  $ST_{>0}$ ,  $KT_{<3}$ , и  $KT_{>3}$ .

Постоје два мотива за коришћење  $p$ -вредности као димензија репрезентационих простора:

- Будући да различити статистички тестови нормалности узимају у обзир различите аспекте узорака, пресликавање узорака у векторе који садрже  $p$ -вредности требало би да омогући мрежама да класификациона правила науче из комплементарних атрибута и тако постигну боље резултате него појединачни тестови.
- Да су уместо  $p$ -вредности коришћене статистике наведених тестова, претпоставка је да би мрежи био потребан велик скуп за тренирање како би научила расподеле тих статистика, односно, како би научила које вредности ових статистика су добри индикатори нормалности (или јаки показатељи не-нормалности). Коришћењем  $p$ -вредности, које се рачунају на основу познате расподеле статистике теста, мрежа прескаче корак учења тих дистрибуција, па је оправдано претпоставити да ће јој бити довољни релативно мали скупови за обучавање.

### Постстандардизациони атрибути

У постстандардизационе атрибуте спадају дескриптивне статистике стандардизоване форме  $z$ .

Приликом генерисања јединки у популацији GA, прва група тих дескриптивних статистика насумично се бира као подскуп следећег скупа статистика:

$$\{n, \bar{z}, \text{med}(z), \text{sd}(z), \text{mad}(z), \sqrt{\beta_1}(z), \beta_2(z)\}. \quad (3.11)$$

Другу групу статистика чине квантили редова  $q, 2q, \dots, \lfloor 1/q \rfloor q$  који се добијају из стандардизованог узорка  $z$ , при чему је  $q$  један од репрезентационих метапараметара. За задату позитивну вредност метапараметра  $q$ , у постстандардизационе атрибуте укључују се сви наведени квантили.

У трећој групи статистика налазе се  $\text{LM}(z)$  и  $\text{RM}(z)$ , које представљају аритметичку средину левог и десног репа стандардизованог узорка  $z$ . Нека је  $\hat{F}_{n,z}$  емпиријска функција расподеле стандардизованог узорка  $z$ . Његов леви и десни реп дефинишемо преко

метапараметара  $m_L$  и  $m_R$  на следећи начин:

$$\begin{aligned} \text{LT}(z) &= \left\{ z_i \in z : \widehat{F}_{n,z}(z_i) \leq m_L \right\} \\ \text{RT}(z) &= \left\{ z_i \in z : \widehat{F}_{n,z}(z_i) \geq 1 - m_R \right\}. \end{aligned} \quad (3.12)$$

Позитивне вредности ових метапараметара показују да се у постстандардизационе атрибуте укључују и средина левог, односно десног репа.

### Преглед метапараметара репрезентације

Метапараметри репрезентације које оптимизујемо генетским алгоритмом су:

- $R \in \{s_1, s_2, s_3, s_4\}$  – избор стандардизационог поступка.
- $PV$  – подскуп скупа тестова  $\{SW, SF, DP, ZC, ZA, ST_{<0}, ST_{>0}, KT_{<3}, KT_{>3}\}$  чије ће  $p$ -вредности бити употребљене као предстандардизациони атрибути.
- $F$  – подскуп скупа статистика  $\{n, \bar{z}, \text{med}(z), \text{sd}(z), \text{mad}(z), \sqrt{\beta_1}(z), \beta_2(z), \text{LM}(z), \text{RM}(z)\} \cup h_q(z)$  које ће бити искоришћене као постстандардизациони атрибути.
- $q \in \{0, 0.01, 0.02, 0.05, 0.1, 0.2\}$  – разлика редова узастопних квантила. Вредност нула значи да квантили нису одабрани као постстандардизациони атрибути.
- $m_L, m_R \in \{0, 0.1, 0.15, 0.2\}$  – тежине левог и десног репа за израчунавање статистика  $\text{LM}(z)$  и  $\text{RM}(z)$ . Вредност нула значи да статистика није одабрана.

### 3.3.3 Метапараметри мрежне архитектуре

У литератури су предложене разне архитектуре неуронских мрежа. Класичне мреже су једносмерне неуронске мреже (енгл. *feedforward neural networks*). У њиховој архитектури разликујемо следеће елементе.

- Улазни слој, чији сваки неурон прихвата на улазу вредност само једне димензије улазних вектора и прослеђује је непромењену на излазу. Овај слој садржи и један додатан неурон који није повезан ни са једном димензијом улаза. Његова излазна вредност увек је једнака јединици (тзв. слободан неурон; енгл. *the bias unit*).
- Низ од  $d$  скривених (унутрашњих) слојева, који сви садрже исти број неурона. У сваком слоју, један неурон је слободан и на излазу увек враћа јединицу, а преосталих  $w$  неурона, који се називају везани, повезано је са свим неуронима из претходног слоја. Сваки од тих  $w$  неурона у скривеном слоју прихвата на улазу тежинску суму излаза свих неурона из претходног слоја и враћа на излазу резултат примене активационе функције на ту суму. У најједноставнијим архитектурама, сви неурони користе исту активациону функцију за израчунавање излаза. Нека је  $a_i^{(j)}$  излаз  $i$ . неурона у  $j$ . скривеном слоју ( $i = 0, 1, 2, \dots, w; j = 1, 2, \dots, d$ ), где  $i = 0$  означава слободан неурон ( $a_0^{(j)} = 1$ ). Нека је  $a_i^{(0)}$  ознака за излаз  $i$ -тог неурона улазног слоја, и нека је  $a$  активациона функција. Нека је  $\theta_{ik}^{(j)}$  тежина којом се множи излаз  $a_i^{(j-1)}$  у суми која представља улаз за  $k$ -ти неурон у слоју  $j$  ( $i = 0, 1, \dots, w; k = 1, 2, \dots, w; j = 1, 2, \dots, d$ ). Тада, излаз  $k$ -тог неурона у скривеном слоју  $j = 1, 2, \dots, d$  рачунамо формулом (3.13):

$$a_k^{(j)} = a \left( \sum_{i=0}^w \theta_{ik}^{(j)} a_i^{(j-1)} \right). \quad (3.13)$$

- Последњу компоненту чини излазни слој, чија структура зависи од природе проблема који се решава. У бинарној класификацији, где се класе уобичајено називају негативна и позитивна и означавају редом бројевима 0 и 1, последњи слој се састоји од једног неурона који на излазу даје процену вероватноће да улазни вектор припада класи 1. Алтернатива томе јесте излазни слој са два неурона, од којих један даје процену вероватноће да улазни вектор припада класи 0, а други процену вероватноће да он припада класи 1. Између ове две структуре не постоји разлика ако се претпостави да улазни објекти припадају или негативној или позитивној класи пошто ће тада суме вероватноћа два излазна неурона бити комплементарне.

Метапараметри архитектуре једносмерних мрежа су:

- $d \in [1, 20]$  – број скривених слојева, који зовемо још и дубина мреже.
- $w \in [1, 200]$  – број везаних неурона у сваком скривеном слоју, односно, ширина мреже.
- $a \in \{\text{ReLU}, \tanh, \text{sigmoid}\}$  – активациона функција.

### 3.3.4 Метапараметри тренирања

У метапараметре тренирања спадају индикатор  $A$  који показује који алгоритам ће бити искоришћен за обучавање мреже, параметри одабраног алгоритма, као и метапараметри који дефинишу тип и степен регуларизације.

У литератури је формулисано више алгоритама за тренирање мрежа и других модела машинског учења. За развој једносмерних мрежа у овој дисертацији одабрани су следећи алгоритми: ADAM [62], SGD [63–65] и LBFGS [66, 67].

Регуларизациони метапараметри су регуларизациона константа  $c \in [10^{-5}, 10^{-2}]$  и  $ES \leq 0.2$ , удео скупа за тренирање који ће бити употребљен као валидациони скуп за рано заустављање [68].

Параметри алгоритама су:

- За алгоритам SGD:
  - ★  $LRI \in [10^{-5}, 10^{-1}]$  – почетна стопа учења.
  - ★  $LRU$  – механизам промене стопе учења, при чему стопа може бити адаптивна, инверзна или константна [63, 65].
  - ★  $m \in (0, 1)$  - вредност момента.
  - ★  $NM \in \{\perp, \top\}$  – логички индикатор који показује да ли се користи Нестеровљев момент.
  - ★  $I \in \{10, 11, \dots, 15\}$  – максималан дозвољен број узастопних итерација без побољшања губитка након којих се прекида тренирање.
  - ★  $t \in [1/4, 3/4]$  – експонент за стопу учења.
- За алгоритам ADAM:
  - ★  $LRI$  и  $LRU$  – исти метапараметри као за алгоритам SGD.
  - ★  $b_1, b_2 \in [0, 1)$  – стопе опадања момената.
- За алгоритам LBFGS:
  - ★  $MF$  – максималан број евалуација функције губитка током тренирања.

Нека је  $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^s$  скуп за тренирање ( $y_i \in \{0, 1\}$ ,  $i = 1, 2, \dots, s$ ), при чему је  $\mathbf{x}_i$  репрезентациони вектор  $i$ . узорка, а  $y_i$  индикатор којој класи припада (1 за нормалне, а 0 за не-нормалне узорке). Нека је  $p_i$  излаз мреже за улаз  $\mathbf{x}_i$ , тј. процењена вероватноћа да он представља нормалан узорак. Циљ сваког од алгоритама ADAM, SGD и LBFGS јесте да минимизују просечан логистички губитак, који се рачуна формулом (3.14):

$$-\frac{1}{s} \sum_{i=1}^s (y_i \log p_i + (1 - y_i) \log(1 - p_i)). \quad (3.14)$$

Овај губитак се уобичајено користи приликом тренирања бинарних класификатора [69].

### 3.3.5 Врсте хромозома

Вредности описаних метапараметара представљају гене хромозома јединке у популацији GA. Сваки хромозом представља једну комбинацију вредности метапараметара, па одговара једном решењу. Како сваки од коришћених алгоритама за тренирање има своје метапараметре, разликујемо три врсте хромозома у зависности од изабраног алгорита (SGD, ADAM или LBFGS). Слика 3.3 илуструје три типа хромозома. Треба приметити да разлика постоји само у сегменту хромозома који се односи на тренирање, док су остала три сегмента иста.



Слика 3.3: Три врсте хромозома чија се структура разликује по гену који одређује избор алгорита за тренирање и генима који представљају метапараметре датог алгорита.

Како постоје разлике у дужини и врсти кода хромозома јединки у популацији GA, генетске операторе укрштања и мутације потребно је дизајнирати тако да функционишу и са хромозомима различите дужине. При томе је кључно да оператори узму у обзир вредност гена  $A$  (индикатор алгорита за тренирање), који одређује присуство других гена (оних својствених одабраном алгоритму).

## 3.4 Генетски оператори

Итеративном применом генетских оператора селекције, укрштања и мутације, GA оптимизује хромозоме јединки и повећава им степен прилагођености.

### 3.4.1 Селекција

Као оператор селекције примењен је рулетски одабир. Нека је  $\varphi_i$  вредност функције прилагођености  $i$ -те јединке у тренутној популацији у датом циклусу ( $i = 1, 2, \dots, N$ ).

Вероватноћа  $\pi_i$  да  $i$ -та јединка буде изабрана у пар јединки родитеља пропорционална је њеној прилагођености:

$$\pi_i = \frac{\varphi_i}{\sum_{j=1}^N \varphi_j}. \quad (3.15)$$

Селекција се завршава када се описаном методом генерише  $\sigma$  парова родитеља. Свака јединка може учествовати у више парова родитеља у истом циклусу, али сваки пар мора садржати различите јединке. Поступак селекције представљен је алгоритмом 3.2. Скуп парова родитеља које овај оператор враћа на излазу у  $u$ -тој итерацији GA биће у даљем тексту означен као  $R^{(u)}$ .

---

### Алгоритам 3.2: Оператор селекције

---

**Улаз:**  $P = [\mathbf{x}_i]_{i=1}^n$  – популација од  $N$  јединки из којих се бирају родитељи;  $\sigma$  – број парова родитеља које треба изабрати;  $\varphi_1, \varphi_2, \dots, \varphi_N$  – вредност функције прилагођености јединки  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$  у популацији  $P$ .

**Израз:** Низ  $\sigma$  парова јединки родитеља.

- 1 за  $k \leftarrow 1, 2, \dots, N$  :
  - 2     $\left[ \pi_k \leftarrow \frac{\varphi_k}{\sum_{i=1}^N \varphi_i} \right.$
  - 3  $R \leftarrow \emptyset$
  - 4 док  $|R| < \sigma$  :
  - 5     $\left[ \mathbf{x}_i, \mathbf{x}_j \leftarrow \text{изабрати две јединке } (i \neq j) \text{ у складу са вероватноћама } \pi_1, \pi_2, \dots, \pi_N. \right.$
  - 6     $\left[ R \leftarrow R \cup \{(\mathbf{x}_i, \mathbf{x}_j)\} \right.$
  - 7 **Резултат:**  $R$
- 

### 3.4.2 Укрштање

Укрштање је представљено у алгоритму 3.3.

---

### Алгоритам 3.3: Оператор укрштања

---

**Улаз:**  $(\mathbf{x}_i, \mathbf{x}_j)$  – пар јединки родитеља;  $S_{[0,1]}$  – функција чији сваки позив враћа насумично одабран број из одсечка  $[0, 1]$ .

**Израз:**  $z$  – потомак јединки  $\mathbf{x}_i$  и  $\mathbf{x}_j$ .

- 1  $\mathcal{G}[z] \leftarrow$  генерисати празан хромозом.
  - 2 за сваки метапараметар  $g \in [\beta_{2,\max}, M, L, PD, R, PV, F, q, m_L, m_R, d, w, a, c, ES]$  :
  - 3     $\left[ \text{ако } S_{[0,1]}() \leq 1/2 \text{ :} \right.$
  - 4        $\left[ \mathcal{G}[z][g] \leftarrow \mathcal{G}[\mathbf{x}_i][g] \right.$
  - 5       **иначе :**
  - 6        $\left[ \mathcal{G}[z][g] \leftarrow \mathcal{G}[\mathbf{x}_j][g] \right.$
  - 7 **ако**  $S_{[0,1]}() \leq 1/2$  :
  - 8     $\left[ \mathcal{G}[z][A] \leftarrow \mathcal{G}[\mathbf{x}_i][A] \right.$
  - 9     $\left[ \text{У хромозом } \mathcal{G}[z] \text{ додати све гене алгоритма за тренирање од родитеља } \mathbf{x}_i. \right.$
  - 10 **иначе :**
  - 11     $\left[ \mathcal{G}[z][A] \leftarrow \mathcal{G}[\mathbf{x}_j][A] \right.$
  - 12     $\left[ \text{У хромозом } \mathcal{G}[z] \text{ додати све гене алгоритма за тренирање од родитеља } \mathbf{x}_j. \right.$
  - 13 **Резултат:**  $z$
- 

Нека је  $(\mathbf{x}_i, \mathbf{x}_j) \in R^{(u)}$  ( $\{i, j\} \subseteq \{1, 2, \dots, N\}$ ) пар јединки родитеља састављен током селекције у  $u$ -том циклусу. Хромозом јединке потомка добија се укрштањем хромозома јединки родитеља  $\mathbf{x}_i$  и  $\mathbf{x}_j$  на следећи начин. Итерацијом се пролази кроз низ гена за

генерисање података, репрезентацију, архитектуру, као и гене који одређују метапараметре тренирања  $ES$ ,  $c$  и  $A$ . За сваки ген насумично се генерише број између 0 и 1. Уколико је он мањи од  $1/2$ , вредност одговарајућег гена преузима од родитеља  $x_i$ , а у супротном, од родитеља  $x_j$ .

Услед хијерархијског односа између индикатора алгоритма за тренирање и њему својствених параметара, јединка потомак преузима гене алгоритма за тренирање од родитеља од којег је наследила индикатор алгоритма  $A$ .

### 3.4.3 Мутација

Оператор мутације примењује се на јединке добијене укрштањем парова родитеља. Вероватноћа да ће јединка потомак бити изабрана за мутацију дефинисана је као параметар генетског алгоритма  $\mu_P$ . Петљом се пролази кроз све гене одабране јединке и за сваки од њих се изводи појединачна мутација са вероватноћом  $\mu_G$ . При томе, разликујемо неколико случајева у зависности од природе гена.

- Ако је за мутацију изабран категоријски ген (као што је метапараметар активационе функције  $a$ ), тренутна вредност тог гена мења се насумично одабраном категоријом из домена одговарајућег метапараметра.
- Ако је у питању ген чија је вредност скуп, као што је случај са метапараметрима  $PV$  и  $F$ , нова вредност се добија тако што се сваки елемент гена искључује из њега са вероватноћом  $\Delta$ , а сваки могући елемент, који није присутан у тренутном скупу, са истом вероватноћом додаје том гену. Вредност  $\Delta$  се као параметар задаје генетском алгоритму приликом покретања.
- Код нумеричких гена,  $\Delta$  представља максималну релативну разлику између нове и старе вредности. Ако се добије вредност ван допуштеног домена, она се пресликава у границу домена која јој је ближа.

При томе, ако је за мутацију изабран индикатор алгоритма  $A$ , онда се мутирају и сви метапараметри својствени индикованом алгоритму. Оператор мутације представљен је у алгоритму 3.4.

---

#### Алгоритам 3.4: Оператор мутације

---

**Улаз:**  $x$  – јединка чији хромозом подлеже мутацији;  $\mu_G$  – вероватноћа мутације појединачног гена;  $\Delta$  – параметар мутације појединачних гена;  $S_{[0,1]}$  – функција чији сваки позив враћа насумично одабран број из одсечка  $[0, 1]$ ;

**Издаз:** Јединка  $x$  са мутираним хромозомом.

- 1 **ако**  $S_{[0,1]}() \leq \mu_G$  :
- 2     Мутирати ген  $A$  у хромозому  $\mathcal{G}[x]$ .
- 3     Одстранити из  $\mathcal{G}[x]$  гене својствене старом алгоритму.
- 4     Додати гене својствене новом алгоритму, насумично одређујући њихове вредности.
- 5 **за сваки** ген  $g \in \mathcal{G}[x]$  сем гена  $A$  :
- 6     **ако**  $S_{[0,1]}() \leq \mu_G \wedge \neg(g \text{ је својствен алгоритму за тренирање } \wedge \text{ ген } A \text{ је мутиран})$  :
- 7         Мутирати ген  $g$ .

8 **Резултат:**  $x$

---

## 3.5 Функција прилагођености

Природна дефиниција прилагођености јединке  $\mathbf{x}$  јесте класификациона тачност мреже добијене према спецификацији метапараметара у генетском коду  $\mathcal{G}[\mathbf{x}]$ .

Међутим, тачност моделâ машинског учења може се само проценити на основу скупа података за тестирање, али не и израчунати егзактно. Уколико би се за евалуацију јединки у свим итерацијама GA увек користили узорци из истих расподела, GA би могао да оптимизује метапараметре за генерисање података тако да репродукују скуп за тестирање. Тада би постојала опасност да резултујућа мрежа научи да класификује само тестне дистрибуције, услед чега би јој тачност била знатно нижа на узорцима из других расподела. Овај проблем може се превазићи тако што се у сваком циклусу GA насумично бирају нове расподеле из којих се генеришу узорци за тестирање.

Са друге стране, тачност није једина мера квалитета мреже. Као што је објашњено у одељку 2.4.1, приликом евалуације мрежа за тестирање нормалности у обзир се мора узети и њихова стабилност у односу на варијабилност расподела коришћених за састављање скупа за тренирање. Поред тога, од мрежа се очекује и да им тачност расте са обимом узорака на којима их евалуирамо, тако да прилагођеност треба да урачуна и облик зависности тачности од обима узорака.

У овој дисертацији, функцију прилагођености рачунамо као тежинску разлику тачности и оцена нестабилности и дефективности облика:

$$\text{прилагођеност} = \text{тачност} - \text{тежина} \times (\text{нестабилност} + \text{дефективност облика}). \quad (3.16)$$

Како се евалуациони узорци мењају, неопходно је у сваком циклусу евалуирати како јединке потомке, тако и јединке у постојећој популацији  $P^{(u)}$ . У супротном би се приликом замене генерација поредили резултати на различитим тестним скуповима.

### 3.5.1 Тачност

Нека је  $\mathcal{D}_{eval,0}^{(u)}$  тестни скуп који користимо за евалуацију јединки у  $u$ . циклусу генетског алгоритма. Очекивања и стандардне девијације расподела бирамо насумично из истих интервала као приликом генерисања скупа за тренирање. За дефинисање не-нормалних узорака користимо насумично одабране моменте  $\sqrt{\beta_1}$  и  $\beta_2$  из интервала  $[-10, 10]$  и  $[0, 100]$ , поштујући релацију (3.6) и усмереност мреже. Међутим, за разлику од скупа за тренирање, за скуп  $\mathcal{D}_{eval,0}^{(u)}$  користимо обиме узорака  $[n_v]_{v=1}^9 = \{15, 25, \dots, 95\}$  ( $n_v = 15 + 10(v - 1)$ ,  $v = 1, 2, \dots, 9$ ). Тиме и кроз сам рачун тачности испитујемо способност мреже да успешно класификује узорке другачије од оних на којима је обучавана. За сваку комбинацију обима и типа генерисано је по 10 000 узорака да би евалуација мреже била прецизна.

Нека је  $a^{(u,0)}$  укупна тачност мреже, израчуната на скупу  $\mathcal{D}_{eval,0}^{(u)}$ , и нека је  $a_{n,j}^{(u,0)}$  њена тачност на узорцима из скупа  $\mathcal{D}_{eval,0}^{(u)}$  са  $n$  елемената добијених из расподела типа  $j$  ( $j \in \{\mathcal{N}\} \cup \mathcal{T}$ ). При томе,  $\mathcal{N}$  представља нормалне расподеле, а  $\mathcal{T}$  је скуп не-нормалних типова:  $\{\text{LL, RL, SL, LP, RP, SP, LM, RM}\}$  за омнибус мреже, односно  $\mathcal{T} = \{\mathcal{T}\} \subset \{\text{LL, RL, SL, LP, RP, SP, LM, RM}\}$  за мреже усмерене на не-нормалан тип  $\mathcal{T}$ .

Ако је  $[y_i]_{i=1}^n \in \{0, 1\}^s$  низ правих класа евалуационих узорака у  $u$ -том циклусу, а  $[p_i]_{i=1}^n \in [0, 1]^s$  низ процењених вероватноћа нормалности, тачност рачунамо на следећи начин:

$$a^{(u,0)} = \frac{1}{s} \sum_{i=1}^s \mathbf{1} \{ \mathbf{1} \{ p_i \geq 1/2 \} = y_i \}. \quad (3.17)$$

### 3.5.2 Нестабилност

Нека је  $\mathcal{D}_{eval,1}^{(u)}$  скуп узорака генерисан на исти начин као и  $\mathcal{D}_{eval,0}^{(u)}$ , с том разликом што су очекивања и девијације расподела из одсецака  $[1000, 2000]$  и  $[10^3, 10^4]$ . Слично, нека је  $\mathcal{D}_{eval,2}^{(u)}$  скуп узорака из расподела чија су очекивања и девијације из одсецака  $[-2000, -1000]$  и  $[10^{-4}, 10^{-3}]$ , док је у свему осталом поступак генерисања исти као за скуп  $\mathcal{D}_{eval,0}^{(u)}$ . Нека су  $a_{n,j}^{(u,1)}$  и  $a_{n,j}^{(u,2)}$  дефинисани аналогно  $a_{n,j}^{(u,0)}$ . Уколико је поставка метапараметара дате јединке резултирала стабилном мрежом, тачности  $a_{n,j}^{(u,1)}$  и  $a_{n,j}^{(u,2)}$  не би требало да се разликују много од тачности  $a_{n,j}^{(u,0)}$  ни за једно  $j$  и  $n$ .

Нека је  $c_{\mathcal{N},k}^{(u)}$  оцена нормалне нестабилности дефинисана као највећа апсолутна разлика између тачности дате мреже на нормалним узорцима истог обима у скуповима  $\mathcal{D}_{eval,0}^{(u)}$  и  $\mathcal{D}_{eval,k}^{(u)}$  ( $k = 1, 2$ ):

$$c_{\mathcal{N},k}^{(u)} = \max_n \left| a_{n,\mathcal{N}}^{(u,0)} - a_{n,\mathcal{N}}^{(u,k)} \right|. \quad (3.18)$$

Нека не-нормална нестабилност  $c_{-\mathcal{N},k}^{(u)}$  за омнибус мреже представља максималну апсолутну разлику просечне тачности на узорцима из расподела не-нормалних типова:

$$c_{-\mathcal{N},k}^{(u)} = \max_{j \in \Gamma} \left| \frac{1}{9} \sum_{v=1}^9 a_{n_v,j}^{(u,0)} - \frac{1}{9} \sum_{v=1}^9 a_{n_v,j}^{(u,k)} \right| \quad k = 1, 2. \quad (3.19)$$

За мреже усмерене ка  $\mathcal{T}$ ,  $c_{-\mathcal{N},k}^{(u)}$  рачунамо аналогно као  $c_{\mathcal{N},k}^{(u)}$ :

$$c_{-\mathcal{N},k}^{(u)} = \max_n \left| a_{n,\mathcal{T}}^{(u,0)} - a_{n,\mathcal{T}}^{(u,k)} \right|. \quad (3.20)$$

Оцену укупне нестабилности  $c^{(u)}$  дефинишемо као просек нормалних и не-нормалних нестабилности израчунатих помоћу скупова  $\mathcal{D}_{eval,l}^{(u)}$  ( $k = 1, 2$ ):

$$c = \frac{1}{4} \left( c_{\mathcal{N},1}^{(u)} + c_{-\mathcal{N},1}^{(u)} + c_{\mathcal{N},2}^{(u)} + c_{-\mathcal{N},2}^{(u)} \right) = \frac{1}{2} \sum_{k=1}^2 \frac{1}{2} \left( c_{\mathcal{N},k}^{(u)} + c_{-\mathcal{N},k}^{(u)} \right). \quad (3.21)$$

Дефинишући  $c_k^{(u)} = \frac{1}{2} \left( c_{\mathcal{N},k}^{(u)} + c_{-\mathcal{N},k}^{(u)} \right)$  као нестабилност процењену помоћу скупа  $\mathcal{D}_{eval,k}^{(u)}$  ( $k = 1, 2$ ), видимо да се укупна нестабилност може израчунати и као просек нестабилности на евалуационим скуповима  $\mathcal{D}_{eval,1}^{(u)}$  и  $\mathcal{D}_{eval,2}^{(u)}$ :

$$c_k^{(u)} = \frac{1}{2} \left( c_1^{(u)} + c_2^{(u)} \right). \quad (3.22)$$

### 3.5.3 Дефективност облика

Дефективност облика показује да ли тачност мреже опада са порастом обима узорка,  $n$ . Мреже са тачношћу опадајућом по обиму узорака представљају неинтуивине класификаторе будући да очекујемо да мрежа буде тачнија на већим узорцима.

Укупну дефективност облика рачунамо на основу дефективности одређених за сваки евалуациони скуп  $\mathcal{D}_{eval,k}^{(u)}$  ( $k = 0, 1, 2$ ). Нека је  $b_{\mathcal{N},k}^{(u)}$  дефективност облика на нормалним узорцима у скупу  $\mathcal{D}_{eval,k}^{(u)}$  ( $k = 0, 1, 2$ ):

$$b_{\mathcal{N},k} = \sum_{v=2}^9 \left( \left| a_{n_v,\mathcal{N}}^{(u,k)} - a_{n_{v-1},\mathcal{N}}^{(u,k)} \right| \cdot \mathbf{1} \left\{ a_{n_v,\mathcal{N}}^{(u,k)} < a_{n_{v-1},\mathcal{N}}^{(u,k)} - \delta \right\} \right), \quad (3.23)$$

где је  $\delta$  максимални праг за који тачност на мањим узорцима може бити већа од тачности на већим узорцима, а да се то не урачуна у дефективност.

Дефективност на не-нормалним узорцима дефинишемо на сличан начин:

$$b_{-\mathcal{N},k}^{(u)} = \sum_{j \neq \mathcal{N}} \sum_{v=2}^9 \left( \left| a_{n_v,j}^{(u,k)} - a_{n_{v-1},j}^{(u,k)} \right| \cdot \mathbf{1} \left\{ a_{n_v,j}^{(u,k)} < a_{n_{v-1},j}^{(u,k)} - \delta \right\} \right). \quad (3.24)$$

Дефективност облика на скупу  $\mathcal{D}_{eval,k}^{(u)}$  ( $k = 0, 1, 2$ ) тада је:

$$b_k = \frac{1}{2} (b_{\mathcal{N},k}^{(u)} + b_{-\mathcal{N},k}^{(u)}), \quad (3.25)$$

док укупну дефективност облика  $b^{(u)}$  дефинишемо као аритметичку средину дефективности на појединачним евалуационим скуповима:

$$b^{(u)} = \frac{1}{3} \sum_{k=1}^3 b_k^{(u)}. \quad (3.26)$$

### 3.5.4 Одређивање тежина

Прилагођеност  $\varphi^{(u)}(\mathbf{x})$  јединке  $\mathbf{x}$  у  $u$ -том циклусу генетског алгоритма рачунамо умањујући тачност  $a^{(u,0)}$  мреже јединке  $\mathbf{x}$  на евалуационом скупу  $\mathcal{D}_{eval,0}^{(u)}$  за тежинску суму оцена нестабилности и дефективности облика:

$$\varphi^{(u)}(\mathbf{x}) = a^{(u,0)} - \omega (b^{(u)} + c^{(u)}), \quad (3.27)$$

где је  $\omega$  тежина задата као параметар генетског алгоритма на улазу.

