KAUNO TECHNOLOGIJOS UNIVERSITETAS
MATEMATIKOS IR GAMTOS MOKSLŲ FAKULTETAS

Jovilė Grėbliauskaitė

ILGĄ LAIKĄ STEBIMŲ PROCESŲ PARAMETRŲ TAIKYMAS KLASIFIKAVIME

Baigiamasis magistro projektas

Vadovas
Doc. dr. Vytautas Janilionis

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS
MATEMATIKOS IR GAMTOS MOKSLŲ FAKULTETAS

ILGĄ LAIKĄ STEBIMŲ PROCESŲ PARAMETRŲ TAIKYMAS KLASIFIKAVIME

Baigiamasis magistro projektas
Taikomoji matematika (kodas 621G10003)

Vadovas
Doc. dr. Vytautas Janilionis

Recenzentas
Prof. dr. Viktoras Šaferis

Projektą atliko
Jovilė Grėbliauskaitė

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS
MATEMATIKOS IR GAMTOS MOKSLŲ FAKULTETAS
Jovilė Grėbliauskaitė
Taikomoji matematika (621G10003)

Baigiamojo projekto „Ilgą laiką stebimų procesų parametrų taikymas klasifikavime“

AKADEMINIO SĄŽININGUMO DEKLARACIJA

2015 m. gegužės mėn. 22 d.
Kaunas

Patvirtinu, kad mano, **Jovilės Grėbliauskaitės,** baigiamasis darbas tema „Ilgą laiką stebimų procesų parametrų taikymas klasifikavime“ yra parašytas visiškai savarankiškai, o visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena darbo dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymu nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

_____ _____

(studento vardas ir pavardė, įrašyti ranka) (parašas)

TURINYS

Grebliauskaite, J. Application of Long-Term Monitored Processes Parameters in Classification. Master‘s work in applied mathematics / supervisor assoc. prof. dr. V. Janilionis; Department of Applied mathematics, Faculty of Mathematics and Natural Sciences, Kaunas University of Technology. – Kaunas, 2015. – 41 p.

SUMMARY

The technology of today allows us to monitor various processes for a long time. One of such examples are the monitoring signals obtained from human wearing sensors. Such sensors enable users to capture and accumulate the human heart, respiratory and other related process signals for a long time. Therefore, the creation of methods and techniques that can handle large volumes of data and enable early detection of a variety of health-related changes, becomes a relevant task.

This paper addresses application of long-time monitored processes parameters in classification task to witch it is offered a methodology that allows the classification to use the calculated parameters of the Detrended Fluctuation Analysis method.

For methodology implementation purposes three programs were created: MATLAB function for application of Detrended Fluctuation Analysis and assessment of method appropriateness for the data, SAS macro for analysing self-similarity parameter differences between the groups, and R code for classification trees implementation and comparison. Proposed methodology was applied to electrocardiogram’s RR interval sequences to detect congestive heart failure.

Analysis has shown, that Detrended Fluctuation Analysis method was appropriate for the data. Means of two self-similarity parameters shown significant differences within the groups. The C5.0 algorithm classification tree, based on self-similarity parameters and age attribute, shown the best fit for analysed data and classified 91% of the data correctly and 83% of the congestive heart failure data correctly.

SANTRUMPOS

CHF	Stazinis širdies nepakankamumas (angl. *Congestive heart failure*)
DFA	Linkmės eliminavimo fliuktuacinė analizė (angl. *Detrended fluctuation analysis)*
EKG	Elektrokardiograma
NSR	Normalus sinusinis ritmas
RR	Intervalas tarp dviejų širdies susitraukimų

ĮŽANGA

Šiuolaikinės technologijos leidžia stebėti įvairius procesus ilgą laiką. Vienas iš tokios stebėsenos pavyzdžių yra signalai, gaunami iš žmogaus dėvimų jutiklių. Tokie jutikliai leidžia fiksuoti ir kaupti žmogaus širdies veiklos, kvėpavimo ir panašių procesų signalus ilgą laiką. Todėl aktualus metodų ir metodikų, galinčių apdoroti didelius duomenų kiekius ir leidžiančių kuo anksčiau aptikti įvairius su sveikata susijusius pakitimus, kūrimo uždavinys.

Darbo tikslas – pasiūlyti metodiką ilgą laiką stebimų procesų sekų informacijos panaudojimui klasifikavime. Sukurti pasiūlytos metodikos realizaciją programinėmis priemonėms. Taikant pasiūlytą metodiką, sukurti klasifikavimo modelį, leidžiantį identifikuoti širdies veiklos sutrikimus (stazinį širdies nepakankamumą) iš žmogaus dėvimų jutiklių gautų elektrokardiogramos RR intervalų sekų.

Sprendžiant ilgą laiką stebimų procesų parametrų taikymo klasifikavime uždavinį yra apjungiami linkmės eliminavimo fliuktuacinės analizės ir klasifikavimo metodai. Tai leidžia proceso stebėjimo ilgą reikšmių seką apibūdinti kelias parametrais, kurie įtraukiami į klasifikavimo modelį, be šių parametrų naudojantį ir kitus kintamuosius.

Pirmojoje darbo dalyje pateikta klasifikavimo metodų ir linkmės eliminavimo fliuktuacinės analizės taikymo literatūros apžvalga, bei sekos savipanašumo tyrimo uždavinys. Antrojoje dalyje pateikta metodika, leidžianti apibūdinti ilgą duomenų seką kelias parametrais, ir juos naudojantis klasifikavimo modelis, aprašomos šios metodikos taikymui pasirinktos programinės įrangos. Trečiojoje dalyje aprašytos sukurtos programinės priemonės ir pasiūlytos metodikos taikymas stazinio širdies nepakankamumo identifikavimui iš žmogaus dėvimų jutiklių gaunamų elektrokardiogramos RR intervalų sekų.

Darbo tematika buvo perskaityti pranešimai dvejose konferencijose:

- konferencijoje „Matematika ir matematikos dėstymas 2015" – KTU;
- XIII studentų konferencijoje „Matematika ir gamtos mokslai: teorija ir taikymas" – KTU;

ir paskelbta publikacija konferencijos pranešimų medžiagoje:

- Grėbliauskaitė J., Janilionis V. Elektrokardiogramos RR intervalų sekų panašumo į save tyrimas. Matematika ir gamtos mokslai: teorija ir taikymas. XIII studentų konferencijos pranešimų medžiaga. Kauno technologijos universitetas. Kaunas: Technologija, 2015. ISBN 9786090211342. p. 30-31.

1 TEORINĖ DALIS

Šioje dalyje pateikiama klasifikavimo metodų ir linkmės eliminavimo fliuktuacinės analizės taikymo literatūros apžvalgą, bei sekos savipanašumo tyrimo uždavinys

1.1 KLASIFIKAVIMO METODŲ APŽVALGA

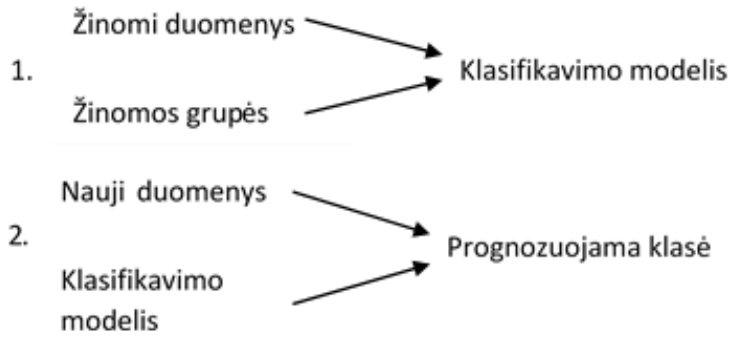
Klasifikavimas, kitaip dar vadinamas mokymu su mokytoju (angl. *supervised learning*). Pagrindinis klasifikavimo tikslas yra nustatyti taisykles, aptinkant vidinę duomenų struktūrą aprašančią informaciją, kurių pagalba objektas pagal savo savybes galėtų būti priskiriamas kuriai nors grupei [1, 2, 3].

Su klasifikavimu susiduriame daugelyje sričių, pavyzdžiui: bankas pagal įvairius rodiklius klasifikuoja klientus į mokius ir nemokius, paskolas į rizikingas ir nerizikingas, gydytojas, remdamasis simptomais siekia nustatyti kokia liga serga pacientas, archeologai pagal kaulų dydį ir formą siekia identifikuoti palaikų lytį, gyvi organizmai pagal ląstelių tipą, struktūrą bei mitybos poreikius skirstomi į penkias karalystes.

Visais aprašytais atvejais pagal tam tikrus objektų požymius bandoma nuspėti kategorinės reikšmės kintamąjį – objekto klasę. Klasių skaičius yra žinomas iš anksto.

Literatūroje [3] yra išskiriami du pagrindiniai klasifikavimo etapai (1.1 pav.):

1) Modelio sudarymas (apmokymas).
2) Modelio naudojimas imties elementų klasės prognozei.



1.1 pav. Klasifikavimo etapai

Pirmajame – modelio konstravimo – etape duomenys padalinami į modelio apmokymo ir modelio testavimo imtis. Kiekvienas modelio apmokymo duomenų aibės stebėjimas X yra reprezentuojamas n-mačių vektoriumi $(x_1, x_n, \dots x_n)$, kurio reikšmės yra atitinkamai požymių $A_1, A_2, \dots A_n$ matavimai. Kiekvienas stebėjimas taip pat turi klasę nurodantį kategorinį kintamąjį [3]. Pagal modelio apmokymo imtį sudaromos klasifikavimo taisyklės, kurios pagal stebėjimų reikšmes prognozuoja jų klasės kintamojo reikšmę.

Diskriminantinė analizė

Tai grupė metodų, skirtų spręsti klasifikavimo uždavinį. Tarkime, objektų populiaciją sudaro g grupių. Turime n diskriminavimo kintamųjų: $A_1, A_2, \dots A_n$. Imties duomenis sudaro stebėjimai x_{ijk}, $i = \overline{1,n}$, $j = \overline{1,g}$, $k = \overline{1,n_j}$, čia n_j- j-tosios grupės stebėjimų skaičius. Diskriminantinės analizės duomenys pateikti 1.1 lentelėje.

1.1 lentelė

Diskriminantinės analizės duomenys

Grupė	Diskriminavimo kintamieji			
	A_1	A_2	…	A_n
1	x_{111} x_{112} … x_{11n_1}	x_{211} x_{212} … x_{21n_1}	…	x_{n11} x_{n12} … x_{n1n_1}
2	x_{111} x_{112} … x_{11n_2}	x_{221} x_{222} … x_{22n_2}	…	x_{n21} x_{n22} … x_{n2n_2}
…	…	…	…	…
n	x_{1g1} x_{1g2} … x_{1gn_g}	x_{2g1} x_{2g2} … x_{2gn_g}	…	x_{ng1} x_{ng2} … x_{ngn_g}

Diskriminantinės analizės prielaidos, pateikto [4] literatūroje:

- Grupių skaičius yra baigtinis.
- Grupės nepriklausomos ir neturi bendrų objektų.
- Diskriminavimo kintamieji nepriklausomi, pasiskirstę pagal normalųjį skirstinį ir matuojami intervalų skalėje.
- Diskriminavimo kintamieji nėra kitų diskriminavimo kintamųjų tiesinės daugdaros.

- Diskriminavimo kintamųjų kovariacijos matricos grupėse lygios.

Diskriminantinėje analizėje yra skaičiuojamas atstumas tarp dviejų imčių vidurkių ir tikrinama ar šis atstumas statistiškai reikšmingai skiriais nuo nulio. Objektams klasifikuoti į grupes yre sudaroma Fišerio arba panašios funkcijos.

Jeigu yra netenkinamos kintamųjų normalumo sąlygos siūloma naudoti logistinę regresiją [4].

Logistinė regresija

Logistinė regresija prognozuoja kategorinio, klasę nurodančio kintamojo reikšmių tikimybes. Pagal grupės kintamojo galimų įgyti reikšmių skaičių logistinė regresija skirstoma į dvinarę, kai grupių skaičius yra 2, ir daugianarę, kai grupių skaičius > 2.

Dvinarėje logistinėje regresijoje priklausomas kintamasis Y gali įgyti dvi reikšmes: 0 ir 1. Kai stebėjimas X yra reprezentuojamas n-mačių vektoriumi $\{x_1, x_n, \dots\ x_n\}$, tai matematinis modelis, kad prognozuojamas klasės kintamasis įgis reikšmę lygią vienetui [4, 5]:

$$P(Y = 1|X) = \frac{e^{z(X)}}{1 + e^{z(X)}},\ \ z(X) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n. \qquad \textbf{(1.1)}$$

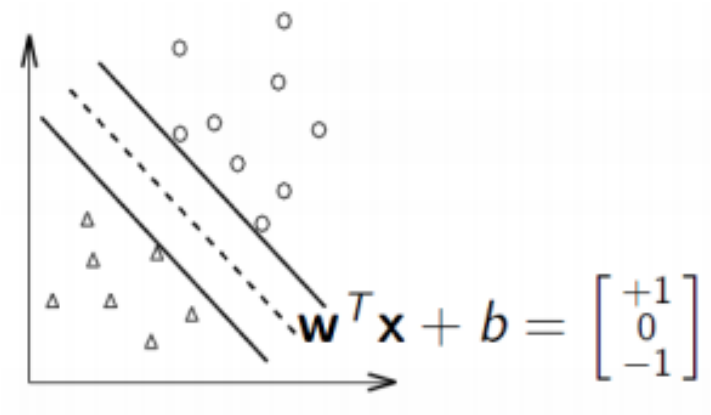
Koeficientų $\beta_0, \beta_1,\ \beta_2, \dots, \beta_n$ įverčiai $\hat{\beta}_0, \hat{\beta}_1,\ \hat{\beta}_2, \dots, \hat{\beta}_n$ randami iš imties duomenų didžiausio tikėtinumo metodu.

Dvinarės logistinės regresijos modelis naudojamas klasių prognozavimui:

- Jei $\hat{P}(Y = 1|X) > 0{,}5$, tai prognozuojama Y reikšmė lygi 1.
- Jei $\hat{P}(Y = 1|X) < 0{,}5$, tai prognozuojama Y reikšmė lygi 0.
- Jei $\hat{P}(Y = 1|X) = 0{,}5$, tai prognozuojamą Y reikšmę siūloma pasirinkti atsitiktinai.

Atraminių vektorių metodas

Atraminių vektorių metodas ieško grupių duomenis atskiriančios hiperplokštumos su kiek galima didesniu atstumu tarp klasifikuojamųjų grupių. Jei pradiniai duomenys neatsiskiria hiperplokštuma, algoritmas transformuoja pradinius duomenis į didesnę dimensiją[3, 6]. 1.2 paveiksle pavaizduotas dvimatis atvejis: tiese atskiriamų plokštumos taškų pavyzdys.



1.2 pav. Tiese atskiriamų plokštumos taškų pavyzdys

Matematiškai taškus skiriantis paviršius užrašomas:

$$\mathrm{w}^{\mathrm{T}}\mathrm{X} + \mathrm{b} = 0, \tag{1.2}$$

čia X – sutartinis tašas, o vektoriaus w ir konstantos b reikšmės apskaičiuojami iš apmokymo duomenų.

Tarkime, turime apmokymo duomenų imtį $(X_i,\ Y_i)$, $i = \overline{1, L}$, $Y_i = \{-1, 1\}$. Klasifikuojant siekiama, kad $\mathrm{w}^{\mathrm{T}}X_i + \mathrm{b} < 0$, kai $Y_i = -1$ ir $w^{\mathrm{T}}X_i + \mathrm{b} > 0$, kai $Y_i = 1$.

