



VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS INSTITUTAS
KOMPIUTERINIO IR DUOMENŲ MODELIAVIMO KATEDRA

Kompiuterinio modeliavimo baigiamasis magistro darbas

Epilepsijos priepuolių aptikimas EEG signaluose
Epileptic seizure detection in EEG signals

Atliko:

Vilija Simonaitytė

parašas

Vadovas:

dr. Andrius Vytautas Misiukas
Misiūnas

Vilnius
2026

Turinys

Sutartinis terminų žodynas	3
Santrauka	4
Summary	5
Įvadas	6
1. Epilepsijos apibrėžimas ir klasifikacija	8
1.1. Epilepsijos diagnostika	8
2. Elektroencefalograma	9
2.1. Tarptautinė 10–20 EEG sistema	9
2.2. Epilepsijos priepuolių EEG charakteristika	10
3. Epilepsijos aptikimo sistema	10
3.1. EEG išankstinis apdorojimas	11
3.2. Požymių išskyrimas	11
3.2.1. Trumpojo lango Furjė transformacija	12
3.2.2. Vilnelės transformacija	12
3.3. Klasifikacija	13
3.3.1. Atraminų vektorių mašina	14
3.3.2. K-artimiausių kaimynų metodas	14
3.3.3. CNN	15
3.3.4. Iš anksto apmokyti CNN	16
4. Metodika	17
4.1. Duomenys ir tiriamieji	18
4.2. Programavimo kalba ir bibliotekos	18
4.3. Metrikos	20
4.4. Epilepsijos priepuolių atpažinimas	21
Išvados ir rekomendacijos	31
Ateities tyrimų planai	33
Priedai	38

Sutartinis terminų žodynas

- EEG – Elektroencefalografija
- DWT – Diskrečioji vilnelių transformacija (angl. *Discrete wavelet transform*)
- SVM – Atraminių vektorių mašina (angl. *Support vector machine*)
- EOG – Elektroomkulografija (angl. *Electrooculogram*)
- ERG – Elektoretinogramos (angl. *Electroretinogram*)
- EKG – Elektrokardiogramos (angl. *Electrocardiogram*)
- EMG – Elektromiogramą (angl. *Electromyogram*)
- STFT – Trumpojo lango Furjė transformacija (angl. *The short-time Fourier transform*)
- WT – Vilnelės transformacija (angl. *Wavelet transform*)
- CWT – Tolydžioji vilnelių transformacija (angl. *Continuous wavelet transform*)
- CNN – Konvoliuciniai neuroniniai tinklai (angl. *Convolutional Neural Networks*)
- ML – mašininis mokymasis (angl. *Machine learning*)
- DL – gilusis mokymasis (angl. *Deep learning*)

Santrauka

Šiame mokslo tiriamajame darbe nagrinėjamas automatizuotų sistemų, skirtų epilepsijos priepuoliams aptikti naudojant elektroencefalogramos (EEG) signalus, kūrimas ir analizė. Tyrime buvo naudojami Bostono vaikų ligoninėje ir Masačusetso technologijos institute (MIT) surinkti EEG duomenys, apimantys 24 vaikų pacientų ilgalaikius įrašus.

Epilepsijos priepuolių aptikimui buvo taikomi keli skirtingi metodai. Pirma, iš EEG signalų naudojant diskrečiąją vilnelės transformaciją (DWT) buvo išgauti požymiai, kurie vėliau klasifikuoti pasitelkiant atraminių vektorių mašiną (SVM) ir artimiausių kaimynų (KNN) metodą. Antra, iš trumpojo lango Furjė transformacijos (STFT) gautos spektogramos buvo naudojamos kaip įvestis įvairioms konvoliucinėms neuroninių tinklų (CNN) architektūroms bei iš anksto apmokytiems modeliams: ResNet50, ResNet50V2, EfficientNetB0 ir EfficientNetV2B0.

Geriausius rezultatus tarp SVM modelių pasiekė konfigūracija su haar vilnelės funkcija ir 5 lygio dekompozicija, kuri parodė 0,86 tikslumą, 0,91 preciziškumą, 0,80 jautrumą ir 0,85 F1 balą. KNN algoritmas geriausiai veikė naudojant bior6.8 vilnelę, tačiau su 2 lygio dekompozicija – šis modelis pasiekė 0,87 tikslumą, 0,80 preciziškumą, 0,98 jautrumą ir 0,88 F1 balą.

Tarp specialiai sukonstruotų CNN modelių optimaliausia architektūra pasiekė 0,844 tikslumą, 0,771 preciziškumą, 0,977 jautrumą ir 0,862 F1 balą. Iš iš anksto apmokytų modelių geriausius rezultatus pademonstravo ResNet50V2, kurio klasifikavimo rodikliai buvo 0,687 tikslumas, 0,702 preciziškumas, 0,687 jautrumas ir 0,682 F1 balas.

Apibendrinant visus gautus rezultatus, geriausias epilepsijos priepuolių aptikimo efektyvumas buvo pasiektas naudojant DTW ir KNN metodų kombinaciją.

Summary

Epileptic seizure detection in EEG signals

This research paper examines the development and analysis of automated systems designed to detect epileptic seizures using electroencephalogram (EEG) signals. The study used EEG data collected at Boston Children’s Hospital and the Massachusetts Institute of Technology (MIT), covering long-term recordings of 24 pediatric patients.

Several different methods were used to detect epileptic seizures. First, features were extracted from the EEG signals using discrete wavelet transform (DWT) and then classified using a support vector machine (SVM) and k-nearest neighbors (KNN) method. Second, spectrograms obtained from short-time Fourier transform (STFT) were used as input to various convolutional neural network (CNN) architectures and pre-trained models: ResNet50, ResNet50V2, EfficientNetB0, and EfficientNetV2B0.

The best results among SVM models were achieved by the configuration with the Haar wavelet function and 5-level decomposition, which showed 0.86 accuracy, 0.91 precision, 0.80 recall, and 0.85 F1 score. The KNN algorithm performed best using the bior6.8 forest, but with level 2 decomposition—this model achieved an accuracy of 0.87, sensitivity of 0.80, precision of 0.98, and an F1 score of 0.88.

Among the pre-trained models, ResNet50V2 demonstrated the best results, with classification metrics of 0.687 accuracy, 0.702 precision, 0.687 sensitivity, and 0.682 F1 score.

Summarizing all the results obtained, the best epilepsy seizure detection efficiency was achieved using a combination of DTW and KNN methods.

Įvadas

Epilepsija yra neurologinis smegenų sutrikimas, pasireiškiantis pasikartojančiais priepuoliais. Viena pagrindinių priemonių epilepsijos diagnostikoje yra elektroencefalografija (EEG), leidžianti fiksuoti smegenų elektrinės veiklos pokyčius, susijusius su epilepsijos priepuoliais. Tačiau tradicinė vizualinė EEG analizė yra sudėtinga, subjektyvi ir reikalaujanti didelės patirties, todėl automatizuotų epilepsijos priepuolių aptikimo sistemų kūrimas yra itin aktuali mokslinė ir klinikinė užduotis.

Šiame darbe analizuojamas EEG duomenų rinkinys, surinktas Bostono vaikų ligoninėje (CHB) ir Masačusetso technologijos institute (MIT), apimantis 24 vaikų pacientų ilgalaikius EEG įrašus.

Epilepsijos priepuolių aptikimas EEG signaluose iš esmės yra signalo segmentų klasifikavimo užduotis, kurios tikslas — atskirti priepuolio būseną nuo normalaus (ne priepuolio) signalo. Ši užduotis gali būti sprendžiama taikant mašininio mokymosi (ML) arba giluminio mokymosi (DL) metodus. Mašininio mokymosi sistemose iš EEG signalų paprastai išgaunamos savybės laiko, dažnio ir laiko–dažnio srityse, kurios vėliau naudojamos kaip įvestis klasifikatoriams. Dažniausiai taikomi klasifikavimo metodai yra atraminių vektorių mašina (SVM), artimiausių kaimynų algoritmas (KNN) ir kiti panašūs metodai [1].

Laiko–dažnio analizė leidžia vienu metu įvertinti tiek laiko, tiek dažnio informaciją, todėl ji ypač tinkama nestacionarių EEG signalų analizei. Atsižvelgiant į daugiakanalę ir nestabilią EEG signalų prigimtį, dažniausiai taikomos trumpojo lango Furjė transformacija (STFT) ir diskrečioji vilnelių transformacija (DWT).

Šiame darbe ypatingas dėmesys skiriamas optimalaus požymių rinkinio parinkimui, leidžiančiam mašininio mokymosi algoritmams pasiekti geriausius klasifikavimo rezultatus. Tačiau rankinis požymių išgavimas yra sudėtingas, imlus laikui ir reikalauja ekspertinių žinių, todėl tai laikoma vienu pagrindinių tradicinių ML metodų trūkumų. Dėl šios priežasties šiame tyrime taip pat nagrinėjamas automatinis epilepsijos priepuolių aptikimas, taikant giluminio mokymosi metodus.

Daugelis ankstesnių tyrimų patvirtino giluminio mokymosi veiksmingumą EEG signalų klasifikavimo srityje [2, 3]. Nors giluminis mokymasis ypač sėkmingai taikomas vaizdų atpažinimo užduotyse, dėl EEG signalų nestacionarumo šių metodų efektyvus pritaikymas priepuolių aptikimui vis dar išlieka aktyvia tyrimų sritimi.