Вредности  $\omega \in (0, 1)$  умањују значај стабилности и облика криве тачности, што значи да тежину  $\omega$  треба поставити на вредност већу од 1. Међутим, тежина не треба да буде ни превелика зато што би се тако изгубиле разлике између тачних, али мало или умерено нестабилних и дефектних мрежа и мрежа које су потпуно неупотребљиве као класификатори. Наиме, јединка чија мрежа сваки узорак проглашава нормалним имала би тачност 0.5, док би јој све оцене нестабилности и дефективности облика биле једнаке нули. Претпоставимо да је  $\mathbf{x}$  таква јединка. Колика год да је тежина  $\omega$ , њена прилагођеност би била  $\varphi(\mathbf{x}) = 0.5$ . Претпоставимо да јединка  $\mathbf{z}$  има високу тачност од 0.8, а нестабилност 0.1 и дефективност облика 0.1. По датим оценама може се закључити да је мрежа јединке  $\mathbf{z}$  релативно стабилна и да њена тачност не опада знатно како обим испитиваних узорака расте. Али, ако би се користила тежина  $\omega = 1.5$ , укупна прилагођеност јединке  $\mathbf{z}$  износила би  $\varphi(\mathbf{z}) = 0.5$ . То значи да би она имала исту шансу као јединка  $\mathbf{x}$  да је током селекције одаберемо као јединку родитеља, иако је  $\mathbf{z}$  очигледно бољи избор. Стога је најбоље тежину  $\omega$  поставити на вредност из интервала  $(1, 1.5)$ .

## 3.6 Замена генерација

Замена генерација представља последњи корак у циклусу генетског алгоритма. У овом кораку се јединке са најнижом вредношћу функције прилагођености у тренутној популацији замењују потомцима добијеним у истом циклусу. У општој формулацији генетског алгоритма, представљеној у псеудокоду 3.1, замена генерација у обзир узима само прилагођеност јединки.

Таква замена адекватна је онда када је прилагођеност детерминистичка функција, што је случај у већини оптимизационих проблема за чије су решавање у литератури

коришћени генетски алгоритми прилагођени конкретном problemu. У problemu који је предмет ове дисертације, функција прилагођености је недетерминистичка: иста јединка у два различита циклуса може имати другачију вредност функције прилагођености. Да би се стекла потпунија слика о квалитету мреже коју носи нека јединка, неопходно је акумулирати све прилагођености израчунате за њу. Такође, важно је размотрити и брзину тренирања мреже, а не само тачност, стабилност и облик криве тачности. У адаптацији генетског алгоритма развијеној у оквиру ове дисертације, замена генерација изводи се у две фазе, које описујемо за  $u$ . циклус.

У првој фази сортирамо јединке  $P^{(u)} \cup MD^{(u)}$  у неоппадајућем поретку по прилагођености израчунатој у тренутном циклусу. При томе, ако се прилагођеност две јединке разликује за мање од  $\eta$ , као боља се сматра јединка чија се мрежа тренира брже. Тако добијамо сортирани низ јединки  $[\mathbf{x}_{(1)}^{(u)}, \mathbf{x}_{(2)}^{(u)}, \dots, \mathbf{x}_{(N+\sigma)}^{(u)}]$ . Нека је  $t_{(i)}^{(u)}$  време за које је тренирана мрежа јединке  $\mathbf{x}_{(i)}^{(u)}$  и нека је  $\varphi_{(i)}^{(u)} = \varphi^{(u)}(\mathbf{x}_{(i)}^{(u)})$ . Тада за сваке две јединке  $\mathbf{x}_{(i)}^{(u)}$  и  $\mathbf{x}_{(j)}^{(u)}$  ( $\{i, j\} \subseteq \{1, 2, \dots, N + \sigma\}$ ) важи следећа импликација:

$$i < j \implies \left( \varphi_{(i)}^{(u)} > \varphi_{(j)}^{(u)} + \eta \right) \vee \left( \left| \varphi_{(i)}^{(u)} - \varphi_{(j)}^{(u)} \right| \leq \eta \wedge t_{(i)}^{(u)} < t_{(j)}^{(u)} \right). \quad (3.28)$$

Позицију на којој се јединка  $\mathbf{x}$  нађе у сортираном низу  $[\mathbf{x}_{(1)}^{(u)}, \mathbf{x}_{(2)}^{(u)}, \dots, \mathbf{x}_{(N+\sigma)}^{(u)}]$  називамо ранг јединке  $\mathbf{x}$  у  $u$ -том циклусу и означавамо као  $r_u(\mathbf{x})$ .

У другој фази, сортирамо јединке  $P^{(u)}$  у неоппадајућем поретку служећи се њиховим укупним просечним ранговима до  $u$ . циклуса.

Нека је  $v(\mathbf{x}_i) \leq u$  циклус у којем је јединка  $\mathbf{x}_i \in P^{(u)}$  генерисана. Тада је њен укупан просечан ранг до  $u$ -тог циклуса, у ознаци  $\bar{r}_u(\mathbf{x}_i)$ , дефинисан као просек свих њених рангова  $r_{v(\mathbf{x}_i)}(\mathbf{x}_i), r_{v(\mathbf{x}_i)+1}(\mathbf{x}_i), \dots, r_u(\mathbf{x}_i)$ . Он се може израчунати следећом рекурзивном формулом:

$$\bar{r}_u(\mathbf{x}_i) = \frac{(u - v(\mathbf{x}_i)) \cdot \bar{r}_{u-1}(\mathbf{x}_i) + r_u(\mathbf{x}_i)}{u - v(\mathbf{x}_i) + 1}. \quad (3.29)$$

Када би се користили само укупни просечни рангови, могло би се десити да јединка  $\mathbf{x}$ , која је евалуирана само једном, истисне јединку  $\mathbf{z}$  чији је укупан просечан ранг незнатно гори од ранга јединке  $\mathbf{x}$ , али која је евалуирана више пута па је њен просечан ранг знатно поузданији. Да би се то избегло, приликом сортирања јединки у другој фази замене генерација може се применити следећа хеуристика. Уколико се просечни рангови две јединке заокругљују на исти умножак параметра  $\xi$ , бољом се сматра она јединка која је евалуирана више пута.

Као резултат друге фазе добија се сортирана популација  $\tilde{P}^{(u)} = [\tilde{\mathbf{x}}_{(1)}^{(u)}, \tilde{\mathbf{x}}_{(2)}^{(u)}, \dots, \tilde{\mathbf{x}}_{(N)}^{(u)}]$ . За сваке две јединке у њој,  $\tilde{\mathbf{x}}_{(i)}^{(u)}$  и  $\tilde{\mathbf{x}}_{(j)}^{(u)}$  ( $\{i, j\} \subseteq \{1, 2, \dots, N + \sigma\}$ ) важи релација (3.30):

$$i < j \implies \left( \left\lfloor \frac{\bar{r}_u(\tilde{\mathbf{x}}_{(i)}^{(u)})}{\xi} \right\rfloor < \left\lfloor \frac{\bar{r}_u(\tilde{\mathbf{x}}_{(j)}^{(u)})}{\xi} \right\rfloor \right) \vee \left( \left\lfloor \frac{\bar{r}_u(\tilde{\mathbf{x}}_{(i)}^{(u)})}{\xi} \right\rfloor = \left\lfloor \frac{\bar{r}_u(\tilde{\mathbf{x}}_{(j)}^{(u)})}{\xi} \right\rfloor \wedge v(\tilde{\mathbf{x}}_{(i)}^{(u)}) < v(\tilde{\mathbf{x}}_{(j)}^{(u)}) \right). \quad (3.30)$$

Када се из популације  $\tilde{P}^{(u)}$  одстрани јединке  $\tilde{\mathbf{x}}_{(N-\sigma+1)}^{(u)}, \tilde{\mathbf{x}}_{(N-\sigma+2)}^{(u)}, \dots, \tilde{\mathbf{x}}_{(N)}^{(u)}$  и замене јединкама из популације потомака  $MD^{(u)}$ , добија се популација  $P^{(u+1)}$ , која представља улаз за наредни циклус GA. Поступак замене генерација представљен је алгоритмом 3.5.

---

**Алгоритам 3.5:** Замена генерација

---

**Улаз:**  $P^{(u)}$  – популација јединки у  $u$ . циклусу;  $MD^{(u)}$  – популација јединки потомака у  $u$ . циклусу;  $\varphi^{(u)}$  – функција прилагођености;  $\xi, \eta$  – параметри за сортирање.

**Израз:**  $P^{(u+1)}$  – популација у којој су јединке са најмањим степеном прилагођености у популацији  $P^{(u)}$  замењене потомцима  $MD^{(u)}$ .

- 1  $[\mathbf{x}_{(1)}^{(u)}, \dots, \mathbf{x}_{(N+\sigma)}^{(u)}] \leftarrow$  сортирати унију  $P^{(u)} \cup MD^{(u)}$  тако да важи релација (3.28).
  - 2 **за**  $i \leftarrow 1, 2, \dots, N + \sigma$  :
  - 3      $r_u(\mathbf{x}_{(i)}^{(u)}) \leftarrow i$
  - 4     **ако**  $\mathbf{x}_{(i)}^{(u)} \in MD^{(u)}$  :
  - 5          $\bar{r}_u(\mathbf{x}_{(i)}^{(u)}) \leftarrow i$
  - 6     **иначе :**
  - 7          $\bar{r}_u(\mathbf{x}_{(i)}^{(u)}) \leftarrow \frac{(u-v(\mathbf{x}_{(i)}^{(u)})) \cdot \bar{r}_{u-1}(\mathbf{x}_{(i)}^{(u)}) + i}{u-v(\mathbf{x}_{(i)}^{(u)})+1}$
  - 8  $P' \leftarrow [\mathbf{x}_{(i)}^{(u)} \mid \mathbf{x}_{(i)}^{(u)} \in P^{(u)}, i = 1, 2, \dots, N + \sigma]$
  - 9  $[\tilde{\mathbf{x}}_{(i)}^{(u)}]_{i=1}^N \leftarrow$  сортирати низ  $P'$  тако да важи релација (3.30).
  - 10  $P^{(u+1)} = \{\tilde{\mathbf{x}}_{(1)}^{(u)}, \dots, \tilde{\mathbf{x}}_{(N-\sigma)}^{(u)}\} \cup MD^{(u)}$
  - 11 **Резултат:**  $P^{(u+1)}$
- 

## 3.7 Критеријуми заустављања

У предложеној имплементацији ГА коришћена су следећа два услова за заустављање, од којих је довољно да један буде испуњен како би алгоритам прекинуо извршавање:

- Достигнут је максималан број циклуса,  $MNI$ .
- Прилагођеност најбоље и најгоре јединке у популацији разликује се за мање од  $\rho$ .

Испуњење другог услова за мале вредности параметра  $\rho$ , који зовемо параметар конвергенције, показује да је алгоритам конвергирао по прилагођености, односно, да је популација јединки врло слична и да су се хромозоми окупили око локалног оптимума функције прилагођености. У том случају је мало вероватно да се генетским операторима може као потомак добити знатно боља јединка, па је оправдано прекинути алгоритам на крају циклуса у којем је конвергенција уочена.

## 3.8 Завршна евалуација и избор мреже

Свака јединка у популацији генетског алгоритма представља протокол за извођење експеримента чији је циљ развој прецизне и стабилне мреже за испитивање нормалности. Генетски алгоритам кроз низ итерација тежи да достигне оптимални протокол, аутоматизујући извођење експеримената по протоколима кодираним у хромозомима јединки. Јединке у завршној популацији представљају најбоље протоколе добијене генетским алгоритмом, тј. најбоља подешавања метапараметара до којих је алгоритам дошао аутоматизујући експерименте. Међутим, да бисмо били сигурни да су пронађена квалитетна подешавања, резултујуће мреже треба испитати још једном.

У ту сврху, по завршетку рада ГА, генеришу се додатни евалуациони скупови  $\mathcal{D}_{eval,0}$ ,  $\mathcal{D}_{eval,1}$  и  $\mathcal{D}_{eval,2}$  на сличан начин као скупове  $\mathcal{D}_{eval,0}^{(u)}$ ,  $\mathcal{D}_{eval,1}^{(u)}$  и  $\mathcal{D}_{eval,2}^{(u)}$ . Разлика је у томе што се за скуп  $\mathcal{D}_{eval,0}$  генерише по 10 000 узорака за сваки обим  $n = 10, 15, \dots, 95, 100$  и сваки

тип расподела, а за скупове  $\mathcal{D}_{eval,1}$  и  $\mathcal{D}_{eval,2}$  користе се узорци обима 12, 17, 22,  $\dots$ , 97. На тај начин детаљније се проверавају перформансе добијених мрежа, испитује њихова способност генерализације тестира да ли им тачност опада са порастом обима узорака. Ако таква мрежа није добијена, извршавање генетског алгоритма треба покренути поново, али са другачијим подешавањем његових параметара, и опет извести завршну евалуацију. Овај поступак прекида се онда када добијемо стабилне мреже високе и неоппадајуће тачности. Ако више мрежа има приближно исту тачност, предност треба дати оној чије је тренирање трајало најкраће. На основу замене генерација у одељку 3.6, таква мрежа би требало да се нађе на првом месту у завршној популацији генетског алгоритма.

### 3.9 Структура предложеног генетског алгоритма

Сви кораци GA адаптираног за развој мрежа које тестирају нормалност расподела детаљно су представљени у алгоритму 3.6.

---

**Алгоритам 3.6:** Адаптација GA за развој мрежа које испитују нормалност расподела

---

**Улаз:**  $N, \sigma, \mu_P, \mu_G, \Delta, MNI, \rho, \omega, \delta, \eta, \xi$

**Излаз:**  $N$  јединки које представљају оптимизоване метапараметре

- 1  $P^{(0)} \leftarrow$  генерисати почетну популацију  $[\mathbf{x}_i^{(0)}]_{i=1}^N$  насумичним подешавањем гена.
  - 2 Генерисати скупове  $\mathcal{D}_{eval,0}^{(0)}, \mathcal{D}_{eval,1}^{(0)}, \mathcal{D}_{eval,2}^{(0)}$  (видети одељак 3.5).
  - 3 **за**  $i \leftarrow 1, 2, \dots, N$  :
  - 4  $\left[ \varphi^{(0)}(\mathbf{x}_i) \leftarrow$  израчунати прилагођеност формулом (3.27) помоћу  $\mathcal{D}_{eval,k}^{(u)}$  ( $k = 0, 1, 2$ ).
  - 5  $\varphi_{\max}^{(0)} \leftarrow \max_{\mathbf{x} \in P^{(0)}} \varphi^{(0)}(\mathbf{x})$
  - 6  $\varphi_{\min}^{(0)} \leftarrow \min_{\mathbf{x} \in P^{(0)}} \varphi^{(0)}(\mathbf{x})$
  - 7  $u \leftarrow 1$
  - 8 **док**  $u \leq MNI \wedge \varphi_{\max}^{(u-1)} - \varphi_{\min}^{(u-1)} > \rho$  :
  - 9  $R^{(u)} \leftarrow$  извршити селекцију (код 3.2) користећи прилагођености  $[\varphi^{(u-1)}(\mathbf{x}) \mid \mathbf{x} \in P^{(u)}]$ .
  - 10  $D^{(u)} \leftarrow \emptyset$
  - 11 **за сваки**  $(\mathbf{x}_i, \mathbf{x}_j) \in R^{(u)}$  :
  - 12  $\left[ \mathbf{z} \leftarrow$  укрстити  $\mathbf{x}_i$  и  $\mathbf{x}_j$  применом оператора датог у псеудокоду 3.3.
  - 13  $\left[ D^{(u)} \leftarrow D^{(u)} \cup \{\mathbf{z}\}$
  - 14  $MD^{(u)} \leftarrow \emptyset$
  - 15 **за сваки**  $\mathbf{z} \in D^{(u)}$  :
  - 16  $\left[ \mathbf{ако}$   $S_{[0,1]}() \leq \mu_P$  :
  - 17  $\left[ \mathbf{z}' \leftarrow$  на јединку  $\mathbf{z}$  применити оператор мутације дефинисан у псеудокоду 3.4.
  - 18  $\left[ MD^{(u)} \leftarrow MD^{(u)} \cup \{\mathbf{z}'\}$
  - 19 Генерисати скупове  $\mathcal{D}_{eval,0}^{(u)}, \mathcal{D}_{eval,1}^{(u)}, \mathcal{D}_{eval,2}^{(u)}$  (видети одељак 3.5).
  - 20 **за сваки**  $\mathbf{x} \in P^{(u)} \cup MD^{(u)}$  :
  - 21  $\left[ \varphi^{(u)}(\mathbf{x}_i) \leftarrow$  израчунати прилагођеност формулом (3.27) помоћу  $\mathcal{D}_{eval,k}^{(u)}$  ( $k = 0, 1, 2$ ).
  - 22  $P^{(u+1)} \leftarrow$  саставити нову популацију од јединки у  $P^{(u)}$  и  $MD^{(u)}$  (псеудокод 3.5).
  - 23  $\varphi_{\max}^{(u)} \leftarrow \max_{\mathbf{x} \in P^{(u)}} \varphi^{(u)}(\mathbf{x})$
  - 24  $\varphi_{\min}^{(u)} \leftarrow \min_{\mathbf{x} \in P^{(u)}} \varphi^{(u)}(\mathbf{x})$
  - 25  $u \leftarrow u + 1$
  - 26 **Резултат:** Генетски код јединки у завршној популацији  $P^{(u-1)}$
- 

Након иницијализације и евалуације почетне популације, GA кроз низ циклуса врши

потрагу за оптималним експерименталним протоколом. У сваком циклусу (итерацији), на тренутну популацију јединки примењују се генетски оператори дизајнирани за хромозоме варијабилних дужина и хијерархијске структуре, која одсликава однос између метапараметара чије се вредности оптимизују. За евалуацију јединки користи се стохастичка функција прилагођености, која поред класификационе тачности резултујућих мрежа, у обзир узима монотоност тачности на узорцима растућих обима, као и стабилност на узорцима различитих обима и из различитих расподела у односу на скуп коришћен за тренирање



## Глава 4

# Извођење статистичких тестова из бинарних класификатора

Стандардним тренирањем на скуповима узорака добијају се мреже којима не можемо унапред задати вероватноћу да нормалне узорке класификују као не-нормалне. Са друге стране, таква контрола је изводљива код статистичких тестова. Методе машинског учења за контролу грешке могу се применити на мреже развијене за испитивање нормалности како би им се ограничила вероватноћа грешке на класи нормалних узорака.

Више таквих метода формулисано је у оквиру машинског учења. Мотивација за њихов развој је контрола грешке бинарних класификатора на класи са озбиљнијим последицама погрешне класификације. На пример, у берзанском пословању, класификација добре прилике за улагање као неисплативе значи пропуштену шансу за увећање капитала. Са друге стране, класификација неисплативе прилике као исплативе доводи до директног губитка финансијских средстава. Да би се у обзир узела таква асиметрија ризика, неопходно је конструисати класификатор који поред максимизације укупне тачности, вероватноћу озбиљније грешке одржава на нивоу задатом унапред. У овом поглављу биће представљене методе које су у досадашњој литератури предложене за контролу вероватноће грешке, као и две нове методе засноване на класичним статистичким тестовима расподела. Поред тога, биће представљени резултати емпиријског поређења ових метода.

### 4.1 Нотација и конвенције

Нека  $\mathcal{X}$  означава простор објеката за класификацију, а  $X$  случајну променљиву која представља објекат из простора  $\mathcal{X}$ . Појединачне објекте простора означавамо малим латиничним словима  $(x, z, \dots)$ , индексираним где је неопходно. Праву класу случајног објекта записујемо као  $Y = y(X)$ . Аналогно томе,  $y(x)$  представља праву класу конкретног објекта  $x$ . Као што је уобичајено у литератури, нула је ознака негативне, а јединица позитивне класе.

Бинарни класификатор дефинишемо као функцију  $\hat{y} : \mathcal{X} \rightarrow \{0, 1\}$ ,  $\hat{y}(x) = \mathbf{1} \{t(x) \geq \tau\}$ , где је  $\tau$  класификациони праг, а  $t : \mathcal{X} \rightarrow \mathbb{R}$  је класификаторов нумерички предизлаз чије више вредности одговарају већој процени вероватноће да је објекат позитиван. При томе, за случајан објекат  $X$  важи и да су  $T = t(X)$ ,  $Y = y(X)$  и  $\hat{Y} = \hat{y}(X)$  такође случајне променљиве.

Вероватноћа (или стопа) лажно позитивних (FPR) и лажно негативних предикција (FNR) класификатора  $\hat{y}$  чији је класификациони праг  $\tau$  дефинисане су на следећи начин:

$$\text{FNR}(\hat{y}, \tau) = \mathbb{P}(\hat{Y} = 0 \mid Y = 1), \quad \text{FPR}(\hat{y}, \tau) = \mathbb{P}(\hat{Y} = 1 \mid Y = 0). \quad (4.1)$$

Нотацију ћемо упростити изостављањем  $\hat{y}$  и  $\tau$  тамо где су класификатор и праг јасни из контекста.

Нека је  $\alpha \in (0, 1)$  максимална прихватљива вероватноћа озбиљније грешке, која се још зове и њен номинални праг. Тада,  $\text{FNR}(\alpha)$  и  $\text{FPR}(\alpha)$  представљају теоријску вероватноћу лажно негативних и лажно позитивних предикција за номинални праг  $\alpha$  који контролише озбиљнију грешку, док су  $\widehat{\text{FNR}}(\alpha)$  и  $\widehat{\text{FPR}}(\alpha)$  одговарајуће емпиријске оцене тих вероватноћа.

## 4.2 Конформална предикција

Нека је  $\alpha$  праг значајности и нека је  $X$  насумичан објекат који треба класификовати. Конформални класификатор као резултат предикције враћа  $\Gamma^\alpha(X)$ , скуп ознака класа за које је процењено да им објекат припада. При томе је вероватноћа најмање  $1 - \alpha$  да се у скупу  $\Gamma^\alpha(X)$  заиста садржи ознака  $Y$  класе из које потиче објекат  $X$  [70]:

$$\mathbb{P} \{Y \in \Gamma^\alpha(X)\} \geq 1 - \alpha. \quad (4.2)$$

Пошто конформални класификатори као предикцију дају скуп класних ознака, релација (4.2) није еквивалентна ограничењима  $\text{FNR} \leq \alpha$  и  $\text{FPR} \leq \alpha$ .

Већина конформалних класификатора развијена је за тзв. онлајн начин употребе. То значи да се након израчунавања скупа  $\Gamma^\alpha(x)$  (и његовог коришћења у даљем одлучивању) може сазнати права класа  $y(x)$ , што омогућава да се скуп за тренирање прошири паром  $(x, y(x))$ . Трансдуктивни онлајн конформални приступ заснива се на принципи-ма трансдуктивног закључивања [70, 71]. Он класификује објекте без индукције општих класификационих правила из скупа за тренирање. То постиже тако што за сваки објекат тренира бар два класификатора, један по класи, који се користе само за класификацију тог објекта. Услед тога, овај приступ је рачунски захтеван. Насупрот њему, индуктивни конформални приступ [72, 73] не захтева да се за сваки нов објекат тренирају нови класификатори. Уместо тога, постојећи класификатори поново пролазе кроз процес тренирања након што класификују одређен број нових објеката (чије се праве ознаке додају постојећем скупу за тренирање). Да би важила релација (4.2), почетни скуп за тренирање мора бити случајан. То значи да у индуктивном приступу горња граница за вероватноћу озбиљније грешке може да се постави на праг  $\alpha$ , али тако да вероватноћа грешке израчуната за све индуктивне класификаторе укупно (један по сваком могућем скупу за тренирање) буде највише  $\alpha$ .

Индуктивни класно-условни мондријански конформални приступ (енгл. *inductive class-conditional Mondrian conformal classification*, ИССМС) [74, 75] је рачунски изводљив као и методе које припадају индуктивном приступу [72, 73], а применљив је и у офлајн режиму (у којем је немогуће накнадно сазнати праву класу већ класификованих објеката).

У првом кораку, офлајн ИССМС приступ (off-ИССМС) раздваја почетни скуп за тренирање на два скупа: активни скуп за тренирање  $\mathcal{D}_{proper}$  и калибрациони скуп  $\mathcal{D}_{cal} = \{(x_i, y_i)\}_{i=1}^m$ . Количник  $\frac{|\mathcal{D}_{proper}|}{|\mathcal{D}_{cal}|}$  је параметар оваквих класификатора [76], па му се вредност мора дефинисати пре њихове конструкције.

Након тога, метода off-ИССМС обучава класификатор  $f$  са нумеричким предизлазом  $t$  користећи активни скуп  $\mathcal{D}_{proper}$ . Нека је  $a(x, \mathcal{D}, c)$  мера неконформалности, која квантификује различитост објекта  $x$  од објеката класе  $c \in \{0, 1\}$  у произвољном скупу  $\mathcal{D}$ . Метода off-ИССМС рачуна степен неконформалности  $a_i^c$  за сваки калибрациони објекат  $x_i \in \mathcal{D}_{cal}$  и обе класе  $c = 0, 1$ . Када се оваквом класификатору зада нов објекат  $x \equiv x_{m+1}$ ,

он прво рачуна степене неконформалности  $a_{m+1}^0$  и  $a_{m+1}^1$  тога објекта служећи се калибрационим скупом. Поредeћи их са већ одређеним степенима, класификатор рачуна класне  $p$ -вредности за објекат  $x$ :

$$p_c(x) = \frac{|\{a_i^c: y_i = c \wedge a_i^c \geq \alpha_{m+1}^c \wedge i \in \{1, 2, \dots, m+1\}\}|}{m+1}. \quad (4.3)$$

Израз  $\Gamma^\alpha(x)$  дефинише се као скуп оних класа чије су  $p$ -вредности изнад прага  $\alpha$ :

$$\Gamma^\alpha(x) = \{c \in \{0, 1\}: p_c(x) > \alpha\}. \quad (4.4)$$

За разлику од онлајн конформалних метода, приступ off-ICCMC нема математичке гаранције да ће вероватноће FNR и FPR бити одозго ограничене прагом  $\alpha$ .

Класификатори који прате приступ назван полуофлајн ICCMC (енгл. *semifoline ICCMC*, so-ICCMC) дефинишу вероватноће FNR и FPR на следећи начин:

$$\text{FNR} = \mathbb{P}(Y \notin \Gamma^\alpha(X) \mid Y = 1) \quad \text{и} \quad \text{FPR} = \mathbb{P}(Y \notin \Gamma^\alpha(X) \mid Y = 0). \quad (4.5)$$

Полуофлајн класификатори испуњавају захтеве  $\text{FNR} \leq \alpha$  и  $\text{FPR} \leq \alpha$ . Услов за то је да се у скуп степена неконформалности које користимо за израчунавање  $p$ -вредности додају степени свих објеката које класификујемо после обучавања. Ипак, треба приметити да предиктивни скуп  $\Gamma^\alpha(x)$  може бити празан или садржати ознаке обе класе. Иако су и такви излази информативни зато што нас упућују да се ти објекти подробније испитају, учестале предикције које су празне или враћају ознаке обе класе ограничавају применљивост приступа so-ICCMC. Празна предикција нам говори да испитивани објекат не припада ниједној класи, док нам предикција  $\{0, 1\}$  говори да припада обема класама. Овакви излази класификатора нису корисни када је неопходно донети одговарајућу одлуку или предузети извесну акцију на основу предиктоване класе датог објекта.

Ако желимо да вероватноћа грешке полуофлајн класификатора на класи  $c \in \{0, 1\}$  буде мања од прага  $\alpha$ , а да он истовремено на излазу враћа или 0 или 1 као предикцију, тада предиктивни скуп полуофлајн класификатора треба модификовати на следећи начин:

$$\hat{y}(x) = \begin{cases} c, & p_c(x) > \alpha \\ 1 - c, & \text{у супротном.} \end{cases} \quad (4.6)$$

Такав конформални класификатор може се користити са гаранцијом да ће вероватноћа погрешне класификације објеката задате класе бити мања од унапред постављеног прага. За више детаља о конформалној предикцији погледати радове [75, 77, 78].

### 4.3 Нејман-Пирсонова класификација

Нејман-Пирсонова класификација (енгл. *Neyman-Pearson classification*, NPC) [79] тежи да конструише класификатор минималне вероватноће грешке на класи за коју је грешка мање озбиљна, а да при томе вероватноћа грешка на класи с озбиљнијом грешком буде ограничена вредношћу прага  $\alpha$ . Прву групу метода у оквиру Нејман-Пирсоновог приступа чине емпиријске методе [80–84], док другу групу чине директне методе [79, 85, 86]. Приликом представљања обе групе Нејман-Пирсонових метода, без губитка општости ћемо претпоставити да лажно позитивна предикција представља озбиљнију грешку од лажно негативне.