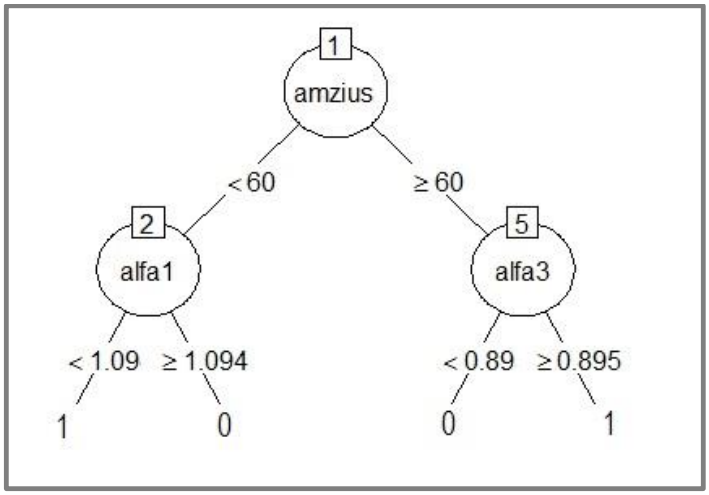
Imties klasifikavimui naudojama klasifikavimo taisyklė f:

$$f(\vec{\mathrm{x}}_i) = \mathrm{sgn}(w^{\mathrm{T}}X_i + \mathrm{b}) \tag{1.3}$$

Skiriamasis paviršius yra konstruojamas naudojant tik tuos taškus kurie nuo skiriamosios linijos nutolę atstumu $\frac{1}{\|\mathrm{w}\|}$, čia $\|\mathrm{w}\|$ – vektoriaus w Euklido norma. Šie taškai yra vadinami atraminiais vektoriais ir yra vieninteliai elementai apmokymo duomenų rinkinyje, kurie daro įtaką klasifikavimo plokštumos radimui,

Klasifikavimo medžiai

Šio metodo rezultatas į medį panaši struktūra, kurią sudaro viršūnės (mazgai) sujungtos perėjimo rodyklėmis, vadinamomis šakomis. Kiekviena viršūnė žymi tam tikrą testą, o iš viršūnės išeinančios perėjimo rodyklės galimas testo reikšmes (klasifikavimo medžio pavyzdys pateiktas 1.3 paveiksle). Sprendimų medžių privalumas — jų aiškumas, grafinio vaizdavimo galimybė, panašumas į žmogaus priimamų sprendimų seką ir galimybė klasifikavimui kartu naudoti diskrečiuosius ir tolydžiuosius kintamuosius. [2].



1.3 pav. Klasifikavimo medžio pavyzdys

Klasifikavimo medžio konstravimą sudaro du žingsniai:

- medžio auginimas (angl. *tree building*). Šiame žingsnyje kiekvienoje viršūnėje parenkant testą yra sudaromas klasifikavimo medis.
- medžio genėjimas (angl. *tree pruning*). Šiame žingsnyje siekiamą klasifikavimo medį supaprastinti.

Klasifikavimo modelio vertinimas

Klasifikavimo modelio vertinimui, pagal sprendžiamo uždavinio specifiką, gali būti naudojami įvairūs metodai. Siekiant įvertinti klasifikavimo modelį visų pirma reikia susidaryti klasifikavimo lentelę 1.2. Klasifikavimo lentelėje:

- TP – klasei 1 priskirtų jos objektų kiekis;
- FN – klasei 1 priskirtų klasės 0 objektų kiekis
- FP – klasei 0 priskirtų klasės 1 objektų kiekis;
- TN – klasei 0 priskirtų jos objektų kiekis;
- P – klasės 1 objektų kiekis;
- N – klasės 0 objektų kiekis;
- P' – klasei 1 priskiriamų objektų kiekis;
- N' – klasei 1 priskiriamų objektų kiekis;
- P + N – visų objektų kiekis.

1.2 lentelė

Klasifikavimo lentelė

		Prognozuojama klasė		Viso:
		1	**0**	
Tikroji klasė	**1**	TP	FN	P
	0	FP	TN	N
Viso:		P'	N'	P + N

Klasifikavimo tikslumo įvertinimui naudojamos metrikos [3] pateiktos 1.3 lentelėje.

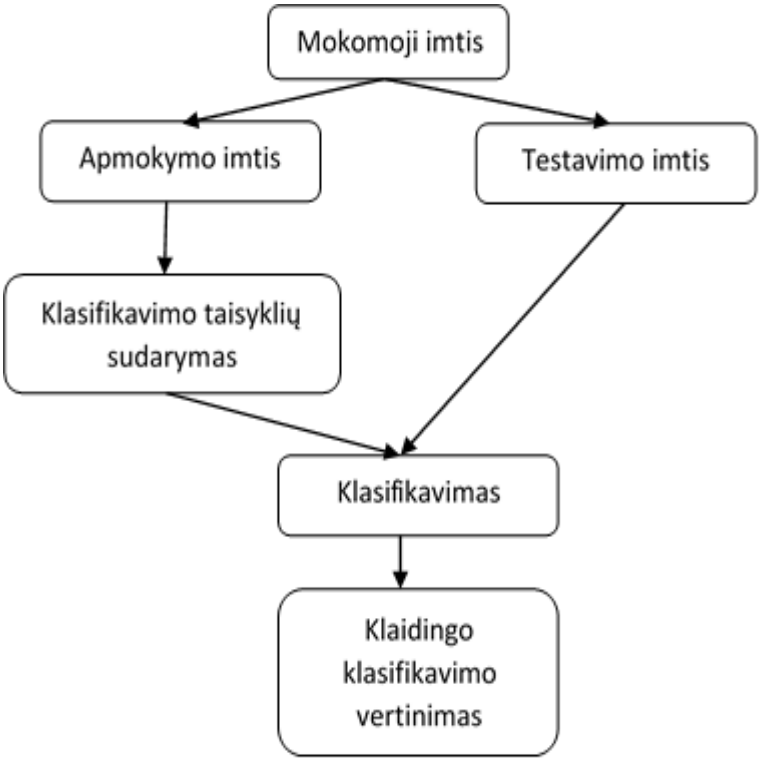
1.3 lentelė

Klasifikavimo tikslumo vertinimo metrikos

Metrikos pavadinimas	Formulė	Siekiama
Teisingai suklasifikuotų objektų dalis (angl. *recognition rate)*	$\frac{TP + TN}{P + N}$	Maksimizuoti
Klaidos santykis (angl. *error rate*)	$\frac{FP + FN}{P + N}$	Minimizuoti
Jautrumas (angl. *sensitivity*)	$\frac{TP}{P}$	Maksimizuoti
Specifiškumas (angl. *specificity*)	$\frac{TN}{N}$	Minimizuoti
Tikslumas (angl. *precision*)	$\frac{TP}{TP + FP}$	Maksimizuoti

Klasifikavimo medžio vertinimui taip pat yra naudojama testavimo imtis (ši imtis yra mokomosios imties dalis, kuri nenaudojama modelio apmokymo žingsnyje). Testavimo imties duomenims, pritaikius iš modelio apmokymo imties duomenų sudarytas klasifikavimo taisykles, pagal jų požymių vektorius priskiriamos klasės kintamojo reikšmės. Medžio tikslumui vertinti sudaroma 1.2 klasifikavimo lentelė ir naudojamos 1.3. lentelėje pateiktos metrikos.

Modelio vertinimo, padalinant mokomąją imtį į apmokymo ir testavimo imtis, schema [7] pateikiama 1.4 pav.



1.4 pav. Klaidingo klasifikavimo vertinimas

Procesų stebėsenos informacijos panaudojimo klasifikavime uždavinys

Klasifikavimui reikalingas $n, n \epsilon \mathbb{N}$ požymių vektorius $X = (x_1, x_2, \dots, x_n)$, pagal kurio kintamųjų reikšmes būtų galima prognozuoti objekto priklausomybę vienai iš galimų klasių. Kartais objektą galima apibūdinti ir kokio nors jo proceso stebėsenos duomenų seka. Pavyzdžiui, tiriamą žmogų galima apibūdinti jo elektrokardiogramos signalu. Siekiant įtraukti tokių signalų (duomenų sekų) informaciją į bendrą klasifikavimo algoritmą, reikia signalą apibūdinti tolydžiais ar diskrečiais parametrais.

1.2 SEKŲ SAVIPANAŠUMO UŽDAVINYS

Norint patikrinti, ar seka yra savipanaši reikia paimti sekos segmentą, transformuoti jį naudojant didinimo parametrus taip, kad segmento mastelis būtų vienodas su pradinės sekos masteliu ir tada palyginti šių dviejų objektų statistines savybes. Siekiant tinkamai palyginti minėtas sekas reikia dviejų mastelio keitimo parametrų:

- Vertikalios ašies mastelio keitimo parametro.
- Horizontalios ašies mastelio keitimas parametro.

Matematiškai laiko eilutė yra vadinama savipanašia, jeigu tenkinama lygybė [8]:

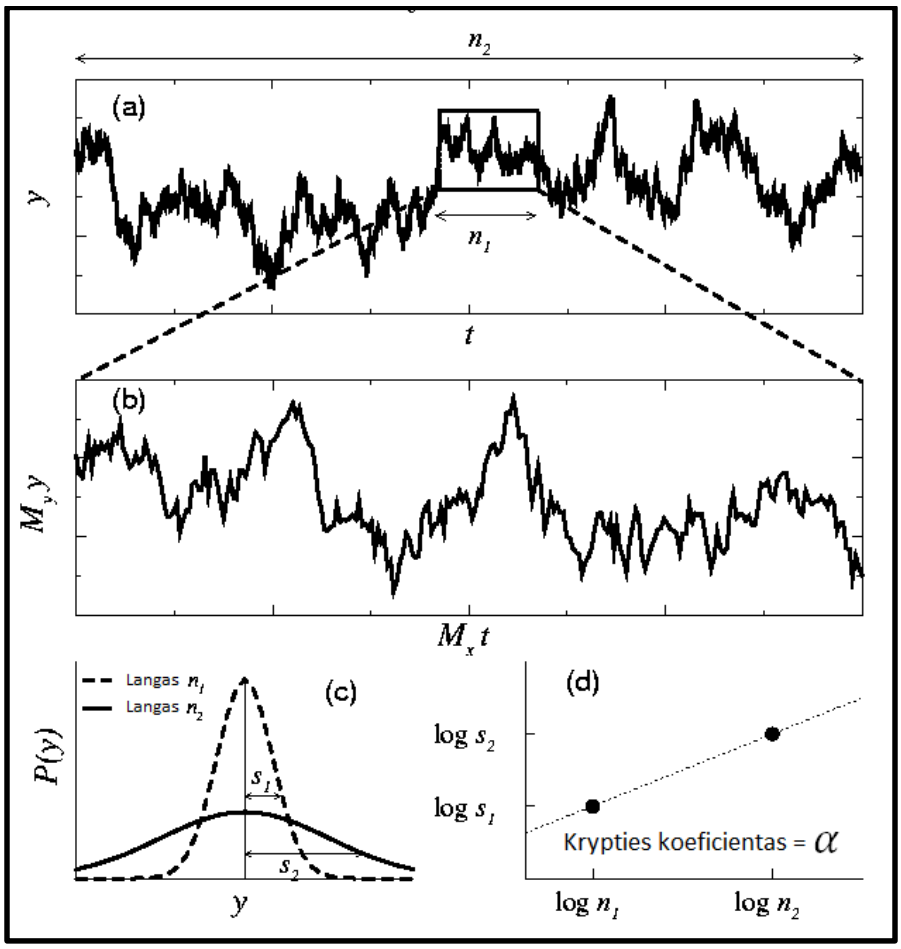
$$y(t) \stackrel{d}{\equiv} a^{\alpha} y\left(\frac{t}{a}\right), \tag{1.4}$$

čia $\stackrel{d}{\equiv}$ reiškia, kad statistinės savybė abejuose lygybės pusėse yra vienodos. Kitais žodžiais tariant, savipanašus procesas $y(t)$, su parametru α, turi tokį patį tikimybinį pasiskirstymą, kaip ir pakeisto mastelio procesas $a^{\alpha} y\left(\frac{t}{a}\right)$. Čia x ašis pakeičiama $t \rightarrow \frac{t}{a}$ ir y ašis $y \rightarrow a^{\alpha} y$. Laipsnio rodiklis α yra vadinamas savipanašumo arba mastelio parametru [8].

Praktikoje yra sunku nustatyti ar procesai yra statistiškai vienodi (tam reikia, kad procesų pasiskirstymo funkcijos būtų vienodos – sutaptų visi momentai), todėl naudojami silpni kriterijai (lyginami vidurkiai ir dispersijos) 1.4 formulės abiejų pusių statistinei lygybei įvertinti [8]. 1.5 paveiksle pavaizduota sekos savipanašumo idėja. Teisingai transformavus x ir y ašis galima gauti seką, panašią į pradinę. Savipanašumo parametras tada gali būti apskaičiuojamas:

$$\alpha = \frac{\ln \mathrm{M_y}}{\ln \mathrm{M_x}} \tag{1.5}$$

čia M_x ir M_y yra x ir y ašių mastelių didinimo parametrai.



1.5 pav. Savipanašios sekos idėja [8]; (a) n_1 ir n_2 – skirtingų mastelių langų dydžiai; (b) mažesniojo mastelio padidintas vaizdas, didinimo parametrai M_x ir M_y (c) tikimybinis dydžio y pasiskirstymas, s_1 ir s_2 – standartiniai nuokrypiai, (d) logaritminiame tinklelyje atidėti standartiniai nuokrypiai prie langų pločių, su kuriais jie gauti

Praktikoje savipanašumo parametras α dažniausiai nežinomas ir yra siekiama jį surasti iš turimų duomenų. Abscisių ašies didinimo parametras yra apskaičiuojamas pagal formulę:

$$M_x = \frac{n_2}{n_1}, \tag{1.6}$$

čia n_1 ir n_2 yra pradinės sekos ir segmento didumai atitinkamai.

Siekiant surasti y ašies didinimo parametrą reikia nusibraižyti pradinės sekos ir jos segmento histogramas. Tada y ašies didinimo parametras randamas iš standartinių nuokrypių santykio:

$$M_y = \frac{s_2}{s_1} \tag{1.7}$$

Įstatę (1.6) ir (1.7) išraiškas į (1.5) formulę gauname:

$$\alpha = \frac{\ln M_y}{\ln M_x} = \frac{\ln s_2 - \ln s_1}{\ln n_2 - \ln n_1} \quad \textbf{(1.8)}$$

(1.8) išraiškoje santykis yra tiesės jungiančios taškus $(\log n_1 , \log s_1)$ ir $(\log n_2 , \log s_2)$ krypties koeficientas.

Vienas iš sekos savipanašumui įvertinti praktikoje naudojamų metodų, kuris yra tinkamas ir nestacionarioms laiko eilutėms – linkmės eliminavimo fliuktuacinės analizės metodas. Šį metodą 1994m. pasiūlė Peng ir kiti straipsnyje, kuris cituotas daugiau kaip 2000 kartų [9, 10].

1995m to paties autoriaus straipsnyje [11] DFA algoritmas buvo taikytas elektrokardiogramos RR intervalų sekų analizėje. Atliktas savipanašumo parametrų tyrimas sveikos širdies ir stazinio širdies nepakankamumo atvejais. Analizuojant RR intervalų sekas pasiūlyta išskirti du savipanašumo parametrus: trumpalaikį α_1 ir ilgalaikį α_2. Atliktas tyrimas su 12-ka sveikų suaugusiųjų (amžius 29 - 64, amžiaus vidurkis 44) ir 15-ka suaugusiųjų turinčių širdies nepakankamumą (amžius 22 - 71, amžiaus vidurkis 56). Įrodyta, kad šių grupių savipanašumo parametrų α_1 ir α_2 vidurkiai statistiškai reikšmingai skiriasi tarp grupių.

Straipsnyje [12], analizuojant EKG RR intervalus DFA algoritmu, pastebėti savipanašumo parametrų skirtumai, susiję su tiriamųjų amžiumi bei aritmija [12]. Analizuojat žmogaus eiseną, pastebėti savipanašumo parametrų skirtumai tarp sveikų ir Huntingtono liga sergančių pacientų [12].

DFA algoritmas [13] literatūroje naudotas miego stadijoms ir miego sutrikimams diagnozuoti [13]. Straipsnyje DFA metodas taikytas elektroencefalogramos (EEG) duomenų sekoms. Parodyta, kad miego apnėja sergančiųjų elektroencefalogramos savipanašumo parametrai statistiškai reikšmingai (su reikšmingumo lygmeniu 0.001) didesnės už sveikų žmonių. Taip pat analizuoti savipanašumo parametrų pokyčiai skirtingų miego stadijų metu. Pastebėta, kad antroje miego stadijoje savipanašumo parametrų reikšmės padidėja, REM miego stadijoje – sumažėja.