Dėl triukšmo ir sudėtingos EEG struktūros, giluminio mokymosi sistemų įvestims dažnai naudojami išankstinės signalo dekompozicijos rezultatai, gauti taikant Furjė arba vilnelių transformacijas. Manoma, kad šios transformacijos leidžia efektyviau išgauti reikšmingas signalo savybes.

Konvoliuciniai neuroniniai tinklai (CNN) suteikia galimybę automatiškai išmokti požymius iš EEG signalų be rankinio jų išgavimo, taip supaprastinant visą analizės procesą ir pakeičiant sudėtingą daugiaetapę mašininio mokymosi darbo eigą vientisu gilaus mokymosi modeliu [4].

Šiame darbe siūlomi keli epilepsijos priepuolių aptikimo modeliai. Pirmasis modelis remiasi skirtingų vilnelių DWT taikymu ir klasifikavimu naudojant SVM bei KNN algoritmus. Antrasis modelis naudoja iš STFT gautas spektrogramas kaip įvestį įvairioms CNN architektūroms.

Darbo tikslas ir uždaviniai

Šio darbo tikslas - atlikti epilepsijos priepuolių klasifikaciją, naudojant EEG duomenis, pritaikius skirtingus kompiuterinio mokymosi įrankius.

Buvo išskirti tokie uždaviniai:

- Literatūros apžvalga: sukurti modelio struktūrą epilepsijos priepuolių aptikimui EEG duomenyse.
- Tyrimui tinkamų duomenų radimas.
- Duomenų įkėlimas: tinkamai nuskaityti EEG duomenis.
- Duomenų apdorojimas: Pritaikyti EEG duomenims diskrečiąją vilnelių transformaciją.
- Savybių išskyrimas: Atlikti duomenų analizę išgaunant naudingus duomenis, kaip maksimumas, minimumas, vidurkis ir pan..
- Klasifikacija: Pritaikyti ir palyginti skirtingų modelių efektyvumą.

1. Epilepsijos apibrėžimas ir klasifikacija

Epilepsija yra lėtinė neurologinė liga, kuria serga virš 50 mln. žmonių visame pasaulyje. Pagal ILEA (anlg. International league against epilepsy) apibrėžimą: Epilepsija yra smegenų liga, kuri atitinka bet vieną iš toliau nurodytų sąlygų: 1. Bent du neprovokuoti (arba refleksiniai) priepuoliai, tarp kurių yra daugiau kaip 24 val. 2. Vienas neišprovokuotas (arba refleksinis) priepuolis ir kitų priepuolių tikimybė, panaši į bendrą pasikartojimo riziką (mažiausiai 60 %) po dviejų neišprovokuotų priepuolių per ateinančius 10 metų [5]. Epilepsiją ar traukulių priepuolius gali sukelti smegenų pažeidimas, atsiradęs dėl tokių priežasčių, kaip galvos smegenų traumos, galvos smegenų auglio, smegenų infekcijos, deguonies trūkumo smegenims ar genetinių sindromų [6].

Pagal pasireiškiančių traukulių tipą išskiriamos tokios epilepsijos formos: generalizuota, židininė ir nežinoma. Sergant generalizuota epilepsija, dažniausiai pasireiškia viso kūno traukuliai. Jų metu paveikiami abu smegenų pusrutuliai. Generalizuoti priepuoliai gali būti motoriniai, kuomet stebimi nevalingi kūno judesiai, arba nemotoriniai, daugiau susiję su sąmonės ar suvokimo sutrikimu. Židininės formos epilepsija sergantiems žmonėms pasireiškia židiniai traukuliai. Kitaip nei generalizuoti, jie paveikia tik vieną smegenų dalį. Progresuojant židininės epilepsijos priepuoliui, priepuolis gali generalizuotis ir pereiti į generalizuotą epilepsiją, tai vadinama židinine epilepsija su antrine generalizacija [7].

Klinikiniuose tyrimuose priepuolio procesą galima suskirstyti į keturias fazes: interiktalinę, preiktalinę, iktalinę ir postiktalinę. Iktalinė fazė - tai epilepsijos protrūkio laikotarpis, kuris pasireiškia kaip priepuolis, insultas ar galvos skausmas. Preiktalinė fazė yra iktalinės fazės pradžia, o postiktalinė fazė yra iktalinės fazės pabaiga. Galiausiai interiktinė fazė yra laikotarpis tarp priepuolių [8].

Vienas iš pagrindinių epilepsijos diagnostikos metodų yra EEG. Tačiau norint nustatyti tikslią epilepsijos diagnozę, dažnai reikia atlikti daug ir ilgai trunkančių EEG įrašų, o jų klinikinė peržiūra reikalauja daug laiko ir išteklių bei kompetencijos [9].

1.1. Epilepsijos diagnostika

Ankstyvas epilepsijos diagnozavimas ir tinkamas vaistų paskyrimas yra labai svarbūs siekiant pagerinti paciento gyvenimo kokybę, sumažinti priepuolių dažnį bei komplikacijų ir sužeidimų riziką. Tiksliai nustatyta diagnozė leidžia paskirti efektyviausią gydymą. Remiantis Tarptautinės Lygos Prieš Epilepsiją (ILAE) ataskaita [5], epilepsijos diagnozavimo procesas apima kelis esminius žingsnius:

1. Klinikinis vertinimas: pirmiausia svarbu išsamiai surinkti paciento ligos istoriją, įvertinti neurologinės apžiūros rezultatus bei gauti liudininkų parodymus apie priepuolio eigą. Jei įmanoma, itin vertingas yra priepuolio vaizdo įrašas, kuris padeda atskirti epilepsinius priepuolius nuo kitų, panašių būklių (pvz., sinkopės, psichogeninių neepilepsinių priepuolių).
2. Papildomi tyrimai: elektroencefalograma (EEG) dažnai naudojama siekiant identifikuoti epileptinį aktyvumą smegenyse. Galimi papildomi galvos vaizdiniai tyrimai — dažniausiai magnetinio rezonanso tomografija (MRT), rečiau – kompiuterinė tomografija (KT), ypač esant įtarimui dėl struktūrinių pažeidimų.
3. Epilepsijos tipo ir sindromo nustatymas: remiantis visais surinktais duomenimis (anamnezės, neurologinio įvertinimo, EEG, vaizdinių bei laboratorinių tyrimų), gydytojas nustato ne tik

tai, ar pacientas serga epilepsija, bet ir kokio tipo epilepsija tai yra – židininė, generalizuota ar mišri. Taip pat, esat galimybei, siekiama identifikuoti specifinį epilepsijos sindromą, kuris padeda parinkti efektyviausią gydymą.

Tinkamai atlikta diagnostika ne tik padeda tiksliai įvertinti būklę, bet ir leidžia išvengti neteisingos diagnozės bei nereikalingo gydymo, todėl yra esminis žingsnis sėkmingame epilepsijos valdyme.

2. Elektroencefalograma

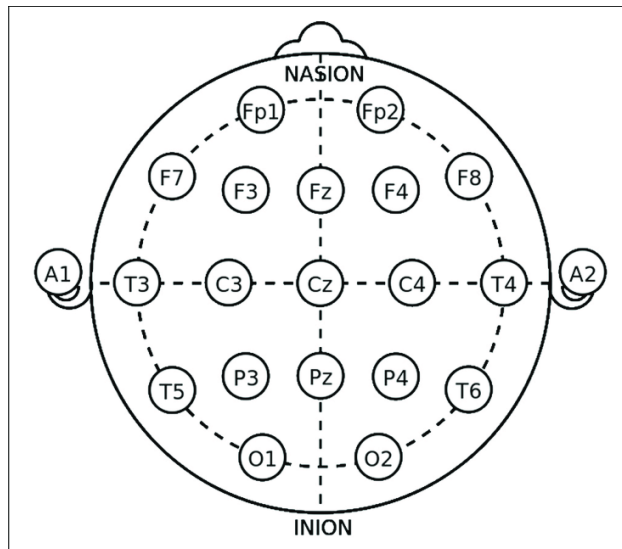
Elektroencefalografija – tai neinvazinis galvos smegenų bioelektrinio aktyvumo tyrimas, kurio metu registruojama galvos smegenų potencialai [10]. Dabartinis supratimas apie EEG šaltinį yra toks, kad ją pirmiausia lemia dideli įtampos svyravimai, atsirandantys dėl sinchroninės postsinapsinės nervų veiklos [11]. Sukėlus veikimo potencialą, į postsinapsinį plyšį išsiskiria neuromediatoriai. Kai šie neuromediatoriai prisijungia prie postsinapsinio neurono, visame neurono kūne pasikeičia krūvis. Siekiant subalansuoti šį krūvį, dendrituose vyksta pasyvus jonų nutekėjimas. Galutinis rezultatas – didelis įtampos skirtumas ląstelėje. Matuojant EEG, stebimus potencialo pokyčius laikui bėgant sukelia ne pavieniai neuronai ar maža grupė, o didelės neuronų grupės, kurios yra panašiai orientuotos ir veikia sinchroniškai [12].

Galvos EEG paprastai matuojama elektrodais, simetriškai išdėstytais ant galvos odos. EEG signalas, arba kanalas, formuojamas iš dviejų elektrodų išmatuotų potencialų skirtumo. Kiekvienas EEG kanalas apibendrina aktyvumą, lokalizuotą tam tikroje smegenų srityje. Židininio priepuolio pradžia susijusi su aktyvumo pokyčiais keliuose galvos EEG kanaluose, esančiuose virš smegenų vietos, kurioje kyla priepuolis, arba šalia jos; kita vertus, generalizuoto priepuolio pradžia susijusi su aktyvumu visuose galvos EEG kanaluose [13].