Емпиријске методе конструишу класификаторе тако што током обучавања решавају следећи оптимизациони проблем:

$$\min_{\hat{y}} \left\{ \widehat{\text{FPR}}(\hat{y}) : \widehat{\text{FNR}}(\hat{y}) \leq \alpha + \varepsilon/2 \right\} \quad (4.7)$$

где су  $\widehat{\text{FNR}}$  и  $\widehat{\text{FPR}}$  емпиријске апроксимације вероватноћа FNR и FPR, израчунате помоћу података за тренирање, а параметар  $\varepsilon$  је позитиван.

Мана ових метода је што се оптимизациони проблем (4.7) мора решити изнова за сваку могућу вредност прага  $\alpha$  коју желимо да користимо приликом класификације. То ове методе чини рачунски захтевним и теже применљивим у пракси.

Директна Нејман-Пирсонова класификација (енгл. *the plug-in (direct) NPC approach*, DNP) [85] заснива се на примени Нејман-Пирсонове леме [87]. Нека су  $q_0$  и  $q_1$  густине негативне и позитивне класе у простору  $\mathcal{X}$ . Према Нејман-Пирсоновој лемі, класификатор са минималном вероватноћом FPR који испуњава услов  $\text{FNR} \leq \alpha$  дат је следећом релацијом:

$$\hat{y}_{NP}^*(x) = \mathbf{1} \left\{ \frac{q_1(x)}{q_0(x)} \geq C_\alpha \right\}, \quad \text{где је} \quad \mathbb{P} \left( \frac{q_1(X)}{q_0(X)} \geq C_\alpha \mid Y = 0 \right) = \alpha. \quad (4.8)$$

Како су густине  $q_0$  и  $q_1$  непознате, директан Нејман-Пирсонов класификатор користи њихове емпиријске апроксимације  $\hat{q}_0$  и  $\hat{q}_1$ , као и емпиријску оцену коефицијента  $C_\alpha$ .

Коришћење емпиријских апроксимација густина представља проблем у високодимензионалним просторима. То је избегнуто у раду [86] где се димензионалност простора смањује селекцијом атрибута пре естимације густина, али уз претпоставку да су атрибути међусобно независни. У раду [88] формулисани су директни параметарски Нејман-Пирсонови класификатори који претпостављају да густине  $q_0$  и  $q_1$  прате вишедимензионалне нормалне расподеле са заједничком матрицом коваријанси. Међутим, та претпоставка је сувише рестриктивна за примену у пракси.

## 4.4 Алгоритам UA

Као у излагању Нејман-Пирсонових метода, и овде ћемо без губитка општости претпоставити да озбиљнију грешку чини лажно негативна предикција. Алгоритам UA (енгл. *Umbrella Algorithm*) [89] је поступак конструкције класификатора који са високом вероватноћом контролишу вероватноћу FNR.

На почетку рада, алгоритам дели податке  $m \geq 1$  пута на скуп за тренирање и скуп за калибрацију. Нека су  $\mathcal{D}_{train}^{(i)}$  и  $\mathcal{D}_{cal}^{(i)}$  скупови добијени у  $i$ -тој подели ( $i = 1, 2, \dots, m$ ). Потом, за сваки пар  $(\mathcal{D}_{train}^{(i)}, \mathcal{D}_{cal}^{(i)})$  ( $i = 1, 2, \dots, m$ ), алгоритам UA обучава класификатор  $f_i$  на скупу за тренирање  $\mathcal{D}_{train}^{(i)}$ , а онда калибрише његов класификациони праг  $\tau_i$  помоћу скупа  $\mathcal{D}_{cal}^{(i)}$ . Циљ калибрације је да се пронађе такав класификациони праг за који је вероватноћа највише  $\delta$  да стопа лажно негативних предикција буде већа од номиналног прага  $\alpha$  (при чему је  $\delta$  параметар који се задаје унапред). Калибрација се заснива на теорему 4.1 [89].

**Теорема 4.1.** *Нека је  $t$  нумерички предизлаз бинарног класификатора  $f$  и нека је  $\mathcal{D}^{(1)} = \{X_1, X_2, \dots, X_n\}$  насумичан скуп  $n$  објеката класе 1, независан од података на којима је класификатор  $f$  обучен. Нека су  $T_{(1)} \leq T_{(2)} \leq \dots \leq T_{(n)}$  сортирани предизлази тог класификатора за објекте  $X_1, X_2, \dots, X_n$ . Сваки нов објекат  $X$ , чији је предизлаз  $T = t(X)$ , можемо класификовати новим класификатором  $f_k = \mathbf{1} \{T \geq T_{(k)}\}$  за дато  $k \in \{1, 2, \dots, n\}$ . Пошто је  $T_{(k)}$  случајна променљива, и вероватноћа FNR класификатора  $f_k$  такође је случајна променљива. Тада важи:*

$$v(k) = P_{X, \mathcal{D}^{(1)}} \{ \text{FNR}(f_k) > \alpha \} \leq \sum_{j=k}^n \binom{n}{j} (1-\alpha)^j \alpha^{n-j}. \quad (4.9)$$

Вероватноћа  $v(k)$  назива се вероватноћа прекорачења [89]. Нека је  $\delta$  најмања прихватљива вероватноћа да се обучавањем добије класификатор чија је стопа FNR већа од номиналног прага  $\alpha$ . Нека је  $T_{(k^*)}$  предизлаз који минимизује вероватноћу  $v(k)$ :

$$k^* = \min\{k \in \{1, 2, \dots, n\} : v(k) \leq \delta\}. \quad (4.10)$$

Калибрисани класификатор дефинисан је тада као  $\mathbf{1}\{T > T_{(k^*)}\}$ . Алгоритам UA за сваки класификатор  $f_i$  ( $i = 1, 2, \dots, n$ ) калибрише класификациони праг на описани начин. Класификатори  $f_i$  ( $i = 1, 2, \dots, m$ ) могу бити сви истих или различитих модела, при чему је могуће конструисати и само један калибрисан класификатор ( $m = 1$ ). Уколико их има више, њихови излази се агрегирају већинском методом.

Иако се алгоритмом UA добијају класификатори код којих важи  $\text{FNR} \leq \alpha$  са вероватноћом од најмање  $1 - \delta$ , нема доказа да овај поступак минимизује стопу FPR [89]. Такође, вероватноћа прекорачења (4.9) зависи и од случајних променљивих  $X_1, X_2, \dots, X_n$  које моделују калибрационе објекте у скупу  $\mathcal{D}^{(1)}$ . Услед тога, за конкретан класификатор добијен минимизацијом (4.10) не важи да је његова стопа грешке FNR већа од  $\alpha$  с вероватноћом од највише  $\delta$ . Исправно тумачење је да од свих класификатора који се могу добити на овај начин, удео оних код којих је вероватноћа  $\text{FNR} > \alpha$  износи највише  $\delta$ .

За задате вредности  $\alpha$  и  $\delta$  и изабран индекс  $k^*$  важи неједнакост (4.11):

$$\sum_{j=k^*}^n \binom{n}{j} (1 - \alpha)^j \alpha^{n-j} \leq \delta. \quad (4.11)$$

За  $k^* = n$ , та неједнакост се своди на  $(1 - \alpha)^n \leq \delta$ . Како је вероватноћа прекорачења  $v(k)$  опадајућа функција по индексу  $k$ , следи да је за задато  $\alpha$  и  $\delta$  неопходно обезбедити најмање  $n \geq \log_{1-\alpha} \delta$  објеката у калибрационом скупу како бисмо могли применити алгоритам UA.

## 4.5 Пермутационе $p$ -вредности

Пермутационе  $p$ -вредности за вишекласну класификацију формулисане су у раду [90]. Овде ћемо их представити за случај бинарне класификације.

Нека је  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^m$  (где је  $y_i = y(x_i)$ ) скуп за тренирање,  $t$  предизлаз класификатора добијеног обучавањем на скупу  $\mathcal{D}$  и  $c \in \{0, 1\}$  циљна класа на којој ограничавамо вероватноћу грешке. Нека је  $x \in \mathcal{X}$  објекат који треба класификовати. Пермутациона  $p$ -вредност хипотезе  $H_c : y(x) = c$ , у ознаци  $p_c(x)$ , рачуна се на следећи начин. Нека је  $\mathcal{D}_i(x)$  модификација скупа  $\mathcal{D}$  која се добија тако што се објекат  $x_i$  замени објектом  $x$ , при чему се као ознака класе задржава  $y_i$ . За сваки индекс  $i$  такав да је  $y_i = c$ , тренира се класификатор на скупу  $\mathcal{D}_i(x)$ . Нека је  $t_i$  предизлаз класификатора обучаваног на скупу  $\mathcal{D}_i(x)$ . Пермутациона  $p$ -вредност  $p_c(x)$  тада је дефинисана на следећи начин:

$$p_c(x) = \frac{1}{m_c + 1} (1 + |\{i : i \in \{1, 2, \dots, m\} \wedge y_i = c \wedge t_i(x_i) \geq t(x)\}|) \quad (4.12)$$

где је  $m_c$  број објеката класе  $c$  у скупу  $\mathcal{D}$ , односно број класификатора који се тренирају на модификованим скуповима. Предикција за објекат  $x$  је класа  $c$  уколико је  $p_c(x) \geq \alpha$ , док га у супротном сврставамо у класу  $1 - c$ . За  $m_c \geq \lceil 1/\alpha \rceil + 1$ , пермутационе  $p_c$ -вредности могу да ограниче вероватноћу грешке на класи  $c$  на задату вредност  $\alpha$ .

Мана овог приступа је што је неопходно изнова тренирати велик број класификатора за класификацију сваког новог објекта, што није изводљиво у пракси. У раду [90]

предложена је формулација пермутационих  $p$ -вредности чије је израчунавање лакше и брже. За сваки нов објекат  $x$  тренира се нов класификатор на скупу  $\mathcal{D} \cup \{(x, c)\}$ . Нека је  $t'$  његов предизлаз. Тада, пермутациону  $p$ -вредност  $p_c(x)$  рачунамо на следећи начин:

$$p_c(x) = \frac{1}{m_c + 1} (1 + |\{i : i \in \{1, 2, \dots, m\} \wedge y_i = c \wedge t'(x_i) \geq t(x)\}|). \quad (4.13)$$

Иако бржа, формулација (4.13) такође је рачунски захтевна. Обе формулације пермутационих  $p$ -вредности могу да контролишу вероватноћу грешке.

Ова приступ еквивалентан је трансдуктивној конформалној предикцији која степен неконформалности објекта  $x_i$  рачуна као  $|t(x_i) - y_i|$ .

## 4.6 Посредно тестирање

Посредно тестирање (енгл. *test-based classification*, ТВС) класификује објекте изводећи статистичке тестове који испитују да ли два узорка потичу из исте расподеле [91–94]. У овом одељку биће представљена примена приступа ТВС на проблем бинарне класификације.

Приступ ТВС првобитно је формулисан у раду [91]. Нека су  $\mathcal{S}_0$  и  $\mathcal{S}_1$  скупови негативних и позитивних објеката. Нека је  $x$  нов објекат који треба класификовати. Ако је он позитиван, претпоставка овог приступа је да ће скуп  $\mathcal{S}_0 \cup \{x\}$  постати сличнији скупу  $\mathcal{S}_1$  него што је то био скуп  $\mathcal{S}_0$ . Аналогна претпоставка важи и у случају да је објекат  $x$  негативан.

Сличност скупова може се квантификовати  $p$ -вредношћу статистика тестова који испитују истоветност две дистрибуције. Нека су  $\Phi_0$  и  $\Phi_1$  расподеле из којих потичу скупови  $\mathcal{S}_0$  и  $\mathcal{S}_1$  и нека су  $\mu_0$  и  $\mu_1$  очекивања расподела  $\Phi_0$  и  $\Phi_1$ . Првобитни приступ ТВС изводи следећа два статистичка теста [91]:

- (а) Тест који скупове  $\mathcal{S}_0 \cup \{x\}$  и  $\mathcal{S}_1$  користи као узорке за испитивање нулте хипотезе  $\mu_0 = \mu_1$ . Нека је  $p_0(x)$  тако добијена  $p$ -вредност.
- (б) Тест који користи скупове  $\mathcal{S}_0$  и  $\mathcal{S}_1 \cup \{x\}$  као узорке за испитивање нулте хипотезе  $\mu_0 = \mu_1$ . Нека је  $p_1(x)$  тако добијена  $p$ -вредност.

Уколико објекат  $x$  припада класи  $c$ , вредност  $p_c(x)$  требало би да буде ниска, за разлику од вредности  $p_{1-c}(x)$  ( $c \in \{0, 1\}$ ). Нека је  $w_c$  априорна вероватноћа догађаја  $y(X) = c$  (пре задавања конкретног објекта  $x$  који треба класификовати). За априорне вероватноће  $w_1$  и  $w_2$  можемо узети заступљеност класа у скупу  $\mathcal{S}_0 \cup \mathcal{S}_1$ , а можемо обе вероватноће поставити на вредност  $1/2$ . У раду [91], објекат  $x$  класификује се као негативан ако важи  $p_0(x)(1 - w_0) < p_1(x)(1 - w_1)$ . У супротном, класификује се као позитиван.

У раду [92] уочено је да предикције првобитне методе ТВС [91] нису поуздане када су обе  $p$ -вредности ниске ( $p_0(x), p_1(x) < \varepsilon$  за малу вредност параметра  $\varepsilon$ ). У том случају се објекту  $x$  додељује класа оног скупа  $\mathcal{S}_c$  чији му је центар ближи [92].

У раду [93] евалуирано је више статистичких тестова, међу којима се истакао Барингхаус–Францов тест (BF) [95]. Уместо  $p$ -вредности, метода предложена у раду [93] пореди реализоване вредности статистика овог теста. Исти тест је коришћен и у раду [94]. Расподела његове статистике зависи од расподела променљивих  $X | Y = 0$  и  $X | Y = 1$ , што значи да се  $p$ -вредности и критичне вредности које одговарају номиналном прагу  $\alpha$  не могу одредити аналитички.

Метода предложена у раду [94] испитује дистанце до најближих суседа. Заснива се на претпоставци да ако објекат  $x$  припада класи  $c \in \{0, 1\}$ , растојања  $\mathcal{R}_{c,x}$  објекта  $x$  до његових најближих суседа класе  $c$  у скупу за тренирање треба да буду мања од

растојања  $\mathcal{R}_{1-c,x}$  до његових најближих суседа класе  $1 - c$  у том скупу. Нека је  $\Phi_{c,x}$  расподела растојања  $\mathcal{R}_{c,x}$  ( $c \in \{0, 1\}$ ). Тада, метода [91] изводи следећа два теста:

- (а) Тест који растојања  $\mathcal{R}_{0,x}$  и  $\mathcal{R}_{1,x}$  користи да испита нулту хипотезу  $\Phi_{0,x} = \Phi_{1,x}$ , при чему је алтернативна хипотеза да је дистрибуција  $\Phi_{0,x}$  стохастички мања од дистрибуције  $\Phi_{1,x}$  (у ознаци  $\Phi_{0,x} \prec \Phi_{1,x}$ ). Нека је  $p_0(x)$  тако добијена  $p$ -вредност.
- (б) Тест који растојања  $\mathcal{R}_{0,x}$  и  $\mathcal{R}_{1,x}$  користи да испита нулту хипотезу  $\Phi_{0,x} = \Phi_{1,x}$ , при чему је алтернативна хипотеза да је дистрибуција  $\Phi_{1,x}$  стохастички мања од дистрибуције  $\Phi_{0,x}$  ( $\Phi_{1,x} \prec \Phi_{0,x}$ ). Нека је  $p_1(x)$  тако добијена  $p$ -вредност.

Потом, објекат  $x$  класификује се као позитиван ако важи  $p_1(x) < p_0(x)$ , а као негативан уколико је  $p_0(x) < p_1(x)$ . У случају да је  $p_0(x) = p_1(x)$ , објекат  $x$  можемо сврстати у класу за коју је грешка озбиљнија, иако то правило није експлицитно наведено у раду [94]. Метода [94] може да ограничи вероватноћу  $\mathbb{P}(Y = 1 - c \mid \hat{Y} = c)$ , где  $\hat{Y}$  моделује предикцију за насумичан објекат  $X$  ( $c \in \{0, 1\}$ ). Међутим, то нису исте вероватноће као FPR и FNR.

Иако досадашње методе развијене у оквиру приступа ТВС не контролишу вероватноће FPR и FNR, оне показују да се статистички тестови расподела могу користити у бинарној класификацији, као и да се њима може ограничити вероватноћа средње грешке  $Y = 1 - c \mid \hat{Y} = c$  (за  $c \in \{0, 1\}$ ). У одељку 4.9 биће показано како се метода ТВС из рада [94] може модификовати тако да под одређеним условима контролише вероватноћу FPR или FNR.

## 4.7 Индекси типичности

Нека су  $q_0$  и  $q_1$  класне густине дефинисане као у директној Нејман-Пирсоновој класификацији. Индекс типичности објекта  $x$  у односу на класу  $c \in \{0, 1\}$  дефинисан је формулом (4.14) [96].

$$p_c(x) = \mathbb{P}(q_c(X) \leq q_c(x) \mid y(X) = c) \quad (4.14)$$

Првобитно су се индекси (4.14) служили претпоставком да су  $q_0$  и  $q_1$  густине нормалних расподела. Међутим, та претпоставка није неопходна за примену ових индекса у пракси. Проблем ове методе је што, као и директна Нејман-Пирсонова класификација, захтева естимацијају класних густина, што је проблематично у просторима с великим бројем димензија и просторима чије су неке димензије непрекидне, а неке категоријске.

## 4.8 Извођење статистичких тестова из бинарних класификатора

У овом одељку биће представљена једноставна метода за извођење статистичких тестова из бинарних класификатора. Сличности и разлике између бинарне класификације у машинском учењу и тестирања хипотеза у статистици могу се лако уочити [97].

- Хипотезе приликом класификације објекта  $x$  су  $H_0 : y(x) = 0$  и  $H_1 : y(x) = 1$ .
- Узорак на којем тестирамо хипотезе статистичких тестова одговара објекту  $x$  који класификујемо применом класификатора у машинском учењу.
- Нумерички предизлаз  $t(x)$  датог класификатора за задати објекат  $x$  еквивалент је реализованој вредности тестне статистике у тестирању хипотеза. То значи да

променљива  $T = t(X)$  представља статистику теста који одговара бинарном класификатору чији је  $t$  предизлаз.

- Класификациони праг  $\tau$  одговара критичној вредности статистика тестова у класичној статистици.

На основу расподеле статистике  $T = t(X)$  можемо израчунати  $p$ -вредност конкретног предизлаза  $t(x)$ . У фреквентистичкој статистици  $p$ -вредност је мера статистичке неизвесности и користи се за контролу вероватноће да тест одбади тачну нулту хипотезу. Појам  $p$ -вредности формализован је у дефиницији 4.1.

**Дефиниција 4.1** ([98, 99]). *Нека је  $T$  статистика теста, а  $\{x_i\}_{i=1}^n$  узорак. Нека су више вредности статистике  $T$  чешиће уколико важи нулта хипотеза  $H_0 : \theta = \theta_0$ , где је  $\theta$  испитивани параметар, а  $\theta_0$  вредност коју нултом хипотезом претпостављамо да он има. Нека је  $\{X_i\}_{i=1}^n$  насумичан узорак. Тада је  $p$ -вредност реализоване вредности статистике  $T$  на узорку  $\{x_i\}_{i=1}^n$  дефинисана на следећи начин:*

$$p(x) = \mathbb{P}(T(\{X_i\}_{i=1}^n) \geq T(\{x_i\}_{i=1}^n) \mid H_0). \quad (4.15)$$

Аналогно можемо дефинисати и  $p$ -вредности за класификацију ако претпоставимо да више вредности предизлаза одговарају хипотези  $H_1$  (односно, класи 1), а ниже вредности хипотези  $H_0$  (тј. класи 0). Услов је да класификатор из којег изводимо статистички тест има нумерички предизлаз, што је случај са већином модела у машинском учењу.

**Дефиниција 4.2** (Класификационе  $p$ -вредности). *Нека је  $T$  статистика класификатора чији је предизлаз функција  $t$ . Нека је  $x$  нов објекат који треба класификовати. Класификациона  $p$ -вредност за хипотезу  $H_0$ , у ознаци  $p_0$ , дефинисана је на следећи начин:*

$$p_0(x) = \mathbb{P}(T \geq t(x) \mid Y = 0). \quad (4.16)$$

*Класификациона  $p$ -вредност за хипотезу  $H_1$ , у ознаци  $p_1$ , може се дефинисати аналогно:*

$$p_1(x) = \mathbb{P}(T \leq t(x) \mid Y = 1). \quad (4.17)$$

Интегралном трансформацијом вероватноће може се показати да овако дефинисане  $p_c$ -вредности прате униформну расподелу на одсечку  $[0, 1]$ . То значи да ако објекат  $x$  класификујемо као негативан када важи  $p_1(x) \leq \alpha$ , вероватноћа лажно позитивних резултата FNR биће једнака прагу  $\alpha$  зато што важи  $\text{FNR} = \mathbb{P}\{\hat{p}_1(X) \leq \alpha\}$ . Вероватноћу лажно позитивних предикција можемо контролисати ако објекат  $x$  класификујемо као позитиван када је  $p_0(x) \leq \alpha$ .

### 4.8.1 Интерпретација

Класификационе  $p_c$ -вредности треба тумачити у складу са принципима фреквентистичке школе у статистици. Вредност  $p_1(x)$  представља граничну вредност фреквенције са којом класификатор  $\mathbf{1}\{t(\cdot) > t(x)\}$ , еквивалентан класификатору  $\mathbf{1}\{p_1(\cdot) > p_1(x)\}$ , да је лажно негативне предикције. Слично можемо протумачити  $p_0(x)$ .

Оправдање за прихватање хипотезе  $H_0$  када је  $p_1(x) \leq \alpha$  лежи у томе што бисмо применом таквог класификационог правила ретко добијали лажно позитивне резултате ако бисмо као номинални праг грешке одабрали малу вредност  $\alpha$  [100]. Аналогни закључак важи за вредност  $p_0(x)$ .

Најчешћи номинални прагови у статистици су 0.01 и 0.05, али можемо користити било коју вредност из интервала  $(0, 1)$  која се на основу теоријских и емпиријских резултата може оправдати као праг.

## 4.8.2 Равнотежа између FNR и FPR

Класификација помоћу  $p_c$ -вредности и номиналног прага  $\alpha$  калибрише класификатор из којег је изведен тест. Нека је  $\tau$  првобитна вредност његовог класификационог прага,  $a(\tau)$  његова тачност, и нека су  $w_1 = \mathbb{P}(y(X) = 1)$  и  $w_0 = \mathbb{P}(y(X) = 0)$  априорне вероватноће хипотеза  $H_1$  и  $H_0$ . Тада важи следећа релација:

$$w_1 \text{FNR}(\tau) + w_0 \text{FPR}(\tau) + a(\tau) = 1. \quad (4.18)$$

Смањујући једну од вероватноћа на  $\alpha$  (нпр. FNR), мењамо класификациони праг са старе вредности  $\tau$  на нову вредност  $\tau'$ . Смањење стопе FNR доводи до тога да се разлика  $w_1(\text{FNR}(\tau) - \alpha)$  дистрибуира између  $w_0 \text{FPR}(\tau')$  и  $a(\tau')$ . График зависности FNR од FPR представља ROC-криву заротирану око праве  $\text{TPR} = 1/2$  (где је TPR вероватноћа тачних позитивних резултата, једнака  $1 - \text{FNR}$ ). Тачке где крива сече праву једначине  $\text{FPR} \mapsto (1-a)/w_0 - (w_1/w_0)\text{FPR}$  за фиксирану тачност  $a \in [0, 1]$  показују класификационе прагове који имају исту тачност ( $a$ ), али другачији однос вероватноћа FNR и FPR.

## 4.8.3 Естимација $p_c$ -вредности

Класификационе  $p_c$ -вредности можемо израчунати непараметарским естиматорима из дефиниције 4.3.

**Дефиниција 4.3.** Нека је  $\mathcal{Z} = \{(X_i, Y_i)\}_{i=1}^n$  скуп насумичних објеката који нису искоришћени за тренирање датог класификатора и нека је

$$\mathcal{Z}_c = \{X_i^{(c)}\}_{i=1}^{n_c} = \{X_j \mid j \in \{1, 2, \dots, n\} \wedge Y_j = c\} \quad (c = 0, 1).$$

Нека је  $x$  објекат који треба класификовати. Естиматори одговарајућих  $p_c$ -вредности дефинисани су на следећи начин:

$$\hat{p}_{0, n_0}(x) = \frac{1}{n_0} \sum_{i=1}^{n_0} \mathbf{1} \left\{ t(X_i^{(0)}) \geq t(x) \right\} \quad (4.19)$$

и

$$\hat{p}_{1, n_1}(x) = \frac{1}{n_1} \sum_{i=1}^{n_1} \mathbf{1} \left\{ t(X_i^{(1)}) \leq t(x) \right\}. \quad (4.20)$$

Да бисмо поједноставили нотацију, изостављаћемо индекс  $n_c$  ( $c \in \{0, 1\}$ ) из израза (4.19) и (4.20) ако је јасан из контекста.

Естиматори (4.19) и (4.20) су по својој дефиницији такође и естиматори класних функција расподеле променљивих  $t(X) \mid Y = 0$  и  $t(X) \mid Y = 1$ . Теорема 4.2 наводи добра својства ових естиматора.

**Теорема 4.2.** Нека случајне променљиве  $X_1, X_2, \dots, X_n$  представљају независне објекте класе  $c \in \{0, 1\}$ , које користимо за израчунавање естиматора  $\hat{p}_{c, n}$ . Важи да естиматор  $\hat{p}_{c, n}$  има следећа својства.

(а) Естиматор  $\hat{p}_{c, n}$  је центриран:

$$(\forall x)(y(x) = c) \quad \mathbb{E}_{X_1, X_2, \dots, X_n} [\hat{p}_{c, n}(x)] = p_c(x). \quad (4.21)$$

(б) Варијанса му је коначна, ограничена одозго и опада линеарно са  $n$ :

$$(\forall x)(y(x) = c) \quad \text{Var}_{X_1, X_2, \dots, X_n} [\hat{p}_{c, n}(x)] = \frac{1}{n} p_c(x)(1 - p_c(x)) \leq \frac{1}{4n}. \quad (4.22)$$

(в) За свако  $x \in \mathcal{X}$ , он асимптотски тежи нормалној расподели:

$$\sqrt{n}(\hat{p}_{c,n}(x) - p_c(x)) \xrightarrow{d} \mathcal{N}(0, p_c(x)(1 - p_c(x))). \quad (4.23)$$

(г) Конвергенција ка правој вредности  $p_c$  је скоро сигурна:

$$\mathbb{P}\left(\lim_{n \rightarrow \infty} \hat{p}_{c,n} = p_c\right) = 1. \quad (4.24)$$

(д) Симултани интервални поверења произвољне ширине  $\varepsilon > 0$  могу се добити за довољно велико  $n$ :

$$\mathbb{P}\left(\sup_x |\hat{p}_{c,n}(x) - p_c(x)| \leq \varepsilon\right) \geq 1 - 2e^{-2n\varepsilon^2}. \quad (4.25)$$

*Доказ.*

(а) Ово својство следи из линеарности очекивања.

(б) Својство (4.22) следи из декомпозиције варијансе  $\text{Var}[aZ_1 + bZ_2] = a^2\text{Var}[Z_1] + b^2\text{Var}[Z_2]$  линеарне комбинације независних скучајних променљивих  $Z_1$  и  $Z_2$ , тога што су индикатори  $\mathbf{1}\{T(X_i) \rho T(x)\}$  ( $\rho \in \{\leq, \geq\}$ ) независни и прате исту расподелу, као и тога што максимум функције  $f(u) = u(1 - u)$  на одсечку  $[0, 1]$  износи  $1/4$ .

(в) Ово својство се доказује на основу централне граничне теореме.

(г) Скоро сигурна конвергенција доказана је Гливенко–Кантелијевом теоремом (в. [101]).

(д) Формула (4.25) представља директну последицу Дворецки–Кифер–Волфовицове неједнакости [102, 103].

□

Теорема 4.2 показује да ће емпиријске оцене класификационих вредности бити врло прецизне уколико се за њихово израчунавање користи довољно велик скуп  $\mathcal{Z}_c$  ( $c \in \{0, 1\}$ ). Естиматори (4.19) и (4.20) тада прате практично униформну расподелу на  $[0, 1]$ , што омогућава прецизну контролу грешке на изабраној класи.

## 4.9 Контрола грешке посредним тестирањем

У овом одељку ће бити представљена модификација методе ТВС из рада [94] која статистичке тестове изводи на дистанцама до најближих позитивних и негативних суседа. Кључна модификација је коришћење композитне нулте хипотезе о расподелама  $\Phi_0$  и  $\Phi_1$ . Такође, биће представљена и њена тежинска верзија WTBS (енгл. *weighted TBS*).