2011 metais apginta daktaro disertacija [14], kurioje vienas iš daugelio metodų miego stadijoms atpažinti iš elektrokardiogramos RR intervalų sekų, yra naudojamas DFA algoritmas.

1.3 DARBO TIKSLAS IR UŽDAVINIAI

Darbo tikslas – pasiūlyti ir programiškai realizuoti metodiką ilgą laiką stebimų procesų sekų informacijos panaudojimui klasifikavime.

Sprendžiami uždaviniai:

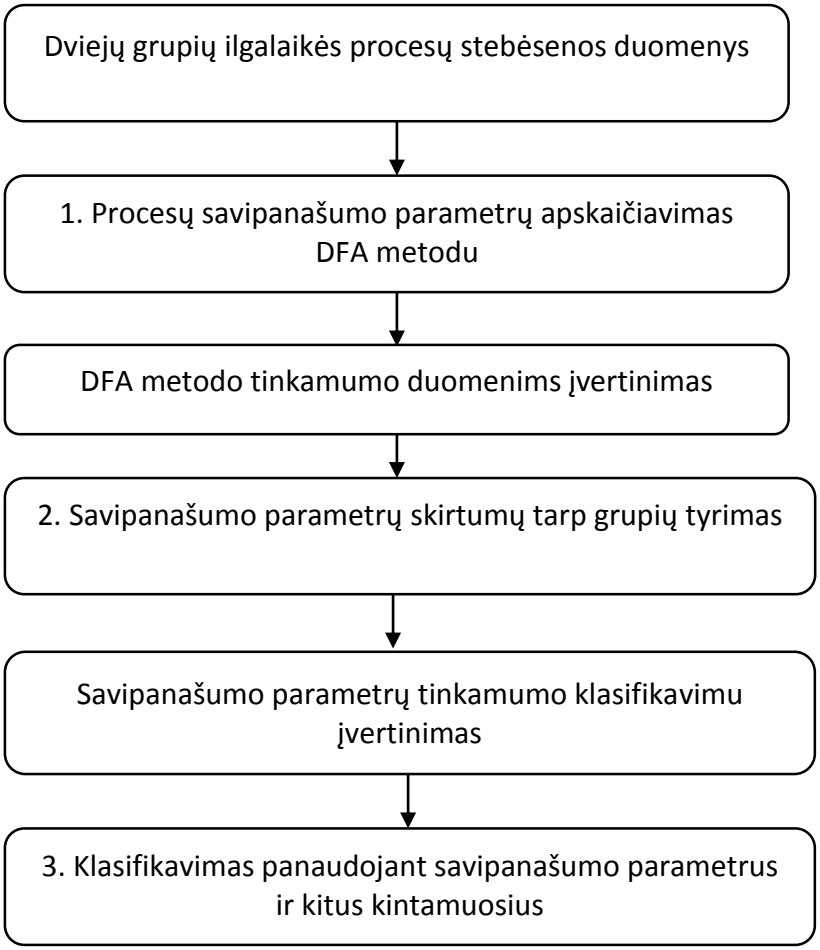
- apžvelgti klasifikavimo metodus, bei su sekos savipanašumo parametrų taikymu susijusią literatūrą;
- pasiūlyti metodiką ilgą laiką stebimų procesų sekų informacijos panaudojimui klasifikavime;
- sukurti programines priemones pasiūlytos metodikos realizavimui, parengti sukurtų programinių priemonių taikymo rekomendacijas;
- pritaikyti pasiūlytą metodiką klasifikavimui, kuris leistų identifikuoti stazinį širdies nepakankamumą iš žmogaus dėvimų jutiklių registruojamų elektrokardiogramos RR intervalų sekų informacijos.

2 TYRIMŲ METODIKA

Šiame skyriuje pateikiama metodiką, leidžianti apibūdinti ilgą duomenų seką kelias parametrais, bei šiuos parametrus naudojantis klasifikavimo moetodas. Aprašomos metodikos taikymui pasirinktos programinės įrangos.

2.1 ILGĄ LAIKĄ STEBIMŲ PROCESŲ PARAMETRŲ TAIKYMO KLASIFIKAVIME METODIKA

Siūloma ilgalaikių procesų stebėsenos savipanašumo parametrus taikyti klasifikavime, kartu su kitais kintamaisiais. Metodikos schema pavaizduota 2.1 paveiksle.



2.1 pav. Ilgalaikės procesų stebėsenos parametrų taikymo klasifikavime metodika

Siūloma klasifikavime panaudoti sekos savipanašumo parametrus, gaunamus DFA metodu. DFA metodo aprašymas ir tinkamumo duomenims vertinimo principai aprašomi 2.1.1 poskyryje.

2.1.2 poskyryje pateikiamas vienfaktorinės dispersinės analizės modelių, skirtų analizuoti savipanašumo parametrų reikšmių skirtumus tarp grupių, aprašymas.

Klasifikavimui atlikti pasirinkti klasifikavimo medžiai, dėl galimybės kartu naudoti diskrečiuosius ir tolydžiuosius kintamuosius. Klasifikavimo medžių sudarymo algoritmai aprašomi 2.1.3 poskyryje.

2.1.1 LINKMĖS ELIMINAVIMO FLIUKTUACINĖ ANALIZĖ

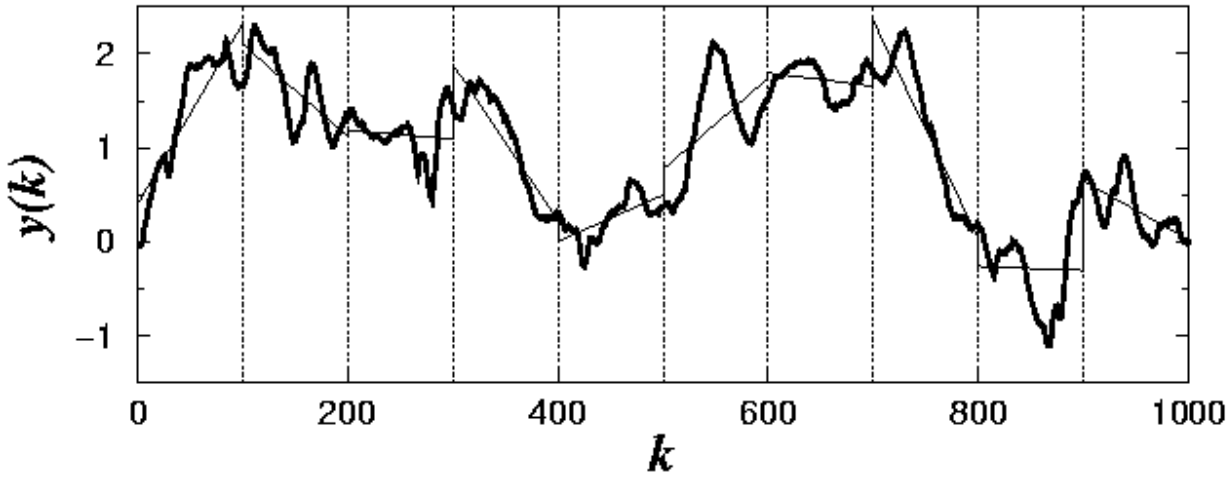
Šiame skyriuje aprašysime sekos savipanašumui tirti naudojamo DFA metodo algoritmą, pateiktą [11] literatūroje.

Tarkime, turime reikšmių seką $\boldsymbol{x_1, x_2, \dots, x_N}$. Prieš atliekant DFA, reikšmių seka yra integruojama:

$$y(k) = \sum_{i=1}^{k} (x_i - \mu), \ \ k = \overline{1, N}, \tag{2.1}$$

čia $\boldsymbol{x_i}$ – i-tasis sekos elementas, μ - sekos vidurkis, N – elementų skaičius sekoje.

DFA algoritmas susideda iš trijų žingsnių. Pirmajame žingsnyje integruota seka padalinama į t vienodo pločio (plotis n) nepersidengiančių segmentų. Antrajame žingsnyje kiekviename segmente mažiausių kvadratų metodu randamas tiesinis trendas, kurio reikšmė taške k yra žymima $y_n(k)$. 2.2 paveiksle pavaizduotas integruotos sekos pavyzdys (integruota EKG RR intervalų ilgių seka), padalinta į 10 nepersidengiančių intervalų po 100 elementų, bei sekos lokalūs trendai .

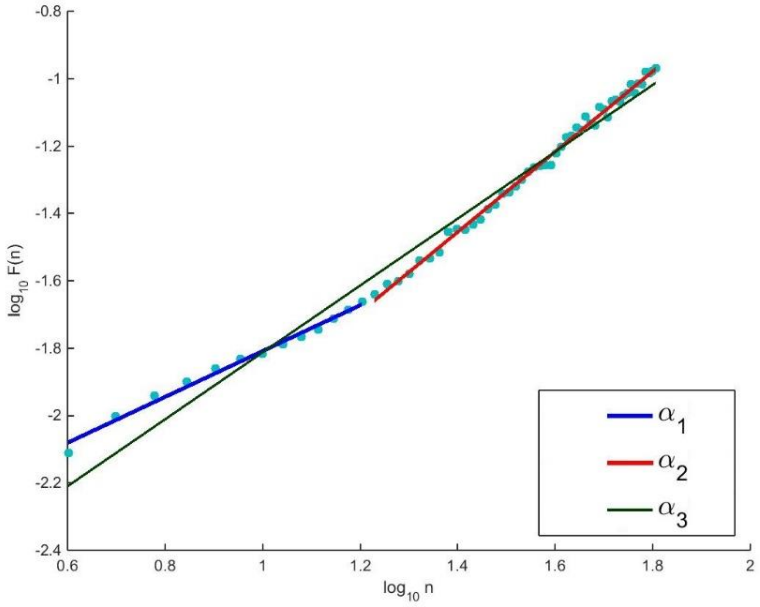


2.2 pav. Lokalūs trendai integruotos sekos segmentuose, $n = 100$

Trečiajame žingsnyje apskaičiuojama vidutinė fliuktuacija $F(n)$, kai segmento plotis lygus n:

$$F(n) = \sqrt{\frac{1}{N}\sum_{k=1}^{N}[y(k) - y_n(k)]^2}. \quad \textbf{(2.2)}$$

Pirmasis, antrasis ir trečiasis žingsniai kartojami pasirenkant skirtingas segmentų pločio reikšmes. Taip yra gaunama $F(n)$ priklausomybė nuo segmentų pločio n. Šią priklausomybę įprasta vaizduoti logaritminiame tinklelyje. Priklausomybės logaritminiame tinklelyje pavyzdys pavaizduotas 2.3 paveiksle.



2.3 pav. Linkmės eliminavimo fliuktuacinės analizės $F(n)$ ir n priklausomybės logaritminiame tinklelyje pavyzdys, savipanašumo parametrai α_1, α_2 ir α_3

Taikant DFA metodą elektrokardiogramos RR intervalų sekoms, priklausomybes logaritminiame tinklelyje siūloma išskirti dvi arba tris sritis: $n = \overline{4{:}\,16}$, $n = \overline{17{:}\,64}$, bei $n = \overline{4{:}\,64}$ [11, 15]. Bendru atvejų pirmosios ir antrosios srities atskyrimo tašką reikia parinkti taip, kad antroji išvestinė

$$\frac{d^2}{d(\log_{10} n)^2}\log_{10} F(n) \quad \textbf{(2.3)}$$

turėtų lokalų ekstremumą tame taške[15].

Kiekviename $F(n)$ ir n priklausomybės logaritminiame tinklelyje segmente mažiausių kvadratų metodu randamos geriausiai taškus aproksimuojančių tiesių lygtys, kurių krypties koeficientai α_1, α_2 ir α_3 – sekos savipanašumo parametrai.

Apie DFA metodo tinkamumą sprendžiama vizualiai iš $F(n)$ ir n priklausomybės logaritminiame tinklelyje, bei apibrėžtumo koeficientų (gaunamų ieškant geriausiai taškus aproksimuojančių tiesių) reikšmių.

Apibrėžtumo koeficientas

Tarkime turime taškus $\{(n_1, F_1), (n_2, F_2), \dots \ (n_k, F_k)\}$, kuriuos geriausiai aproksimuoja tiesė $\hat{F}(n_i) = \hat{\alpha} \cdot n_i + \hat{\beta}, \ \ i = \overline{1,k}$. Tada aproksimuojančios tiesės tikimo duomenims vertinimui naudojamas apibrėžtumo koeficientas apskaičiuojamas pagal formulę [4, 17]:

$$R^2 = \frac{\sum_{i=1}^{k}(\hat{F}(n_i) - \bar{F})^2}{\sum_{i=1}^{k}(F_i - \bar{F})^2} \qquad \textbf{(2.4)}$$

čia $\bar{F} = \frac{1}{k}\sum_{i=1}^{k} F_i$.

Kuo R^2 reikšmė didesnė, tuo labiau stebėjimai yra sukoncentruoti apie mažiausių kvadratų metodu gautą tiesę.

2.1.2 SAVIPANAŠUMO PARAMETRŲ SKIRTUMŲ TARP GRUPIŲ ANALIZĖ

Norint įvertinti, ar skiriasi savipanašumo parametrų reikšmės tarp grupių, tenka spręsti vienfaktorinės dispersinės analizės uždavinį. Tiriamos hipotezės:

$$H_0: \mu_{\alpha_i^{\langle 1 \rangle}} = \mu_{\alpha_i^{\langle 2 \rangle}}, \quad H_a: \mu_{\alpha_i^{\langle 1 \rangle}} \neq \mu_{\alpha_i^{\langle 1 \rangle}} \ i = \overline{1,3}. \qquad \textbf{(2.5)}$$

Čia $\mu_{\alpha_i^{\langle 1 \rangle}}$ yra savipanašumo parametro α_i pirmosios grupės vidurkis ir $\mu_{\alpha_i^{\langle 2 \rangle}}$ – savipanašumo parametro α_i antrosios grupės vidurkis.

Nulinėms hipotezėms tikrinimui yra naudojama Fišerio statistiką, kurios reikšmė šiuo atveju sutampa su Stjudento t statistikos reikšme. Jei Fišerio statistikos p reikšmė didesnė už pasirinktą reikšmingumo lygmenį, hipotezė apie vidurkių lygybė nėra atmetama.

Modelio prielaidos [16]:

- kintamieji pasiskirstę pagal normalųjį skirstinį (suderinamumo hipotezei tikrinti naudosime Šapiro-Vilko, Kolmogorovo-Smirnovo, Kramerio-fon Mises ir Andersono-Darlingo statistikas);
- lygios kintamųjų dispersijos (hipotezei apie dispersijų lygybę tikrinti naudosime Livyno kriterijų, kuris yra ne toks jautrus normalumo prielaidos pažeidimams).

2.1.3 KLASIFIKAVIMO MEDŽIŲ SUDARYMO ALGORITMAI

Darbe naudosime vieną iš populiariausių neparametrinių klasifikavimo metodų – klasifikavimo medžius [2]. Klasifikavimo medžių privalumas, kad daugumoje algoritmų tarp kintamųjų, naudojamų klasifikavimui, gali būti ir diskrečių ir tolydžių kintamųjų. Būtent ši savybė lėmė klasifikavimo medžių pasirinkimą.

Remiantis literatūra [3], darbe pasirinkti trys klasifikavimo algoritmai: ID3, C4.5 ir CART. Pateiksime šių algoritmų aprašymą.

Tegul aibė D, yra duomenų aibė, kurios elementų klasės yra žinomos. Klasės kintamasis gali įgyti m skirtingų reikšmių, kurios žymi m klasių $C_1, C_2, \dots C_n$. Tegul $C_{i,D}$ žymi duomenų aibės D elementus, priklausančius grupei C_i, o $\left|C_{i,D}\right|$ ir |D| –aibių $C_{i,D}$ ir D elementų kiekius atitinkamai.

ID3 algoritmas

Algoritmą 1986 metais pasiūlė J. R. Quinlan [17]. Šiame algoritme testai medžio mazguose parenkami remiantis informacijos teorijos sąvokomis, skaidymo požymio parinkimas paremtas informacijos išlošiu (angl. *Information gain*).

Aibės D informacija (kitaip dar vadinama entropija) apskaičiuojama pagal:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i), \tag{2.6}$$

čia p_i yra apriorinis klasės tikimybės įvertis apskaičiuojamas pagal: $p_i = \frac{|C_{i,D}|}{|D|}$.