Pagrindinis EEG trūkumas – maža erdvinė skyra [14]. Visų pirma, didžiausią įtaką skalpo EEG daro arčiausiai galvos odos paviršiaus esantys neuronai; priešingai, giliai galvos smegenų struktūrose esančių neuronų aktyvumas nepastebimas. Be to, smegenis supantis smegenų skystis ir kaukolė veikia kaip slopintuvai, kurie labai sumažina aukštesnio dažnio nervinių virpesių amplitudę. Svarbi šių fizinių apribojimų pasekmė yra ta, kad tam tikrų rūšių priepuolių, būtent tų, kurie apima mažą, giliai esančią smegenų sritį, negalima stebėti naudojant galvos EEG.

2.1. Tarptautinė 10–20 EEG sistema

EEG aktyvumui registruoti paprastai naudojama standartizuota tarptautinė 10–20 sistema. Ši sistema turi 21 elektrodą, simetriškai išdėstytą ant galvos odos paviršiaus (1 pav.). Šių padėtyų apskaičiuojamos kaip standartinių atstumų procentai, gautus įrašus galima palyginti tarp skirtingų pacientų. Matuojant EEG gaunama laikinė ir erdvinė informacija apie sinchroninę daugelio neuronų sužadavinimą smegenyse, kurio dominuojantis dažnis atitinka smegenų ritmus [15].



1 pav. EEG elektrodų išdėstymas pagal 10–20 tarptautinę sistemą

2.2. Epilepsijos priepuolių EEG charakteristika

Epilepsija yra vienas iš labiausiai paplitusių neurologinių sutrikimų, atsirandančių dėl nenormalios smegenų elektrinės veiklos, kuri dažniausiai pasireiškia kaip epilepsijos priepuoliai. Šių priepuolių analizė ir diagnozė dažniausiai atliekama naudojant EEG signalus. EEG signalų interpretacija, kurią atlieka gydytojai specialistai, yra sudėtingas ir daug laiko reikalaujantis procesas. Todėl automatinės priepuolių aptikimo sistemos gali ženkliai palengvinti šį darbą, sumažindamos analizės laiką ir padidindamos diagnostikos tikslumą [16].

Daugumoje iki šiol pasiūlytų metodų epilepsijos priepuoliai aptinkami analizuojant EEG signalus laiko, dažnio arba laiko–dažnio domene. Atsižvelgiant į EEG signalų charakteristikas, epilepsijos eiga gali būti suskirstyta į tris pagrindines fazes: normalią (tarppriepuolinę), preiktinę (priešpriepuolinę) ir iktinę (priepuolio). Priepuolio metu (iktinėje fazėje) EEG signaluose atsiranda anomalijų, tokių kaip [17]:

- Pikas (Spike) – trumpalaikis, aiškiai matomas signalas, kurio trukmė nuo 20 iki mažiau nei 70 ms, taip pat turintis smailų viršūnės tašką.
- Staigi banga (Sharp wave) – trumpalaikis, aiškiai išsiskiriantis iš foninės veiklos signalas, turintis smailų viršūnės tašką. Staigios bangos yra panašios į pikus, bet trunka ilgiau – nuo 70 iki 200 ms).
- Piko–bangos kompleksas (Spike-wave complex) – signalų seka, susidedanti iš piko, po kurio eina lėta banga. Lėta banga yra ilgesnės trukmės nei alfa banga, t. y. trunka daugiau nei 1/8 sekundės.

3. Epilepsijos aptikimo sistema

Dirbtinis intelektas (DI) atlieka svarbų vaidmenį teikiant pagalbą medicinos srityje. Ypač EEG pagrįstų priepuolių aptikime. Priepuolių aptikimo procesą galima suskirstyti į tris pagrindinius etapus: pirminį apdorojimą, požymių išskyrimą ir klasifikavimą. Šie etapai bus išsamiai pristatyti.

3.1. EEG išankstinis apdorojimas

Neapdorotas EEG signalas yra nestacionarus ir pasižymi maža erdvine skiriamąja geba. Taip pat EEG signalai yra jautrūs įvairiems trikdžiams, tokiems kaip akių judesiai, raumenų aktyvumas, elektros tinklo triukšmas. Šie artefaktai gali iškreipti EEG įrašus ir privesti prie klaidingų išvadų. Todėl artefaktų identifikavimas ir pašalinimas yra svarbus EEG duomenų analizavimo etapas, kuris padeda užtikrinti kokybišką požymių išskyrimą bei patikimą klasifikaciją ar diagnostiką [18, 19].

Vienas iš esminių artefaktų valdymo komponentų yra filtravimas. EEG įrašymo metu taikomas bent vienas filtras, kuris pašalina aukštesnius nei tam tikra riba dažnius – tai vadinama žemųjų dažnių praleidimo filtru (angl. low-pass filter). Šis filtras praleidžia žemus dažnius, bet slopina aukštesnius, tokiu būdu užkirsdamas kelią klaidoms dėl spektrų sanklotos (angl. aliasing).

Be filtravimo, vienas iš svarbiausių metodų siekiant sumažinti EEG signalų nestacionarumą yra duomenų segmentacija. Segmentuojant duomenis į trumpesnius, laikinai stabilius langus, galima tiksliau analizuoti signalų struktūrą bei atpažinti su epilepsijos priepuoliais susijusius pokyčius. Remiantis naujausiais tyrimais, viename iš jų nurodoma, kad „kritiškiausi žingsniai EEG valymo procese yra duomenų segmentacija, o nepriklausomų komponentų analizės (ICA) algoritmas turi nedidelę įtaką“ [20].

3.2. Požymių išskyrimas

Po to, kai EEG signalai yra išvalomi nuo triukšmų ir artefaktų per išankstinio apdorojimo (preprocessing) etapą, būtina iš jų išgauti reikšmingus požymius. Požymių išskyrimui taikomi įvairūs metodai, kaip nepriklausomų komponentų analizė (ICA), greitoji Furjė transformacija (FFT), vilnelių transformacijos (WT), automatinė regresija (AR) ir pagrindinių komponentų analizė (PCA).

Šiame darbe požymių išskyrimui pasirinkti du plačiai taikomi laiko–dažnio analizės metodai: diskrečioji vilnelių transformacija (DWT) ir Trumpojo lango Furjė transformacija (STFT) [21, 22, 23, 1]. Šių metodų pasirinkimą lėmė tiek jų teorinis tinkamumas nestacionariems EEG signalams analizuoti, tiek jų veiksmingumas, įrodytas kituose moksliniuose tyrimuose, susijusiuose su epilepsijos signalų analize.

STFT metodas pasirinktas dėl jo gebėjimo pateikti laiko–dažnio vaizdą, leidžiantį analizuoti, kaip signalo spektras keičiasi laike. Viename iš naujesnių tyrimų, Aurore Bussalb ir kt. pritaikė modelį, kuris susidėjo iš STFT, po kurio sekė 2D–CNN ir transformatoriaus blokas. Šis modelis, klasifikuojant EEG įrašus į normalius ir abnormalius, pasiekė 90,92% tikslumą [24]. Toraman ir kt. savo tyrime EEG signalus konvertavo į spektrogramas naudodami STFT. Naudodami tris skirtingus iš anksto apmokytus CNN modelius (VGG19, ResNet, DenseNet) gautas spektrogramas suklasifikavo į dvi būsenas: tarp priepuolių ir priepuolio. Modeliai pasiekė 91,05% tikslumą [3].

DWT leidžia efektyviai dekomponuoti signalą į skirtingus dažnio sub-juostų lygius, išlaikant laiko domeną. Tai ypač svarbu analizuojant epilepsinius priepuolius, kurie dažnai pasireiškia kaip staigūs ir trumpalaikiai signalų pokyčiai. Daugylyje darbų DWT buvo sėkmingai pritaikyta epilepsijos diagnostikoje. Pavyzdžiui, M. Li ir kt. pasiūlė neuroninį tinklą, kuris veikė su DWT išgautais požymiais ir pasiekė 98,78% klasifikacijos tikslumą aptinkant epilepsijos priepuolius [25]. Kitame tyrime panaudojus DWT ir dirbtinius neuroninius tinklus su LM algoritmu buvo pasiektas 99,44% tikslumas [26]. Shen ir kt. savo tyrime pritaikė DWT ir SVM metodus ir pasiekė 97% tikslumą [23].

3.2.1. Trumpojo lango Furjė transformacija

Furjė transformacija (angl. Fourier Transform) – tai matematinė operacija, kuri transformuoja duomenis iš laiko (arba erdvės) domenų į dažnio domeną. [27]. Kadangi Furjė koeficientai nustatomi pagal visą signalą, Furjė analizė teikia tik bendrą laiko vidurkį ir užgožia bet koki lokalų signalo elgesį.

Šis apribojimas gali būti iš dalies išspręstas įvedant trumpojo lango Furjė transformatą (STFT). Pagrindinė STFT idėja – pradinį signalą išskaidyti į nedidelius laiko langus ir kiekvienam laiko segmentui taikyti FT, kad būtų išreikštas signalo dažnio pokytis laike, kuris vyko tame segmente. STFT matematinė lygtis (3.1) yra tokia:

$$S_{f,\tau} = \sum_{t=0}^{N-1} x(t)\omega t - \tau e^{-j2\pi f t} \quad (3.1)$$

kur $x(t)$ – analizuojamas signalas, $\omega(t)$ – lango funkcija, kurios centras yra ties $\tau = t$.