### 4.9.1 Тестирање растојања до најближих суседа

Нека је  $c \in \{0, 1\}$  ознака критичне класе, тј. нека је озбиљнија грешка сврставање објекта класе  $c$  у класу  $1 - c$ . Ако важи  $y(x) = c$ , растојања  $\mathcal{R}_{c,x}$  објекта  $x$  до његових најближих суседа у скупу  $\mathcal{S}_c$  треба да буду мања од дистанци  $\mathcal{R}_{1-c,x}$  до његових најближих суседа у скупу  $\mathcal{S}_{1-c}$ . Другим речима, требало би да важи  $\mathbb{P}(\Phi_{c,x} < \Phi_{1-c,x}) >$

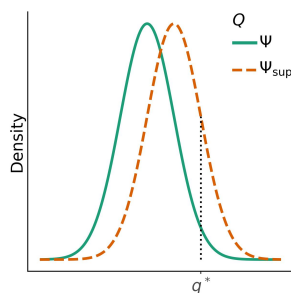
$\mathbb{P}(\Phi_{c,x} > \Phi_{1-c,x})$ , што означавамо као  $\Phi_{c,x} \triangleleft \Phi_{1-c,x}$  (релације  $\triangleleft$  и  $\triangleright$  дефинишу се аналогно). Случај  $y(x) = 0$  одговара релацији  $\Phi_{0,x} \triangleleft \Phi_1(x)$ , а случај  $y(x) = 1$  покрива сценарио  $\Phi_{1,x} \triangleleft \Phi_{0,x}$ . Ако је класификација позитивног објекта као негативног озбиљнија грешка, тада у граничном случају  $\mathbb{P}(\Phi_{c,x} < \Phi_{1-c,x}) = \mathbb{P}(\Phi_{c,x} > \Phi_{1-c,x})$  треба сврстати објекат  $x$  у позитивну класу, и обрнуто. То је оправдано зато што се тако умањује ризик да се направи озбиљнија грешка.

Статистички тест  $\mathbf{T}$  композитне нулте хипотезе  $H_0 : \Phi_{c,x} \triangleleft \Phi_{1-c,x}$  и алтернативне хипотезе  $H_a : \Phi_{c,x} \triangleright \Phi_{1-c,x}$  одбацује тачну нулту хипотезу са вероватноћом не вишом од задатог прага  $\alpha$  уколико су испуњене претпоставке о дистрибуцијама  $\Phi_{c,x}$  и  $\Phi_{1-c,x}$  за примену тог теста. Нека је  $p_c(x)$   $p$ -вредност израчуната извођењем таквог теста за произвољан објекат  $x$  и нека је, за задати праг статистичке значајности  $\alpha$ , класификатор  $\hat{y}_{c,\alpha}^{\mathbf{T}}$  дефинисан на следећи начин:

$$\hat{y}_{c,\alpha}^{\mathbf{T}}(x) = \begin{cases} c, & p_c(x) > \alpha \\ 1 - c, & p_c(x) \leq \alpha \end{cases} \quad (x \in \mathcal{X}). \quad (4.26)$$

Ако за сваки објекат  $x$  који припада класи  $c$  важи да су претпоставке теста  $\mathbf{T}$  испуњене и да је  $\Phi_{c,x} \triangleleft \Phi_{1-c,x}$ , тада ће класификатор  $\hat{y}_{c,\alpha}^{\mathbf{T}}$  погрешно класификовати објекте класе  $c$  са вероватноћом од највише  $\alpha$ .

У пракси је за примену класификатора (4.26) и израчунавање  $p_c$ -вредности довољно да се пронађе супремум  $\Psi_{\text{sup}}$  (у односу на релацију  $\triangleleft$ ) свих расподела  $\Psi$  које статистика  $Q$  теста  $\mathbf{T}$  може пратити ако важи композитна нулта хипотеза  $\Phi_{c,x} \triangleleft \Phi_{1-c,x}$ . Пример је представљен на слици 4.1. На њој је приказано како за произвољан објекат  $x$  статистика  $Q$  прати (непознату) расподелу  $\Psi$  која је стохастички мања од познате расподеле  $\Psi_{\text{sup}}$ . Без губитка општости, претпоставимо да су више вредности статистике  $Q$  у складу са алтернативном хипотезом. У том случају, за сваку реализацију  $q^*$  статистике  $Q$ , важи да је  $p$ -вредност израчуната на основу дистрибуције  $\Psi$  мања или једнака од  $p$ -вредности израчунате помоћу расподеле  $\Psi_{\text{sup}}$ . Последица је да ће вероватноћа одговарајуће класификационе грешке износити највише  $\alpha$  ако користимо  $p$ -вредности добијене путем расподеле  $\Psi_{\text{sup}}$ . Супремум  $\Psi_{\text{sup}}$  углавном одговара граничном случају  $\Phi_{c,x} = \Phi_{1-c,x}$ . То



**Слика 4.1:** За реализовану вредност  $q^*$  статистике  $Q$  важи  $\mathbb{P}_{Q \sim \Psi}(Q \geq q^*) \leq \mathbb{P}_{Q \sim \Psi_{\text{sup}}}(Q \geq q^*)$ .

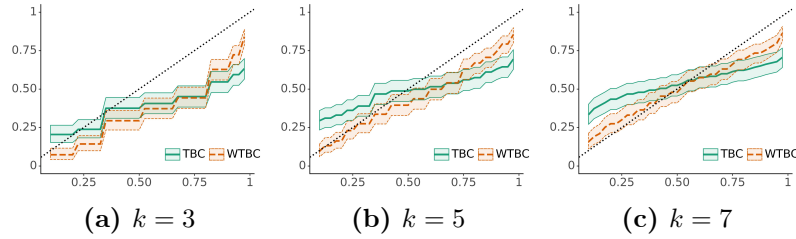
важи код стандардних тестова који се користе за испитивање расподела на основу два узорка, као што су независни и упарени  $t$  тест и Ман–Витнијев  $U$  тест (MWU). Иако они нису формулисани тако да тестирају релацију  $\triangleleft$ , лако се показује да се они могу користити као тестови таквог односа расподела.

## 4.9.2 Тестирање тежинских растојања до најближих суседа

У раду [94] користи се  $k = 3$  суседа, али је у складу са мотивацијом приступа оправдано претпоставити да коришћење већег броја суседа даје боље резултате. Ипак, то није

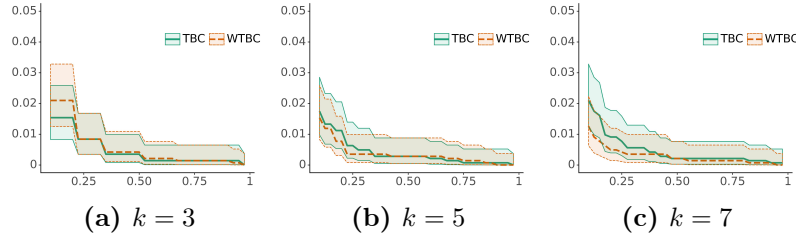
нужно случај, већ је могуће и да дође до погоршања перформанси. Претпоставка је да то тога може доћи уколико су објекти критичне класе ближи објектима друге него своје.

У методи WTBC, тај проблем решава се тако што се растојања  $\mathcal{R}_{0,x}$  и  $\mathcal{R}_{1,x}$  множе тежинама које су реципрочне њиховим ранговима. Нека је  $\tilde{\mathcal{R}}_{c,x} = [r_i^{(c)}]_{i=1}^k$  неопадајуће сортиран низ растојања објекта  $x$  до његових  $k$  најближих суседа класе  $c \in \{0, 1\}$ . Уместо тим дистанцама, у методи WTBC служимо се тежинским растојањима  $\{\frac{1}{i}r_i^{(c)} \mid r_i \in \tilde{\mathcal{R}}_{c,x}\}$  ( $c = 0, 1$ ). То нам омогућава да користимо више суседа и остваримо прецизнију контролу грешке на критичној класи, што показује слика 4.2. На њој су приказани резултати примене метода TBC и WTBC на скуп HMEQ из репозиторијума Kaggle<sup>1</sup>. У обе методе у овом примеру користе се косинусно растојање, тест MWU и  $k = 3, 5, 7$  суседа за дефинисање скупова  $\mathcal{R}_0(x)$  и  $\mathcal{R}_1(x)$ . Емпиријска грешка  $\widehat{\text{FNR}}(\alpha)$  значајно ближе прати номинални праг  $\alpha$  у случају WTBC него у методи TBC из одељка 4.9.1.



**Слика 4.2:** Зависност вероватноће  $\widehat{\text{FNR}}$  од прага  $\alpha$ . Интервали поверења израчунати су с поверењем од 99%.

Примена тежина у методи WTBC не утиче негативно на вероватноћу друге грешке,  $\widehat{\text{FPR}}$ , што видимо на слици 4.3. Поклапање интервала поверења за вероватноће  $\widehat{\text{FPR}}(\alpha)$  показује да нема значајних разлика између метода WTBC и TBC.



**Слика 4.3:** Зависност вероватноће  $\widehat{\text{FPR}}$  од прага  $\alpha$ . Интервали поверења израчунати су с поверењем од 99%.

Сличан ефекат уочава се и уколико се уместо Ман-Витнијевог теста користи независни  $t$  тест.

### 4.9.3 Пример

Наводимо пример из скупа HMEQ. Нека су сортирана косинусна растојања од позитивног објекта до његових пет најближих суседа у обе класе:

$$\mathcal{R}_1 = [0.267, 0.283, 0.289, 0.295, 0.307] \quad \text{и} \quad \mathcal{R}_0 = [0.233, 0.241, 0.242, 0.243, 0.245].$$

Реализована вредност статистике теста MWU износи  $U = 25$ . Њена  $p$ -вредност је 0.004. За  $t$  тест се добија вредност статистике од 6.836, чија је  $p$ -вредност мања од  $10^{-4}$ . Осим

<sup>1</sup><https://www.kaggle.com/datasets/ajay1735/hmeq-data>

уколико не користимо веома ниске номиналне прагове  $\alpha$ , овај објекат ће бити погрешно класификован као негативан.

Ако применимо тежине реципрочне ранговима, добијају се следећа растојања:

$$\mathcal{R}_1 = [0.267, 0.142, 0.096, 0.074, 0.061] \quad \text{и} \quad \mathcal{R}_0 = [0.233, 0.12, 0.081, 0.061, 0.049].$$

Сада тест MWU резултира статистиком  $U = 15.5$  чија је  $p$ -вредност једнака 0.3. Слично,  $t$  тест реализује 0.383 као статистику и враћа  $p$ -вредност од 0.356. Видимо да примена тежинског приступа омогућава да се дати објекат исправно класификује као позитиван (осим уколико се праг  $\alpha$  не постави на веома високу вредност).

## 4.10 Евалуација

Евалуиране су следеће методе за контролу грешке: конформална предикција (формулација (4.6), CPF), директна Нејман-Пирсонова класификација (DNP) (при чему се количник густина процењује кернелима [104]), индекси типичности (TI), алгоритам UA, тежинско посредно тестирање (WTBC), као и варијанта посредног тестирања (TBC) заснована на најближим суседима, предложена у одељку 4.9.1. Као референтна метода је због једноставности и интуитивности узето извођење статистичких тестова из бинарних класификатора (енгл. *statistical classification testing*, SCT) представљено у одељку 4.8.

### 4.10.1 Скупови за евалуацију

Наведене методе тестиране су на четири скупа различите величине. У питању су:

- HMEQ<sup>2</sup> - скуп за обуку детектора ризичних пријава за кредит, односно, апликаната који неће моћи да врате кредит финансијској институцији. Скуп садржи податке о 5960 захтева, од којих 1189 чине захтеви странака које нису исплатиле кредит.
- SPAM<sup>3</sup> - скуп мејлова за обуку класификатора који проналазе нежељене мејлове у електронској пошти. Скуп садржи 4601 мејл, од којих 2788 нису непожељни. Озбиљнија грешка овде је класификација регуларне поруке као непожељне.
- PIMA<sup>4</sup> [105] - скуп о медицинским подацима особа које имају и особа које немају дијабетес. Обухвата 768 особа, од којих 268 са дијабетесом.
- NORM - скуп који садржи 1700 нормалних узорака од 20 елемената и 1700 узорака из разних симетричних платикуртичних расподела са 20 елемената. Узорци су сортирани и нормализовани min-max трансформацијом тако да припадају одсечку  $[-3, 3]$ , док су минимуми и максимуми изостављени након трансформације.

За сва четири скупа, циљ је био проценити која метода најбоље контролише вероватноћу FNR.

### 4.10.2 Подешавање параметара

За сваки скуп и метод су пронађена најбоља подешавања параметара тог метода на том скупу. Испробане конфигурације представљене су у табели 4.1.

<sup>2</sup><https://www.kaggle.com/datasets/ajay1735/hmeq-data>

<sup>3</sup><https://www.kaggle.com/datasets/venky73/spam-mails-dataset>

<sup>4</sup><https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database>

Будући да алгоритам UA користи пола објеката критичне класе за калибрацију класификационог прага, половина критичног подскопа резервисана је за калибрацију и у другим методама у којима је то неопходно. Тај параметар означен је као  $\rho$ .

За методу CPF су евалуиране следеће функције као мера неконформалности:

- $avgdev(x, \mathcal{D}, c)$  - Менхетн растојање објекта  $x$  до центра објеката класе  $c \in \{0, 1\}$  у скупу  $\mathcal{D}$ .
- $nearest(x, \mathcal{D}, c)$  - Менхетн растојање између објекта  $x$  и његовог најближег суседа класе  $c \in \{0, 1\}$  у скупу  $\mathcal{D}$ .
- $score(x, c) = |c - t(x)|$  - разлика између предизлаза  $t(x) \in [0, 1]$  и идеалног предизлаза за класу  $c \in \{0, 1\}$ .

У прелиминарним експериментима утврђено је да тип кернела не утиче на перформансе метода TI и DNP, па су одабрани јединични RBF кернел за методу TI и квадратни за метод DNP.

**Табела 4.1:** Испробане конфигурације метода за контролу грешке

Метода	Параметри
SCT	$\rho = 0.5$
TI	кернел: јединични RBF
CPF	степен неконформалности: $\{avgdev, nearest, score\}$ $\rho = 0.5$ уколико је степен $score$
DNP	$\rho = 0.5$ кернел: квадратни
UA	$\delta \in \{0.01, 0.05\}$ $m \in \{1, 5\}$ $\rho = 0.5$
(W)TVC	$k \in \{3, 5, 7, 10\}$ растојање: Менхетн или косинусно тест: Ман-Витнијев (MWU) или $t$ -тест

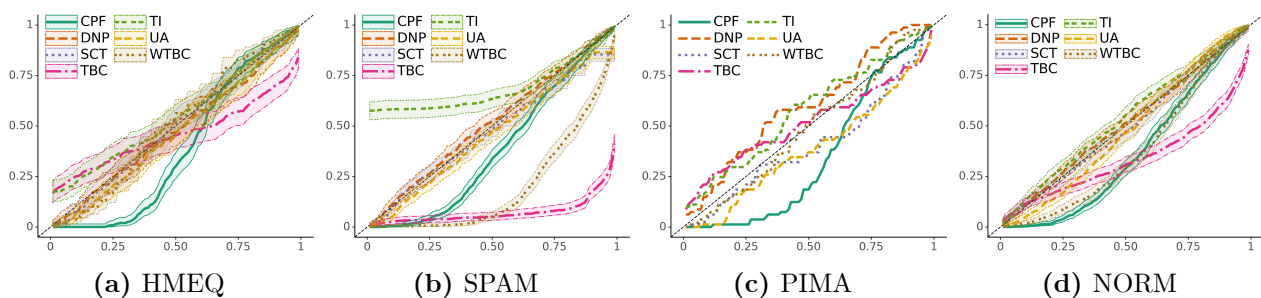
На сваком скупу је искоришћен исти бинарни класификатор за методе SCT и UA, као и за технику CPF која користи функцију  $score$  као степен неконформалности. За скуп HMEQ су то машине потпорних вектора са Платовим скалирањем, за скуп SPAM случајне шуме, а за NORM неуронске мреже са два скривена слоја од по девет неурона и активационом функцијом ReLU. Логистичка регресија је примењена као класификатор за скуп PIMA.

### 4.10.3 Резултати и дискусија

У првом кораку евалуације испитана је способност датих метода да вероватноћу грешке на критичној класи (FNR) поставе на задати номинални праг  $\alpha \in \{0.01, 0.02, \dots, 0.99\}$ . У сваком скупу је 70% података искоришћено за тренирање, а преосталих 30% за евалуацију.

Резултати су приказани на слици 4.4. На основу њих може се закључити следеће.

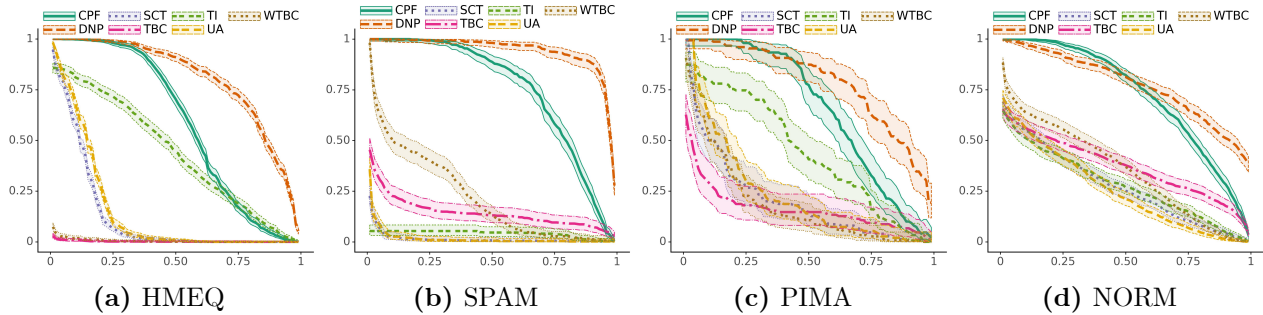
- За скупове HMEQ, SPAM и NORM, методе SCT, UA и DNP практично су ostvarile egzaktnu kontrolu greške  $((\forall \alpha \in (0, 1)) \widehat{FNR}(\alpha) = \alpha)$ .
- На скупу PIMA је најбоље резултате ostvarila metoda WTBC, док је друга најбоља metoda TBC.
- Metoda WTBC ostvarila је egzaktnu kontrolu на скупу HMEQ. Међутим, на скуповима NORM и SPAM, методе WTBC и TBC нису успеле у постављању вероватноће  $\widehat{FNR}(\alpha)$  на задате вредности прагова  $\alpha$ , иако су их одржале на вредностима нижим од прагова.
- Metoda CPF на свим скуповима одржава релацију  $\widehat{FNR}(\alpha) < \alpha$  са значајно нижим емпиријским вероватноћама  $\widehat{FNR}(\alpha)$  од номиналних прагова  $\alpha$ . Metoda TI није ostvarila kontrolu greške на скуповима SPAM и HMEQ, али јесте на скуповима NORM и PIMA.



**Слика 4.4:** Зависност емпиријске оцене вероватноће  $\widehat{FNR}$  од задатог прага  $\alpha$ . Клопер-Пирсонови интервали поверења (99%) приказани су за скупове HMEQ, SPAM и NORM, док су за скуп PIMA изостављени зарад прегледности.

Потом је испитана зависност  $\alpha \mapsto \widehat{FPR}(\alpha)$  вероватноће грешке на негативној класи од номиналног прага за позитивну класу. Та зависност приказана је на слици 4.5. Закључци су следећи.

- Методе SCT и UA су на скуповима HMEQ, NORM и SPAM показале боље резултате од свих других метода. На скупу PIMA је metoda WTBC имала исте перформансе као SCT и UA, док је metoda TBC дала најбоље резултате.
- Методе SCT и UA су готово изједначене, с тим што metoda SCT постиже нешто ниже вероватноће  $\widehat{FPR}$  на скуповима HMEQ и NORM.
- Metoda TI је ostvarila nižu verovatnoћу greške него методе DNP и CPF на свим скуповима. На скупу SPAM, ова metoda готово да дала лажно позитивне предикције, што је у равнотежи са високом вероватноћом лажно негативних предикција.
- Вероватноћа лажно позитивне грешке је код метода WTBC и TBC између вероватноћа методе TI и метода SCT и UA. Metoda TBC има nižu verovatnoћу  $\widehat{FPR}$  од WTBC на свим скуповима сем на скупу HMEQ.
- Методе DNP и CPF су имале највећу вероватноћу лажно позитивних предикција на сва четири скупа.



**Слика 4.5:** Зависност емпиријских оцена вероватноће FPR од номиналних прагова ( $\alpha$ ) за вероватноћу FNR. Приказани су интервали поверења са (99%).

За сваку методу су израчунате следеће мере перформанси ( $[u]_+ = u \cdot \mathbf{1}\{u > 0\}$ ,  $[u]_- = u \cdot \mathbf{1}\{u < 0\}$  за  $u \in \mathbb{R}$ ):

$$a = 2 \int_0^1 [\widehat{\text{FNR}}(\alpha) - \alpha]_+ d\alpha, \quad b = 2 \int_0^1 [\widehat{\text{FNR}}(\alpha) - \alpha]_- d\alpha \quad \text{и} \quad \ell = \int_0^1 \widehat{\text{FPR}}(\alpha) d\alpha.$$

Вредност  $a$  представља нормализовану површину испод криве  $\alpha \mapsto \widehat{\text{FNR}}(\alpha)$ , а изнад графика  $\alpha \mapsto \alpha$ . Она квантификује прекорачење номиналног прага. Нормализација се изводи тако што се дата површина подели својом максималном вредношћу (1/2). Вредност  $b$  дефинише се аналогно као нормализована површина изнад графика  $\alpha \mapsto \widehat{\text{FNR}}(\alpha)$ , а испод  $\alpha \mapsto \alpha$ . Она представља меру у којој дата метода остварује контролу грешке, тј. испуњава услов  $\widehat{\text{FNR}}(\alpha) \leq \alpha$ , али тако што вероватноћу грешке поставља на вредност нижу од прага и тако одступа од идеалне контроле грешке ( $\widehat{\text{FNR}}(\alpha) = \alpha$  за  $\alpha \in (0, 1)$ ). Вредност  $\ell$  дефинисана је као површина испод графика криве  $\alpha \mapsto \widehat{\text{FPR}}(\alpha)$  и  $\alpha$ -осе и квантификује укупну вероватноћу лажно позитивних предикција.

Дате оцене перформанси одређују тродимензионални простор у  $[0, 1]^3$ . Свака метода се на основу израчунатих вредности  $a$ ,  $b$  и  $\ell$  може пресликати у једну његову тачку. Ако је циљ минимизација вероватноће  $\text{FPR}(\alpha)$  под условом  $\text{FNR}(\alpha) \leq \alpha$ , идеалну контролу остварују класификатори пресликани у тачке на правој  $a = 0 \wedge \ell = 0$ . Растојање до те праве (у ознаци  $d_{a,\ell}$ ) представља одступање дате методе од идеалног класификатора. Уколико желимо да минимизујемо одступање вероватноће  $\text{FNR}(\alpha)$  од задатог прага  $\alpha$ , неопходно је да у обзир узмемо и вредност  $b$ . У том случају, идеалан класификатор пресликава се у координатни почетак. Квалитет дате методе тада можемо изразити њеном удаљеношћу до координатног почетка ( $d_{a,b,\ell}$ ).

Растојања испитаних метода представљена су у табели 4.2. Поред дистанци, израчунати су и њихових рангови у односу на оба растојања. Метода WTBC има најбоље рангове на скуповима HMEQ и PIMA. Најбоље перформансе на скуповима SPAM и NORM показује метода SCT. Алгоритам UA је на сва четири скупа остварио идеалну вредност  $a = 0$ .

Просечни рангови приказани су у табели 4.3. Метода SCT има најбоље просечне рангове у односу на растојања  $d_{a,\ell}$  и  $d_{a,b,\ell}$ , као и вредност  $a$ . Најбоље рангиране методе у односу на растојање  $d_{a,\ell}$  су SCT, WTBC, TBC, и UA. За растојање  $d_{a,b,\ell}$ , најбоље методе су SCT, UA и WTBC.

**Табела 4.2:** Рангови метода по вредностима  $a$ ,  $b$  и  $\ell$ . Рангови одређени према еуклидским растојањима  $d_{a,\ell}$  и  $d_{a,b,\ell}$  означени су као  $r_{a,\ell}$  и  $r_{a,b,\ell}$ . Методе су сортиране по просечном рангу  $\bar{r} = \frac{1}{2}(r_{a,\ell} + r_{a,b,\ell})$ . За сваки скуп су подебљане најбоље вредности  $a, b, \ell, d_{a,\ell}$  и  $d_{a,b,\ell}$ .

Скуп	Метода	$a$	$b$	$\ell$	$d_{a,\ell}$	$d_{a,b,\ell}$	$r_{a,\ell}$	$r_{a,b,\ell}$	$\bar{r}$
HMEQ	WTBC	0.0459	0.0396	0.0050	<b>0.0461</b>	<b>0.0608</b>	1	1	1.0
	SCT	0.0059	0.0141	0.1150	0.1151	0.1160	3	2	2.5
	TBC	0.0779	0.1567	<b>0.0021</b>	0.0779	0.1750	2	4	3.0
	UA	<b>0.0000</b>	0.0734	0.1408	0.1408	0.1588	4	3	3.5
	TI	0.0979	<b>0.0009</b>	0.4341	0.4450	0.4450	5	5	5.0
	CPF	0.0101	0.2399	0.5805	0.5805	0.6282	6	6	6.0
	DNP	0.0000	0.0470	0.7955	0.7955	0.7969	7	7	7.0
NORM	UA	0.0173	0.0626	<b>0.2458</b>	<b>0.2464</b>	<b>0.2542</b>	1	1	1.0
	SCT	0.0019	0.0242	0.2657	0.2657	0.2668	2	2	2.0
	TI	0.0704	<b>0.0000</b>	0.2742	0.2831	0.2831	3	3	3.0
	WTBC	0.0032	0.2426	0.3444	0.3445	0.4213	4	4	4.0
	TBC	0.0041	0.3248	0.3639	0.3639	0.4877	5	5	5.0
	CPF	<b>0.0002</b>	0.2653	0.7158	0.7158	0.7634	6	7	6.5
	DNP	0.0199	0.0025	0.7621	0.7624	0.7624	7	6	6.5
PIMA	WTBC	0.0171	0.0441	0.1753	<b>0.1762</b>	<b>0.1816</b>	1	1	1.0
	TBC	0.0923	0.0646	<b>0.1566</b>	0.1818	0.1929	2	2	2.0
	SCT	0.0002	0.1833	0.1964	0.1964	0.2686	4	3	3.5
	UA	<b>0.0000</b>	0.2268	0.1925	0.1925	0.2975	3	4	3.5
	TI	0.1552	<b>0.0000</b>	0.4361	0.4629	0.4629	5	5	5.0
	CPF	0.0016	0.2983	0.6461	0.6461	0.7116	6	6	6.0
	DNP	0.1978	0.0000	0.7731	0.7980	0.7980	7	7	7.0
SPAM	SCT	0.0050	0.0515	<b>0.0128</b>	<b>0.0137</b>	<b>0.0533</b>	1	1	1.0
	UA	<b>0.0000</b>	0.0806	0.0146	0.0146	0.0819	2	2	2.0
	TI	0.3672	<b>0.0013</b>	0.0406	0.3694	0.3694	5	3	4.0
	WTBC	<b>0.0000</b>	0.5978	0.2136	0.2136	0.6349	4	4	4.0
	TBC	<b>0.0000</b>	0.8375	0.1398	0.1398	0.8491	3	6	4.5
	CPF	<b>0.0000</b>	0.2309	0.7451	0.7451	0.7801	6	5	5.5
	DNP	0.0316	0.0007	0.9286	0.9291	0.9291	7	7	7.0

**Табела 4.3:** Просечни рангови  $\bar{r}_{a,\ell}$ ,  $\bar{r}_{a,b,\ell}$  и  $\bar{r}_a$ , израчунати у односу на растојања  $d_{a,\ell}$  и  $d_{a,b,\ell}$ , као и вредност  $a$ .

Метода	$\bar{r}_{a,\ell}$	$\bar{r}_{a,b,\ell}$	$\bar{r}_a$
CPF	6.00	6.00	<b>2.00</b>
DNP	7.00	6.75	4.25
SCT	<b>2.50</b>	<b>2.00</b>	<b>2.00</b>
TBC	3.00	4.25	3.75
TI	4.50	4.00	5.75
UA	<b>2.50</b>	2.50	<b>2.00</b>
WTBC	<b>2.50</b>	2.50	3.00



## Глава 5

# Тестови нормалности засновани на једносмерним мрежама

Из мрежа добијених генетским алгоритмом изведени су статистички тестови методом SCT, која се показала најбољом у експерименталној студији техника за контролу грешке.

Параметри генетског алгоритма постављени су на следеће вредности:  $N = 20$ ,  $\sigma = 8$ ,  $\mu_P = 0.8$ ,  $\mu_G = 0.4$ ,  $\delta = 0.005$ ,  $\eta = 0.02$ ,  $\xi = 0.25$  и  $\omega = 1$ . Та конфигурација одређена је прелиминарним тестовима за одређивање комбинације параметара која резултује брзим мрежама високих перформанси.

За испитивање моћи искоришћени су узорци из расподела у Пирсоновом систему дистрибуција, као и узорци из не-нормалних расподела које се у литератури најчешће користе за евалуацију статистичких тестова нормалности.

### 5.1 Усмерени мрежни тестови

#### 5.1.1 Конфигурације метапараметара

За сваки тип алтернативних расподела дефинисан у табели 3.1 су применом GA подешени сви метапараметри за конструкцију усмерене мреже. Добијене конфигурације представљене су у табели 5.1. Можемо видети да скуп стандардизационих атрибута ни у једној конфигурацији није празан, већ садржи  $p$ -вредности различитих статистичких тестова. Како су  $p$ -вредности већ калибрисане за сваки обим узорака, мрежама нису били потребни велики скупови за тренирање. Оне су разноврсних архитектура умерене сложености, са највише три унутрашња слоја и највише 100 неурона по слоју, сем мреже за тип алтернативних расподела LM, чији слојеви имају 146 вештачких неурона. Добијене конфигурације резултирају мрежама које се брзо тренирају и чији се излаз може брзо израчунати.