Tarkime požymis A turi v skirtingų reikšmių $\{a_1, a_2, \dots a_v\}$. Požymis A gali būti naudojamas padalinti aibę D į poaibius $\{D_1, D_2, \dots D_v\}$. Aibėje D_j yra tie aibės D stebėjimai, kuriuose A reikšmė yra lygi a_j.

Požymio testo informacija apskaičiuojama pagal:

$$Info_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} Info(D_j). \quad \textbf{(2.7)}$$

Tada kiekvienam požymiui yra skaičiuojamas informacijos išlošis, žymintis informacijos kiekio sumažėjimą po aibės padalinimo pagal požymį A:

$$Gain(A) = Info(D) - Info_A(D). \quad \textbf{(2.8)}$$

Aibės padalinimui parenkamas tas požymis, kurio informacijos išlošio reikšmė yra didžiausia. ID3 algoritmas medžio genėjimo neatlieka.

C4.5 algoritmas

Šis algoritmas yra ID3 algoritmo išplėtimas[18]. Medžio auginimo etape C4.5 algoritme mazgo testo parinkimo kriterijus išplečiamas, įvedant santykinį informacijos išlošį (angl. *Gain Ratio*). Algoritme panaudojama padalinimo informacija:

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \log_2\left(\frac{|D_j|}{|D|}\right). \quad \textbf{(2.9)}$$

Informacijos išlošis apskaičiuojamas pagal:

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo(A, D)}. \quad \textbf{(2.10)}$$

Viršūnės testu parenkamas tas požymis, su kuriuo gaunama didžiausia $GainRatio$ reikšmė.

Jei požymis A yra tolydusis, mazgo testas turi pavidalą: $A \leq A^{threshold}$ ir turi tik dvi galimas baigtis. Sprendžiamas geriausiai grupes atskiriančio padalinimo taško $A^{threshold}$ radimo uždavinys. Jeigu požymių vektoriuje A yra k skirtingų reikšmių, tai siekiant surasti geriausio padalinimo tašką reikės atlikti $k - 1$ patikrinimų. Aibės A reikšmes reikia išrikiuoti didėjimo tvarka, tada padalinimo taškai parenkami taip:

$$\frac{a_i + a_{i+1}}{2}. \quad \textbf{(2.11)}$$

Kiekvienu atveju gautam padalinimo taškui apskaičiuojamas $Info_A(D)$, su padalinimų skaičiumi lygiu dviem ((2.7) išraiškoje $v = 2$). Mažiausią informacijos testo reikšmę turintis taškas parenkamas kaip padalinimo taškas. Gaunami du aibės D poaibiai : D_1, kuriame $A \leq A^{threshold}$, ir D_2, kuriame $A > A^{threshold}$.

Algoritmas C4.5 naudoja paklaidomis pagrįstą genėjimą (angl. *Error-based pruning*) [19], kuris atliekamas po klasifikavimo medžio sudarymo. Naudojamas paklaidos santykio, viršutinis pasikliautinojo intervalo rėžis:

$$\bar{\varepsilon}(T,S) = \varepsilon(T,S) + Z_\alpha \cdot \sqrt{\frac{\varepsilon(T,S)\cdot(1-\varepsilon(T,S))}{|S|}}, \quad \textbf{(2.12)}$$

čia $\varepsilon(T,S)$ – medžio T klaidingai klasifikuojamų stebėjimų ir visų klasifikuojamų stebėjimų S ($|S|$-stebėjimų skaičius), Z_α- standartinio normaliojo skirstinio α lygmens kvantilis.

Tegul $subtree(T,t)$ žymi medžio T pomedį, kurio pagrindinė viršūnė yra mazge t, $maxchild(T,t)$ – mazgas, į kuri patenka daugiausiai aibes S stebėjimų po viršūnėje t atlikto testo ir S_t – visi aibės S stebėjimai, patenkantys į viršūnę t. Tada, einant iš apačios į viršų, viršūnėse apskaičiuojami:

- $\bar{\varepsilon}(subtree(T,t), S_t)$;
- $\bar{\varepsilon}(pruned(subtree(T,t),t), S_t)$;
- $\bar{\varepsilon}\big(pruned(subtree(T, maxchild(T,t)), S_{maxchild(T,t)}\big)$;

Jei mažiausia reikšmę įgyja pirmoji išraiška – medis paliekamas toks koks yra, jei antroji – viršūnė t yra genėjama, jei trečioji – viršūnė t yra pakeičiama viršūne $maxchild(T,t)$.

CART algoritmas

CART algoritme padalinimo požymiui parinkti naudojamas Gini indeksas. Naudojant analogišką žymėjimą kaip ir aprašytuose anksčiau algoritmuose, aibės Gini indeksui apskaičiuoti naudojama formulė:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2 \quad \textbf{(2.13)}$$

čia p_i analogiškai – apriorinis klasės tikimybės įvertis apskaičiuojamas pagal: $p_i = \frac{|C_{i,D}|}{|D|}$. Gini indeksas atlieka binarinį požymių padalinimą – kiekvienas testas turi tik dvi galimas reikšmes.

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2) \quad \textbf{(2.14)}$$

Kiekvienam požymiui apskaičiuojamas $Gini_A(D)$ su visais galimais požymio reikšmių padalinimais į dvi aibes jeigu požymis diskretusis ir visais galimais padalinimo taškais, jei požymis tolydusis.

Aibės skaidymui parenkamas tas požymis, su kuriuo maksimizuojama išraiška:

$$\Delta Gini(A) = Gini(D) - Gini_A(D) \quad \textbf{(2.15)}$$

CART algoritmas taip pat naudoja genėjimo po klasifikavimo medžio sudarymo metodiką, pagrįstą medžio sudėtingumo vertinimu. Medžio sudėtingumas yra funkcija, priklausanti nuo lapų medyje skaičiaus ir klaidų santykio. Genėjimas atliekamas iš apačios į viršų. Kiekvienai vidinei viršūnei yra skaičiuojamas pomedžio sudėtingumas ir sudėtingumas, jeigu pomedis būtų apgenėtas (pakeistas lapu). Jei apgenėjus pomedį gaunama mažesnė sudėtingumo reikšmė nei pradinė – pomedis pakeičiamas lapu.

Sprendimų medžiams sudaryti yra naudojamas kryžminio patikrinimo metodas. Trečiojoje dalyje sprendžiamam uždaviniu naudosime kryžminio patikrinimo metodą. Padalinsime duomenų aibę į tris dalis.

Sudarydami sprendimų medį, pirmiausiai naudosime pirmą ir antrą duomenų aibes, o modeliui testuoti – trečiąją aibę. Tada medžiui sudaryti naudosime pirmą ir trečiąją duomenų aibes, o testuoti – antrąją. Ir trečiąjį kartą sprendimų medį sudarysime iš antrosios ir trečiosios duomenų aibių, o testuosime naudodami pirmąją. Sprendimų medžio sudarymo metodui įvertinti skaičiuosime klaidų kiekį.

2.2 PROGRAMINĖS ĮRANGOS PASIRINKIMAS

Aprašytos metodikos realizacijai pasirinktos trys programinės įrangos: MATLAB, SAS ir R.

DFA algoritmui realizuoti pasirinkta MATLAB [20] programinė įranga. Pasirinkimą lėmė MATLAB vidinių funkcijų gausa, palengvinanti algoritmo aprašymą, ir paelementės operacijos, leidžiančios adresuoti elementus naudojant indeksų masyvus.

Pasirinktos MATLAB funkcijos:

- `textread` ir `fprintf` – funkcijos skirtos duomenų iš tekstinio failo nuskaitymui ir duomenų išvedimui į tekstinį failą atitinkamai;
- `polyfit(x,y,n)` – grąžina *n*-tojo laipsnio polinomo, geriausiai tinkančio duomenims (mažiausių kvadratų prasme) koeficientų reikšmes. Pirma reikšmė grąžinamame masyve – polinomo reikšmė prie didžiausio laipsnio;
- `plot(x,y, parameters)` – funkcija skirta grafiniam duomenų vaizdavimui;
- kitos.

Savipanašumo parametrų skirtumų tarp grupių analizei pasirinkta duomenų analizės sistema SAS [21]. Šioje sistemoje gausu procedūrų, kurios leidžia greitai ir lengvai atlikti statistinę duomenų analizę.

Pasirinktos procedūros:

- PROC SGPLOT – procedūra skirta grafiniam rezultatų vaizdavimui. Naudosime braižyti savipanašumo parametrų reikšmių pasiskirstymą grupėse.
- PROC UNIVARIATE – procedūra skirta detalei kintamojo pasiskirstymo statistinei analizei. Procedūrą naudosime normalumo ir vidurkio lygybės nuliui hipotezių tikrinimui.
- PROC GLM – procedūra skirta atlikti paprastą ir daugialypę regresiją, dispersinę analizę (ANOVA), kovariacinę analizę, daugiamatę dispersinę analizę (MANOVA) ir kt. Procedūrą naudosime hipotezių apie savipanašumo parametrų vidurkių lygybes grupėse tikrinimui.

Kadangi SAS/Enterprise Minner modulio, kuriame yra klasifikavimo medžių sudarymo procedūros, licencijos Kauno technologijos universitetas neturi, klasifikavimo medžiams sudaryti pasirinkta atviro kodo programinė įranga R [22]. Šioje programoje yra realizuoti klasifikavimo algoritmai aprašyti 2.1.3.skyriuje. Klasifikavimo medžių sudarymui skirti paketai:

- Paketas „C5.0“, naudojantis patobulintą C4.5 algoritmo versiją C5.0;
- Paketas „rpart“, naudojantis patobulintas CART algoritmo versijas;

3 TYRIMO REZULTATAI

Šioje dalyje aprašomi rezultatai, gauti taikant 2 skyriuje pasiūlytą metodiką stazinio širdies nepakankamumo identifikavimo iš elektrokardiogramos RR intervalų sekų uždaviniui. Pateikiami sukurtų programinių priemonių aprašymai ir rekomendacijos tolimesniems tyrimams.

3.1 ANALIZUOJAMI DUOMENYS

Skyriuje 2.1 pasiūlytą ilgą laiką stebimų procesų parametrų panaudojimo klasifikavime metodiką taikysime spręsti tiriamųjų klasifikavimo į pasižyminčius normaliu sinusiniu ritmu (toliau šiame darbe NSR, angl. *Normal Sinus Rhythm*) ir turinčius stazinį širdies nepakankamumą (toliau šiame darbe CHF, angl. *Congestive Heart Failure*) uždavinį. Tiriamuosius apibūdinantys duomenys yra jų amžius ir EKG RR ~ 2 valandų (8192 įrašų) stebėsenos intervalų sekos. Tokio ilgio EKG RR intervalų sekos yra pakankamos DFA algoritmo taikymui ir savipanašumo parametrų radimui [11].

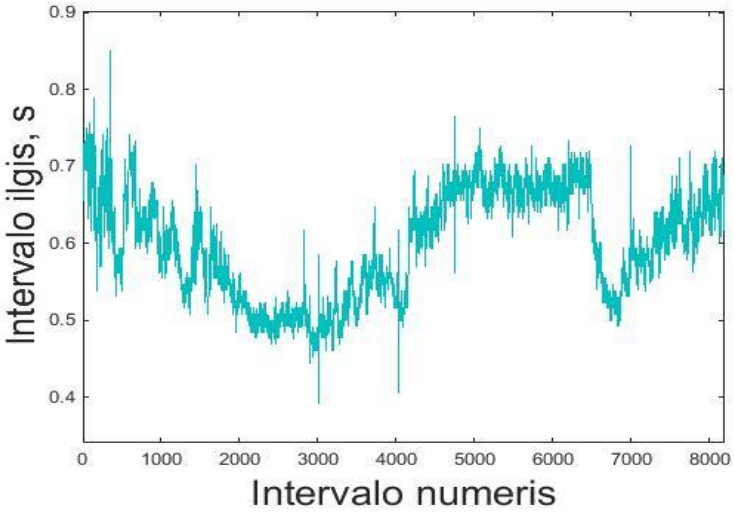
- NSR grupėje yra 29-nių tiriamųjų duomenys. Tiriamųjų amžiaus intervalas $[29, 76]$, vidurkis 62 metai, standartinis nuokrypis 11 metų.
- CHF grupėje yra 29-nių tiriamųjų duomenys. Tiriamųjų amžiaus intervalas $[34, 79]$, vidurkis 55 metai, standartinis nuokrypis 11 metų.

Duomenis paimti iš didelio ir šiuo metu sparčiai augančio svetainės [23] skaitmeninių įrašų archyvo „PhysioBank". Šiame archyve yra tiek sveikų, tiek įvairiomis ligomis sergančių tiriamųjų ilgalaikių ir trumpalaikių stebėsenų įvairūs biomedicininiai signalai.

Plačiau apie pasirinktus duomenis, jų rinkimo metodikas ir su šiais duomenimis atliktų kitų tyrimų sąrašus galima rasti [24, 25].

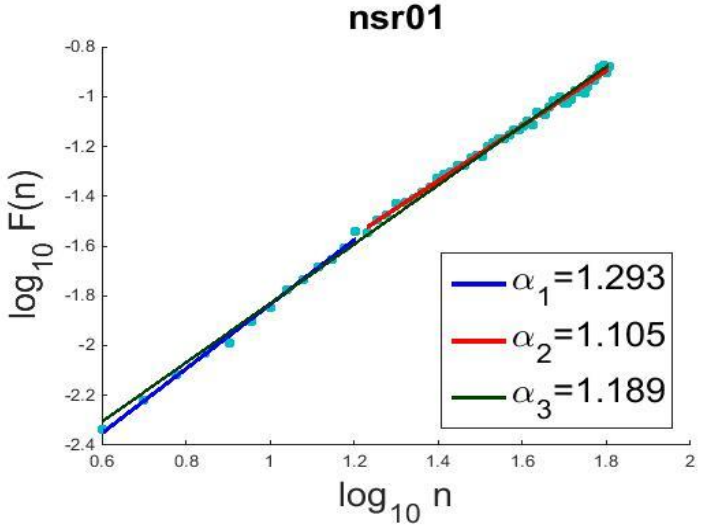
3.2 ILGĄ LAIKĄ STEBIMŲ PROCESŲ PARAMETRŲ TAIKYMO KLASIFIKAVIME METODIKOS PANAUDOJIMAS STAZINIO ŠIRDIES NEPAKANKAMUMO IDENTIFIKAVIMO UŽDAVINIO SPRENDIMUI

Analizuojame vieno iš NSR grupės tiriamųjų elektrokardiogramos parametro RR intervalų seką (seka pavaizduota 3.1 paveiksle).



3.1 pav. NSR grupės tiriamojo elektrokardiogramos RR intervalų seka

DFA metodu gautas fliuktuacijų dydžio ($F(n)$) nuo segmento pločio (n) priklausomybė logaritminiame tinklelyje pavaizduota 3.2 paveiksle. Gautos savipanašumo parametrų reikšmės:



3.2 pav. NSR grupės tiriamojo DFA metodu gautas fliuktuacijų ($F(n)$) nuo segmento pločio (n) priklausomybė logaritminiame tinklelyje

Skaičiuojant pirmosios regresijos tiesės ($n = \overline{4,16}$) krypties koeficientą $\alpha_1 = 1{,}293$ apibrėžtumo koeficiento reikšmė $R_1^2 = 0{,}997$. Skaičiuojant antrosios regresijos tiesės ($n = \overline{17,64}$) krypties koeficiento reikšmę $\alpha_2 = 1{,}105$ apibrėžtumo koeficiento reikšmė $R_2^2 = 0{,}994$. Bendros regresijos tiesės savipanašumo parametro reikšmė 1,189, o apibrėžtumo koeficientas $R_3^2 = 0{,}997$.

Visų tiriamųjų DFA metodu gauti rezultatai ir fliuktuacijų dydžio nuo segmentų pločio priklausomybės logaritminiai tinkleliai pateikti 1 priede.

1P. 1 lentelės paaiškinimai pateikti 3.1 lentelėje.

3.1 lentelė.