STFT yra laiko dažnio atvaizdavimas, dažnai naudojamas signalų analizėje, taip pat skaitmeninių vaizdų apdorojime, balso apdorojime, biologijoje ir medicinoje, elektroencefalografinių (EEG) signalų analizėje (STFT leidžia atvaizduoti EEG signalus laiko dažniu, o tai rodo kitoki vizualizavimą analizei) [28, 29]. Priešingai nei STFT, kur naudojamas fiksuotas laiko langas, vilnelės transformacija (WT) naudoja kintamo dydžio langus.

3.2.2. Vilnelės transformacija

Vilnelė yra į bangą panašus svyravimas, lokalizuotas laike [30]. Vilnelės pasižymi dviem pagrindinėmis savybėmis: masteliu ir vieta. Mastelis apibrėžia, kiek „ištempta“ arba „suspausta“ yra vilnelė – ši savybė susijusi su dažniu. Vieta apibrėžia, kurioje laiko (arba erdvės) vietoje yra vilnelė. Pagrindinis vilnelių transformacijos tikslas – apskaičiuoti, kiek tam tikro mastelio ir vietos signalu yra vilnelių. Tai atliekama signalui pritaikius konvoliucijos metodą su įvairių mastelių vilnelių rinkiniu. Yra du pagrindiniai vilnelių analizės tipai: nuolatinė (3.2) ir diskrečioji (3.4). Pagrindinis skirtumas yra, kad priešingai nei nuolatinėje, diskrečiojoje vilnelių transformacijoje naudojami signalai, kurių trukmė yra baigtinė.

$$X_{CWT}(a, b) = \int_{-\infty}^{+\infty} s(t)\psi_{a,b}^*(t)dt \quad (3.2)$$

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \quad (3.3)$$

a – mastelio koeficientas, b – poslinkis.

$$X_{DWT}(j, k) = \int_{-\infty}^{+\infty} s(t)\psi_{j,k}^*(t)dt \quad (3.4)$$

$$\psi_{j,k}(t) = a_0^{-\frac{j}{2}}\psi(a_0^{-j}t - kT) \quad (3.5)$$

j – mastelio koeficientas, k – poslinkio parametras

Vilnelių transformacijos taikymas EEG signalų analizei yra svarbus signalų analizės procesas, leidžiantis išgauti specifinę informaciją iš įvesties duomenų. Biomedicininuose signaluose savybių išskyrimas yra būtinas siekiant sumažinti duomenų dimensiją ir atpažinti svarbias signalo charakteristikas.

EEG signalai yra pateikiami laiko srityje, todėl tiesioginė jų analizė ne visada leidžia gauti naudingą informaciją. Vilnelių transformacija leidžia konvertuoti laiko srities signalus į dažnio sritį, kas yra itin naudinga nestacionarių signalų, tokių kaip EEG, analizei. Diskrečioji vilnelių transformacija (DWT) leidžia išskirti signalo komponentus skirtinguose dažnių diapazonuose, taip suteikdama aiškesnį supratimą apie signalo struktūrą.

3.3. Klasifikacija

Klasifikavimo algoritmų kokybė labai priklauso nuo išskirtos savybės, kuri pateikiama klasifikatoriui. Klasifikatorius naudojamas signalui klasifikuoti į įvairias klases, todėl klasifikatorius laikomas sprendimų priėmimo sistema.

Klasifikavimo etape galima atlikti du pagrindinius etapus, t. y. mokymą ir testavimą. Išskirti požymiai suskirstomi į šiuos etapus, o apmokius klasifikatorių, naujus duomenis galima klasifikuoti naudojant apmokytą tinklą, tai vadinama testavimo etapu. Epilepsijos priepuolių aptikimo sistemoje naudojami klasifikatoriai apima mašininio mokymosi metodus bei, pastaruoju metu, giliuosius neuroninius tinklus [31, 29].

Tarp daugelio taikomų mašininio mokymosi metodų, atraminių vektorių mašinos (SVM) ir artimiausių kaimynų metodas (k-NN) išsiskiria kaip vieni populiariausių ir patikimiausių klasifikatorių šioje srityje.

SVM dažniausiai pasirenkamas dėl savo gebėjimo efektyviai spręsti tiek linijinius, tiek nelineinius klasifikavimo uždavinius, ypač kai duomenų kiekis nėra labai didelis, bet jų požymiai turi aiškią atskirtį. Naudojant tinkamus branduolio (angl. *kernel*) metodus, SVM pasiekia labai aukštą tikslumą. Daugybė tyrimų (1 lentelė) patvirtina šio metodo efektyvumą epilepsijos atvejų klasifikacijoje.

Tuo tarpu **k-NN** klasifikatorius yra paprastas, tačiau labai efektyvus, ypač kai duomenys yra tinkamai paruošti ir turi aiškius požymių skirtumus tarp klasių. Jo pagrindinis pranašumas – nereikia ilgo treniravimo proceso, nes klasifikacija atliekama remiantis atstumu iki artimiausių treniravimo aibės pavyzdžių. Dėl savo paprastumo ir gero veikimo daugelyje atvejų, k-NN dažnai naudojamas kaip atskaitos taškas (baseline) tiriant kitų metodų efektyvumą. Tyrimai parodė, kad k-NN gali pasiekti konkurencingą tikslumą, ypač kai požymiai parinkti naudojant DWT, PCA ar kitus tinkamus metodus (žr. 1 lentelę).

Tiek SVM, tiek k-NN pasižymi savo privalumais: SVM yra tikslesnis sudėtingesniuose klasifikavimo uždaviniuose, o k-NN – greitesnis ir lengviau įgyvendinamas praktiniuose sprendimuose, ypač kai siekiama realaus laiko analizės. Dėl šių savybių, abu metodai yra plačiai naudojami EEG signalų analizėje ir epilepsijos aptikimo sistemų kūrimo.

1 lentelė. Skirtingų klasifikatorių tikslumai pagal literatūros šaltinius

Šaltiniai	Klasifikatorius	Tikslumas (Accuracy)	Metai
Choudhury et al. (2018) [32]	KNN	100%	2018
Feng, Zhao and Fu (2018) [33]	SVM	98.67%	2018
Shivarudhrappa et al. (2019) [34]	SVM	100%	2019
Karabiber Cura et al. (2020) [35]	KNN	97.00%	2020

Pastaraisiais metais giluminio mokymo (angl. *deep learning*, DL) metodai tapo dominuojančiais automatizuoto epilepsijos priepuolių aptikimo sistemų kūrimo srityje, nes jie padeda išspręsti tradicinių mašininio mokymosi metodų apribojimus. Priešingai nei klasikiniai metodai, kuriems būtina rankiniu būdu parinkti ir išskirti požymius, giluminio mokymo architektūros geba automatiškai išmokyti svarbiausius signalų reprezentacinius požymius iš papildomai neapdorotų duomenų, ypač kai dirbama su dideliais duomenų rinkiniais.

CNN architektūros plačiai naudojamos dėl gebėjimo automatiškai išgauti reikšmingus požymius iš EEG signalų, ypač kai šie signalai paverčiami į laiko-dažnio reprezentacijas, tokias kaip spektrogramos. Suat Toraman atliktame tyrime [3], naudotam iš anksto apmokytam CNN buvo pateiktos STFT pagrindu gautos 2D spektrogramos. Modelis pasiekė apie 90% tikslumą.

3.3.1. Atraminių vektorių mašina

Atraminių vektorių mašina (angl. Support vector machine, SVM) – mašininio mokymosi algoritmas, naudojamas klasifikavimui ir regresijai. Naudojant SVM siekiama nustatyti hiperplokštumą su didžiausiu atstumu (marža) tarp kiekvienos klasės duomenų. Tiesinė hiperplokštuma apibrėžiama šia (3.6) funkcija:

$$w^T x + b = 0 \quad (3.6)$$

kur w yra atraminis vektorius.

Ši hiperplokštuma turi pasižymėti dviem specifinėmis savybėmis: (1) ji turi turėti mažiausią įmanomą duomenų atskyrimo paklaidą ir (2) jos atstumas iki artimiausių kiekvienos klasės duomenų turi būti maksimalus. Šiomis aplinkybėmis kiekvienos klasės duomenys gali būti tik kairėje arba dešinėje hiperplokštumos pusėje. Todėl duomenų atskiriamumui kontroliuoti galima apibrėžti dvi ribas (3.7):

$$\begin{aligned} w^T x + b &\geq 1, \text{ kai } y_i = 1 \\ w^T x + b &\leq -1, \text{ kai } y_i = -1 \end{aligned} \quad (3.7)$$

3.3.2. K-artimiausių kaimynų metodas

K-artimiausių kaimynų (KNN) algoritmas yra paprastas, bet veiksmingas algoritmas, turintis platų pritaikymą biomedicinos kompiuterijoje. KNN yra neparametrinis algoritmas, kuriame objektas klasifikuojamas remiantis k-artimiausių kaimynų balsų dauguma. Šiame algoritme išsaugomas visas mokymo rinkinys ir apskaičiuojamas atstumas tarp nežinomo objekto ir k-artimiausių kaimynų. Dažniausiai šis atstumas matuojamas naudojant Euklidinį atstumą ir Manhatano atstumą. Tada nežinomo objekto klasė nustatoma pagal k-artimiausių narių balsų daugumą ir atitinkamai klasifikuoja naują tašką.