Развијено је укупно осам усмерених мрежа са следећим значењем излаза:

- 1 - задати узорак потиче из нормалне расподеле произвољног очекивања  $\mu \in \mathbb{R}$  и произвољне варијансе  $\sigma^2 > 0$ .
- 0 - задати узорак потиче из не-нормалне расподеле која припада типу из чијих су дистрибуција добијени не-нормални узорци за тренирање.

Мреже за алтернативне типове RP и RL исте су као за типове LP и LL, с том разликом што узорци пре пресликавања у репрезентациони простор пролазе кроз додатан корак промене знака. Тако се мрежама усмереним ка улево искошеним расподелама омогућава да препознају удесно искошене узорке, а задрже исту тачност на класи нормалних.

**Табела 5.1:** Најбоље конфигурације метапараметара за конструкцију усмерених мрежа, добијене генетским алгоритмом. За алтернативне типове RP и RL, свим елементима узорака се прво мења знак, па се онда примењују мреже развијене за типове LP и LL.

Тип	Група метапара-метара	Метапараметри
LP	Подаци	$M = 19, L = 1, PD = \perp$
	Репрезентација	$R = s_4, PV = \{KT_{<3}, ST_{<0}, ST_{>0}, SW, DP\}, F = \{n, \bar{z}, \text{median}(z), \text{sd}(z), \text{mad}(z), \beta_2(z)\}, m_L = 0.2, m_R = 0.1, q = 0.02$
	Архитектура	$d = 3, w = 5, a = \text{ReLU}$
	Тренирање	$A = \text{SGD}, c = 0.00897, ES = 0, LRI = 0.034, LRU = \text{constant}, m = 0.3686, NM = \top, I = 12, t = 0.63026$
SP	Подаци	$M = 49, L = 1, PD = \perp$
	Репрезентација	$R = s_4, PV = \{KT_{<3}, KT_{>3}, SK_{<0}, SK_{>0}, ZC, ZA, SW, SF, DP\}, F = \{n, \sqrt{\beta_1}(z), \beta_2(z), \bar{z}, \text{median}(z), \text{sd}(z), \text{mad}(z)\}, q = 0.1, m_L = 0, m_R = 0.2$
	Архитектура	$d = 2, w = 51, a = \text{ReLU},$
	Тренирање	$A = \text{ADAM}, c = 0.00617, ES = 0.2, LRI = 0.02767, b_1 = 0.41951, b_2 = 0.72279, I = 13$
LL	Подаци	$M = 54, L = 7, PD = \perp, \beta_{2,\max} = 57.8$
	Репрезентација	$R = s_1, PV = \{KT_{<3}, SK_{<0}, SK_{>0}, SW, DP\}, F = \{n, \sqrt{\beta_1}(z), \beta_2(z), \text{median}(z), \text{sd}(z), \text{mad}(z)\}, q = 0.2, m_L = 0.1, m_R = 0.1$
	Архитектура	$d = 2, w = 51, a = \tanh$
	Тренирање	$A = \text{ADAM}, ES = 0.176, c = 0.00428, LRI = 0.03162, b_1 = 0.41805, b_2 = 0.49718, I = 14$
SL	Подаци	$M = 10, L = 8, \beta_{2,\max} = 28.97, PD = \perp$
	Репрезентација	$R = s_3, PV = \{KT_{<3}, KT_{>3}, ST_{>0}, ZC, SW, DP\}, F = \{n, \sqrt{\beta_1}(z), \beta_2(z), \bar{z}, \text{sd}(z)\}, q = 0.2, m_L = 0.1, m_R = 0$
	Архитектура	$d = 2, w = 39, a = \tanh$
	Тренирање	$A = \text{SGD}, c = 0.00723, ES = 0.11476, LRI = 0.04406, m = 0.2258, NM = \top, I = 11, t = 0.68203$
LM	Подаци	$M = 66, L = 3, PD = \perp$
	Репрезентација	$R = s_4, PV = \{KT_{>3}, SK_{<0}, SK_{>0}, ZC, ZA, SW, DP\}, F = \{n, \beta_2(z), \bar{z}, \text{median}(z), \text{sd}(z), \text{mad}(z)\}, q = 0.02, m_L = 0.2, m_R = 0.2$
	Архитектура	$d = 3, w = 146, a = \text{ReLU}$
	Тренирање	$A = \text{SGD}, c = 0.00428, ES = 0.19045, LRI = 0.04794, LRU = \text{constant}, m = 0.98663, NM = \perp, I = 11$
RM	Подаци	$M = 65, L = 1, PD = \top$
	Репрезентација	$R = s_4, PV = \{KT_{>3}, SK_{<0}, SK_{>0}, ZC, ZA, SW, DP\}, F = \{n, \sqrt{\beta_1}(z), \beta_2(z), \text{median}(z), \text{sd}(z), \text{mad}(z)\}, q = 0.02, m_L = 0.2, m_R = 0.1$
	Архитектура	$d = 1, w = 5, a = \text{logistic}$
	Тренирање	$A = \text{SGD}, c = 0.00308, ES = 0, LRI = 0.09908, LRU = \text{adaptive}, m = 0.87187, NM = \perp, I = 13$

### 5.1.2 Стабилност

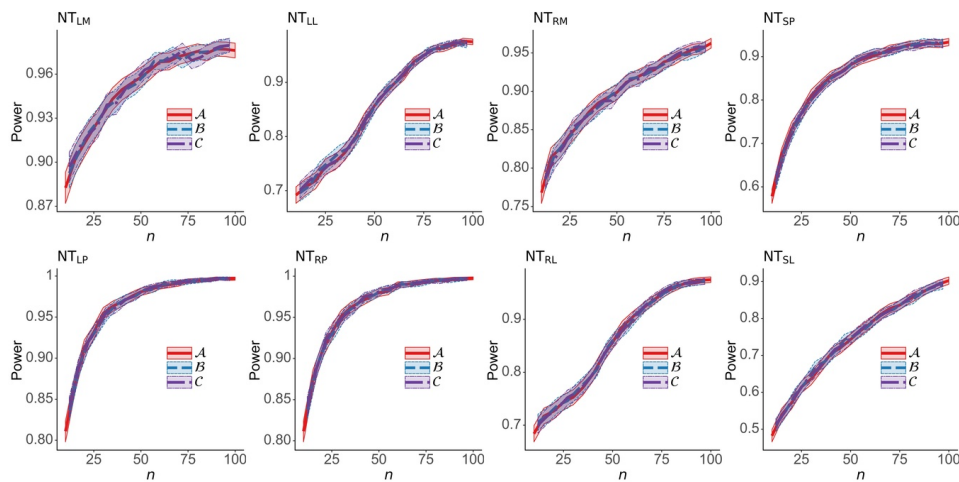
Први корак у евалуацији мрежа било је испитивање њихове стабилности и монотоности тачности. У ту сврху су за сваку мрежу састављена три скупа са узорцима из

расподела знатно различите варијабилности и са различитим очекивањима и обимима:

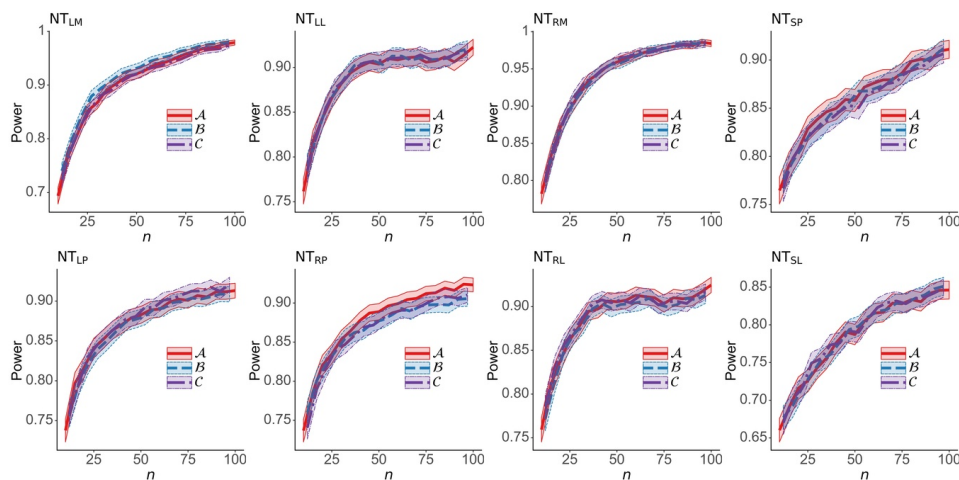
- $\mathcal{A}$  – очекивања  $\mu$  и девијације  $\sigma$  су насумично бирани из истих интервала као за тренирање ( $\mu \in [-100, 100]$ ,  $\sigma \in [10^{-2}, 10^2]$ ). Узорци су обима  $n \in \{10, 15, \dots, 100\}$ .
- $\mathcal{B}$  – очекивања су насумично бирана из интервала  $[1000, 2000]$ , а девијације из интервала  $[10^{-4}, 10^{-3}]$ . Узорци су обима  $n \in \{12, 17, \dots, 97\}$ .
- $\mathcal{C}$  – очекивања су бирана из интервала  $[-2000, -1000]$ , а девијације из  $[10^3, 10^4]$ . Узорци су обима  $n \in \{12, 17, \dots, 97\}$ .

Сваки скуп садржи по 10 000 нормалних и 10 000 не-нормалних узорака за сваки обим.

Резултати евалуације приказани су на сликама 5.1 и 5.2. На њима се види да варијације у очекивању, стандардној девијацији и обиму узорака не утичу значајно на тачност мрежа, као и да тачност није опадајуће монотоности у односу на обим узорака.



**Слика 5.1:** Тачност усмерених мрежа на нормалним узорцима из скупова  $\mathcal{A}$ ,  $\mathcal{B}$  и  $\mathcal{C}$ , добијених из дистрибуција различите варијабилности и других параметара. Приказани су интервали поверења од 99.9%.



**Слика 5.2:** Тачност усмерених мрежа на не-нормалним узорцима из скупова  $\mathcal{A}$ ,  $\mathcal{B}$  и  $\mathcal{C}$ , добијених из дистрибуција различите варијабилности и других параметара. Приказани су интервали поверења од 99.9%.

### 5.1.3 Извођење усмерених мрежних тестова нормалности

Из мрежа су методом SCT изведени статистички тестови. За сваки узорак  $x$  који добије на улазу, усмерена мрежа враћа  $p_{\mathcal{N}}(x)$ , процену вероватноће да узорак  $x$  прати нормалну расподелу  $\mathcal{N}(\mu, \sigma^2)$  са произвољним параметрима  $\mu$  и  $\sigma^2$ . За сваку мрежу и сваки обим  $n = 5, 10, \dots, 95, 100$ , коришћењем 10 000 узорака датог обима одређени су квантили процене  $p_{\mathcal{N}}$  чији су редови  $\alpha = 0.01, 0.05$  и  $0.1$ . Ти квантили представљају класификационе прагове који одговарају номиналним праговима вероватноће да се нормалан узорак класификује као не-нормалан, што је у статистици познато као грешка прве врсте. На тај начин је остварена контрола грешке на три нивоа значајности који се најчешће користе у научним истраживањима и анализи података.

У наставку ћемо мрежни тест усмерен ка типу  $T \in \{LL, LM, LP, RL, RM, RP, SL, SP\}$  означавати као  $NT_T$ . За задати узорак  $x = \{x_i\}_{i=1}^n$ , чије елементе моделује случајна променљива  $X$ , хипотезе теста  $NT_T$  су:

$H_0$  Узорак  $x$  потиче из нормалне расподеле:  $(\exists \mu \in \mathbb{R}, \sigma^2 > 0) X \sim \mathcal{N}(\mu, \sigma^2)$ .

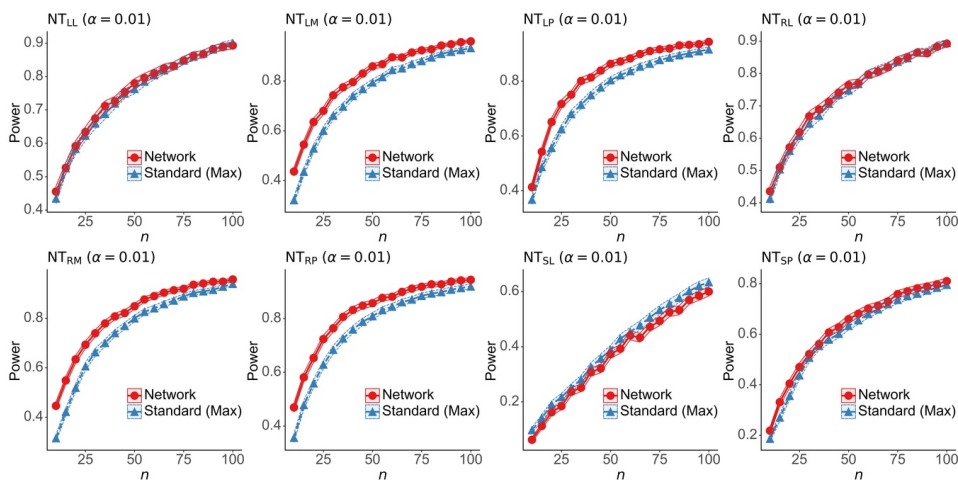
$H_a$  Узорак  $x$  потиче из расподеле типа  $T$ :  $(\exists \Psi \in T) X \sim \Psi$ .

### 5.1.4 Поређење са стандардним статистичким тестовима

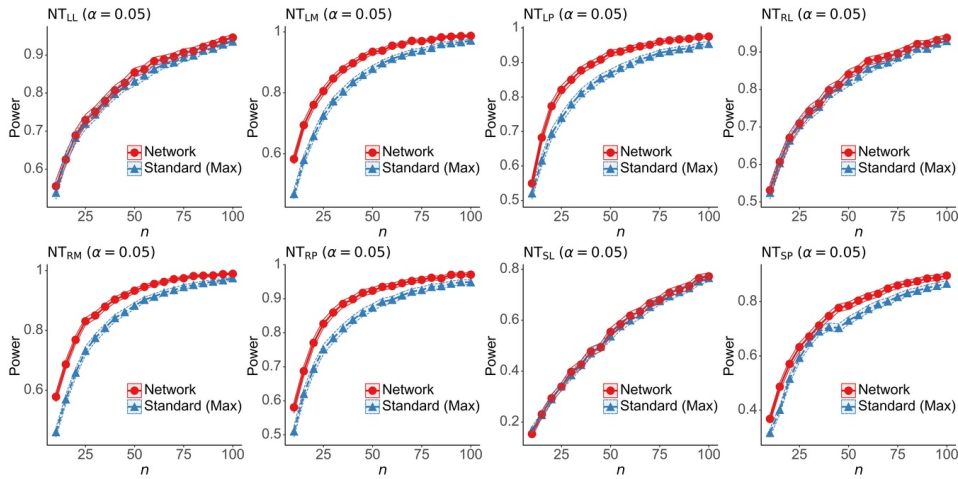
Након потврде стабилности добијених усмерених мрежа, извршено је њихово поређење са одговарајућим усмереним тестовима ST и KT, као и следећим омнибус тестовима нормалности: SW, SF, JB, CVM, AD, DP, LF, CS, ZC и ZA.

За сваки не-нормални тип насумично је дефинисано 1000 парова  $(\sqrt{\beta_1}, \beta_2)$  у складу са ограничењима тог типа. За сваки пар су насумично генерисани очекивање  $\mu \in [-100, 100]$  и стандардна девијација  $\sigma \in [10^{-2}, 10^2]$ , чиме су одређени моменти који одговарају не-нормалним расподелама тог типа. Из сваке такве расподеле генерисано је по десет узорака обима  $n = 10, 15, \dots, 95, 100$ . Тиме је добијен скуп који садржи 10 000 узорака обима  $n = 10, 15, \dots, 95, 100$  из сваког од осам основних не-нормалних типова.

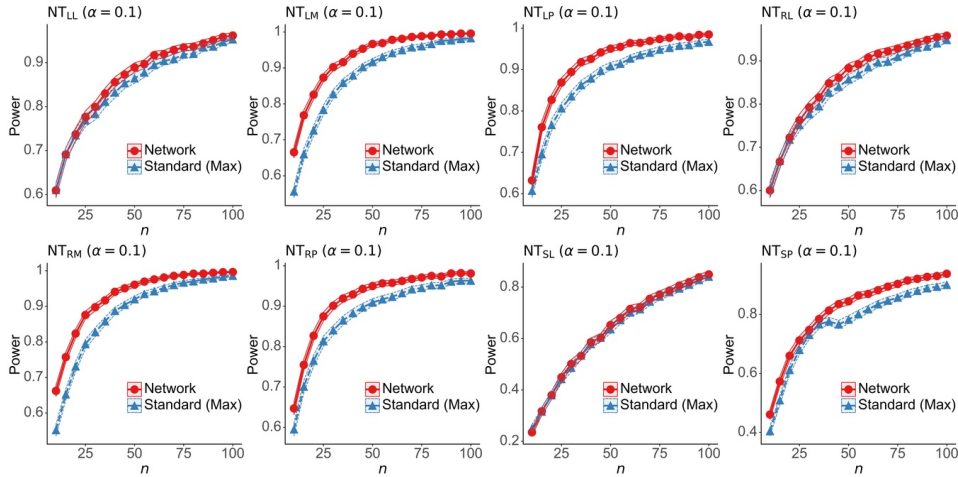
Резултати су представљени на сликама 5.3–5.5. За сваки обим  $n$  и алтернативни тип  $T$  приказана је моћ мрежног теста  $NT_T$ , као и највећа моћ коју је остварио неки класичан статистички тест на узорцима обима  $n$  и типа  $T$ . Видимо да мрежни тестови нормалности имају упоредиву или већу моћ од стандардних статистичких тестова на свим не-нормалним типовима и за сва три прага значајности.



**Слика 5.3:** Моћ усмерених мрежних тестова у поређењу с максималном моћи стандардних статистичких тестова, уз праг значајности  $\alpha = 0.01$ . Приказани су интервали поверења од 99.9%.



Слика 5.4: Моћ усмерених мрежних тестова у поређењу с максималном моћи стандардних статистичких тестова, уз праг значајности  $\alpha = 0.05$ . Приказани су интервали поверења од 99.9%.



Слика 5.5: Моћ усмерених мрежних тестова у поређењу с максималном моћи стандардних статистичких тестова, уз праг значајности  $\alpha = 0.1$ . Приказани су интервали поверења од 99.9%.

### 5.1.5 Дискусија

Резултати показују следеће. У случају платикуртичних и мезокуртичних алтернативних расподела, усмерени мрежни тестови имају већу моћ од стандардних статистичких тестова нормалности. За искошене лептокуртичне расподеле, усмерени мрежни тестови имају исту моћ као најмоћнији статистички тест при прагу значајности  $\alpha = 0.01$ , док је за  $\alpha = 0.05$  и  $\alpha = 0.1$  моћ мрежних тестова већа. Што се тиче симетричних лептокуртичних расподела, за  $\alpha = 0.01$  је максимална моћ стандардних тестова мало изнад моћи теста  $NT_{SL}$ , док су  $NT_{SL}$  и најмоћнији стандардни тест изједначени за  $\alpha \in \{0.05, 0.1\}$ .

На основу наведеног можемо извести следеће препоруке за мале узорке ( $n \leq 100$ ):

- 1) Ако је алтернативна расподела мезокуртична или платикуртична, треба користити одговарајући мрежни тест.
- 2) Ако је алтернативна расподела лептокуртична:
  - 2.1) Ако је симетрична:
    - 2.1.1) За  $\alpha = 0.01$  треба користити DP ако је  $n \leq 30$ , а JB за веће узорке.
    - 2.1.2) Ако је  $\alpha = 0.05$  или  $\alpha = 0.1$ , најбољи тестови су  $NT_{SL}$  и  $KT_{>3}$ .
  - 2.2) Ако је алтернативна расподела искошена, мрежни тестови имају исту моћ као најмоћнији стандардни тест за дати тип расподела.

## 5.2 Омнибус мрежни тестови

### 5.2.1 Конфигурација метапараметара

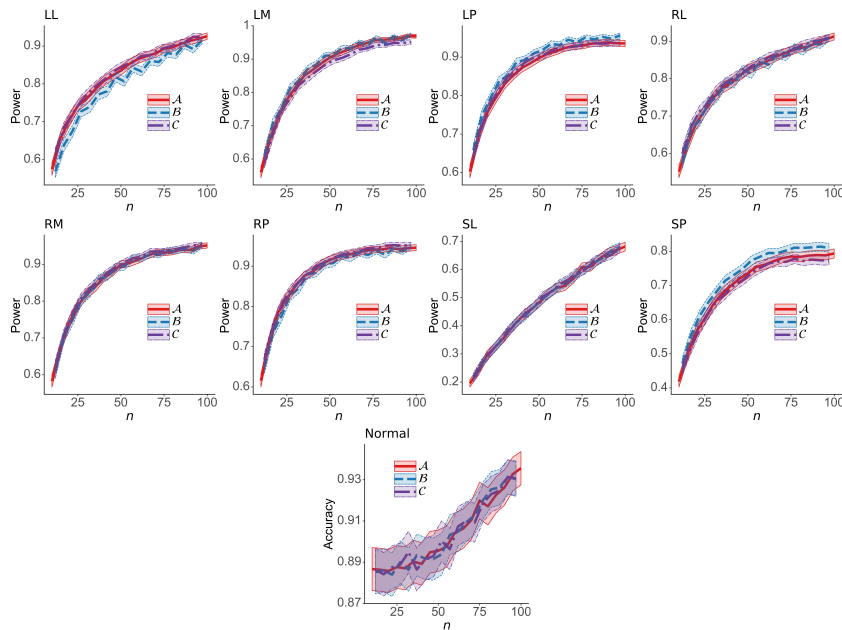
Најбоља конфигурација метапараметара добијена помоћу GA представљена је у табели 5.2. Као и усмерене мреже, и омнибус мрежа нема много унутрашњих слојева (само четири), али су јој ти слојеви знатно шири (170 неурона). Репрезентациони простор укључује  $p$ -вредности стандардних тестова и бројне дескриптивне постстандардизационе статистике. Адекватна репрезентација омогућила је коришћење релативно малог скупа за тренирање (1760 узорака).

Табела 5.2: Конфигурација метапараметара за развој омнибус мреже

Подаци	$M = 11, L = 1, \beta_{2,\max} = 47.42, PD = \top$
Репрезентација	$R = s_2, PV = \{KT_{>3}, ST_{<0}, ST_{>0}, ZC, SW, SF, DP\}, F = \{n, \beta_2(z), \bar{z}, \text{median}(z), \text{sd}(z), \text{mad}(z)\}, q = 0.05, m_L = 0.2, m_R = 0$
Архитектура	$d = 4, w = 170, a = \text{ReLU}$
Тренирање	$A = \text{SGD}, c = 0.00318, ES = 0, LRI = 0.0089, LRU = \text{adaptive}, m = 0.0246, NM = \perp, I = 10, t = 0.4831$

### 5.2.2 Стабилност

Стабилност омнибус мреже испитана је на исти начин као стабилност усмерених мрежа. На слици 5.6 приказано је поклапање крива тачности на скуповима различитих обима узорака, варијабилности и очекивања, што значи да добијена мрежа има високу стабилност и добру способност генерализације. Тачност је у свим случајевима неопадajuћа по обиму  $n$ .



Слика 5.6: Тачност омнибус мреже на не-нормалним узорцима из скупова  $A, B$  и  $C$ , добијених из дистрибуција различите варијабилности и других параметара. Приказани су интервали поверења од 99.9%.

### 5.2.3 Групе расподела за поређење

Моћ мрежног теста испитана је на седам група расподела које се најчешће користе у литератури за евалуацију омнибус тестова нормалности. Коришћене групе расподела представљене су у табели 5.3. Прву групу чине лептокуртичне и симетричне расподеле неограниченог домена. У другој групи су искошене, неограничене и лептокуртичне расподеле, док се трећа група од друге разликује само по ограничености домена. Четврта група обухвата ограничене платикуртичне расподеле, а пету, шесту и седму групу чине контаминирани нормалне расподеле. Нека је  $(1 - \rho_c)\mathcal{N}(0, 1) + \rho_c\mathcal{N}(\mu_c, \sigma_c^2)$  контаминирана нормална расподела, где је  $\mathcal{N}(0, 1)$  основна, а  $\mathcal{N}(\mu_c, \sigma_c^2)$  контаминирајућа компонента. При томе, за параметре  $\mu_c$  и  $\sigma_c$  важи да нису исти као код основне компоненте (тј. важи услов  $\mu_c \neq 0 \vee \sigma_c \neq 1$ ). Параметар  $\rho_c \in (0, 1)$  представља степен контаминације, односно вероватноћу да насумично генерисан број из комбиноване расподеле потиче из контаминирајуће компоненте  $\mathcal{N}(\mu_c, \sigma_c^2)$ . За  $\mu_c = 0 \wedge \sigma_c \neq 1$ , контаминација је варијационог типа, пошто основна и контаминирајућа компонента деле исто очекивање, а разликују се само по варијанси. За  $\mu_c \neq 0 \wedge \sigma_c = 1$  кажемо да је контаминација локацијског типа пошто компоненте имају исту варијансу. Контаминација је комбинованог типа ако је истовремено и локацијска и варијациона ( $\mu_c \neq 0 \wedge \sigma_c \neq 1$ ).

Табела 5.3: Групе расподела за евалуацију.

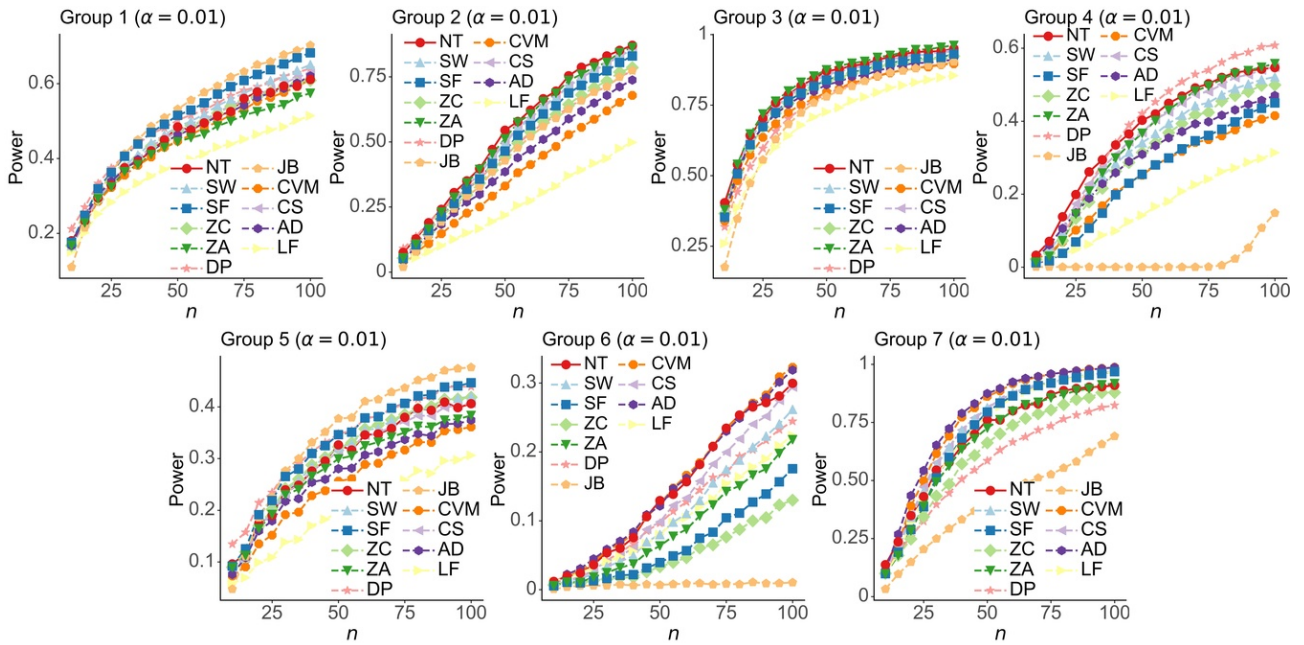
Група	Опис	Расподеле
1	Симетричне, неограничене, лептокуртичне	Стандардна логистичка, стандардна Лапласова, Студентова $t(1)$ и $t(3)$
2	Искошене, неограничене, лептокуртичне	Гумбелове $(0, 1)$ , $(0, 2)$ и $(0, 0.5)$
3	Искошене, ограничене, лептокуртичне	Експоненцијална $(0, 1)$ , Гама $(2, 1)$ и $(0.5, 1)$ , логнормалне $(0, 1)$ , $(0, 2)$ и $(0, 0.5)$ , Вајбулове $(0.5, 1)$ и $(2, 1)$
4	Платикуртичне, ограничене	Униформна $(0, 1)$ , Бета $(2, 2)$ , $(0.5, 0.5)$ и $(4, 4)$
5	Варијацијски контаминирана $\mathcal{N}(0, 1)$	$(\rho_c, \sigma_c) \in \{0.1, 0.2\} \times \{0.25, 2, 4\}$ , $\mu_c = 0$
6	Локацијски контаминирана $\mathcal{N}(0, 1)$	$(\rho_c, \mu_c) \in \{0.3, 0.4, 0.5\} \times \{1, 3, 5\}$ , $\sigma_c = 1$
7	Комбиновано контаминирана $\mathcal{N}(0, 1)$	$(\rho_c, \sigma_c, \mu_c) \in \{0.3, 0.4, 0.5\} \times \{0.25, 4\} \times \{1, 3\}$

Из сваке групе је за сваки обим  $n = 10, 15, \dots, 95, 100$  генерисано по 10 000 узорака обима  $n$  из те групе. Расподеле у оквиру исте групе смењиване су циклично приликом генерисања узорака како би се обезбедила њихова равномерна заступљеност.