1 priedo duomenų lentelės paaiškinimai

Žymėjimas duomenų faile ir programoje	**Žymėjimas modelyje**	**Paaiškinimas**
id	**-**	Duomenų failo pavadinimas, tiriamojo identifikacijos numeris
alfai	$\boldsymbol{\alpha_i}$	Savipanašumo parametrai, i=1,2,3
ri	$\boldsymbol{R_i^2}$	Tiesės, kurios krypties koeficientu α_i, apibrėžtumo koeficientai, i=1,2,3
ind	**grupė**	Dvireikšmis kintamasis: 0- tiriamasis priklauso pirmajai grupei (NSR), 1- tiriamasis priklauso antrajai grupei (CHF)
am	**amžius**	Tiriamojo amžius

Visų tiriamųjų atvejais gautos apibrėžtumo koeficientų R_1^2, R_2^2 ir R_3^2 reikšmės, ieškant $F(n)$ ir n priklausomybes logaritminiame tinklelyje regresijos tiesių, yra nemažesnės už 0,717, vidurkis – 0.993, standartinis nuokrypis - 0.035, mediana – 0,996. Darėme išvadą, kad DFA metodas yra tinkamas RR intervalų sekų tyrimui. Ši išvada patvirtina [11, 12] straipsniuose gautus rezultatus apie DFA metodo taikymo RR intervalų sekoms tinkamumą.

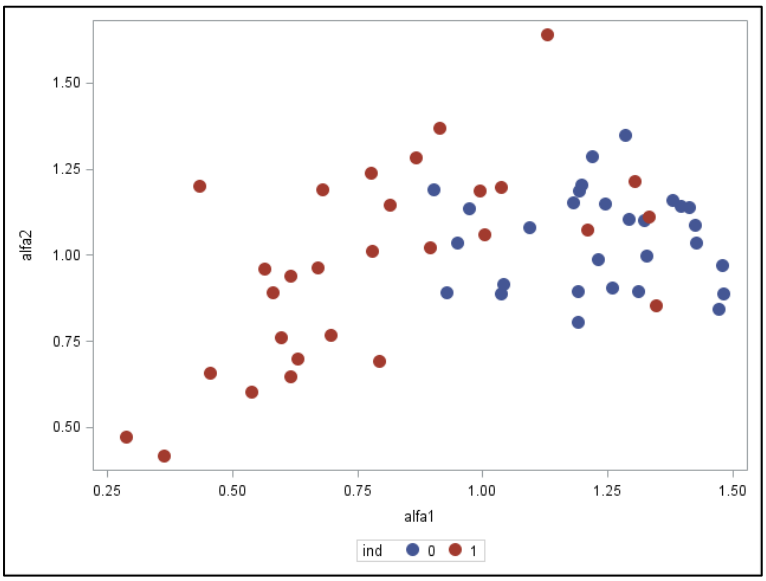
Apskaičiuotos savipanašumo parametrų statistinės charakteristikos pateiktos 3.2 lentelėje.

3.2 lentelė

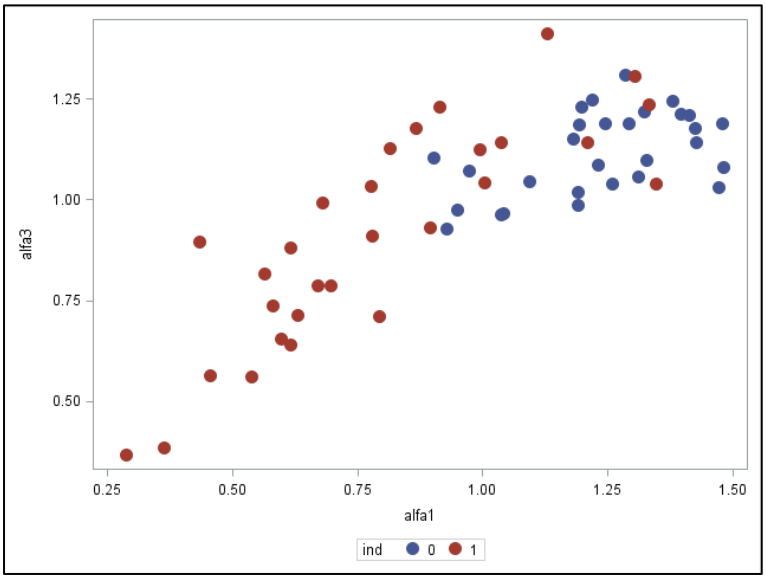
Savipanašumo parametrų statistinės charakteristikos

Grupė	**NSR**			**CHF**		
Parametras	**α1**	**α2**	**α3**	**α1**	**α2**	**α3**
Mažiausia reikšmė	0,90	0,80	0,96	0,29	0,42	0,88
Didžiausia reikšmė	1,48	1,35	1,31	1,35	1,64	1,00
Vidurkis	1,24	1,06	1,13	0,79	0,97	0,98
Standartinis nuokrypis	0,16	0,14	0,10	0,28	0,28	0,03
Mediana	1,24	1,09	1,14	0,78	1,01	0,99

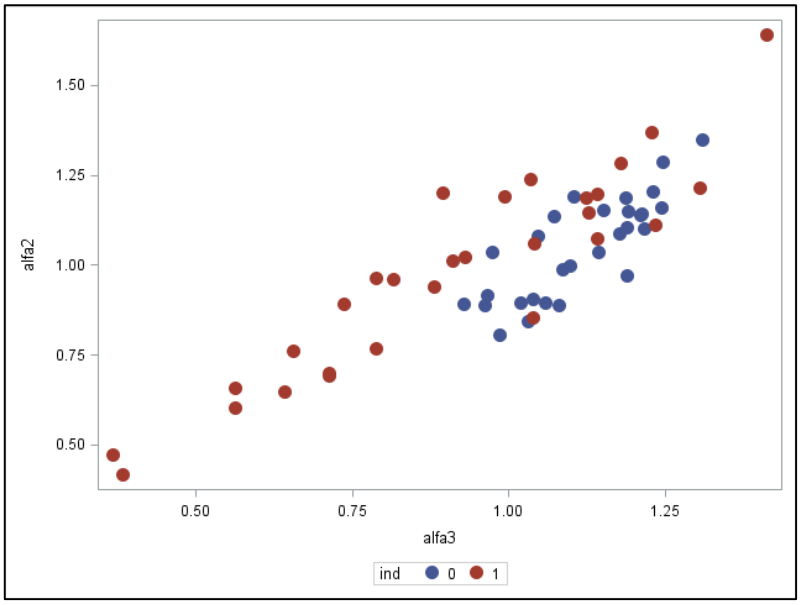
Turėdami visų tiriamųjų savipanašumo parametrų reikšmes, analizavome kaip šios reikšmės skiriasi NSR ir CHF grupėse. Savipananašumo parametrų taškų sklaido diagramos pavaizduotos 3.3-3.5 paveiksluose.



3.3 pav. Savipanašumo parametrų $\alpha_1$ir α_2 taškų sklaidos diagramos pagal grupę



3.4 pav. Savipanašumo parametrų $\alpha_1$ir α_3 taškų sklaidos diagramos pagal grupę



3.5 pav. Savipanašumo parametrų α_3 ir α_2 taškų sklaidos diagramos pagal grupę

Tyrėme hipotezes apie savipanašumo parametrų α_1, α_2 ir α_3 pasiskirstymą pagal normalųjį dėsnį NSR ir CHF grupėse. Abejose grupėse hipotezės buvo neatmestos (3.3 ir 3.4 lentelėse pateiktos normalumo hipotezių tikrinimo rezultatai)

3.3 lentelė

Hipotezių apie NSR grupės savipanašumo parametrų pasiskirstymų suderinamumo su normaliuoju skirstiniu tikrinimo rezultatai

Savipanašumo parametras:	α_1	α_2	α_3
Statistika | p reikšmė:	p	p	p
Šapiro-Vilko W	0,144	0,268	0,347
Kolmogorovo-Smirnovo D	>0,150	>0,150	>0,150
Kramerio-fon Mises W-sq	>0,250	0,150	0,236
Andersono-Darlingo A-sq	>0,250	0,146	>0,250

3.4 lentelė

Hipotezių apie CHF grupės savipanašumo parametrų pasiskirstymų suderinamumo su normaliuoju skirstiniu tikrinimo rezultatai

Savipanašumo parametras:	α_1	α_2	α_3
Statistika| p reikšmė:	p	p	p
Šapiro-Vilko W	0,364	0,783	0,806
Kolmogorovo-Smirnovo D	>0,150	>0,150	>0,150
Kramerio-fon Mises W-sq	>0,250	>0,250	>0,250
Andersono-Darlingo A-sq	>0,250	>0,250	>0,250

Kadangi normalumo prielaidos tenkinamos, tikrinome hipotezes apie savipanašumo parametrų vidurkių skirtumus tarp NSR ir CHF grupių. Hipotezė:

$$H_0: \mu_{\alpha_1^{\langle 1\rangle}} = \mu_{\alpha_1^{\langle 2\rangle}}, \quad H_a: \mu_{\alpha_1^{\langle 1\rangle}} \neq \mu_{\alpha_1^{\langle 1\rangle}} \qquad \textbf{(3.1)}$$

Gauta $p < 0.0001$, todėl atmetėme hipotezę apie parametro α_1 vidurkių lygybę grupėse. Analogiškai atmetėme ir hipotezę apie parametro α_3 vidurkių lygybę grupėse ($p = 0.000$), tačiau hipotezė apie parametro α_2 vidurkių lygybę grupėse neatmetama ($p = 0.213$).

Kadangi prielaidos apie dispersijų lygybę buvo atmestos (tikrinome Livyno variacijos homogeniškumo kriterijumi, rezultatai pateikti 3.5 lentelėje), (3.1) nulinės hipotezės tikrinimui naudojome Stjudento kriterijų, skirtą patikrinti hipotezei apie vidurkių lygybę, kai dispersijos nėra lygios. Hipotezės apie savipanašumo parametrų α_1 ir α_3 vidurkių grupėse lygybes buvo atmestos, su reikšmingumo lygmeniu $\alpha = 0{,}05$.

3.5 lentelė

Hipotezių apie dispersijų lygybę tikrinimo rezultatai (Livyno kriterijaus p reikšmė)

Parametras, apie kurio dispersijų lygybe grupėse daroma hipotezė	$\boldsymbol{\alpha_1}$	$\boldsymbol{\alpha_2}$	$\boldsymbol{\alpha_3}$
Livyno kriterijaus *p* reikšmė	0,0072	0,0035	0,0002

Tada taikėme gautuosius tiriamųjų EKG RR intervalų sekų savipanašumo parametrus ir tiriamųjų amžių klasifikavimo medžiams sudaryti.

Klasifikavimo medžiams sudaryti ir jų tinkamumo duomenims vertinimui naudojome kryžminio patikrinimo metodą. Duomenų aibę dalinome į tris dalis: dvi dalys po 10 tiriamųjų iš CHF ir NSR grupių ir trečioji dalis po 9-s tiriamuosius iš grupių.

Klasifikavimo medžiams sudaryti naudoti algoritmai:

- I metodas - patobulintas C4.5 metodas (C5.0).
- II metodas patobulintas CART, naudojantis apibendrintą gini indeksą;
- III metodas patobulintas CART, naudojantis informacinį skaidymą;

Sprendimų medžio sudarymo metodui įvertinti skaičiavome vidutinį klaidų kiekį procentais ir vidutinį klaidingo CHF grupės narių priskyrimo NSR grupei procentą, kadangi pagal sprendžiamą uždavinį svarbiau identifikuoti stazinį širdies nepakankamumą.

Gauti klasifikavimo medžių klaidingo klasifikavimo procentai pateikti 3.6 lentelėje, klaidingo CHF grupės kintamojo priskyrimo NSR grupei procentai pateikti 3.7 lentelėje.

3.6 lentelė

Klaidingo klasifikavimo procentai

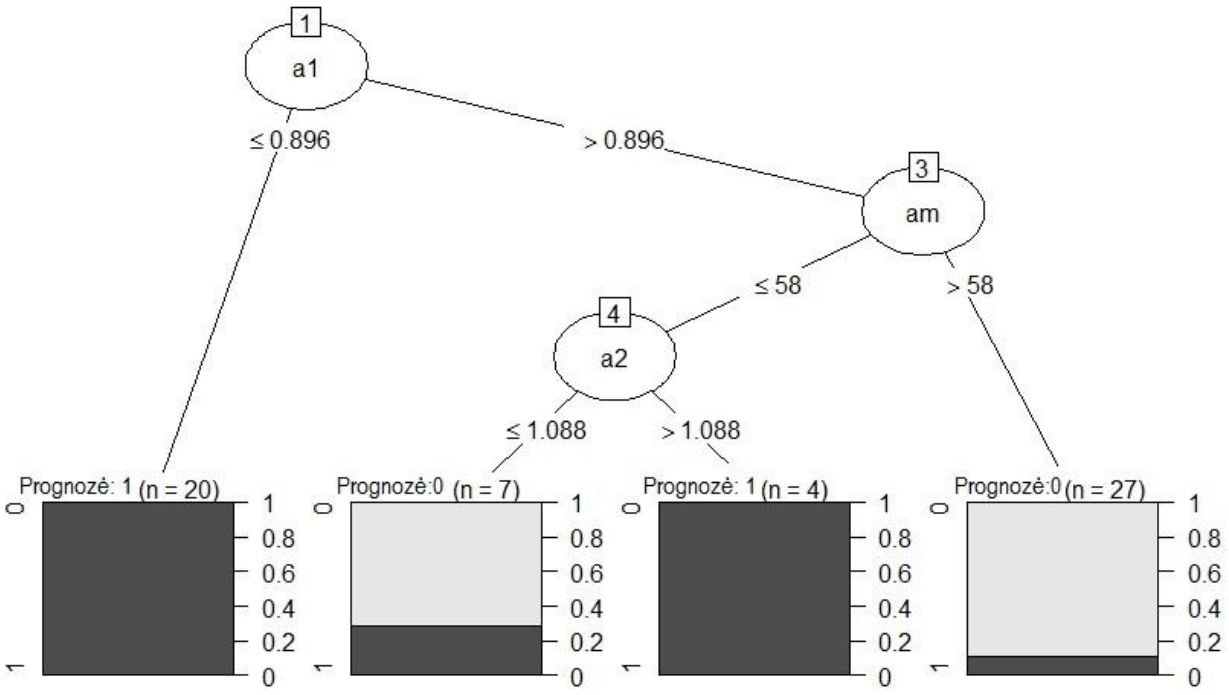
Metodas	**I**	**II**	**III**
Klaidingo klasifikavimo procentas	**10**	**17**	**10**

3.7 lentelė

Klaidingo CHF grupės kintamojo priskyrimo NSR grupei procentai

Metodas	**I**	**II**	**III**
Klaidingo CHF grupės klasifikavimo procentas	**13**	**13**	**16**

Tiriamiems duomenims klasifikuoti labiausiai tinkamas C5.0 metodas. Šiuo metodu sudarytas klasifikavimo medis, medžio sudarymui naudojant visą mokomąją imtį, klaidingai klasifikuoja 9% visų objektų ir 17% CHF grupės objektų. C5.0 klasifikavimo medis pavaizduotas 3. 7 paveiksle.



3.7 pav. C5.0 metodo klasifikavimo medis, čia a1 – α_1 , am – amžius ir a2 – α_2, 1 – CHF grupė, 0 – NSR grupė

Gauto klasifikavimo modelio panaudojimas

Tyrimo metu gautas klasifikavimo medis gali būti taikomas žmogaus dėvimų jutiklių analizėje. Sukūrus papildomas priemones, leidžiančias jutiklio naudotojui pačiam patikrinti savo EKG RR intervalų sekų parametrus, galima anksti pastebėti stazinį širdies nepakankamumą.

3.3 METODIKOS PROGRAMINĖ REALIZACIJA

DFA metodo algoritmui realizuoti parašyta universali MATLAB funkcija „**DFA**“(programos tekstas pateiktas 2 priede.). Kreipiantis į funkciją reikia nurodyti parametrus:

- ***data*** – norimos analizuoti sekos reikšmių vektorius;
- ***scale*** – sekos padalinimo į segmentus segmentų pločių vektorius;

- ***order*** – linkmės eliminavimo segmentuose polinomo eilė (šiame darbe analizuoti tik pirmos eilės polinomai, tačiau bendru atveju gali būti naudojami ir aukštesnės eilės polinomai);
- ***break_point*** – kintamasis, nurodantis dviejų regresijos tiesių (su krypties koeficientu α_1 ir su krypties koeficientu α_2) atskyrimo tašką;
- ***plot_mf*** – dvireikšmis kintamasis nurodantis, ar braižyti logaritminių ašių grafiką (1 – braižyti, 0 – nebraižyti),
- ***name*** – tekstinio tipo kintamasis, nurodantis $F(n)$ ir n priklausomybes logaritminio tinklelio pavadinimą.