3.3.3. CNN

Konvoliucinis neuroninis tinklas (angl. *Convolutional Neural Network*, CNN) yra pažangi dirbtinių neuroninių tinklų (ANN) architektūra, skirta efektyviam tinklo struktūros duomenų, ypač vaizdų ir vaizdo įrašų, apdorojimui. CNN pasižymi gebėjimu automatiškai išgauti reikšmingus požymius iš įvesties duomenų, todėl nereikalauja iš anksto apibrėžtų vaizdo savybių. Skirtingai nei tradiciniai metodai, CNN savybių išgavimą atlieka mokymosi proceso metu, identifikuojamas svarbiausius vaizdo bruožus, tokius kaip kraštai, tekstūros ir formos. Naudojant dešimtis ar net šimtus paslėptų sluoksnių, tinklas geba išmokyti vis sudėtingesnes vaizdų reprezentacijas. CNN veikimo principas iš dalies paremtas žmogaus regos žievės funkcionavimu, kur vizualinė informacija apdorojama keliais nepriklausomais sluoksniais, o vėliau integruojama bendram sprendimui priimti.

CNN veikimas susideda iš dviejų pagrindinių etapų: požymių išskyrimo ir klasifikavimo. Požymių išskyrimo etapas atliekamas naudojant konvoliucijos, aktyvacijos ir jungimo sluoksnius, o klasifikavimo etapas vykdomas visiškai sujungtuose sluoksniuose. Įvesties sluoksnis yra pirmasis tinklo sluoksnis, kuris priima neapdorotus duomenis iš išorinio pasaulio ir transformuoja juos į tinklui suprantamą formatą. Įvesties duomenų gylis atitinka vaizdo kanalų skaičių, pavyzdžiui, RGB vaizdų atveju jis lygus trimis, o gilesniuose tinklo sluoksniuose – anksčiau pritaikytų filtrų skaičiumi.

Pagrindinis CNN elementas yra konvoliucijos sluoksnis, kuris atlieka dvimatę konvoliucijos (kryžminės koreliacijos) operaciją. Šios operacijos metu maža matrica, vadinama branduoliu (angl. *kernel*) arba filtru, slenka per įvesties matricą ir išgauna lokalius požymius. Matematiškai 2D konvoliucija aprašoma taip (3.8):

$$out(N_i, C_{out}) = bias(C_{out}) + \sum_{k=0}^{C_{in}-1} weight(C_{out}, k) \star input(N_i, k) \quad (3.8)$$

kur $\star$ žymi dvimatės kryžminės koreliacijos operatorių, N – grupės dydį, C – kanalų skaičių, H – įvesties plokštumų aukštį pikseliais, o W – plotį pikseliais.

Šios operacijos rezultatas yra savybių žemėlapis, kuriame fiksuojami tokie vaizdo bruožai kaip kraštai, tekstūros ir formos. Išvesties matricos aukštis (3.9) ir plotis (3.10) apskaičiuojami pagal šias formules:

$$H_{out} = \left\lfloor \frac{H_{in} + 2 \cdot padding[0] - dilation[0] \cdot (kernel_size[0] - 1) - 1}{stride[0]} + 1 \right\rfloor, \quad (3.9)$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2 \cdot padding[1] - dilation[1] \cdot (kernel_size[1] - 1) - 1}{stride[1]} + 1 \right\rfloor. \quad (3.10)$$

Po konvoliucijos sluoksnio dažniausiai taikomas jungimo (angl. *pooling*) sluoksnis, kurio paskirtis yra sumažinti savybių žemėlapio matmenis, išlaikant svarbiausią informaciją ir sumažinant skaičiavimų sudėtingumą. Dažniausiai naudojami du jungimo tipai: maksimalus jungimas (angl. *max pooling*) ir vidutinis jungimas (angl. *average pooling*). Maksimalus jungimas atrenka didžiausią reikšmę iš kiekvieno lokalaus bloko, taip pabrėždamas ryškiausius vaizdo elementus, o vidutinis jungimas apskaičiuoja reikšmių vidurkį, leidžiantį išsaugoti daugiau informacijos apie mažiau reikšmingas savybes.

Po požymių išskyrimo etapo duomenys perduodami į visiškai sujungtą (angl. *fully connected*) sluoksnį, kuris atlieka klasifikavimo funkciją. Kadangi konvoliucijos sluoksniai generuoja dvimačes matricas, jos pirmiausia transformuojamos į vienmatį vektorių. Visiškai sujungtame sluoksnyje atliekama linijinė transformacija (3.11), aprašoma lygtimi:

$$Z = W \cdot X + b, \quad (3.11)$$

kur W yra svorių matrica, X – įvesties vektorius, o b – nuokrypis.

Siekiant apdoroti sudėtingus netiesinius ryšius, po linijinės transformacijos taikoma aktyvacijos funkcija. Viena dažniausiai naudojamų aktyvacijos funkcijų CNN architektūroje yra ReLU (angl. *Rectified Linear Unit*)(3.12), kuri apibrėžiama taip:

$$f(x) = \max(0, x). \quad (3.12)$$

ReLU funkcija suteikia tinklui netiesiškumo, pagreitina mokymosi procesą ir padeda išvengti nykstančio gradiento problemos.

3.3.4. Iš anksto apmokyti CNN

Iš anksto apmokyti CNN modeliai yra neuroniniai tinklai, kurie buvo apmokyti naudojant didelį duomenų rinkinį, paprastai bendrai vaizdų atpažinimo užduočiai, ir yra prieinami kitiems naudoti. Šie modeliai fiksuoja sudėtingas struktūras ir savybes, todėl yra labai veiksmingi vaizdų klasifikavimui.

Iš anksto apmokyti modeliai yra svarbi šiuolaikinio giliojo mokymosi dalis. Šie modeliai iš pradžių apmokomi naudojant didelius, bendrosios paskirties duomenų rinkinius, pvz., ImageNet. Jie mokosi atpažinti įvairias savybes, nuo paprastų kraštų iki sudėtingų tekstūrų ir objektų. Šis išsamus mokymas leidžia jiems gerai apibendrinti, todėl jie tampa veiksmingu atspirties tašku naujoms užduotims. Tiksliai pritaikydami šiuos modelius konkretiems duomenų rinkiniams, kūrėjai gali pasiekti aukštą našumą naudodami mažiau duomenų ir skaičiavimų.

Iš anksto apmokytų modelių architektūra skiriasi, tačiau jie turi bendrų bruožų. Jie susideda iš kelių sluoksnių, kurie palaipsniui išgauna savybes iš įvesties vaizdų. Ankstyvieji sluoksniai fiksuoja žemo lygio savybes, o gilesnieji sluoksniai atpažįsta aukšto lygio struktūras. Iš anksto apmokyti modeliai gali būti pritaikyti įvairioms sritims, nuo medicininių vaizdų iki autonominio vairavimo. Dėl savo universalumo ir veiksmingumo jie yra neįkainojami įrankiai kompiuterinio matymo srityje.

Tačiau giliųjų neuroninių tinklų mokymas yra daug laiko reikalaujantis procesas ir jie yra linkę į persimokymą. Jiekiant išspręsti šį trūkumą, Microsoft komanda pristatė liekamąją mokymosi sistemą, skirtą pagerinti tinklų, kurie yra žymiai gilesni nei anksčiau naudoti, mokymą. Šis tyrimas buvo paskelbtas 2015 m. straipsnyje „Deep Residual Learning for Image Recognition” (Gilus liekamasis mokymasis vaizdų atpažinimui). Taip atsirado žinomas „ResNet” (trumpinys nuo „Residual Network”).

Pagrindinė problema, kurią išsprendė ResNet, buvo giliųjų neuroninių tinklų degradacijos problema. Tinklams tampant gilesniais, jų tikslumas pasiekia persisotinimo tašką ir tada sparčiai mažėja. Ši degradacija nėra susijusi su pernelyg dideliu pritaikymu, o su sunkumais optimizuojant mokymo procesą. ResNet išsprendė šią problemą naudodamas liekamuosius blokus (angl. *Residual blocks*),

kurie leidžia tiesiogiai perduoti informaciją per praleidžiamuosius ryšius. Šio tipo praleidimo jungties privalumas yra tas, kad jei kuris nors sluoksnis pakenks architektūros našumui, jis bus praleistas reguliarizavimo metu. Taigi, tai leidžia treniruoti labai gilų neuroninį tinklą be problemų, kurias sukelia išnykstantis/sprogstantis gradientas [36].

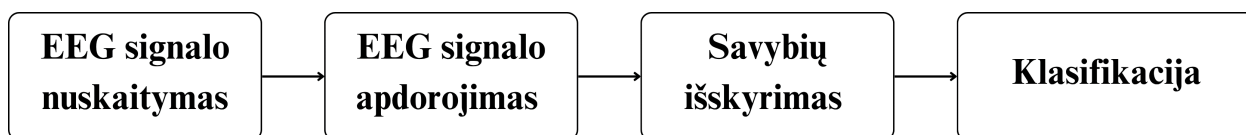
Kitas dažnai taikomas iš anksto apmokytas CNN modelis yra EfficientNet. EfficientNet yra CNN šeima, kurios tikslas – pasiekti aukštą našumą naudojant mažiau skaičiavimo išteklių, palyginti su ankstesnėmis architektūromis. Ją pristatė Mingxing Tan ir Quoc V. Le iš „Google Research“ savo 2019 m. straipsnyje [37].