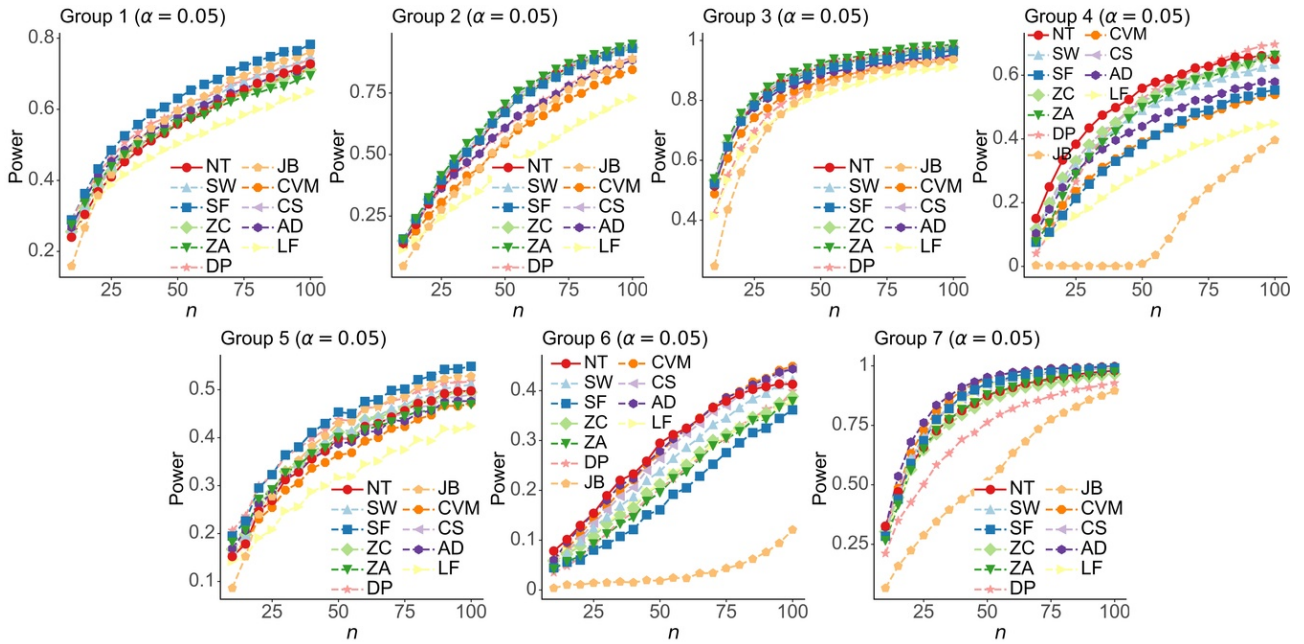
### 5.2.4 Поређење са стандардним статистичким тестовима

Мрежни омнибус тест NT изведен је на исти начин као и усмерени тестови из усмерених мрежа.

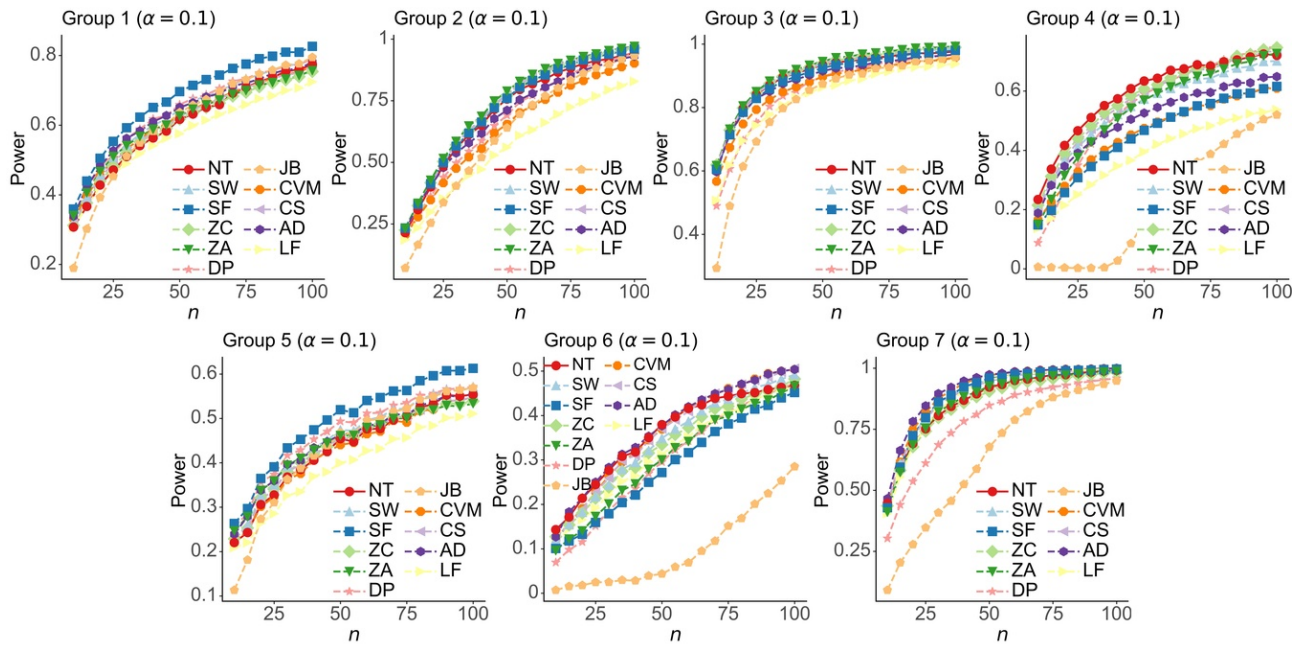
Криве моћи теста NT и најмоћнијих стандардних омнибус тестова за  $\alpha = 0.01, 0.05, 0.1$  приказане су на сликама 5.7–5.9, а анализа рангова представљена је у табели 5.4. Тестови су ранжирани за сваку комбинацију групе, обима  $n$  и прага  $\alpha = 0.01, 0.05, 0.1$ , а потом је израчунат њихов просечан ранг за свако  $\alpha$ , као и за сваку комбинацију групе и прага  $\alpha$ .



Слика 5.7: Моћ омнибус тестова за праг  $\alpha = 0.01$



Слика 5.8: Моћ омнибус тестова за праг  $\alpha = 0.05$



Слика 5.9: Моћ омнибус тестова за праг  $\alpha = 0.1$

Табела 5.4: Просечни рангови мрежног омнибус теста NT и стандардних омнибус тестова. Најбољи рангови су подебљани. Вредности у блоку 1–7 представљају просек свих рангова за исти праг  $\alpha$ , док су вредности у редовима изнад просечни рангови за дату групу и дати праг  $\alpha$ .

Група	$\alpha$	AD	CS	CVM	DP	JB	LF	NT	SF	SW	ZA	ZC
1	0.01	6.84	5.79	9.13	2.68	2.24	10.95	6.32	<b>2.13</b>	4.34	9.24	6.34
	0.05	4.61	6.63	7.95	2.63	4.21	10.84	8.11	<b>1.00</b>	4.13	8.21	7.68
	0.1	3.79	7.47	6.32	2.53	5.95	10.74	8.45	<b>1.00</b>	3.89	6.79	9.08
2	0.01	8.84	3.55	9.92	5.47	8.13	10.95	<b>1.16</b>	5.32	4.18	2.21	6.26
	0.05	7.58	2.79	9.68	7.16	9.32	10.89	5.11	5.11	2.95	<b>1.11</b>	4.32
	0.1	7.11	3.03	9.42	7.68	9.68	10.79	6.16	4.58	2.66	<b>1.05</b>	3.84
3	0.01	6.74	3.05	8.47	9.05	9.63	10.84	1.92	5.21	3.95	<b>1.08</b>	6.05
	0.05	6.95	2.08	8.55	8.47	10.24	10.74	4.53	5.89	3.76	<b>1.00</b>	3.79
	0.1	7.00	2.97	8.84	8.42	10.16	10.53	5.21	5.84	3.95	<b>1.05</b>	2.03
4	0.01	6.11	3.26	8.26	2.37	11.00	9.79	<b>1.89</b>	8.53	4.63	4.00	6.16
	0.05	6.58	2.74	8.16	3.37	11.00	9.89	<b>1.47</b>	8.53	5.37	5.37	3.53
	0.1	6.47	3.16	8.05	4.16	11.00	9.95	<b>1.74</b>	8.63	5.21	5.47	2.16
5	0.01	8.89	6.53	9.95	2.21	<b>2.00</b>	10.95	5.79	2.47	5.37	7.63	4.21
	0.05	7.74	6.82	9.68	2.26	4.00	10.89	7.18	<b>1.16</b>	4.21	6.79	5.26
	0.1	4.53	7.11	9.16	2.00	6.11	10.89	7.89	<b>1.00</b>	3.68	6.68	6.95
6	0.01	2.00	4.58	<b>1.79</b>	5.11	11.00	5.95	2.53	9.11	6.05	8.05	9.84
	0.05	<b>2.05</b>	3.84	2.11	8.00	11.00	6.53	2.16	9.84	5.11	8.68	6.68
	0.1	<b>1.42</b>	3.11	2.84	8.58	11.00	7.26	3.21	9.74	4.79	8.37	5.68
7	0.01	<b>1.26</b>	4.13	2.00	9.53	11.00	5.87	6.26	5.68	3.42	7.89	8.95
	0.05	<b>1.34</b>	5.00	1.82	10.00	11.00	5.39	6.79	4.32	4.03	7.79	8.53
	0.1	<b>1.34</b>	5.34	1.82	10.00	11.00	5.05	7.11	3.66	4.42	7.53	8.74
1–7	0.01	5.81	4.41	7.08	5.20	7.86	9.33	<b>3.70</b>	5.49	4.56	5.73	6.83
	0.05	5.26	4.27	6.85	5.98	8.68	9.31	5.05	5.12	<b>4.22</b>	5.56	5.68
	0.1	4.52	4.60	6.64	6.20	9.27	9.32	5.68	4.92	<b>4.09</b>	5.28	5.50

### 5.2.5 Дискусија

Резултати евалуације омнибус тестова у складу су са постојећом литературом. Ниједан тест не може се издвојити као најмоћнији на свим групама алтернативних расподела, али се могу дефинисати препоруке када треба применити који тест.

Мрежни омнибус тест НТ има највећу моћ у следећим случајевима:

- (а) Уколико је расподела симетрична и платикуртична.
- (б) Уколико узорак потиче из искошене, лептокуртичне и неограничене расподеле, а праг статистичке значајности је  $\alpha = 0.01$ .

Такође, мрежни тест је други најбољи избор за искошене, лептокуртичне и ограничене расподеле при прагу значајности  $\alpha = 0.01$ , а добре перформансе има и на класи локацијски контаминираних нормалних расподела, нарочито за узорке са мање од 70 елемената.

На симетричним расподелама најбоље се показао тест SF. На искошеним лептокуртичним расподелама највећу моћ има тест ZA, сем за случај неограничених расподела при прагу  $\alpha = 0.01$ , где најбољи ранг има тест NT.

За варијацијски контаминираних нормалних расподела, најбоље перформансе је на прагу значајности  $\alpha = 0.01$  дао тест JB, док за прагове  $\alpha = 0.05, 0.1$  најбоље резултате има тест SF. За случај локацијски контаминираних нормалних расподела, највећу моћ при прагу  $\alpha = 0.01$  има тест SVM. За све друге комбинације прага и типа контаминације, најбољи је тест AD.

Најбољи просечан ранг за праг  $\alpha = 0.01$ , посматрано на свим групама расподела, има мрежни тест NT. Иако ни на једној групи нема најбољи просечни ранг, тест SW има најбољи свеукупан просечни ранг за прагове  $\alpha = 0.05, 0.1$ , што је у складу са ранијим истраживањима и препорукама да се овај тест користи због високе моћи на многим класама не-нормалних расподела [13, 14]. Тест CS има нешто лошији свеукупни просечни ранг од теста SW за  $\alpha = 0.05, 0.1$ , али му је моћ приближно иста. За праг значајности  $\alpha = 0.01$ , тест CS је у просеку моћнији од теста SW.

## Глава 6

# Омнибус тест нормалности заснован на дугорочној рекурентној мрежи типа LSTM

Генетски алгоритам предложен у глави 3 конструише и тренира класичне једносмерне мреже. Овај тип архитектура имају све неуронске мреже које су до сада развијене за испитивање нормалности. Примена таквих мрежа тражи да се узорци прво пресликају у коначнодимензионалне векторе унапред дефинисаног репрезентационог простора.

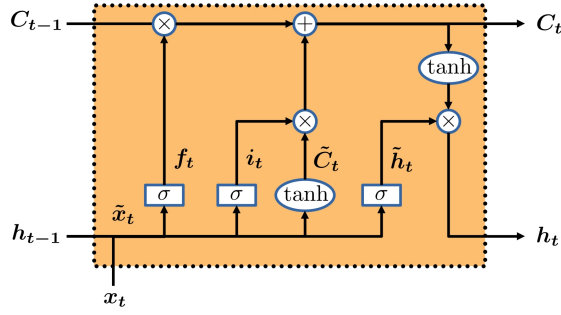
Генетски алгоритам може да оптимизује све метапараметре, укључујући и оне који одређују репрезентациони простор, али тиме није у потпуности решен проблем репрезентације узорака. Наиме, сви предстандардизациони и постстандардизациони атрибути, који су у глави 3 наведени као могуће димензије репрезентационог простора, представљају посредне информације о задатом узорку. То значи да једносмерне мреже уче класификациона правила из информација о узорцима уместо директно из њих.

У овој глави је представљено како се директно учење може реализовати применом рекурентних неуронских мрежа са дугорочном меморијом, чија је архитектура у стању да обради улазне податке различите кардиналности.

### 6.1 Мреже типа LSTM

Дугорочне рекурентне мреже (енгл. *long short-term memory*, LSTM) [106] имају примену у предикцији и класификацији временских серија. Ове мреже на улазу прихватају низове произвољне дужине, а у стању су да детектују и за предикцију искористе релације између удаљених елемената, а не само суседних. Пример таквих података су текстови написани на природним језицима, где једна реч или синтагма може у потпуности одредити значење друге речи од које је деле читаве реченице или знатан број других лексема. Мреже типа LSTM процесирају такве секвенце елемент по елемент помоћу слоја посебне структуре који релације између улазних елемената памти у дугорочној меморији. Меморија је реализована као скривено стање, односно вектор који називамо вектор контекста и чији се садржај мења током секвенцијалне обраде улазних података у складу са наученим дугорочним и краткорочним релацијама. Структура таквог слоја, који зовемо слој типа LSTM, представљена је на слици 6.1. Он садржи три унутрашња једносмерна слоја са по  $\ell$  неурона, од којих сваки има сопствену тежинску матрицу реда  $(\ell, \ell + d + 1)$ .

Нека је  $[x_t]_{t=1}^n$  улазни низ, при чему  $d$ -димензионални вектор  $x_t = [x_i^{(t)}]_{i=1}^d$  представља улаз у кораку  $t = 1, 2, \dots$  ( $d \geq 1$ ). Нека  $\ell$ -димензионални вектор  $h_t = [h_i^{(t)}]_{i=1}^\ell$  означава



**Слика 6.1:** Слој типа LSTM. Он секвенцијалне податке обрађује елемент по елемент, односно, корак по корак:  $x_t$  је улаз у кораку  $t$ ,  $h_t$  је излаз слоја у кораку  $t$ ,  $\tilde{x}_t$  је вектор добијен надовезивањем излаза  $h_{t-1}$  и улаза  $x_t$ , а  $C_t$  је вектор контекста у кораку  $t$ . Правоугаоници означавају класичне једносмерне слојеве са сигмоидалном активацијом ( $\sigma$ ), а кругови и елипсе представљају Адамарове операције.

излаз слоја LSTM након процесирања елемента  $x_t$ , и нека је

$$\tilde{x}_t = [h_1^{(t-1)}, \dots, h_\ell^{(t-1)}, x_1^{(t)}, \dots, x_d^{(t)}, 1]^T$$

вектор који добијамо надовезивањем вектора  $x_t$  и  $h_{t-1}$  и којем зарад компактности нотације додајемо константу 1 као последњу димензију. Нека је  $C_t$  ознака за  $\ell$ -димензионални вектор контекста у кораку  $t$ .

Вектор  $C_t$  одређује се на следећи начин. Прво се вектор контекста из претходног корака,  $C_{t-1}$ , пропушта кроз тзв. маску заборављања  $f_t \in [0, 1]^\ell$  (енг. *forget gate*) тако што се сваки елемент вектора  $C_{t-1}$  множи одговарајућим елементом маске. Маска  $f_t$  добија се као излаз класичног једносмерног слоја са тежинском матрицом  $W_f \in \mathbb{R}^{(\ell, \ell+d+1)}$  и сигмоидалном активационом функцијом ( $\sigma$ ) за улаз  $\tilde{x}_t$ , тј.  $f_t = \sigma(W_f \tilde{x}_t)$ . Тако се одређује који део контекста из претходног корака треба заборавити. Које информације из тренутног улаза треба запамтити одређује маска меморизације,  $i_t = \sigma(W_i \tilde{x}_t) \in [0, 1]^\ell$ , која се рачуна слично као маска заборављања, пропуштањем проширеног улазног вектора кроз сигмоидални једносмерни слој тежинске матрице  $W_i$ . Множењем улазног контекста  $\tilde{C}_t = \tanh(\tilde{x}_t)$  маском  $i_t = \sigma(W_i \tilde{x}_t) \in [0, 1]^\ell$ , добијамо вектор који ће у датом кораку  $t$  заменити информације заборављене из контекста претходног корака. Та замена се имплементира као сабирање:

$$C_t = (f_t \otimes C_{t-1}) \oplus (i_t \otimes \tilde{C}_t). \quad (6.1)$$

па елементе вектора контекста  $C_t$  у кораку  $t$  можемо сматрати линеарном комбинацијом одговарајућих елемената вектора  $C_{t-1}$  и  $\tilde{C}_t$ .

Вектор контекста  $C_t$  директно одређује излаз  $h_t$  на следећи начин:

$$\tilde{h}_t = \sigma(W_h \tilde{x}_t) \quad h_t = \tilde{h}_t \otimes \tanh(C_t). \quad (6.2)$$

У наредном кораку, излаз  $h_t$  комбинује се са улазом  $\tilde{x}_{t+1}$  и цео поступак се понавља док се не обради цео улазни низ  $[x_t]_{t=1}^n$ . Уместо  $\tanh$  може се користити и друга функција за израчунавање улазног контекста и излазног вектора.

Излаз  $h_t$  у регресионом проблемима користимо као предикцију за неки наредни елемент  $x_{t+r}$  ( $r \geq 1$ ) улазног низа. У бинарној класификацији, последњи излаз  $h_n$  пропуштамо кроз сигмоидални неурон да бисмо добили излазну вредност између 0 и 1, коју потом користимо као процену вероватноће да читава секвенца  $[x_t]_{t=1}^n$  припада позитивној класи. Између коначног сигмоидалног неурона и почетног слоја LSTM могу се налазити додатни слојеви типа LSTM, као и класични једносмерни слојеви.

## 6.2 Трансформација узорака

Нека је  $x = \{x_i\}_{i=1}^n$  узорак. Сортирање је природан начин да се он преведе у низ. Нека је  $[x_{(i)}]_{i=1}^n$  неоппадајућа пермутација узорка  $x$  ( $x_{(i)} \leq x_{(i+1)}$  за  $i = 1, 2, \dots, n-1$ ).

Како узорци могу потицати из расподела различитих очекивања и варијансе, опсег могућих вредности елемената  $x_{(i)}$  је цео скуп  $\mathbb{R}$ . При томе, неопходно је обезбедити нумеричку стабилност током тренирања и осигурати да мрежа има исте перформансе и на узорцима из расподела знатно другачијих очекивања и варијансе од оних искоришћених за тренирање. Стога је потребно трансформисати сортирани узорак  $[x_{(i)}]_{i=1}^n$  тако да се првобитна узорачка средина и варијанса не могу израчунати након трансформације, при чему је потребно да она очува остатак структуре датог узорка. У ту сврху се могу користити следеће трансформације:

$$s_1 : x_{(i)} \mapsto (x_{(i)} - \bar{x})/\text{sd}(x) \quad (6.3)$$

$$s_2 : x_{(i)} \mapsto (x_{(i)} - \text{med}(x))/\text{mad}(x) \quad (6.4)$$

$$s_3 : x_{(i)} \mapsto -3 + 6 \cdot (x_{(i)} - \min(x))/(\max(x) - \min(x)). \quad (6.5)$$

Трансформација  $s_3$  има кодомен  $[-3, 3]$ , који садржи 99.97% стандардне нормалне расподеле. Она открива релативну удаљеност елемената од минимума и максимума трансформисаног узорка. Са друге стране, трансформације  $s_1$  и  $s_2$  показују релативну удаљеност до централне вредности узорка (аритметичке средине или медијане), изражену у оценама варијабилности. Трансформација  $s_2$  је мање осетљива на екстремне вредности него трансформација  $s_1$  будући да користи медијану и медијалну апсолутну девијацију.

Ако применимо све три трансформације истовремено, добијамо тродимензионални низ  $[z_{(i)}]_{i=1}^{(n)}$  чији је општи члан:

$$z_{(i)} = [s_1(x_{(i)}) \quad s_2(x_{(i)}) \quad s_3(x_{(i)})]^T \quad i = 1, 2, \dots, n \quad (6.6)$$

а можемо применити и ма који непразан подскуп трансформација  $\{s_1, s_2, s_3\}$ .

## 6.3 Метапараметри

При примени другог типа мреже и другачијег претпроцесирања узорака, метапараметри генерисања података могу остати исти, али је неопходно променити метапараметре архитектуре и репрезентације.

Нови метапараметри репрезентације су три логичке променљиве  $S_1, S_2, S_3 \in \{\perp, \top\}$  које показују које од трансформација  $s_1, s_2$  и  $s_3$  користимо у припреми узорака за мрежу. Барем једна трансформација мора бити примењена, што значи да за ове променљиве важи услов  $S_1 \vee S_2 \vee S_3$ .

Што се тиче архитектуре, мреже смо ограничили на највише три класична једносмерна слоја после слоја типа LSTM. Последња компонента је неурон са сигмоидалном активацијом. Оваква фамилија мрежа дефинисана је следећим метапараметрима:

- $\ell \in [1, 128]$  – број неурона у унутрашњим слојевима слоја LSTM.
- $d \in \{0, 1, 2, 3\}$  – број скривених једносмерних слојева.
- $w \in \{32, 64, 128\}$  – број неурона у скривеним једносмерним слојевима.
- $a \in \{\tanh, \sigma\}$  – активациона функција у скривеним слојевима, као и функција за одређивање улазног контекста и излазног вектора у слоју LSTM.

Како су се за развој једносмерних мрежа алгоритми ADAM и SGD показали боље од алгоритма LBFGS, он није разматран у оптимизацији метапараметара у овом експерименту. Оптимизовани су следећи метапараметри тренирања.

- $A$  – индикатор алгоритма (ADAM или SGD).
- $l$  – функција губитка (логистичка или функција фокалног губитка [107]).
- $c \in [10^{-4}, 10^{-2}]$  – регуларизациона константа.
- $r \in \{l_1, l_2\}$  – индикатор типа регуларизације ( $l_1$  или  $l_2$ ).
- $b \in \{32, 64, 128\}$  – величина групе у једној итерацији (епохи) алгоритма за тренирање (енгл. *batch size*).
- $LR \in [10^{-4}, 10^{-1}]$  – стопа учења.
- $CN \in (0.1, 5)$  – граница норме градијената.
- $ES \in \{10, 20\}$  – број итерација након којих се тренирање зауставља ако се перформансе не побољшају.
- Метапараметри својствени алгоритмима:
  - ★ За алгоритам ADAM:  $b_1, b_2 \in (0, 1)$ , стопе слабљења момената.
  - ★ За алгоритам SGD:  $m \in (0, 1)$  – вредност момента и  $NM \in \{\perp, \top\}$  – индикатор да ли се користи Нестеровљев момент.

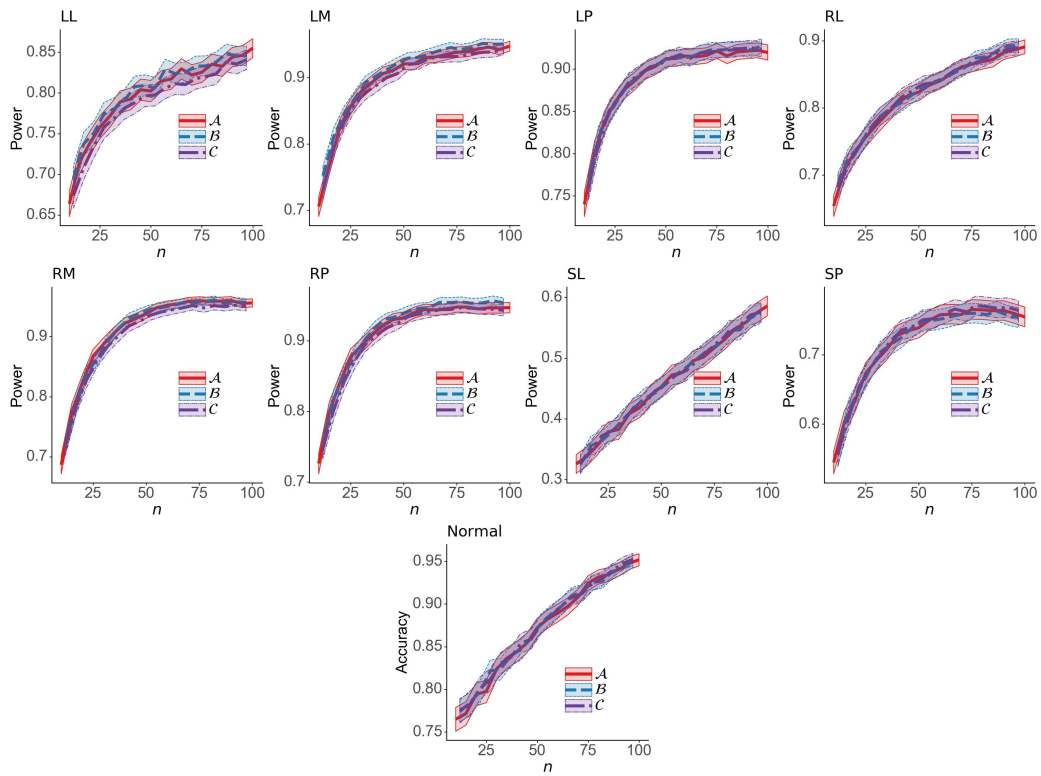
Тренирање се реализује у највише 200 епоха, при чему се 20% скупа за тренирање користи за валидацију. Примењена су иста подешавања GA као за једносмерне мреже, сем тежине која је постављена на вредност  $\omega = 1.33$ . Добијена конфигурација представљена је у табели 6.1. Она користи све три трансформације, што се може објаснити њиховом комплементарношћу. Пошто не користи  $p$ -вредности, мрежи је био потребан већи скуп за тренирање.

**Табела 6.1:** Најбоља пронађена подешавања метапараметара за развој мрежа типа LSTM за испитивање нормалности

Вредности	Метапараметри
Подаци	$M = 35, L = 7, \beta_{2,\max} = 16.63, PD = \perp$
Репрезентација	$s_1 = s_2 = s_3 = \top$
Архитектура	$d = 1, w = 16, a = \tanh, \ell = 10$
Тренирање	$A = \text{SGD}, l = \text{логистичкигубитак}, c = 0.00246,$ $CN = 1.449, LR = 0.0089, m = 0.79776, NM = \top,$ $b = 128, r = l_2, ES = 10$

## 6.4 Стабилност

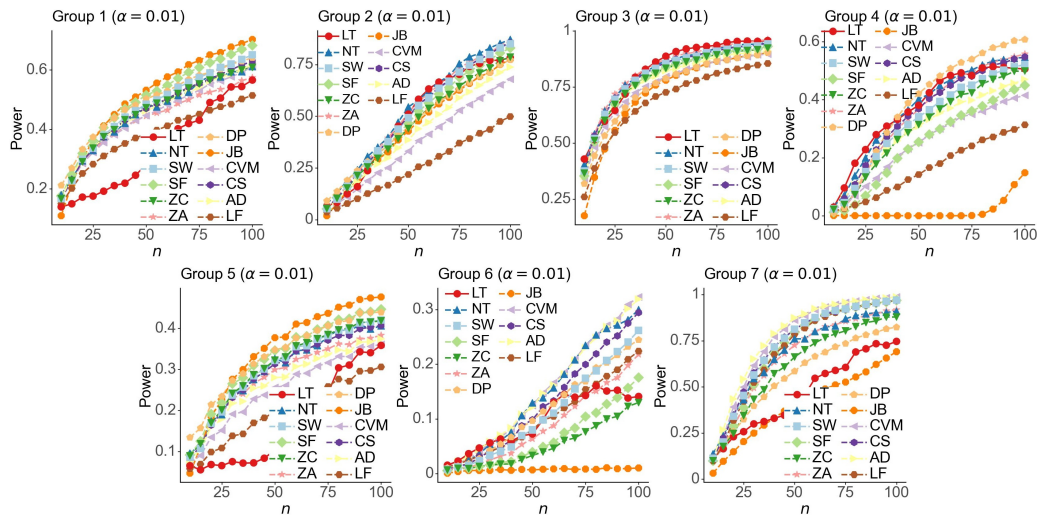
Стабилност је испитана на исти начин као стабилност једносмерних мрежа. Криве тачности на скуповима различитих стандардних девијација представљене су на слици 6.2. Криве се поклапају за сваки тип расподела, а ниједна није опадајуће монотоности у односу на обим узорака.