„**DFA**“ funkcijos rezultatai, tiriant duomenų seką:

- trys sekos savipanašumo parametrų reikšmės;
- trys apibrėžtumo koeficientų, gaunamų ieškant regresijos tiesių logaritminiame $F(n)$ ir n priklausomybės tinklelyje, reikšmės;
- $F(n)$ ir n priklausomybė logaritminiame tinklelyje.

Konkretus „DFA“ funkcijos panaudojimo sekų tyrimui pavyzdžio programinis tekstas pateiktas 3 priede. Šioje programoje kiekvienoje iteracijoje:

- nuskaitomas tekstinis duomenų failas ir suformuojamas duomenų vektorius;
- su pasirinktais parametrais kreipiamasi į DFA funkciją;
- DFA metodu gauto $F(n)$ ir n priklausomybės logaritminio tinklelio paveikslas išsaugomas aplanke „Paveikslai“.
- DFA metodo rezultatai bei informacija apie duomenų failo pavadinimą ir tiriamojo grupės numeris įrašomi į duomenų failą „DFA rezultatai.txt“.

Programos vykdymo rezultatas – duomenų failas „DFA rezultatai.txt“. Įrašai šiame duomenų faile atskirti kableliais, kintamųjų žymėjimo paaiškinimai pateikti 3.1 lentelėje.

DFA rezultatų analizei – savipanašumo parametrų skirtumų pirmoje ir antrojoje grupėje tyrimui – parašyta SAS makro programa „DFA_rezultatu_analize“. Kreipiantis į makroprogramą reikia nurodyti:

- **l** – privalomas kintamasis, nurodantis vietą kompiuteryje, kurioje yra MATLAB analizės rezultatų tekstinis failas „DFA rezultatai.txt“.
- **grafikai** – dvireikšmis kintamasis: 1 – braižyti parametrų grafikus ir 0 – nebraižyti (numatytoji reikšmė – 1);
- **normalumas** – dvireikšmis kintamasis: 1 – tikrinti hipotezes apie savipanašumo parametrų reikšmių grupėse pasiskirstymą pagal normalųjį skirstinį ir 0 – netikrinti;
- **skirtumai** – kintamasis galinti įgyti dvi reikšmės: 1 – analizuoti parametru skirtumus tarp grupių ir 0 – neanalizuoti.

Kreipimosi į SAS makrokomandą pavyzdys pavaizduotas 3.5 paveiksle.

```
%let location=%str(C:\Users\Jovile\Desktop\DFA\I Matlab);
%DFA_rezultatu_analize(&location)
```

3.6 pav. Kreipimosi į SAS makro komandą „DFA_rezultatų_analize“ pavyzdys, kintamasis „location“ nurodo vietą kompiuteryje, kur yra saugomas tekstinis failas „DFA rezultatai.txt“

Klasifikavimo medžiams palyginti parašyta R programa. Programos tektas pateiktas 4 priede. Programinėje realizacijoje pirmosiose eilutėse nurodoma darbinė aplinka, kurioje yra Microsoft Excel duomenų failas „Duomenys.xlsx“. Šiame faile yra „DFA rezultatai.txt“ duomenų failo informacija ir papildomi kintamieji, kurie bus naudojami klasifikavimo medžiams sudaryti. Programos rezultatas – du klaidingo klasifikavimo vektoriai ir geriausiai klasifikuojantis klasifikavimo medis.

Rekomendacijos tolesniems tyrimams

Darbe pasiūlyta ilgą laiką stebimų procesų parametrų taikymo klasifikavime metodika (2.1 skyrius) galėtų būti taikoma įvairiuose klasifikavimo uždaviniuose. Taikymui atlikti galima panaudoti sukurtas programines priemones, pakoregavus MATLAB RR intervalų sekų analizei skirtos programos duomenų nuskaitymo iš tekstinių failų žingsnį.

IŠVADOS

1. Apžvelgus klasifikavimo ir ilgą laiką stebimų procesų analizės metodus pasiūlyta klasifikavime panaudoti DFA metodu gaunamus sekos savipanašumo parametrus.

2. Sudaryta DFA metodu gaunamų savipanašumo parametrų naudojimo klasifikavime metodika.

3. Įgyvendinant pasiūlytą metodiką, sukurtos šios programinės priemonės:
   - MATLAB funkcija skirta linkmės eliminavimo fliuktuacinės analizės metodo realizacijai ir metodo tinkamumo duomenims įvertinimui;
   - SAS makrokomanda skirta analizuoti savipanašumo parametrų skirtumus tarp grupių;
   - R programa skirta palyginti skirtingais algoritmais sudarytų klasifikavimo medžių, naudojančių savipanašumo parametrus ir kitus kintamuosius, kokybę.

4. Pritaikius pasiūlytą metodiką širdies stazinio nepakankamumo identifikavimui, pagal elektrokardiogramos RR intervalų ilgių sekas pastebėta, kad:
   - linkmės eliminavimo fliuktuacinės analizės metodas yra tinkamas elektrokardiogramos RR intervalų sekų analizei;
   - savipanašumo parametrai α_1 ir α_3 statistiškai reikšmingai skiriasi normalaus sinusinio ritmo ir stazinio širdies nepakankamumo grupių duomenims;
   - sudarytas klasifikavimo medis, teisingai klasifikuoja 91% sveikų žmonių ir 83% stazinį širdies nepakankamumą turinčių žmonių;
   - sudarytas klasifikavimo modelis gali būti taikomas žmogaus dėvimų jutiklių analizėje, siekiant kuo anksčiau pastebėti su širdies darbu (staziniu širdies nepakankamumu) susijusius pakitimus.

LITERATŪROS SĄRAŠAS

1. Sugiyama, M.; Kawanabe, M. Adaptive Computation and Machine Learning: Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation. Cambridge, MA, USA: MIT Press

2. Yang, Z.R. Machine Learning Approaches to Bioinformatics. River Edge, NJ, USA: World Scientific Publishing Co., 2010. ISBN 9789814287319

3. Han, J., Kamber, M.; Pei, J. Data Mining: Concepts and Techniques. Third Edition ed. Morgan Kaufmann. 2012. ISBN 978-0-12-381479-1.

4. Čekanavičius V.; Murauskas, G. Statistika ir jos taikymai II. TEV, Vilnius, 2002. ISBN 9955-491-16-7

5. Bergerud, W.A. Introduction to logistic regression models. Biometrics information Handbook, 1996, No. 7.

6. Shawe-Taylor, J.; Cristianini, N. Support Vector Machines and other kernel-based learning methods. Cambridge University Press, 2000

7. Šimkevičius, S. Klasifikavimo su mokytoju metodų lyginamoji analizė: magistro baigiamasis darbas. Kauno technologijos universitetas Fundamentaliųjų mokslų fakultetas Taikomosios matematikos katedra, 2006

8. Goldberger, A. L. et al; Fractal Dynamics in Physiology: Alterations with Disease and Aging. *Proceedings of the National Academy of Sciences of the United States of America*, Feb 19, vol. 99 Suppl 1, pp. 2466-2472 ISSN 0027-8424; 0027-8424

9. Peng, C.-K., et al. Mosaic Organization of DNA Nucleotides. PHYSICAL REVIEW E, vol. 49, no. 2, 1994 pp. 1685-1689.

10. Citations amount of the article “Mosaic Organization of DNA Nucleotides” [online]. [viewed 5 04 2015]. Available from: https://scholar.google.com/citations?view_op=view_citation&hl=lt&user=Z0fBt9oAAAAJ&citation_for_view=Z0fBt9oAAAAJ:u5HHmVD_uO8C,

11. Peng, C.K.; Havlin, S.; Stanley, H.E.; Goldberger, A.L. Quantification of Scaling Exponents and Crossover Phenomena in Nonstationary Heartbeat Time Series. *Chaos (Woodbury, N.Y.)*, vol. 5, no. 1, 1995, pp. 82-87 ISSN 1054-1500; 1054-1500

12. Peng, C.K.; Hausdorff, J.M.; Goldberger, A.L. Fractal mechanisms in neural control: Human heartbeat and gait dynamics in health and disease. In: Walleczek J, ed. Self-Organized Biological Dynamics and Nonlinear Control. Cambridge: Cambridge University Press, 2000

13. Zhou, J.; Wu, X.M.; Zeng, W.J.; 2015. Automatic Detection of Sleep Apnea Based on EEG Detrended Fluctuation Analysis and Support Vector Machine. *Journal of Clinical Monitoring and Computing*, 2015, ISSN 1573-2614; 1387-1307

14. Varoneckas, A. Hibridinis RR intervalų sekų modelis miego stadijoms atpažinti. Fiziniai mokslai, informatika (09 P) ed. Kaunas: Vytauto Didžiojo universitetas, Vilniaus universiteto Matematikos ir informatikos institutas. 2011, ISBN 978-9955-12-651-5

15. Iyengar, N.L.; Peng, C.K.; Morin, R.; Goldberger, A.L., Lipsitz, L.A. Age-related alterations in the fractal scaling of cardiac interbeat interval dynamics.

16. Čekanavičius, V. Taikomoji regresinė analizė socialiniuose tyrimuose, Kaunas, 2011 m. [interaktyvus]. [žiūrėta 2015m. gegužės 10d.]. Prieiga per:http://www.lidata.eu/files/mokymai/trast/Regresine_Analize_soc_tyrimuose.pdf

17. Quinlan, J.R. Induction of decision trees. Mach. Learn., 1(1):81–106, 1986.

18. Quinlan, J.R. C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993

19. Rokach, L.; Maimon, O. Data Mining with Decision Trees : Theory and Applications. River Edge, NJ, USA: World Scientific, 2007. ProQuest ebrary. Web. 6 June 2015.

20. MATLAB - The Language of Technical Computing, documentation [online]. [viewed 10 02 2015]. Available from: http://se.mathworks.com/help/matlab/index.html

21. SAS documentation [online]. [viewed 10 04 2015]. Available from: http://www.sas.com/en_us/home.html

22. R documentation [online]. [viewed 02 05 2015]. Available from: http://cran.r-project.org/

23. PhysioNet - The research resource for complex physiologic signals. [online]. [viewed 25 02 2015]. Available from: http://physionet.org/

24. Normal Synus Rytm RR intervals database database [online]. [viewed 25 02 2015]. Available from: http://physionet.org/physiobank/database/nsr2db/

25. Congestive Heart Failure RR intervals database [online]. [viewed 25 02 2015]. Available from: http://physionet.org/physiobank/database/chf2db/