Pagrindinė EfficientNet idėja yra naujas mastelio keitimo metodas, kuris vienodai keičia visus gylio, pločio ir skiriamosios gebos matmenis, naudojant sudėtinį koeficientą. EfficientNet naudoja techniką, vadinamą sudėtinio koeficientu, kad modelius būtų galima paprastai, bet efektyviai padidinti. Vietoj to, kad atsitiktinai didintų plotį, gylį ar skiriamąją gebą, sudėtinis mastelio keitimas vienodai keičia kiekvieną matmenį, naudojant tam tikrą fiksuotą mastelio koeficientų rinkinį.

Autoriai, prieš kurdami sudėtinį mastelio keitimo metodą, nuodugniai ištyrė kiekvienos mastelio keitimo strategijos poveikį modelio efektyvumui ir našumui. Jie priėjo prie išvados, kad nors vieno matmens mastelio keitimas gali padėti pagerinti modelio našumą, geriausias būdas padidinti bendrą modelio našumą yra subalansuoti visų trijų matmenų (pločio, gylio ir vaizdo skiriamosios gebos) mastelį, atsižvelgiant į kintamus turimus išteklius [38].

4. Metodika

Šiame darbe buvo taikomos statistinės ir mašininio mokymosi technikos epilepsijos priepuolių aptikimui iš EEG signalų. Tyrimo metodika apima tris pagrindinius etapus: duomenų paruošimą, savybių išskyrimą ir klasifikavimą. Pagrindinė tyrimo idėja – transformuoti neapdorotus EEG signalus į informatyvius požymius, leidžiančius efektyviai atskirti priepuolių ir ne priepuolių būsenas (žr. 3 pav.).



2 pav. Epilepsijos priepuolių aptikimo algoritmo struktūrinė schema

Tyrime buvo analizuojami ir tarpusavyje lyginami du skirtingi priepuolių aptikimo metodai. Pirmasis — klasikinis požymių inžinerijos metodas, kuriame savybės iš EEG signalų išgaunamos taikant diskrečiąją vilnelių transformaciją (DWT), o klasifikavimui naudojami tradiciniai mašininio mokymosi algoritmai: atraminių vektorių mašina (SVM) ir artimiausių kaimynų metodas (KNN).

Antrasis metodas — giliojo mokymosi pagrindu sukurtas modelis, kuriame iš EEG signalų generuojamos laiko–dažnio domeno reprezentacijos, taikant trumpojo lango Furjė transformaciją (STFT). Gauti spektrogramų vaizdai naudojami kaip įvestis konvoliuciniams neuroniniams tinklams (CNN), įskaitant tiek savarankiškai sukurtas architektūras, tiek iš anksto apmokytus modelius.

Toks dviejų skirtingų požiūrių taikymas leidžia įvertinti tiek klasikinio požymių inžinerijos metodo, tiek šiuolaikinių gilaus mokymosi sprendimų tinkamumą epilepsijos priepuolių aptikimo uždaviniui.

4.1. Duomenys ir tiriamieji

CHB-MIT galvos EEG duomenų bazė – tai 22 vaikų, sergančių sunkiai gydomais traukuliais, EEG įrašų rinkinys [39]. Ji dažnai naudojamas priepuolių aptikimo ar prognozavimo tyrimams. Duomenų bazė surinkta Bostono vaikų ligoninėje [40]. Visi tiriamieji buvo stebimi iki kelių dienų po vaistų nuo traukulių nutraukimo, kad būtų galima apibūdinti jų priepuolius ir įvertinti jų tinkamumą chirurginiam įsikišimui. Siekiant apsaugoti tiriamųjų privatumą, visa originaliuose įrašuose esanti saugoma sveikatos informacija (PHI) buvo pakeista pakaitine informacija. Galvos EEG įrašai buvo gauti naudojant 256 Hz diskretizavimo dažnį ir 16 bitų skiriamąją gebą. Įrašams buvo naudojama tarptautinė 10-20 EEG elektrodų padėčių ir nomenklatūros sistema.

Duomenų bazėje yra 24 galvos EEG įrašai, vadinami chb01, chb02, chb03 ir pan. Pirmaisiais 23 atvejais, išskyrus chb24 atvejį, buvo įrašyti 22 tiriamųjų (5 vyrų, 3-22 metų amžiaus, ir 17 moterų, 1,5-19 metų amžiaus) įrašai. Chb01 ir chb21 atvejai buvo gauti iš to paties tiriamojo. Iš viso užregistruoti 198 epilepsijos priepuoliai.

Kadangi CHB-MIT duomenų bazėje esantys duomenys yra labai nesubalansuoti (apytikriai iš viso 11 693 s ir 3 417 165 s traukulių ir netraukių atvejų atitinkamai) šiame tyrime buvo naudojamas viešai prieinamas, iš anksto apdorotas ir subalansuotas CHB-MIT duomenų rinkinys, gautas iš IEEE DataPort (<https://ieee-dataport.org/open-access/preprocessed-chb-mit-scalp-eeeg-database>). Duomenų rinkinys buvo sudarytas remiantis originaliais PhysioNet CHB-MIT EEG įrašais. Išankstinis apdorojimas apėmė subalansuotų prieš priepuolio ir priepuolio duomenų išgavimą iš 24 pacientų „.edf“ failų, kurių kiekvienas apėmė maždaug 68 min. (4098 s) EEG aktyvumo. Pažymėtina, kad kiekvienam pavyzdžiui buvo naudojami tik 10 sekundžių segmentai, todėl buvo gauti CSV failai, kurie yra tiesiogiai paruošti naudoti įvairiuose mašinų ir giliųjų mokymosi modeliuose. Kiekviename faile išlaikomi standartiniai 23 bipoliniai EEG kanalai, įrašyti 256 Hz dažnio imtimi, o paskutinėje skiltyje nurodyta klasės etiketė (0 – be priepuolių, 1 – su priepuoliais) [41].

4.2. Programavimo kalba ir bibliotekos

EEG duomenų analizei buvo naudojama python programavimo kalba. Python yra nemokama, atviro kodo programavimo kalba, kuri yra paprasta ir lengvai suprantama. Python turi platų bibliotekų ir plėtinių pasirinkimą, kurie palengvina darbą įvairiose srityse. Python ekosistemoje yra tokios bibliotekos kaip NumPy ir SciPy, kurios padeda efektyviai apdoroti ir analizuoti duomenis. Be to, Python integruojasi su mašininio mokymosi įrankiais, kaip scikit-learn, leidžiantčiais atlikti pažangias EEG duomenų analizes.

Pandas

Pandas yra atvirojo kodo duomenų tvarkymo ir analizės biblioteka, skirta Python programavimo kalbai, teikianti lanksčias duomenų struktūras bei funkcijas darbui su struktūrizuotais duomenimis. Ši biblioteka buvo naudojama EEG įrašų duomenims tvarkyti, valdyti ir organizuoti, taip pat išskirtoms savybėms paruošti tolesnei mašininio mokymosi analizei [42].

NumPy

NumPy yra pagrindinė skaitinių skaičiavimų biblioteka Python aplinkoje, palaikanti didelius daugiamačius masyvus ir matricas bei platų matematinių funkcijų rinkinį darbui su jais. Ši biblioteka buvo naudojama efektyvioms masyvų manipuliacijoms ir matematiniams skaičiavimams

atlikti, kurie yra būtini savybių išgavimui iš EEG signalų [43].

PyWavelets

PyWavelets yra Python biblioteka, skirta bangelinėms transformacijoms, kurios yra plačiai taikomos signalų apdorojime, ypač analizuojant nestacionarius ir pereinamuosius signalus. Šiame tyrime PyWavelets buvo naudojama EEG signalų diskrečiosioms vilnelių transformacijoms (DWT) atlikti, siekiant išgauti informatyvias signalo savybes [44].

SciPy

SciPy yra atvirojo kodo Python biblioteka, skirta moksliniams ir techniniams skaičiavimams. Ji apima įvairias funkcijas, tokias kaip optimizavimas, integravimas, interpoliavimas, signalų ir vaizdų apdorojimas bei kitos pažangios matematinės operacijos. Ši biblioteka buvo naudojama įvairiems skaičiavimo ir duomenų analizės etapams vykdyti, užtikrinant tikslų ir efektyvų EEG duomenų apdorojimą, reikalingą epilepsijos priepuolių aptikimui [45].

Scikit-learn

Scikit-learn yra Python skirta mašininio mokymosi biblioteka, siūlanti paprastas ir efektyvias duomenų analizės bei modeliavimo priemones, sukurtas remiantis „NumPy“, „SciPy“ ir „Matplotlib“ [46]. Šiame tyrime ši biblioteka buvo naudojama klasikiniams mašininio mokymosi metodams taikyti:

- **Atraminų vektorių klasifikatorius (SVC)** – klasifikavimo metodas, kuris veikia ieškodamas hiperplokštumos, geriausiai atskiriančios skirtingas klases savybių erdvėje. Šis klasifikatorius buvo naudojamas epilepsijos priepuolių ir nepriepuolinių būsenų atskyrimui.
- **Artimiausių kaimynų metodas (KNN)** – klasifikavimo algoritmas, grindžiamas principu, kad naujo duomenų taško klasė nustatoma pagal jam artimiausių mokymo pavyzdžių klases. KNN taip pat buvo taikytas priepuoliams ir nepriepuoliams diferencijuoti.