Слика 6.2: Стабилност мрежа типа на скуповима различитих обима узорака, добијених из расподела различитих очекивања и варијанси. Приказани су интервали поверења од 99.9%.

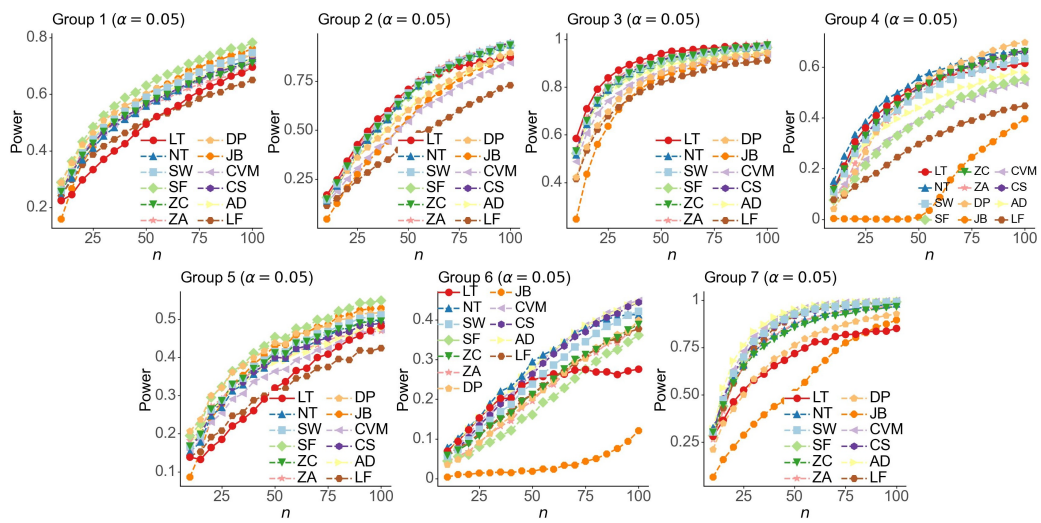
## 6.5 Евалуација

Моћ омнибус теста изведеног из мреже LSTM, у ознаци LT, процењена је на истим групама расподела и на исти начин као и моћ теста NT изведеног из једносмерне омнибус мреже. Резултати су представљени на сликама 6.3–6.5 и у табели 6.2. Тест LT показује високу моћ на расподелама ограничених домена (групе 3 и 4), а компетитиван је са већином стандардних тестова на групи искошених лептокуртичних расподела са неограниченим доменима (група 2).

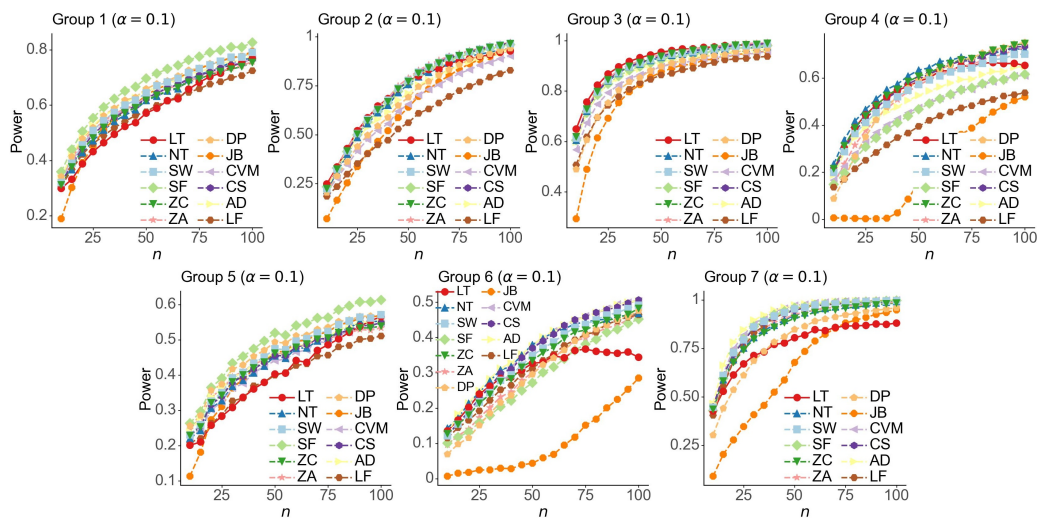


Слика 6.3: Моћ омнибус тестова за праг  $\alpha = 0.01$

Међутим, ако се фокусирамо на изузетно мале узорке ( $n \leq 30$ ), видећемо да тест LT постиже знатно боље резултате, што показује анализа рангова у табели 6.3.



Слика 6.4: Моћ омнибус тестова за праг  $\alpha = 0.05$



Слика 6.5: Моћ омнибус тестова за праг  $\alpha = 0.1$

**Табела 6.2:** Просечни рангови омнибус тестова за  $n = 10, 15, \dots, 95, 100$ . Блокови 1–7, 1–4 и 5–7 показују просечне рангове за групе обухваћене тим блоком. Најбоље вредности су подебљане.

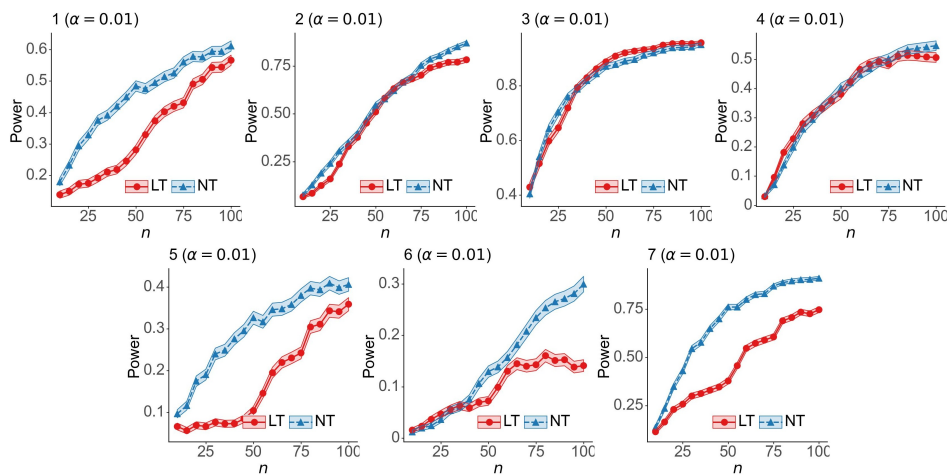
Група	$\alpha$	AD	CS	CVM	DP	JB	LF	LT	NT	SF	SW	ZA	ZC
1	0.01	6.84	5.79	9.13	2.68	2.29	11.21	11.68	6.32	<b>2.13</b>	4.34	9.24	6.34
	0.05	4.61	6.63	7.95	2.63	4.26	11.32	11.42	8.11	<b>1.00</b>	4.13	8.26	7.68
	0.1	3.79	7.53	6.37	2.53	6.05	11.16	11.05	8.45	<b>1.00</b>	3.89	6.95	9.24
2	0.01	9.68	3.92	10.92	6.21	8.97	11.95	5.16	<b>1.26</b>	5.95	4.55	2.42	7.00
	0.05	8.53	3.37	10.68	8.05	10.26	11.89	3.74	5.68	5.79	3.53	<b>1.58</b>	4.89
	0.1	8.05	3.45	10.42	8.47	10.63	11.79	4.74	6.74	5.05	3.03	<b>1.32</b>	4.32
3	0.01	7.58	3.84	9.47	10.05	10.63	11.84	2.53	2.71	6.05	4.74	<b>1.66</b>	6.89
	0.05	7.95	2.97	9.55	9.47	11.24	11.74	<b>1.47</b>	5.53	6.89	4.71	1.74	4.74
	0.1	8.00	3.87	9.84	9.42	11.16	11.53	1.74	6.21	6.84	4.84	<b>1.68</b>	2.87
4	0.01	7.11	4.05	9.26	2.68	12.00	10.79	2.84	<b>2.37</b>	9.53	5.53	4.68	7.16
	0.05	7.58	3.21	9.16	3.79	12.00	10.89	4.26	<b>1.47</b>	9.53	6.05	5.95	4.11
	0.1	7.47	3.53	9.05	4.58	12.00	10.95	4.63	<b>1.74</b>	9.63	6.00	6.11	2.32
5	0.01	8.89	6.53	9.95	2.21	<b>2.05</b>	11.26	11.63	5.79	2.47	5.37	7.63	4.21
	0.05	7.79	6.82	9.79	2.26	4.05	11.47	11.11	7.18	<b>1.16</b>	4.21	6.89	5.26
	0.1	4.63	7.21	9.32	2.00	6.21	11.47	10.47	8.00	<b>1.00</b>	3.68	6.89	7.11
6	0.01	2.21	4.89	<b>2.00</b>	5.53	12.00	6.58	5.68	2.84	10.00	6.58	8.84	10.84
	0.05	<b>2.11</b>	4.16	2.42	8.58	12.00	7.11	6.84	2.16	10.53	5.58	9.26	7.26
	0.1	<b>1.47</b>	3.37	3.05	9.16	12.00	7.74	7.53	3.21	10.37	5.11	8.95	6.05
7	0.01	<b>1.32</b>	4.18	2.05	9.58	11.84	5.92	10.68	6.26	5.74	3.47	7.95	9.00
	0.05	<b>1.34</b>	5.00	1.82	10.21	11.84	5.45	10.84	6.79	4.32	4.03	7.84	8.53
	0.1	<b>1.34</b>	5.34	1.82	10.26	11.68	5.11	10.95	7.11	3.66	4.42	7.58	8.74
1–7	0.01	6.23	4.74	7.54	5.56	8.54	9.94	7.17	<b>3.94</b>	5.98	4.94	6.06	7.35
	0.05	5.70	<b>4.59</b>	7.34	6.43	9.38	9.98	7.10	5.27	5.60	4.61	5.93	6.07
	0.1	4.97	4.90	7.12	6.63	9.96	9.96	7.30	5.92	5.36	<b>4.42</b>	5.64	5.80
1–4	0.01	7.80	4.40	9.70	5.41	8.47	11.45	5.55	<b>3.16</b>	5.91	4.79	4.50	6.85
	0.05	7.16	<b>4.05</b>	9.34	5.99	9.44	11.46	5.22	5.20	5.80	4.61	4.38	5.36
	0.1	6.83	4.59	8.92	6.25	9.96	11.36	5.54	5.78	5.63	4.44	<b>4.01</b>	4.68
5–7	0.01	<b>4.14</b>	5.20	4.67	5.77	8.63	7.92	9.33	4.96	6.07	5.14	8.14	8.02
	0.05	<b>3.75</b>	5.32	4.68	7.02	9.30	8.01	9.60	5.38	5.33	4.61	8.00	7.02
	0.1	<b>2.48</b>	5.31	4.73	7.14	9.96	8.11	9.65	6.11	5.01	4.40	7.81	7.30

**Табела 6.3:** Просечни рангови тестова за  $n = 10, 15, \dots, 30$ . Блокови 1–7, 1–4 и 5–7 показују просечне рангове за групе обухваћене тим блоком. Најбоље вредности су подебљане.

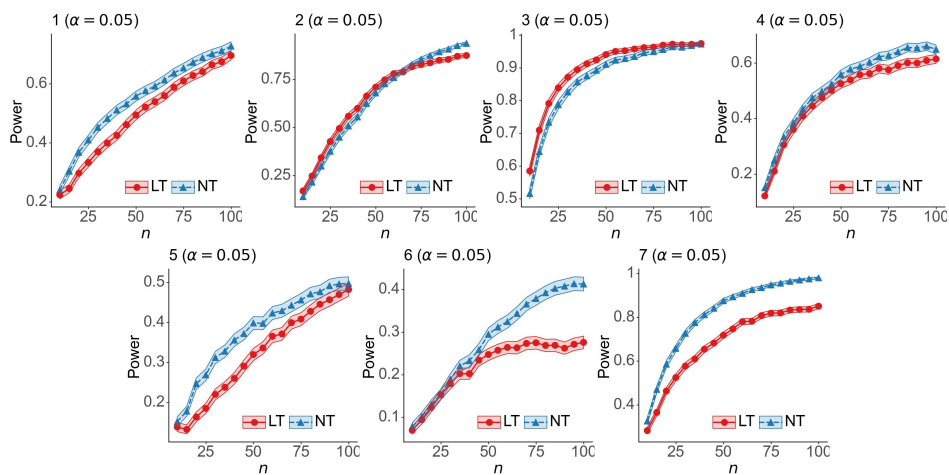
$G$	$\alpha$	AD	CS	CVM	DP	JB	LF	LT	NT	SF	SW	ZA	ZC
1	0.01	5.00	6.80	9.30	<b>1.20</b>	5.90	10.80	11.80	5.20	2.50	5.90	7.50	6.10
	0.05	3.20	6.40	7.20	2.00	9.40	10.40	11.80	9.60	<b>1.00</b>	4.80	4.40	7.80
	0.1	2.40	7.00	5.60	2.80	11.00	10.00	11.60	9.40	<b>1.00</b>	5.20	4.00	8.00
2	0.01	8.80	5.00	10.70	<b>1.60</b>	9.50	11.80	8.20	<b>1.60</b>	6.20	5.20	3.00	6.40
	0.05	8.00	5.00	10.00	7.60	11.40	11.60	<b>1.00</b>	7.60	4.20	4.80	2.40	4.40
	0.1	6.80	5.60	9.80	9.20	11.80	11.20	<b>1.00</b>	7.60	4.40	4.00	2.20	4.40
3	0.01	6.40	3.40	9.00	10.00	11.60	11.40	6.00	1.90	6.20	4.00	<b>1.50</b>	6.60
	0.05	7.80	3.30	9.00	10.00	12.00	11.00	<b>1.00</b>	6.60	6.60	4.50	2.00	4.20
	0.1	7.80	4.20	9.00	10.40	12.00	10.60	<b>1.00</b>	6.20	7.00	4.80	2.20	2.80
4	0.01	5.00	3.40	8.00	7.00	12.00	10.20	<b>1.20</b>	1.80	10.20	4.60	7.40	7.20
	0.05	6.40	3.60	8.00	8.80	12.00	10.60	2.00	<b>1.00</b>	9.80	5.00	7.40	3.40
	0.1	6.00	4.00	8.20	9.20	12.00	10.80	2.40	<b>1.00</b>	9.60	5.00	7.20	<b>2.60</b>
5	0.01	8.60	6.60	9.80	<b>1.20</b>	5.00	11.00	11.60	4.60	2.60	6.40	6.60	4.00
	0.05	5.60	6.80	9.20	<b>1.40</b>	8.60	10.60	11.80	8.80	1.60	4.80	3.60	5.20
	0.1	3.20	6.60	8.80	2.00	10.60	10.60	11.60	8.40	<b>1.00</b>	5.00	3.80	6.40
6	0.01	2.60	6.80	3.40	5.80	12.00	4.40	<b>1.40</b>	3.80	10.40	7.80	9.20	10.40
	0.05	2.20	5.40	3.80	10.20	12.00	6.60	2.80	<b>1.20</b>	10.40	6.60	9.40	7.40
	0.1	<b>1.60</b>	4.80	4.40	10.80	12.00	8.00	3.00	1.80	10.00	6.20	9.20	6.20
7	0.01	<b>1.60</b>	4.00	2.80	8.40	12.00	7.20	9.20	3.20	7.60	4.20	8.80	9.00
	0.05	<b>1.20</b>	3.60	2.40	10.80	12.00	8.00	9.80	5.20	5.80	3.60	8.40	7.20
	0.1	<b>1.00</b>	4.20	2.60	11.00	12.00	7.80	9.60	6.20	4.40	3.60	7.60	8.00
1–7	0.01	5.43	5.14	7.57	5.03	9.71	9.54	7.06	<b>3.16</b>	6.53	5.44	6.29	7.10
	0.05	4.91	<b>4.87</b>	7.09	7.26	11.06	9.83	5.74	5.71	5.63	4.87	5.37	5.66
	0.1	<b>4.11</b>	5.20	6.91	7.91	11.63	9.86	5.74	5.80	5.34	4.83	5.17	5.49
1–4	0.01	6.30	4.65	9.25	4.95	9.75	11.05	6.80	<b>2.62</b>	6.28	4.92	4.85	6.58
	0.05	6.35	4.58	8.55	7.10	11.20	10.90	<b>3.95</b>	6.20	5.40	4.78	4.05	4.95
	0.1	5.75	5.20	8.15	7.90	11.70	10.65	4.00	6.05	5.50	4.75	<b>3.90</b>	4.45
5–7	0.01	4.27	5.80	5.33	5.13	9.67	7.53	7.40	<b>3.87</b>	6.87	6.13	8.20	7.80
	0.05	<b>3.00</b>	5.27	5.13	7.47	10.87	8.40	8.13	5.07	5.93	5.00	7.13	6.60
	0.1	<b>1.93</b>	5.20	5.27	7.93	11.53	8.80	8.07	5.47	5.13	4.93	6.87	6.87

## 6.6 Поређење са мрежним омнибус тестом NT

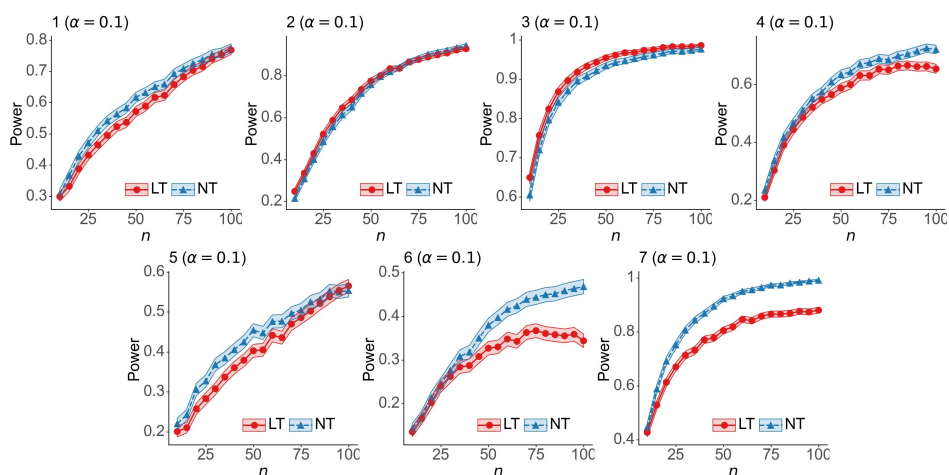
На сликама 6.6–6.8 дат је упоредни приказ моћи тестова LT и NT са интервалима поверења од 99.9% за све групе алтернативних расподела из табеле 5.3.



Слика 6.6: Моћ тестова NT и LT за праг  $\alpha = 0.01$



Слика 6.7: Моћ тестова NT и LT за праг  $\alpha = 0.05$



Слика 6.8: Моћ тестова NT и LT за праг  $\alpha = 0.1$

Тест NT има већу моћ од теста LT у готово свим случајевима. Главни изузетак представља група 3 за  $\alpha \in \{0.05, 0.1\}$  и  $n \leq 100$ , где је тест LT моћнији. Највећа разлика уочава се код контаминираних нормалних расподела, на којима моћ теста NT знатно превазилази моћ теста LT, као и на јако малим узорцима ( $n \leq 30$ ) из група 2 и 3, где тест LT превазилази тест NT.

## 6.7 Дискусија

Тест LT је на узорцима са највише 100 елемената показао високу моћ за лептокуртичне искошене и платикуртичне симетричне расподеле. Међутим, за изузетно мале узорке ( $n \leq 30$ ), тест LT рангира се знатно боље у поређењу са другим тестовима. Он има најбољи ранг на класи искошених лептокуртичних расподела за праг  $\alpha \in \{0.05, 0.1\}$  (групе 2 и 3), као и на класи симетричних платикуртичних расподела (група 4) за  $\alpha = 0.01$ , док за  $\alpha \in \{0.05, 0.1\}$  има други најбољи просечни ранг на узорцима из групе 4. Тест LT је у групи 6 показао врло добре резултате: најбољи је за  $\alpha = 0.01$  и трећи најбољи за  $\alpha \in \{0.05, 0.1\}$ . Међутим, има нижу моћ од других тестова на класи варијацијски контаминираних нормалних расподела (група 5), чији тип контаминације чува симетричност основне нормалне компоненте, а врло низак ранг има и на групи 1. Моћ му је већа на групи 7, коју чине комбиновано контаминиране нормалне расподеле, али су за такве расподеле и други тестови моћнији.

Мрежни тестови имају најбољи просечан ранг на симетричним платикуртичним расподелама без обзира на величину узорка. Ако је задати узорак мали ( $n \leq 30$ ), највећу моћ за највећи број алтернативних расподела имају управо мрежни тестови.

На основу наведених резултата, формулишемо следеће препоруке за примену омнибус тестова на узорке са највише 100 елемената:

- 1) Ако је  $\alpha = 0.01$ , треба применити тест NT.
- 2) Ако је  $\alpha = 0.05$ , треба применити тест CS или SW.
- 3) Ако је  $\alpha = 0.1$ , треба применити тест SW.

За узорке са највише 30 елемената важи следеће:

- 1) Ако је  $\alpha = 0.01$ , треба применити тест NT.
- 2) Ако је  $\alpha = 0.05$ , треба применити тест CS или SW.
- 3) Ако је  $\alpha = 0.1$ , треба применити тест AD.

# Глава 7

## Дискусија

Истраживање приказано у оквиру ове дисертације полази од следећих хипотеза:

- ( $\mathcal{H}_1$ ) Може се формирати скуп података на којем је могуће тренирати неуронску мрежу тако да она по перформансама надмаши стандардне тестове нормалности.
- ( $\mathcal{H}_2$ ) Може се развити неуронска мрежа која нулту хипотезу нормалности испитује наспрам алтернативне хипотезе да задати узорак потиче из тачно одређене расподеле или фамилије расподела, а не из скупа свих расподела које нису нормалне.
- ( $\mathcal{H}_3$ ) Могуће је оптимизовати не само параметре неуронске мреже, већ и начин репрезентације узорака и структуру података за тренирање.
- ( $\mathcal{H}_4$ ) Применом метода контроле грешке, развијеним у оквиру машинског учења, могуће је задати горњу границу вероватноће да се нормална расподела не класификује као таква, чиме би се омогућило коришћење неуронске мреже као да је у питању стандардан статистички тест. При томе, услов је да се горња граница може накнадно мењати тако да не захтева поновно тренирање мреже.

Експериментални резултати показују да су полазне хипотезе  $\mathcal{H}_2$ ,  $\mathcal{H}_3$  и  $\mathcal{H}_4$  доказане у потпуности. Генетским алгоритом је могуће оптимизовати параметре свих корака у конструкцији неуронске мреже за испитивање нормалности расподела, а то су метапараметри генерисања података, репрезентације, архитектуре и тренирања. Метода SCT успешно контролише вероватноћу да нормалну расподелу прогласимо не-нормалном.

Хипотеза  $\mathcal{H}_1$  доказана је делимично. У одређеним случајевима, неуронске мреже и тестови из њих изведени превазилазе класичне статистичке тестове нормалности по перформансама. Усмерени мрежни тестови изведени из једносмерних мрежа надмашили су стандардне тестове за скоро све типове не-нормалних расподела и прагове значајности. На узорцима са највише 100 елемената и за праг значајности  $\alpha = 0.01$ , мрежни омнибус тест NT има моћ већу од стандардних статистичких тестова с којима смо га упоредили. Исто важи и за веома мале узорке (са највише 30 елемената). Код њих се рекурентни мрежни тест LT показао као најбољи избор за искошене лептокуртичне и симетричне платикуртичне расподеле. Добијени резултати у складу су с постојећом литературом. Бројне емпиријске студије о моћи тестова нормалности за разне обиме узорака и типове не-нормалних расподела показују да ниједан статистички тест по моћи не надмашује друге у свим случајевима.

### 7.1 Тумачивост мрежних тестова

Поред моћи, важна карактеристика тестова нормалности је и њихова тумачивост. Предност мрежних тестова је што као статистику користе нумерички предизлаз неур-

ронске мреже, који има једноставну и интуитивну интерпретацију. У питању је процена вероватноће да задати узорак потиче из нормалне расподеле. Насупрот томе, статистике већине стандардних статистичких тестова нормалности не могу се тако лако протумачити, већ коначну интерпретацију добијају кроз  $p$ -вредност. То је проблематично зато што је у свакодневной пракси погрешно тумачење  $p$ -вредности прилично распрострањено [7, 9, 10].

Ипак, иако статистика мрежних тестова има интуитивну интерпретацију, начин на који се она рачуна је теже објаснити него што је то случај код статистичких тестова. Наиме, неуронске мреже су нелинеране и врло сложене функције аргумената чак и када имају мали број скривених слојева и неурона у њима. Тај недостатак може се отклонити применом метода за објашњавање модела машинског учења, које се користе све чешће (нпр. [108–112]).

## 7.2 Мреже као емпиријски индуковане статистике

Статистике класичних статистичких тестова нормалности заснивају се на теоретским својствима нормалних расподела. Њихове расподеле се у већини случајева могу извести аналитички, што убрзава како развој тих тестова, тако и израчунавање  $p$ -вредности.

Насупрот њима, код мрежних тестова морамо индуковати статистику (односно, неуронску мрежу) из скупова за тренирање који садрже како узорке из нормалних, тако и узорке из бројних не-нормалних расподела. Такав емпиријски приступ омогућава мрежама да приликом обучавања пронађу оптималан скуп својстава по којима се ове две класе дистрибуција највише разликују. То је могуће објашњење зашто је мрежни тест NT надмашио све статистичке тестове нормалности по свеукупној просечној моћи при прагу значајности  $\alpha = 0.01$  на свим класама не-нормалних расподела за узорке са највише 30 елемената и свим класама сем контаминираних нормалних дистрибуција за узорке са максимално 100 елемената. Слично се може објаснити и висока моћ рекурентног мрежног теста LT на типовима не-нормалних расподела на којима је превазишао класичне статистичке тестове нормалности, као и то што су усмерени мрежни тестови моћнији од класичних статистичких.

Ипак, емпиријска природа мрежних тестова отежава контролу грешке пошто је расподелу излазне вероватноће мрежа немогуће извести аналитички. Да би се контролисала вероватноћа грешке да се нормалан узорак прогласи не-нормалним, неопходно је извести статистички тест из мреже путем метода машинског учења за контролу грешке. У оквиру ове дисертације је показано да најбоље перформансе има метода SCT, која излазну вероватноћу мрежа третира као статистику и њену расподелу одређују емпиријски помоћу нормалних узорака који нису употребљени за обучавање тих мрежа. Ипак, расподела се не може аналитички одредити ни код појединих статистичких тестова нормалности, као што је случај са тестовима CS, ZC и ZA, тако да мрежни тестови нису једини изузетак по овом питању.

## 7.3 Величина скупа за тренирање

У табели 7.1 дат је упоредни приказ величина скупова за тренирање омнибус мрежа из којих су у овој дисертацији изведени тестови NT и LT, као и скупова за конструкцију мрежа из радова [5, 6].

Сви наведени скупови су балансирани. Мрежа из рада [6] обучавања је на скупу који садржи укупно 26 100 узорака различитих обима (10, 20, ..., 100). Мрежа из рада [5] тренирана је на узорцима обима 25, 50, 100 и 200. За сваки обим је генерисано 5000

**Табела 7.1:** Величина скупа за тренирање ( $N$ ) омнибус мрежа за тестирање нормалности (NT - класична једносмерна мрежа, LT - дугорочна рекурентна мрежа типа LSTM).

Мрежа	[5]	[6]	NT	LT
$N$	40000	26100	<b>1760</b>	39200

нормалних и 5000 не-нормалних узорака. Скуп за мрежу из рада [4] није обухваћен зато што садржи само узорке са 30 елемената, за разлику од осталих мрежа које су конструисане са циљем да се примењују на узорке различитих обима.

Да су се скупови за тренирање у оквиру ове дисертације састављали на исти начин као у раду [5], садржали би чак 100 000 узорака. Истовремена оптимизација свих мета-параметара помоћу генетског алгоритма омогућила је да се репрезентација, архитектура и алгоритам за тренирање подесе тако да 1760 узорака буде довољно за обучавање прецизне и стабилне једносмерне мреже из које је изведен мрежни тест NT. Претпоставка је да значајну улогу у томе игра избор  $p$ -вредности као предстандардизационих димензија репрезентационог простора. Могуће објашњење је да  $p$ -вредности садрже информације о томе колико се често дата вредност статистике јавља код узорака из нормалних расподела. Мреже које не користе  $p$ -вредности морају да науче расподелу репрезентационих димензија из података. Да би у томе успеле, потребни су им велики скупови узорака за тренирање.