1 PRIEDAS. RR INTERVALŲ SEKŲ LINKMĖS ELIMINAVIMO FLIUKTUACINĖS ANALIZĖS REZULTATAI

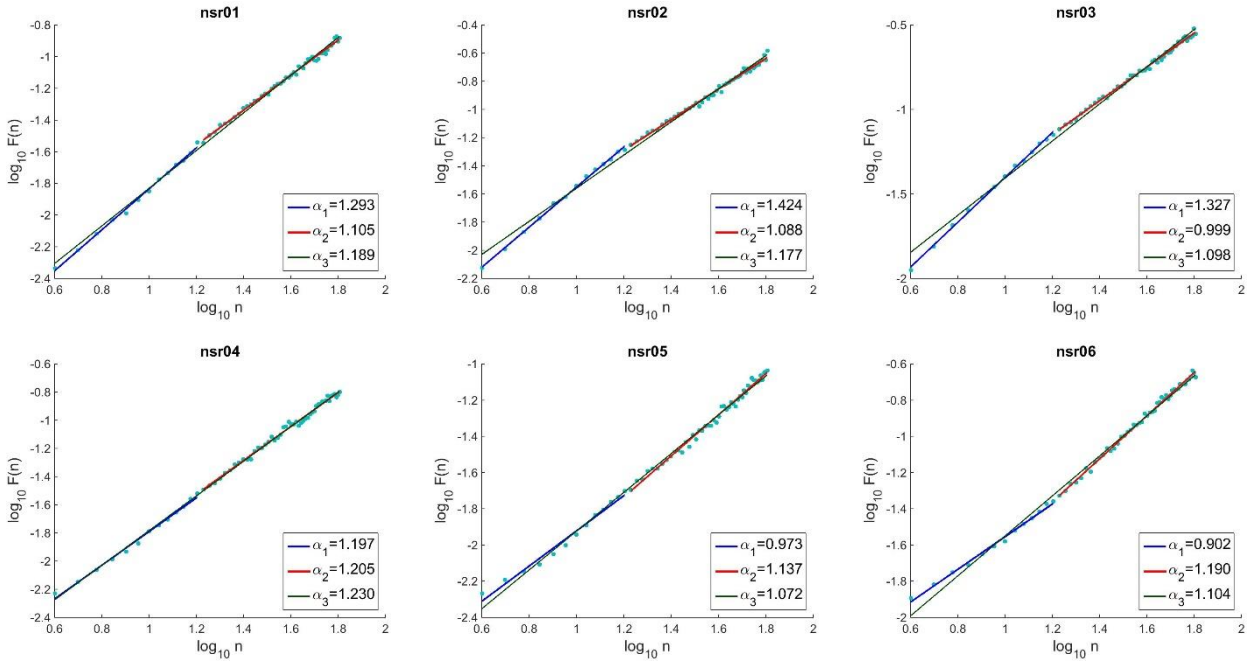
1 P. 1 lentelė

RR intervalų sekų DFA algoritmo rezultatai ir tiriamojo amžius

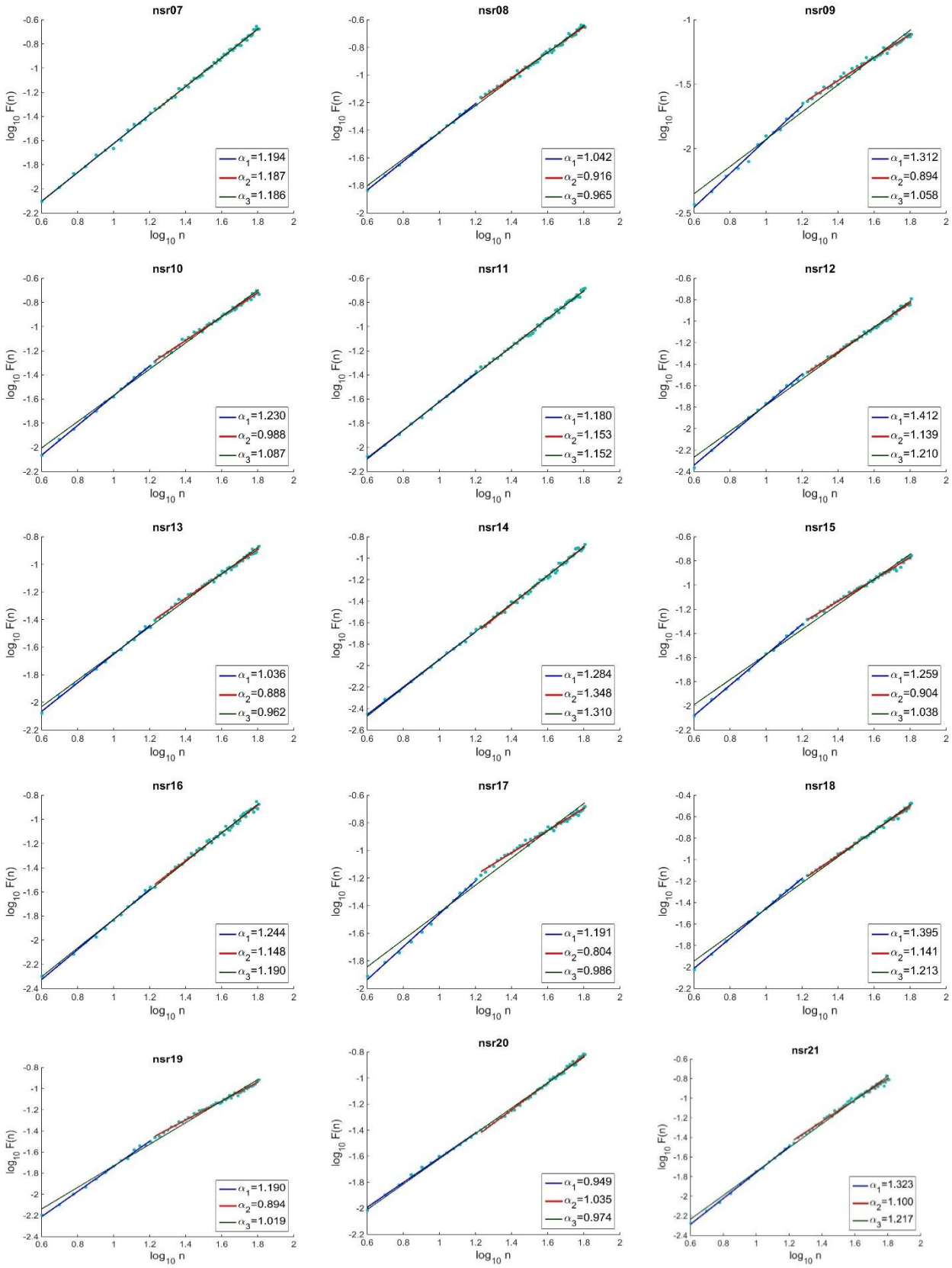
nr	id	alfa1	alfa2	alfa3	r1	r2	r3	ind	am
1	nsr01	1,293	1,105	1,189	0,997	0,994	0,997	0	64
2	nsr02	1,424	1,088	1,177	0,998	0,993	0,994	0	67
3	nsr03	1,327	0,999	1,098	0,998	0,995	0,994	0	62
4	nsr04	1,197	1,205	1,230	0,993	0,992	0,998	0	76
5	nsr05	0,973	1,137	1,072	0,982	0,991	0,995	0	63
6	nsr06	0,902	1,190	1,104	0,994	0,995	0,994	0	65
7	nsr07	1,194	1,187	1,186	0,993	0,996	0,999	0	73
8	nsr08	1,042	0,916	0,965	0,999	0,993	0,997	0	68
9	nsr09	1,312	0,894	1,058	0,993	0,989	0,987	0	65
10	nsr10	1,230	0,988	1,087	0,999	0,995	0,996	0	58
11	nsr11	1,180	1,153	1,152	0,999	0,996	0,999	0	59
12	nsr12	1,412	1,139	1,210	0,997	0,996	0,996	0	66
13	nsr13	1,036	0,888	0,962	0,997	0,994	0,997	0	75
14	nsr14	1,284	1,348	1,310	0,998	0,994	0,998	0	64
15	nsr15	1,259	0,904	1,038	0,999	0,994	0,991	0	65
16	nsr16	1,244	1,148	1,190	0,996	0,992	0,998	0	70
17	nsr17	1,191	0,804	0,986	0,996	0,991	0,986	0	67
18	nsr18	1,395	1,141	1,213	0,998	0,996	0,996	0	65
19	nsr19	1,190	0,894	1,019	0,997	0,994	0,993	0	66
20	nsr20	0,949	1,035	0,974	0,996	0,996	0,998	0	60
21	nsr21	1,323	1,100	1,217	0,999	0,990	0,995	0	63
22	nsr22	1,479	0,972	1,188	0,998	0,990	0,985	0	62
23	nsr23	1,219	1,286	1,247	0,983	0,986	0,995	0	70
24	nsr24	1,098	1,275	1,221	0,914	0,717	0,904	0	64
25	nsr25	1,379	1,158	1,245	0,995	0,995	0,998	0	67
26	nsr26	1,426	1,037	1,143	0,995	0,996	0,992	0	29
27	nsr27	1,482	0,889	1,081	0,996	0,995	0,981	0	38
28	nsr28	1,094	1,081	1,046	0,995	0,994	0,998	0	40
29	nsr29	1,471	0,843	1,031	0,992	0,997	0,977	0	35
30	chf01	1,333	1,110	1,235	0,997	0,992	0,995	1	55
31	chf02	0,681	1,192	0,993	0,987	0,995	0,981	1	59
32	chf03	0,597	0,759	0,656	0,984	0,973	0,983	1	68
33	chf04	0,776	1,237	1,034	0,988	0,909	0,954	1	62
34	chf05	0,814	1,147	1,127	0,875	0,885	0,959	1	39
35	chf06	0,616	0,941	0,880	0,967	0,784	0,924	1	38
36	chf07	0,287	0,473	0,367	0,983	0,959	0,959	1	62

RR intervalų sekų DFA algoritmo rezultatai ir tiriamojo amžius

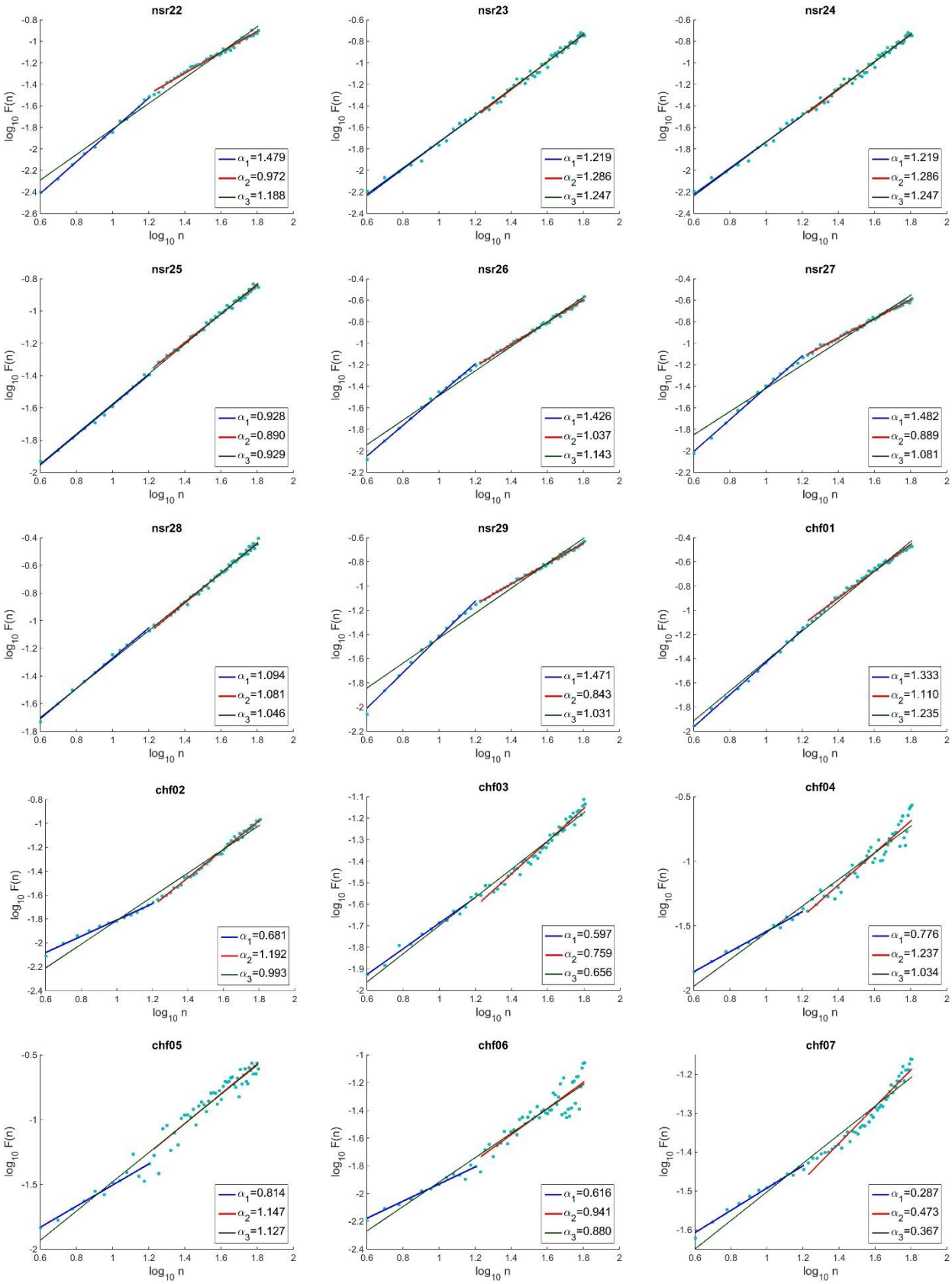
37	chf08	0,631	0,698	0,713	0,979	0,953	0,987	1	62
38	chf09	0,580	0,891	0,736	0,984	0,989	0,981	1	65
39	chf10	0,779	1,010	0,911	0,994	0,992	0,993	1	43
40	chf11	1,346	0,852	1,039	0,997	0,995	0,984	1	34
41	chf12	0,539	0,603	0,562	0,987	0,960	0,987	1	54
42	chf13	1,004	1,060	1,041	0,995	0,991	0,997	1	53
43	chf14	0,364	0,416	0,384	0,992	0,980	0,992	1	79
44	chf15	0,671	0,965	0,787	0,935	0,979	0,974	1	43
45	chf16	1,303	1,213	1,305	0,995	0,994	0,997	1	58
46	chf17	0,995	1,188	1,123	0,996	0,995	0,997	1	50
47	chf18	0,565	0,960	0,816	0,973	0,979	0,978	1	72
48	chf19	1,209	1,072	1,142	0,997	0,994	0,997	1	62
49	chf20	0,913	1,370	1,229	0,994	0,995	0,990	1	64
50	chf21	0,617	0,648	0,641	0,999	0,995	0,999	1	37
51	chf22	0,896	1,023	0,930	0,990	0,881	0,960	1	63
52	chf23	0,793	0,693	0,712	0,996	0,994	0,997	1	56
53	chf24	0,868	1,284	1,178	0,971	0,996	0,990	1	35
54	chf25	1,037	1,196	1,141	0,992	0,946	0,984	1	66
55	chf26	0,697	0,768	0,787	0,937	0,933	0,979	1	51
56	chf27	0,456	0,659	0,563	0,988	0,987	0,985	1	64
57	chf28	0,434	1,200	0,895	0,979	0,994	0,950	1	51
58	chf29	1,128	1,641	1,412	0,923	0,917	0,959	1	58



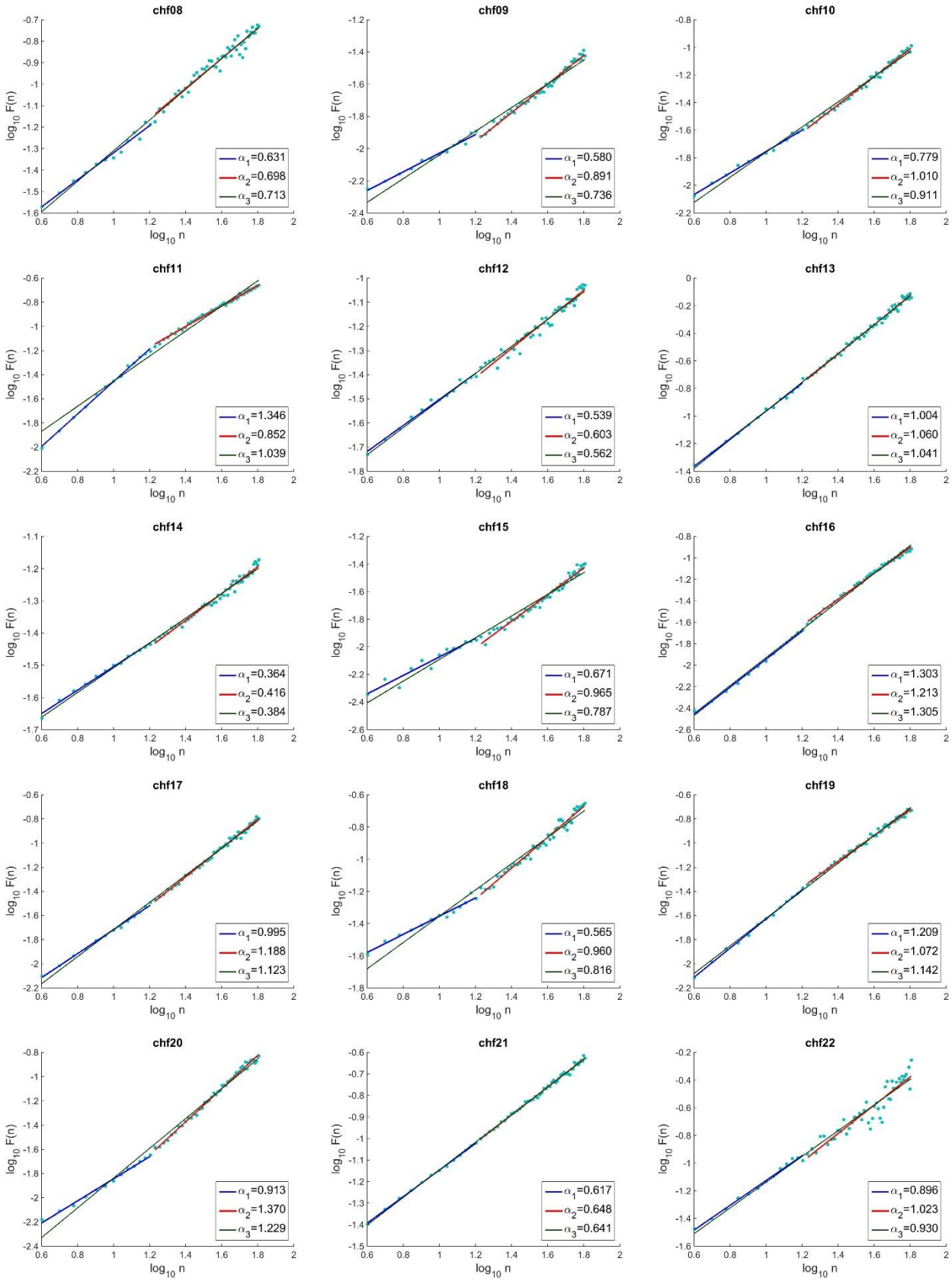
1 P. 1 pav. RR intervalų sekų $F(n)$ ir n priklausomybės logaritminiuose tinkleliuose



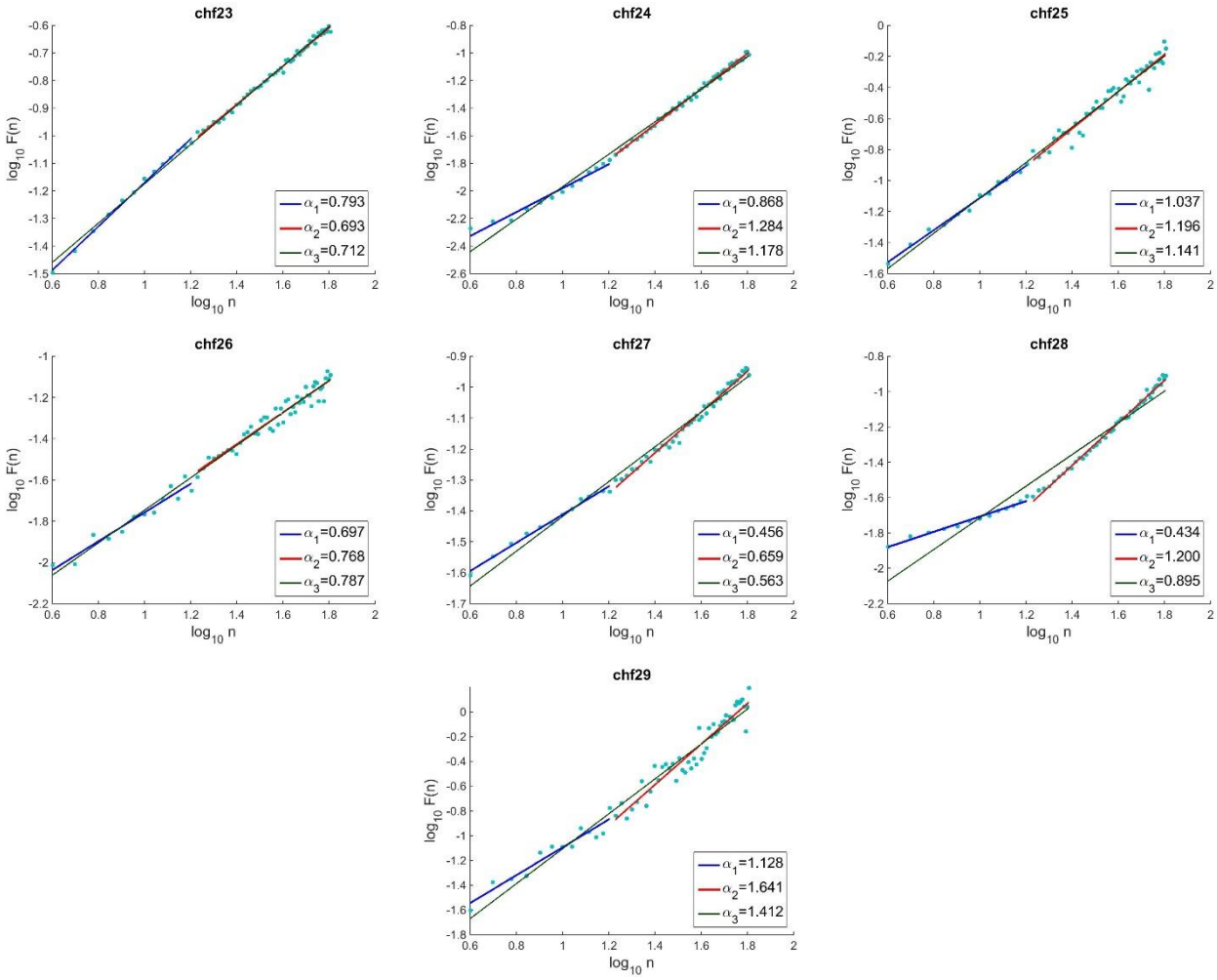
1P. 1 pav. tęsinys I. RR intervalų sekų F(n) ir n priklausomybės logaritminiuose tinkleliuose