TensorFlow

TensorFlow yra atvirojo kodo biblioteka, skirta didelio masto skaitiniams skaičiavimams, mašininio ir giluminio mokymosi modelių kūrimui bei optimizavimui. Ji leidžia efektyviai apdoroti didelius duomenų kiekius ir spartinti modelių mokymą [47]. Ši technologija buvo naudojama giluminio mokymosi modelių įgyvendinimui šiame darbe.

- **Keras** – aukšto lygio TensorFlow API, suteikianti patogų ir produktyvų interfeisą giluminio mokymosi modeliams kurti, apmokyti ir optimizuoti. Šiame darbe Keras buvo naudojama pritaikyti iš anksto apmokytus konvoliucinius neuroninius tinklus (CNN), tokius kaip *ResNet50* ir *EfficientNetB0*.

PyTorch

PyTorch yra atvirojo kodo giluminio mokymosi sistema, jungianti Torch mašininio mokymosi biblioteką su Python pagrindu sukurta aukšto lygio API. Ji pasižymi lankstumu ir dinamine skaičiavimų grafų struktūra, todėl yra ypač tinkama neuroninių tinklų tyrimams ir eksperimentams.

Šiame darbe PyTorch buvo naudojama konvoliucinio neuroninio tinklo (CNN) modelio kūrimui ir mokymui [48].

4.3. Metrikos

Kuriant naujus ML modelius, būtina palyginti jų našumą su jau esamais modeliais. Vertinimas turi du tikslus: metodai, kurie neveikia gerai, gali būti atmesti, o tie, kurie atrodo perspektyvūs, gali būti toliau optimizuojami. Naudojant mašininio mokymosi algoritmus pirmiausia padalijame duomenis į mokymo ir bandymo rinkinius, mokymo duomenis naudojame modelio mokymui ir patvirtinimui, prognozuojame visus bandymo duomenų atvejus ir gautas prognozes lyginame su atitinkamomis bandymo rinkinio tikrosiomis vertėmis. Tokiu būdu galime įvertinti, ar naujo mašininio mokymosi modelio prognozės yra geresnės už žmogaus ar esamų modelių prognozes mūsų bandymo rinkinyje.

Binarinės klasifikacijos užduotyje duomenų atvejai paprastai prognozuojami kaip teigiami arba neigiami, taigi teigiamas žymėjimas interpretuojamas kaip ligos, anomalijos ar kito nukrypimo buvimas, o neigiamas atvejis šiuo atžvilgiu nesiskiria nuo bazinio lygio. Todėl kiekviena numatyta dvejetainė žymė turi keturias galimas reikšmes: tikrasis teigiamas (TP) yra teisingai numatytas teigiamas rezultatas, tikrasis neigiamas (TN) yra teisingai numatytas neigiamas rezultatas, klaidingas teigiamas (FP) yra neigiamas atvejis, numatytas kaip teigiamas, o klaidingas neigiamas (FN) yra teigiamas atvejis, numatytas kaip neigiamas. Supainiojimo matrica, kurioje yra TP, TN, FP ir FN stebėjimų skaičiai (3 pav.), gali būti naudojamas kelioms metrikoms apskaičiuoti, siekiant įvertinti binarinį klasifikatorių.

		Predicted	
		Negative	Positive
Actual	Negative	TN	FP
	Positive	FN	TP

3 pav. Supainiojimo matrica

Klasifikatorių efektyvumui įvertinti šiame darbe buvo naudojami šie rodikliai:

Tikslumas (angl. *Accuracy*) apibrėžiamas kaip visų teisingų prognozių santykis su bendru prognozių skaičiumi. Nors jis dažnai siejamas su modelio kokybe, tikslumas yra tik viena iš jos sudedamųjų dalių ir vien tik šis rodiklis nesuteikia išsamaus vaizdo apie modelio veikimą. Siekiant detaliau įvertinti modelio efektyvumą, naudojami ir kiti rodikliai, tokie kaip jautrumas (angl. *Sensitivity/ Recall*), kuris parodo teisingai atpažintų teigiamų atvejų dalį iš visų realiai teigiamų atvejų. Jis yra ypač svarbus sveikatos priežiūros srityje, pavyzdžiui, ligų diagnostikoje, kur itin svarbu sumažinti klaidingų neigiamų rezultatų skaičių. Kitas svarbus rodiklis – preciziškumas (angl. *Precision*), kuris nusako, kokia dalis visų modelio priskirtų teigiamų atvejų iš tikrųjų yra teisingi, todėl padeda sumažinti klaidingų teigiamų prognozių kiekį. Šių dviejų rodiklių pusiausvyrą apibendrina F1 rodiklis – jautrumo ir preciziškumo harmoninis vidurkis, leidžiantis subalansuotai įvertinti

modelio veikimą, ypač dirbant su nesubalansuotais duomenų rinkiniais, pavyzdžiui, nustatant retas ligas, kai neigiamų atvejų yra gerokai daugiau nei teigiamų.

$$\text{Tikslumas} = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.1)$$

$$\text{Preciziškumas} = \frac{TP}{TP + FP} \quad (4.2)$$

$$\text{Jautrumas} = \frac{TP}{TP + FN} \quad (4.3)$$

$$F1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (4.4)$$

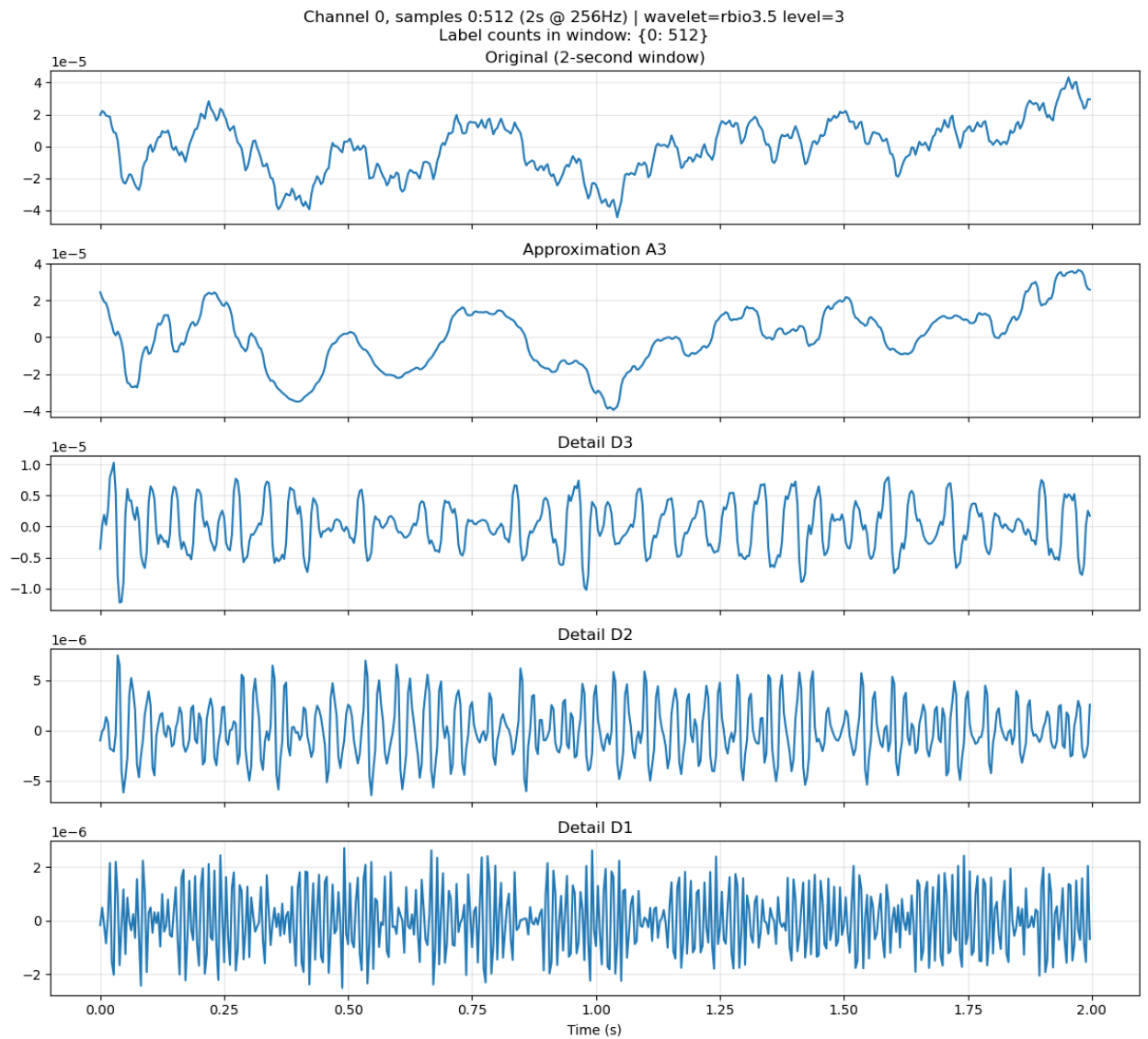
4.4. Epilepsijos priepuolių atpažinimas

Duomenų apdorojimas

Kadangi naudojami duomenys jau buvo subalansuoti ir iš jų pašalinti nereikalingi kanalai, pirmiausia buvo atliktas signalo triukšmo pašalinimas. Naudotas siaurajuostis slopinimo filtras (angl. notch filter), skirtas 50 Hz dažniui (ir jo harmonikoms) pašalinti, kurį dažniausiai sukelia elektros tinklo įtampa (Raghu, Shivarudhrappa, et al., 2020). Šis filtravimo žingsnis būtinas siekiant sumažinti fiziologinius ir techninius artefaktus, kaip ir rekomenduojama neurologinių EEG analizių standartuose. Remiantis ankstesnių tyrimų aprašytu duomenų apdorojimu, duomenų diskretizacijos dažnis buvo sumažintas iš 256Hz į 128Hz [22, 24]. Toliau gauti EEG įrašai buvo suskirstyti į 2 sekundžių epochas, siekiant pagerinti duomenų stacionarumą [49]. Gauta 4095 epochų, tarp kurių buvo 2048 be priepuolio ir 2047 su priepuoliu.