## 7.4 Стабилност и монотоност тачности

Резултати показују да методологија заснована на генетским алгоритмима проналази стабилне мреже. Стабилност је неопходан услов за примену мрежа и извођење тестова из њих. Ако тачност мреже зависи од варијансе расподела из којих се генеришу узорци за тренирање, процена вероватноће нормалности пратиће различите расподеле за узорке из дистрибуција различите варијабилности. То значи да методе контроле грешке већ тренираних класификатора не могу да се примене на такве мреже. Досадашњи приступи нису узимали у обзир стабилност у односу на варијансу, тако да постојеће мреже нису стабилне.

Такође, важно је и да тачност буде неоппадајућа функција обима узорака. Што је узорак већи, то је више информација о расподели доступно мрежним и статистичким тестовима, тако да је оправдано очекивати да им тачност расте са обимом испитиваних узорака. То је случај и са класичним једносмерним, и са дугорочним рекурентним мрежама добијеним генетским алгоритмом.

## 7.5 Контрола вероватноће грешке

Битан корак у развоју мрежних тестова био је одабир методе за контролу грешке на класи нормалних узорака. Изведено је опсежно истраживање постојећих техника контроле у области машинског учења. Као референтна метода је формулисано извођење статистичких тестова из бинарних класификатора (SCT), а предложене су и нове верзије посредничког тестирања: метода која користи тестове са композитном нултом хипотезом и метода која користи тежинска растојања (WTBC).

Када се у обзир узму прецизност у контроли грешке на критичној и вероватноћа грешке на некритичној класи на свим евалуационим скуповима, најбоље резултате има управо референтна метода SCT. Међутим, експериментална студија представљена у

одељку 4.10 показује да метода WTBC може да оствари боље резултате на скуповима података специфичног тополошког својства. У питању су простори код којих су позитивни објекти ближи својим негативним него позитивним суседима (без губитка општости можемо претпоставити да је циљ контрола грешке на позитивној класи).

## 7.6 Аутоматизација нумеричких експеримената

Генетски алгоритам може се посматрати не само као техника за оптимизацију метапараметара, већ и као метода која аутоматизује експерименте на основу принципа еволуције. Свака јединка у популацији GA представља експеримент чији је циљ развој неуронске мреже за испитивање нормалности. При томе, функција прилагођености коју користимо за евалуацију јединки квантификује успешност једног експеримента. Итеративно примењујући генетске операторе и вршећи замену генерација, GA изводи низ све успешнијих експеримената. Он аутоматизовано одређује, на основу успешности већ извршених експеримената, конфигурације нових експеримената које ће извести у наредном кораку.

Предност аутоматизације је што можемо брзо и паралелно извести велик број експеримената. Предуслов за то је да генетски оператори, функција прилагођености и замена генерација буду адекватно адаптирани за извођење експеримената и прате сличну логику којом би се у традиционалном секвенцијалном приступу водили истраживачи када на основу резултата извршених огледа бирају конфигурацију за наредни.

Мана аутоматског приступа је што он не може да објасни и тумачи експерименталне резултате, већ их првенствено оптимизује. У стандардној теорији дизајна и анализе експеримената, од истраживача би се очекивало да статистичким методама утврде какав утицај имају метапараметри на успешност експеримента (односно на вредност функције прилагођености). Међутим, те методе нису лако применљиве на огледе са великим бројем фактора, што је случај са експериментима које представљају јединке у популацији GA. Разлог за то је велик број могућих интеракција међу метапараметрима, које је тешко испитати и протумачити без увођења рестриктивних претпоставки о њиховом утицају на прилагођеност.

## 7.7 Изазови

Да би се конструисала стабилна и прецизна мрежа за испитивање нормалности, морамо пажљиво подесити све метапараметре. Примена генетских алгоритама захтева знатне рачунарске ресурсе, што није случај код аналитичког извођења класичних тестова нормалности у статистици.

Такође, ако се параметри генетског алгоритма не подесе адекватно или ако на одговарајући начин не формулишемо његове компоненте (операторе, функцију прилагођености и замену генерација), он неће моћи ефикасно да претражи простор решења и пронађе задовољавајућу конфигурацију метапараметара.

Како је дисертација фокус истраживања ставила на мале узорке ( $n \leq 100$ ), перформансе добијених тестова и мрежа на већим узорцима нису испитане. Ипак, на основу неоппадајуће тачности мрежа у односу на обим узорака, може се претпоставити да им моћ неће опадати на значајно већим узорцима.

# Глава 8

## Закључак

У оквиру ове дисертације, формулисана је нова методологија за развој неуронских мрежа за испитивање нормалности расподела на основу малих узорака, са највише 100 елемената. Методологија се заснива на примени генетског алгоритма на проблем истовремене оптимизације параметара генерисања података, репрезентације узорака, мрежне архитектуре и алгоритма за тренирање. Мрежама добијеним на тај начин вероватноћа грешке прве врсте контролисана је како би се оне могле користити као статистички тестови.

Експериментални резултати показују да су полазне хипотезе  $\mathcal{H}_2$ ,  $\mathcal{H}_3$  и  $\mathcal{H}_4$  доказане у целости, а да је хипотеза  $\mathcal{H}_1$  доказана делимично. Генетски алгоритам успешно оптимизује све параметре од којих зависе перформансе мреже, а изабраном методом може се контролисати вероватноћа да мрежа погрешно класификује нормалну расподелу као не-нормалну. Развијени су усмерени и мрежни омнибус тестови нормалности на основу класичних једносмерних мрежа и дугорочних рекурентних неуронских мрежа типа LSTM. Усмерени мрежни тестови су надмашили класичне статистичке тестове на узорцима из алтернативних расподела за које су развијени. Омнибус мрежни тестови показали су се компетитивним са стандардним статистичким омнибус тестовима нормалности, а успели су да их по моћи превазиђу за велики број типова расподела.

Поред тога, предложена је метода статистичке класификације као референтна техника за поређење метода за контролу грешке бинарних класификатора на циљној класи. Формулисана је и верзија посредничког тестирања заснована на композитној нултој хипотези, као и тежинска верзија посредничког тестирања (WTBC), која дистанце до најближих суседа множи тежинама пре извођења статистичког теста о истоветности расподела. Метода WTBC има боље перформансе како од нетежинске формулације, тако и од других метода контроле грешке на скуповима код којих су објекти циљне класе ближи најближим суседима из супротне него суседним објектима из исте класе.

Могући правци будућих истраживања у оквиру ове области су:

- Примена генетског алгоритма на проблем испитивања припадности другим расподелама, нпр. тестирање униформности.
- Комбинација класичних једносмерних мрежа које користе  $p$ -вредности са мрежама типа LSTM.
- Примена других метахеуристика у оптимизацији метапараметара података, репрезентације, архитектуре и тренирања.



# Литература

- [1] Vasicek, O. A test for normality based on sample entropy. *Journal of the Royal Statistical Society. Series B (Methodological)*, 38(1):54–59, 1976.
- [2] Noughabi, H. A., Arghami, N. R. Monte Carlo comparison of seven normality tests. *Journal of Statistical Computation and Simulation*, 81(8):965–972, 2011.
- [3] Esteban, M. D., Castellanos, M. E., Morales, D., Vajda, I. Monte Carlo comparison of four normality tests using different entropy estimates. *Communications in Statistics - Simulation and Computation*, 30(4):761–785, 2001.
- [4] Wilson, P., Engel, A. Testing for normality using neural networks. In *[1990] Proceedings. First International Symposium on Uncertainty Modeling and Analysis*, 700–704, 1990.
- [5] Sigut, J., Piñeiro, J., Estévez, J., Toledo, P. A neural network approach to normality testing. *Intelligent Data Analysis*, 10(6):509–519, 12 2006.
- [6] Simić, M. Testing for normality with neural networks. *Neural Computing and Applications*, 33(23):16279–16313, July 2021.
- [7] Goodman, S. A dirty dozen: Twelve p-value misconceptions. *Seminars in Hematology*, 45(3):135–140, 2008. Interpretation of Quantitative Research.
- [8] Demidenko, E. The p-value you can't buy. *The American Statistician*, 70(1):33–38, January 2016.
- [9] Wasserstein, R. L., Lazar, N. A. The ASA statement on p-values: Context, process, and purpose. *The American Statistician*, 70(2):129–133, 2016.
- [10] Wasserstein, R. L., Schirm, A. L., Lazar, N. A. Moving to a world beyond  $p < 0.05$ . *The American Statistician*, 73(sup1):1–19, 2019.
- [11] Thode, H. *Testing For Normality*. Statistics, textbooks and monographs. Taylor & Francis, 2002.
- [12] Shapiro, S. S., Wilk, M. B. An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611, 1965.
- [13] Ghasemi, A., Zahediasl, S. Normality tests for statistical analysis: A guide for non-statisticians. *International Journal of Endocrinology and Metabolism*, 10(2):486–489, December 2012.
- [14] Gupta, A., Mishra, P., Pandey, C. M., Singh, U., Sahu, C., Keshri, A. Descriptive statistics and normality tests for statistical data. *Annals of Cardiac Anaesthesia*, 22(1):67, 2019.

- [15] Shapiro, S. S., Francia, R. S. An approximate analysis of variance test for normality. *Journal of the American Statistical Association*, 67(337):215–216, 1972.
- [16] Chen, L., Shapiro, S. S. An alternative test for normality based on normalized spacings. *Journal of Statistical Computation and Simulation*, 53(3-4):269–287, 1995.
- [17] Weisberg, S., Bingham, C. An approximate analysis of variance test for non-normality suitable for machine calculation. *Technometrics*, 17(1):133–134, 1975.
- [18] Jarque, C. M., Bera, A. K. Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics Letters*, 6(3):255 – 259, 1980.
- [19] Bowman, K. O., Shenton, L. R. Omnibus test contours for departures from normality based on  $\sqrt{b_1}$  and  $b_2$ . *Biometrika*, 62(2):243–250, 08 1975.
- [20] Urzua, C. On the correct use of omnibus tests for normality. *Economics Letters*, 53(3):247–251, 1996.
- [21] D’Agostino, R., Pearson, E. S. Tests for departure from normality. Empirical results for the distributions of  $b_2$  and  $\sqrt{b_1}$ . *Biometrika*, 60(3):613–622, 1973.
- [22] Ralph B., D., Belanger, A. A suggestion for using powerful and informative tests of normality. *The American Statistician*, 44(4):316–321, 1990.
- [23] Kolmogorov, A. Sulla determinazione empirica di una legge di distribuzione. *Giornale dell’ Istituto Italiano degli Attuari*, 4:83–91, 1933.
- [24] Lilliefors, H. W. On the Kolmogorov-Smirnov test for normality with mean and variance unknown. *Journal of the American Statistical Association*, 62(318):399–402, 1967.
- [25] van Soest, J. Some experimental results concerning tests of normality\*. *Statistica Neerlandica*, 21(1):91–97, 1967.
- [26] Wijekularathna, D. K., Manage, A. B. W., Scariano, S. M. Power analysis of several normality tests: A Monte Carlo simulation study. *Communications in Statistics - Simulation and Computation*, 1–17, September 2019.
- [27] Anderson, T. W., Darling, D. A. A test of goodness of fit. *Journal of the American Statistical Association*, 49(268):765–769, 1954.
- [28] Gross, J., Ligges, U. nortest: Tests for normality, 2015. R package version 1.0-4.
- [29] Cramér, H. On the composition of elementary errors. *Scandinavian Actuarial Journal*, 1928(1):13–74, January 1928.
- [30] von Mises, R. *Wahrscheinlichkeit Statistik und Wahrheit*. Springer Berlin Heidelberg, 1928.
- [31] von Mises, R. *Wahrscheinlichkeitsrechnung und Ihre Anwendung in der Statistik und Theoretischen Physik*. F. Deuticke, 1931.
- [32] Smirnov, N. V. Sur la distribution de  $\omega^2$ . *CR Acad. Sci. Paris*, 202(S 449), 1936.
- [33] Zhang, J., Wu, Y. Likelihood-ratio tests for normality. *Computational Statistics & Data Analysis*, 49(3):709–721, 2005.

- [34] Zhang, J. *Powerful goodness-of-fit and multi-sample tests*. PhD thesis, York University, Toronto, Ontario, Canada, July 2001.
- [35] Zhang, J. Powerful goodness-of-fit tests based on the likelihood ratio. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 64(2):281–294, 2002.
- [36] D’Agostino, R. B. An omnibus test of normality for moderate and large size samples. *Biometrika*, 58(2):341–348, 1971.
- [37] Anscombe, F. J., Glynn, W. J. Distribution of the kurtosis statistic  $b_2$  for normal samples. *Biometrika*, 70(1):227–234, 1983.
- [38] Johnson, N. L. Systems of frequency curves generated by methods of translation. *Biometrika*, 36(1/2):149, June 1949.
- [39] Lin, C.-C., Mudholkar, G. S. A simple test for normality against asymmetric alternatives. *Biometrika*, 67(2):455–461, 1980.
- [40] Pearson, K. Contributions to the mathematical theory of evolution. II. Skew variation in homogeneous material. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 186:343–414, 1895.
- [41] Pearson, K. X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 50(302):157–175, July 1900.
- [42] Pearson, K. Mathematical contributions to the theory of evolution. X. Supplement to a memoir on skew variation. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 197(287-299):443–459, 1901.
- [43] Pearson, K. Mathematical contributions to the theory of evolution. XIX. Second supplement to a memoir on skew variation. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 216(538-548):429–457, 1916.
- [44] Brownlee, J. *Clever algorithms: nature-inspired programming recipes*. Jason Brownlee, 2011.
- [45] Mirjalili, S. *Genetic Algorithm*, 43–55. Springer International Publishing, Cham, 2019.
- [46] Kramer, O. *Genetic Algorithm Essentials*. Springer International Publishing, 2017.
- [47] Rudolph, G. Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Networks*, 5(1):96–101, 1994.
- [48] Schmitt, L. M. Theory of genetic algorithms. *Theoretical Computer Science*, 259(1):1–61, 2001.
- [49] Socała, J., Kosiński, W. On convergence of a simple genetic algorithm. In Rutkowski, L., Tadeusiewicz, R., Zadeh, L. A., Zurada, J. M., editors, *Artificial Intelligence and Soft Computing – ICAISC 2008*, 489–498, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
- [50] Kar, A. K. Bio inspired computing – a review of algorithms and scope of applications. *Expert Systems with Applications*, 59:20–32, 2016.

- [51] Sohail, A. Genetic algorithms in the fields of artificial intelligence and data sciences. *Annals of Data Science*, 10(4):1007–1018, August 2021.
- [52] Alhijawi, B., Awajan, A. Genetic algorithms: theory, genetic operators, solutions, and applications. *Evolutionary Intelligence*, 2023.
- [53] White, D., Ligomenides, P. GANNet: A genetic algorithm for optimizing topology and weights in neural network design. In Mira, J., Cabestany, J., Prieto, A., editors, *New Trends in Neural Computation*, 322–327, Berlin, Heidelberg, 1993. Springer Berlin Heidelberg.
- [54] Ding, S., Xu, L., Su, C., Jin, F. An optimizing method of RBF neural network based on genetic algorithm. *Neural Computing and Applications*, 21(2):333–336, July 2011.
- [55] Idrissi, M. A. J., Ramchoun, H., Ghanou, Y., Eттаouil, M. Genetic algorithm for neural network architecture optimization. In *2016 3rd International Conference on Logistics Operations Management (GOL)*, 1–4, 2016.
- [56] Zhang, T., Lei, C., Zhang, Z., Meng, X.-B., Chen, C. L. P. AS-NAS: Adaptive scalable neural architecture search with reinforced evolutionary algorithm for deep learning. *IEEE Transactions on Evolutionary Computation*, 25(5):830–841, 2021.
- [57] Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Chen, X., Wang, X. A comprehensive survey of neural architecture search: Challenges and solutions. *ACM Computing Surveys*, 54(4), may 2021.
- [58] Mellor, J., Turner, J., Storkey, A., Crowley, E. J. Neural architecture search without training. In Meila, M., Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 7588–7598. PMLR, 18–24 Jul 2021.
- [59] Liu, Y., Sun, Y., Xue, B., Zhang, M., Yen, G. G., Tan, K. C. A survey on evolutionary neural architecture search. *IEEE Transactions on Neural Networks and Learning Systems*, 34(2):550–570, 2023.
- [60] Karvanen, J., Eriksson, J., Koivunen, V. Pearson system based method for blind separation. In *Proceedings of Second International Workshop on Independent Component Analysis and Blind Signal Separation (ICA2000), Helsinki, Finland*, 585–590, 2000.
- [61] Kallner, A. *Laboratory Statistics*. Elsevier, 2 edition, 2018.
- [62] Kingma, D. P., Ba, J. Adam: A method for stochastic optimization. In Bengio, Y., LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.
- [63] Bottou, L. Stochastic gradient descent (v.2). <https://leon.bottou.org/projects/sgd>, 2011. Accessed: January 5, 2024.
- [64] Shalev-Shwartz, S., Singer, Y., Srebro, N. Pegasos: Primal Estimated sub-GrAdient Solver for SVM. In *Proceedings of the 24th International Conference on Machine Learning, ICML '07*, 807–814, New York, NY, USA, 2007. Association for Computing Machinery.

- [65] Tsuruoka, Y., Tsujii, J., Ananiadou, S. Stochastic gradient descent training for L1-regularized log-linear models with cumulative penalty. In Su, K.-Y., Su, J., Wiebe, J., Li, H., editors, *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, 477–485, Suntec, Singapore, August 2009. Association for Computational Linguistics.
- [66] Nocedal, J. Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, 35(151):773–782, 1980.
- [67] Liu, D. C., Nocedal, J. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1–3):503–528, August 1989.
- [68] Morgan, N., Bourlard, H. Generalization and parameter estimation in feedforward nets: Some experiments. In Touretzky, D., editor, *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1989.
- [69] Zafar, I., Tzanidou, G., Burton, R., Patel, N., Araujo, L. *Hands-On Convolutional Neural Networks with TensorFlow*. Packt Publishing, 2018.
- [70] Gammerman, A., Vovk, V., Vapnik, V. Learning by transduction. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI’98, 148–155, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.
- [71] Vapnik, V. N. *Statistical learning theory*. Adaptive and Cognitive Dynamic Systems: Signal Processing, Learning, Communications and Control. John Wiley & Sons, Nashville, TN, September 1998.
- [72] Papadopoulos, H., Proedrou, K., Vovk, V., Gammerman, A. Inductive confidence machines for regression. In Elomaa, T., Mannila, H., Toivonen, H., editors, *Machine Learning: ECML 2002*, 345–356, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.
- [73] Papadopoulos, H., Vovk, V., Gammerman, A. Qualified prediction for large data sets in the case of pattern recognition. In Wani, M. A., Arabnia, H. R., Cios, K. J., Hafeez, K., Kendall, G., editors, *Proceedings of the 2002 International Conference on Machine Learning and Applications - ICMLA 2002, June 24-27, 2002, Las Vegas, Nevada, USA*, 159–163. CSREA Press, 2002.
- [74] Toccaceli, P., Gammerman, A. Combination of inductive mondrian conformal predictors. *Machine Learning*, 108(3):489–510, August 2018.
- [75] Vovk, V., Gammerman, A., Shafer, G. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.
- [76] Papadopoulos, H. Inductive conformal prediction: Theory and application to neural networks. In Fritzsche, P., editor, *Tools in Artificial Intelligence*, chapter 18. IntechOpen, Rijeka, 2008.
- [77] Balasubramanian, V., Ho, S.-S., Vovk, V. *Conformal prediction for reliable machine learning*. Morgan Kaufmann, Oxford, England, 4 2014.
- [78] Angelopoulos, A. N., Bates, S. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021.

- [79] Tong, X., Feng, Y., Zhao, A. A survey on Neyman-Pearson classification and suggestions for future research. *Wiley Interdisciplinary Reviews: Computational Statistics*, 8(2):64–81, February 2016.
- [80] Cannon, A., Howse, J., Hush, D., Scovel, C. Learning with the Neyman–Pearson and min–max criteria. Technical report, Los Alamos National Laboratory, June 2002.
- [81] Scott, C., Nowak, R. A Neyman-Pearson approach to statistical learning. *IEEE Transactions on Information Theory*, 51(11):3806–3819, 2005.
- [82] Scott, C., Bellala, G., Willett, R. Generalization error analysis for FDR controlled classification. In *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*, 792–796, 2007.
- [83] Han, M., Chen, D., Sun, Z. Analysis to Neyman-Pearson classification with convex loss function. *Analysis in Theory and Applications*, 24(1):18–28, March 2008.
- [84] Rigollet, P., Tong, X. Neyman-Pearson classification, convexity and stochastic constraints. *Journal of Machine Learning Research*, 12(Oct):2831–2855, 2011.
- [85] Tong, X. A plug-in approach to Neyman-Pearson classification. *J. Mach. Learn. Res.*, 14(1):3011–3040, January 2013.
- [86] Zhao, A., Feng, Y., Wang, L., Tong, X. Neyman-Pearson classification under high-dimensional settings. *J. Mach. Learn. Res.*, 17(1):7469–7507, January 2016.
- [87] Lehmann, E. L., Romano, J. P. *Testing statistical hypotheses*. Springer Science & Business Media, 2006.
- [88] Tong, X., Xia, L., Wang, J., Feng, Y. Neyman-Pearson classification: parametrics and sample size requirement. *Journal of Machine Learning Research*, 21(12):1–48, 2020.
- [89] Tong, X., Feng, Y., Li, J. J. Neyman-Pearson classification algorithms and NP receiver operating characteristics. *Science Advances*, 4(2), 2018.
- [90] Dümbgen, L., Igl, B.-W., Munk, A. P-values for classification. *Electron. J. Statist.*, 2:468–493, 2008.
- [91] Liao, S.-M., Akritas, M. Test-based classification: A linkage between classification and statistical testing. *Statistics & Probability Letters*, 77(12):1269–1281, July 2007.
- [92] Ghimire, S., Wang, H. Classification of image pixels based on minimum distance and hypothesis testing. *Computational Statistics & Data Analysis*, 56(7):2273–2287, July 2012.
- [93] Guo, L., Modarres, R. Interpoint distance classification of high dimensional discrete observations. *International Statistical Review*, 87(2):191–206, 2019.
- [94] He, Z., Sheng, C., Liu, Y., Zou, Q. Instance-based classification through hypothesis testing. *IEEE Access*, 9:17485–17494, 2021.
- [95] Baringhaus, L., Franz, C. On a new multivariate two-sample test. *Journal of Multivariate Analysis*, 88(1):190–206, January 2004.

- [96] McLachlan, G. J. *Discriminant Analysis and Statistical Pattern Recognition*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. John Wiley & Sons, Inc., New York, 3 1992.
- [97] Li, J. J., Tong, X. Statistical hypothesis testing versus machine learning binary classification: Distinctions and guidelines. *Patterns*, 1(7):100115, 2020.
- [98] Abell, M. L., Braselton, J. P., Rafter, J. A. *Statistics with Mathematica*. Academic Press, 1999.
- [99] Biau, D. J., Jolles, B. M., Porcher, R. P value and the theory of hypothesis testing: An explanation for new researchers. *Clinical Orthopaedics and Related Research*®, 468(3):885–892, November 2009.
- [100] Mayo, D. G., Spanos, A. Severe testing as a basic concept in a Neyman–Pearson philosophy of induction. *The British Journal for the Philosophy of Science*, 57(2):323–357, 2006.
- [101] Tucker, H. G. A generalization of the Glivenko-Cantelli theorem. *The Annals of Mathematical Statistics*, 30(3):828–830, 1959.
- [102] Dvoretzky, A., Kiefer, J., Wolfowitz, J. Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. *The Annals of Mathematical Statistics*, 27(3):642–669, 1956.
- [103] Massart, P. The tight constant in the Dvoretzky-Kiefer-Wolfowitz inequality. *The Annals of Probability*, 18(3):1269–1283, 1990.
- [104] Kanamori, T., Suzuki, T., Sugiyama, M. Statistical analysis of kernel-based least-squares density-ratio estimation. *Machine Learning*, 86(3):335–367, November 2011.
- [105] Smith, J. W., Everhart, J., Dickson, W., Knowler, W., Johannes, R. Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the Symposium on Computer Applications and Medical Care*, 261–265. IEEE Computer Society Press, November 1988.
- [106] Hochreiter, S., Schmidhuber, J. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- [107] Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollar, P. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [108] Goldstein, A., Kapelner, A., Bleich, J., Pitkin, E. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65, 2015.
- [109] Ribeiro, M. T., Singh, S., Guestrin, C. "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16*, 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery.
- [110] Lundberg, S. M., Lee, S.-I. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 4765–4774, 2017.

- [111] Ribeiro, M. T., Singh, S., Guestrin, C. Anchors: High-precision model-agnostic explanations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018.
- [112] Dhurandhar, A., Chen, P.-Y., Luss, R., Tu, C.-C., Ting, P.-S., Shanmugam, K., Das, P. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. In *NeurIPS*, 590–601, 2018.

## Биографија Аутора

Милош Симић рођен је 1991. године у Крагујевцу, где је завршио основну школу „Живадинка Дивац” и природно-математички смер Прве крагујевачке гимназије као носилац Вукове дипломе. Основне академске студије информатике завршио је 2014. године на Природно-математичком факултету у Крагујевцу са просечном оценом 10,00 и као студент генерације, са дипломским радом „*Бореалис – друштвена мрежа за књижев-но стваралаштво и критику*” под менторством професора доктора Владимира Цвјетковића. На истој установи је 2015. године завршио мастер студије информатике са просеком 10,00, одбранивши мастер рад „*Примена оптимизације колонијом пчела у испитивању задовољности формула вероватности логике  $LPP_2^{ext}$* ” под менторством професора доктора Небојше Икодиновића. Добитник је стипендије Фонда за младе таленте за најбоље студенте завршних година основних и мастер академских студија, као и Универзитета у Крагујевцу за најбоље студенте завршних година основних академских студија. Од 2018. године је студент докторских студија Интелигентних система при Универзитету у Београду, где се бави комбиновањем статистике и вештачке интелигенције под менторством доктора Срђана Станковића, професора емеритуса Универзитета у Београду, и др Зорице Станимировић, редовног професора Математичког факултета у Београду. Изабран је у научно звање истраживач приправник на Електротехничком факултету у Београду 2020. године. Аутор је два објављена научна рада. Године 2020. је његов рад *How to Control the Error Rates of Binary Classifiers* представљен као један од десет најбољих на догађају *New in ML* организованом у оквиру конференције *NeurIPS 2020*. Његово решење за објашњив модел одобравања кредита постиже четврти најбољи резултат на хакатону *Нове љубљанске банке 2021*. године. Године 2022. је добио награду *Kaggle ML Research Spotlight* компаније *Kaggle*. Од 2021. године ради као аутор, а од 2022. као уредник техничких чланака из области вештачке интелигенције и рачунарских наука у *Baeldung*-у. Добитник је више награда из области књижевности. Заступљен је у антологијама и књижевној периодици, а 2021. године његову драму „*Европеја – пут малог народа у Европу*”, објавила је издавачка кућа Кораци.

## Изјава о ауторству

Име и преизме аутора: Милош Симић

Број индекса: 46/2018

### Изјављујем

да је докторска дисертација под насловом

#### **Оптимизација алгоритамских тестова нормалности заснованих на неуронским мрежама**

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

У Београду,

Потпис аутора

---

---

## **Изјава о истоветности штампане и електронске верзије докторског рада**

Име и преизме аутора: Милош Симић

Број индекса: 46/2018

Студијски програм: Интелигентни системи

Наслов рада: Оптимизација алгоритамиких тестова нормалности заснованих на неуронским мрежама

Ментори: др Зорица Станимировић и др Срђан Станковић

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

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

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

У Београду,

**Потпис аутора**

---

---

## **Изјава о коришћењу**

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

### **Оптимизација алгоритамских тестова нормалности заснованих на неуронским мрежама**

која је моје ауторско дело.

Дисертацију са свим прилозима предао/ла сам у електронском формату погодном за трајно архивирање. Моју докторску дисертацију похрањену у Дигиталном репозиторијуму Универзитета у Београду и доступну у отвореном приступу могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons) за коју сам се одлучио/ла.

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

(Молимо да заокружите само једну од шест понуђених лиценци. Кратак опис лиценци је саставни део ове изјаве).

У Београду,

**Потпис аутора**

---

---

1. **Ауторство.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце, чак и у комерцијалне сврхе. Ово је најслободнија од свих лиценци.

2. **Ауторство – некомерцијално.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела.

3. **Ауторство – некомерцијално – без прерада.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела. У односу на све остале лиценце, овом лиценцом се ограничава највећи обим права коришћења дела.

4. **Ауторство – некомерцијално – делити под истим условима.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца не дозвољава комерцијалну употребу дела и прерада.

5. **Ауторство – без прерада.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца дозвољава комерцијалну употребу дела.

6. **Ауторство – делити под истим условима.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца дозвољава комерцијалну употребу дела и прерада. Слична је софтверским лиценцама, односно лиценцама отвореног кода.