1 P. 1 pav. tęsinys II. RR intervalų sekų F(n) ir n priklausomybės logaritminiuose tinkleliuose



1 P. 1 pav. tęsinys III. RR intervalų sekų F(n) ir n priklausomybės logaritminiuose tinkleliuose



1 P. 1 pav. tęsinys IV. RR intervalų sekų F(n) ir n priklausomybės logaritminiuose tinkleliuose

2 PRIEDAS. LINKMĖS ELIMINAVIMO FLIUKTUACINĖS ANALIZĖS METODO MATLAB FUNKCIJA DFA

```
function [ Hq, R] = DFA( data, scale, order , break_point, plot_mf, name)
%Funkcija, atliekanti linkmes eliminavimo fliuktuacine analize;
%   PARAMETRAI: data-analizuojama seka,
%               scale - segment? plociai,
%               order - polinomo eile,
%               k-skiriamasis taskas atskirti alfa1 ir alfa2 krypties
koeficientu tieses
%               plot_mf - dvireiksmis grafiko braizymo kintamasis(1 - braizyti,
0 - nebraizyti);
%               name - priklausomybes logaritminiame tinklelyje grafiko
%               pavadinimas
%   GRAŽINAMI PARAMETRAI:
%               Hq - savipanašumo parametr? vektorius: [alfa1 alfa2 alfa3];
%               R - apibreztumo koeficientu vektorius: [r1 r2 r3].

data=data'; data=cumsum(data-mean(data)); ls=length(scale);
F=zeros(1,ls); coef=zeros(ls,order+1);

for i =1:ls;
    s=scale(i);                              %Segmentu plotis;
    segments_n=floor(length(data)/s);        %Segmentu skaicius;
    RMS=zeros(1, segments_n);
    for j = 1:segments_n;
        index=((j-1)*s+1 : j*s);
        coef(i,:)=polyfit(index, data(index), order);
        RMS(j) = mean((data(index)- polyval(coef(i,:), index)).^2);
    end;
    F(i)=(mean(RMS))^(1/2);
    clear RMS;
end;

X=log10(scale); Fn=log10(F);
o=find(scale > break_point, 1, 'first');
X1=X(1:o-1);    F1=Fn(1:o-1);

C1=polyfit(X1, F1, 1);
Hq(1) = C1(1);
RegLine_1=polyval(C1, X1);
SSE1 = sum((F1- RegLine_1).^2);
SST1 = sum((F1-mean(F1)).^2);
R(1)=1-SSE1/SST1;

X2=X(o:end);  F2=Fn(o:end);
C2=polyfit(X2, F2, 1);
Hq(2) = C2(1);
RegLine_2=polyval(C2, X2);
SSE2 = sum((F2- RegLine_2).^2); SST2 = sum((F2-mean(F2)).^2);
R(2)=1-SSE2/SST2;

C3=polyfit(X, Fn, 1);
Hq(3) = C3(1);
RegLine_3=polyval(C3,X);
```

```
SSE = sum((Fn - RegLine_3).^2);
SST = sum((Fn  - mean(Fn)).^2);
R(3)=1-SSE/SST;

if plot_mf==1;
    co = [ 0 0.75 0.75; 0 0 1; 1 0 0; 0 0.255 0];
    set(groot,'defaultAxesColorOrder',co)
    figure
    xlabel('log_{10} n','Fontsize', 20)
    ylabel('log_{10} F(n)','Fontsize', 20)
    title(name, 'Fontsize', 20)
    hold on
    plot(X, Fn, '.','markersize', 20);
    qline(1)=plot(X1, RegLine_1, 'linewidth', 3);
    qline(2)=plot(X2, RegLine_2,  'linewidth', 3);
    qline(3)= plot(X, RegLine_3,  'linewidth', 2);
    legend([qline(1) qline(2) qline(3)], {sprintf('\\alpha_{1}=%.3f', Hq(1)),
sprintf('\\alpha_{2}=%.3f', Hq(2)), sprintf('\\alpha_{3}=%.3f', Hq(3))},
'Location', 'southeast', 'Fontsize', 20)
    set(gca, 'Fontsize', 16)
end
```

3 PRIEDAS. MATLAB PROGRAMA RR INTERVALŲ SEKŲ TYRIMUI LINKMĖS ELIMINAVIMO FLIUKTUACINĖS ANALIZĖS METODU

```
%DFA metodo panaudojimas tirti RR intervalu sekas;
clear all; warning off;

fileID = fopen('DFA rezultatai.txt','w');
fprintf(fileID,  '%s\r\n' , 'nr,id,alfa1,alfa2,alfa3,r1,r2,r3,ind');
SSP  = zeros(83, 3); R = zeros(83, 3);
scale=4:1:64; order=1; k=16;

for i=1:58;
    cd Duomenys
    if i < 10;
        ind=0;
        name = sprintf('nsr0%d.txt',i);
        RR=textread(name, '%*s %*s %f %*s %*s', 'headerlines',27);
     elseif i >= 10 && i <= 29;
        ind=0;
        name = sprintf('nsr%d.txt',i);
        RR=textread(name, '%*s %*s %f %*s %*s', 'headerlines',27);
    elseif i > 29 && i < 39;
        ind=1;
        name = sprintf('chf0%d.txt',i-29);
        RR=textread(name, '%*s %*s %f %*s %*s', 'headerlines',27);
    else
        ind=1;
        name = sprintf('chf%d.txt',i-29);
        RR=textread(name, '%*s %*s %f %*s %*s', 'headerlines',27);
    end
    cd ../

    z=0; for p=3:length(RR); if RR(p)> 2*RR(p-1)|| RR(p) < RR(p-1)/2 ;
    RR(p)=(RR(p-1)+RR(p-2))/2; z=z+1; end;   end;

   RR=RR(1:8192);
   %figure();
   %plot(RR);
   %axis([0 8192 min(RR)-0.05 max(RR)+0.05])
   %title('RR intervalu seka', 'Fontsize', 20)
   %xlabel('Intervalo numeris','Fontsize', 20)
   %ylabel('Intervalo ilgis, s', 'Fontsize', 20)

  [SSP(i,:), R(i, :)] = DFA(RR, scale, order, k, 1, strtok(name, '.'));

  cd Paveikslai
    saveas(figure(i),strcat(strtok(name, '.'), '.jpg'));
  cd ../

  fprintf(fileID,'%d,%s,%.3f,%.3f,%.3f,%.3f,%.3f,%.3f,%d \r\n', i, strtok(name,
'.'), SSP(i,:), R(i,:), ind);
  clear RR;
end;
fclose('all');
```

4 PRIEDAS. SAS MAKROKOMANDA SKIRTUMŲ TARP GRUPIŲ ANALIZEI

```
*************************************************************************;
* Makro komanda skirta DFA algoritmu gautu savipanašumo parametru analizei;
*************************************************************************;
%MACRO DFA_rezultatu_analize(l, grafikai=1, normalumas=1,skirtumai=1);

      proc import datafile="&l.\DFA rezultatai.txt" out=duomenys dbms=dlm
replace;
            delimiter=",";
            getnames=yes;
      run;

      %IF &grafikai %THEN %DO;
            title "SAVIPANAŠUMO PARAMETRU PASISKIRSTYMAI";  title;
            proc sgplot; scatter y=alfa2 x=alfa1 /group=ind
MARKERATTRS=(symbol=CircleFilled) MARKERATTRS=(SIZE=12); run;
            proc sgplot; scatter y=alfa3 x=alfa1 /group=ind
MARKERATTRS=(symbol=CircleFilled) MARKERATTRS=(SIZE=12); run;
            proc sgplot; scatter y=alfa2 x=alfa3 /group=ind
MARKERATTRS=(symbol=CircleFilled) MARKERATTRS=(SIZE=12); run;
      %END;

      %IF &normalumas %THEN %DO;
            title "PIRMOSIOS GRUPES SAVIPANAŠUMO PARAMETRU ANALIZE"; title;
            proc univariate data=duomenys normal;
                  var alfa1 alfa2 alfa3;
                  where ind=0;
                  histogram;
            run;

            title "ANTRISIOS GRUPES SAVIPANAŠUMO PARAMETRU ANALIZE"; title;
            proc univariate data=duomenys normal;
                  var alfa1 alfa2 alfa3;
                  where ind=1;
                  histogram;
            run;
      %END;

      %IF &skirtumai %THEN %DO;
            title "VIENFAKTORINE DISPERSINE ANALIZE, ALFA1";
            proc glm data=duomenys;
               class ind;
               model alfa1=ind/ ss3;
               output out=Tab
                              r = r;
               means ind / hovtest=levene;
            run;
            quit;
            title;

            title "LIEKANU ANALIZE, APIE ALFA1";
            proc univariate data=Tab normal;
               var r;
            run;
```

```
            title;

            title "VIENFAKTORINE DISPERSINE ANALIZE, ALFA2";
            proc glm data=duomenys;
               class ind;
               model alfa2=ind/ ss3;
               output out=Tab
                              r = r;
               means ind / hovtest=levene;
            run;
            quit;
            title;

            title "LIEKANU ANALIZE, APIE ALFA2";
            proc univariate data=Tab normal;
               var r;
            run;
            title;

            title "VIENFAKTORINE DISPERSINE ANALIZE, ALFA3";
            proc glm data=duomenys;
               class ind;
               model alfa3=ind/ ss3;
               output out=Tab
                              r = r;
               means ind / hovtest=levene;
            run;
            quit;
            title;

            title "LIEKANU ANALIZE, APIE ALFA3";
            proc univariate data=Tab normal;
               var r;
            run;
            title;
      %END;
%MEND DFA_rezultatu_analize;

ods html close;
ods html;
%let location=%str(C:\Users\Jovile\Desktop\DFA\I Matlab);
%DFA_rezultatu_analize(&location)
run;
```

5 PRIEDAS. R PROGRAMA KLASIFIKAVIMO MEDŽIAMS SUDARYTI IR PALYGINTI

```
# Klasifikavimo medziai DFA metodo duomenims;
rm(list=ls())
setwd("C:/Users/Jovile/Desktop/DFA")          ## Nurodyti, kelią iki duomenų failo "Duomenys.xlsx"


## Duomenu nuskaitymas;
library(XLConnect)
pradiniai_duomenys <- readWorksheetFromFile("Duomenys.xlsx",sheet ='Duomenu_lentele', header = TRUE)
duomenys_0 <- subset(pradiniai_duomenys,  pradiniai_duomenys$ind=='0')
duomenys_1 <- subset(pradiniai_duomenys,  pradiniai_duomenys$ind=='1')
ns=round(nrow(duomenys_0)/3, 0)

## Duomenu isskaidymas į tris vienodai subalansuotas atsitiktines imtis kryžminiam patikrinimui;
s1={set.seed(11111);sample(1:nrow(duomenys_0), ns, replace=FALSE)}
D10=duomenys_0[s1,]; D11=duomenys_1[s1,]
duomenys_0=duomenys_0[-s1, ];duomenys_1=duomenys_1[-s1, ]
D1=rbind(D10, D11); D1_ind=subset(D1, select=c('ind')); D1=D1[,c(3,4,5,13:ncol(D1))]
s2={set.seed(11111);sample(1:nrow(duomenys_0), ns, replace=FALSE)}
D20=duomenys_0[s2,]; D21=duomenys_1[s2,]
duomenys_0=duomenys_0[-s2, ]; duomenys_1=duomenys_1[-s2, ]
D2=rbind(D20, D21); D2_ind=subset(D2, select=c('ind')); D2=D2[,c(3,4,5,13:ncol(D2))]
D3=rbind(duomenys_0, duomenys_1); D3_ind=subset(D3, select=c('ind')); D3=D3[,c(3,4,5,13:ncol(D3))]



DD1=rbind(D1, D2); DD_ind1=rbind(D1_ind, D2_ind); DD_ind1=DD_ind1[,]
DD2=rbind(D1, D3); DD_ind2=rbind(D1_ind, D3_ind); DD_ind2=DD_ind2[,]
```

```
DD3=rbind(D2, D3); DD_ind3=rbind(D2_ind, D3_ind); DD_ind3=DD_ind3[,]
D= list(DD1, DD2, DD3); I=list(DD_ind1, DD_ind2, DD_ind3)
Test=rbind(D1,D2,D3);
Test_ind=rbind(D1_ind, D2_ind, D3_ind)
MC_all <- vector("numeric", 3); MC_1 <- vector("numeric", 3)

# 1 KLASIFIKAVIMO MEDIS
library(C50)
If <- list(as.factor(DD_ind1), as.factor(DD_ind2), as.factor(DD_ind3))
mc_all1 <- vector("numeric", 3); mc_11 <- vector("numeric", 3)
for (i in 1:3){
   CT <- C5.0( D[[i]], If[[i]])
   TP <- predict(CT, Test, type='class')
   Table <- table(TP, t(Test_ind))
   a <- Table['0', '1'];    b <- Table['1', '0']
   mc_all1[i] <- round((a+b)*100/nrow(Test),0)
   mc_11[i] <- round(a/(nrow(Test)/2)*100)
   remove(CT, TP, Table, a, b)
}

MC_all[1] <- round(mean(mc_all1),0) ; MC_1[1] <- round(mean(mc_11),0)

# 2 KLASIFIKAVIMO MEDIS
mc_all2 <- vector("numeric", 3); mc_12 <- vector("numeric", 3)
for (i in 1:3){
   CT <- rpart(I[[i]]~., data=D[[i]] , method = 'class', parms = list(split = "information"))
   TP <- predict(CT, Test, type='class')
   Table <- table(TP, t(Test_ind))
   a <- Table['0', '1'];    b <- Table['1', '0']
   mc_all2[i] <- round((a+b)*100/nrow(Test),0)
   mc_12[i] <- round(a/(nrow(Test)/2)*100)
   remove(CT, TP, Table, a, b)
```

```
}
MC_all[2] <- round(mean(mc_all2),0); MC_1[2] <- round(mean(mc_11),0)

# 3 KLASIFIKAVIMO MEDIS
library(rpart)
mc_all3 <- vector("numeric", 3); mc_13 <- vector("numeric", 3)
for (i in 1:3){
   CT <- rpart(I[[i]]~., data=D[[i]] , method = 'class', parms = list(split = "gini"))
   TP <- predict(CT, Test, type='class')
   Table <- table(TP, t(Test_ind))
   a <- Table['0', '1'];    b <- Table['1', '0']
   mc_all3[i] <- round((a+b)*100/nrow(Test),0);   mc_13[i] <- round(a/(nrow(Test)/2)*100)
   remove(CT, TP, Table, a, b)
}
MC_all[3] <- round(mean(mc_all1),0); MC_1[3] <- round(mean(mc_13),0)
Klaidingo_klasifikavimo_procentas <-MC_all
Klaidingo_1_priskyromo_0 <-MC_1
Klaidingo_klasifikavimo_procentas
Klaidingo_1_priskyromo_0
```