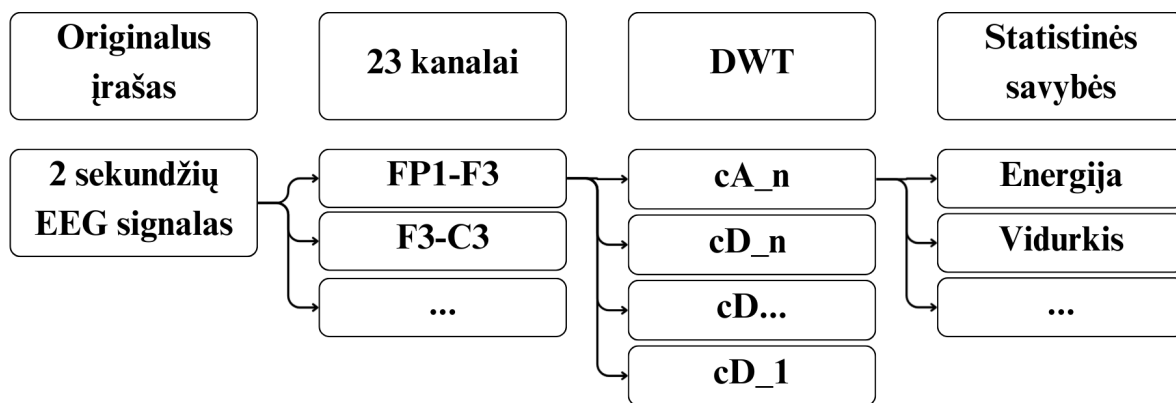
Pirmas modelis: DWT + SVM / KNN

Pirmajame modelyje EEG signalas buvo pirmiausia transformuojamas į kelių dažninių juostų reprezentaciją naudojant diskrečiąją vilnelių transformaciją (DWT), o po to iš kiekvienos juostos apskaičiuoti statistiniai požymiai. Kiekvienai 2 sekundžių epochai atlikta skirtingų šeimų (bior6.8, coif3, db10, dmey, haar, rbio3.5 ir sym5) bangų transformaciją su skirtingų lygių dekompozicija. Parametrai pasirinkti remiantis Chen et al. (2017) darbu, kuris parodė, kad mūsų tyrime naudotiems duomenims (CHB–MIT) išvardintos šeimos pasiekė geriausius rezultatus. Didžiausias galimas dekompozicijos lygis kiekvienai vilnelės šeimai buvo nustatomas naudojant funkciją `pywt.dwt_max_level`. Ši funkcija parenka didžiausią dekompozicijos lygį, kuriame bent vienas iš išvesties koeficientų nėra reikšmingai paveiktas kraštinių efektų, atsirandančių dėl signalo išplėtimo procedūros. Diskrečioji vilnelių transformacija buvo realizuota naudojant PyWavelets bibliotekos funkciją `wavedec`, kuri grąžina vieną aproksimacijos koeficientų rinkinį ir kelis detalių koeficientų rinkinius, atitinkančius pasirinktą dekompozicijos gylį (žr. 4 pav.).



4 pav. Originalus, 2 sekundžių trukmės, vieno kanalo EEG segmentas (viršuje) ir 3-lygių diskrečios vilnelės transformacijos koeficientai

Iš kiekvienos epochos gautų aproksimacijos ir vilnelės koeficientų buvo apskaičiuotos aštuonios statistinės savybės: energija, vidurkis, standartinis nuokrypis, dispersija, vidutinis absoliutus nuokrypis (MAD), maksimali vertė, minimali vertė, diapazonas (angl. *range*). Visos gautos savybės apjungtos į bendrą savybių vektorių atitinkamai duomenims be epilepsijos priepuolio ir su epilepsijos priepuoliu (5 pav.).



5 pav. Savybių išskyrimo diagrama: 1) Iš kiekvienos epochos išskiriami 23 EEG kanalai, 2) Kiekvienam kanalui pritaikoma DWT, 3) Iš kiekvieno gauto DWT koeficiento apskaičiuojamos statistinės savybės.

Klasifikavimui buvo taikoma atraminių vektorių mašina (SVM) ir K artimiausių kaimynų metodas (KNN). Siekiant objektyviai parinkti hiperparametrus, abiem klasifikatoriams buvo taikyta *GridSearchCV* procedūra su kryžmine patikra (CV), testuojant keletą parametrų kombinacijų: SVM atveju (C , γ , $kernel$), o KNN atveju ($n_neighbors$, $weights$, $metric$). Duomenys buvo padalinti santykiu 80/20 į mokymo ir testavimo aibes. Kadangi kiekvieno paciento epochos buvo išdėstytos nuosekliai, rankiniu būdu atrinktos pirmosios 1638 epochos be priepuolio ir su priepoliu, skirtos mokymui, bei sekančios 409 epochos testavimui, taip sumažinant tų pačių pacientų pasikartojimą tarp rinkinių. Gautų rinkinių požymiai buvo standartizuoti naudojant *StandardScaler*. Toks standartizavimas yra svarbus, nes tiek SVM (ypač su RBF branduoliu), tiek KNN jautriai reaguoja į skirtingų požymių skales.

Atlikus modelių mokymą ir optimizaciją, geriausius rezultatus pasiekė SVM klasifikatorius su parametrais $C = 10$, $\gamma = scale$ ir $kernel = rbf$, kuriam buvo pateiktos statistinės savybės, gautos taikant *rbio3.5* vilnelės transformaciją su 3 lygių dekompozicija. Šis modelis pasiekė 0.86 tikslumą, 0.91 preciziškumą, 0.80 jautrumą ir 0.85 F1 balą. Aukštas preciziškumas rodo, kad modelis patikimai prognozuoja epilepsijos priepuolius ir retai pateikia klaidingus teigiamus atvejus, t. y. nepriepuolinius EEG segmentus klaidingai priskiria priepuoliams. Tuo tarpu žemesnė jautrumo reikšmė parodo, kad nors dauguma priepuolių yra sėkmingai aptinkami, dalis tikrųjų teigiamų atvejų vis dar lieka neaptikti.

KNN klasifikatorius su parametrais $metric = manhattan$, $n_neighbors = 7$ ir $weights = uniform$ pasiekė geriausius rezultatus naudojant *rbio3.5* vilnelės transformacijos su 4 dekompozicijos lygiu gautas statistines savybes. Šio modelio rezultatai buvo: 0.87 tikslumas, 0.80 preciziškumas, 0.98 jautrumas ir 0.88 F1 balas. Lyginant su SVM, KNN pasižymėjo ženkliai didesniu jautrumu, t. y. itin efektyviai aptiko epilepsijos priepuolius ir beveik nepraleido teigiamų atvejų, kas yra ypač svarbu klinikinėje praktikoje. Tačiau mažesnis preciziškumas rodo, kad šis modelis generavo daugiau klaidingų teigiamų sprendimų, t. y. nepriepuolinius EEG segmentus dažniau klaidingai priskyrė priepuolių.

Apibendrinant, KNN modelis yra tinkamesnis užduotims, kuriose kritiškai svarbu maksimaliai aptikti epilepsijos priepuolius, kadangi jis pasižymi itin aukštu jautrumu (0.98) ir labai retai praleidžia teigiamus atvejus. Tuo tarpu SVM modelis labiau tinka scenarijams, kuriuose prioritetas teikiamas klaidingų pavojaus signalų mažinimui, nes jo aukštas preciziškumas (0.91) rodo, kad jis generuoja mažiau klaidingų teigiamų sprendimų. Šis kompromisas tarp jautrumo ir preciziškumo

aiškiai atskleidžia praktinius abiejų modelių taikymo skirtumus.

Diskretizacijos dažnio įtaka modeliui

Atliekant tyrimą, modelių mokymas ir optimizacija buvo vykdomi naudojant iš anksto sumažinto diskretizacijos dažnio (128 Hz) duomenis. Papildomai buvo patikrinta, ar šis sprendimas buvo pagrįstas, palyginus modelių veikimą su rezultatais, gautais naudojant originalaus diskretizacijos dažnio (256 Hz) signalus.

Gauti rezultatai (žr. 2 lentelę) parodė, kad diskretizacijos dažnio sumažinimas turėjo reikšmingą įtaką modelių veikimui, tačiau ši įtaka buvo nevienoda skirtingiems klasifikatoriams.

SVM klasifikatoriaus atveju, nors naudojant 256 Hz duomenis buvo pasiektas didesnis jautrumas (0.90), tačiau tai įvyko reikšmingai sumažėjus preciziškumui (0.79) ir bendram tikslumui (0.83), palyginti su modeliu, apmokytu naudojant 128 Hz signalus, kuris pasižymėjo aukštesniu tikslumu (0.86), preciziškumu (0.91) ir F1 balu (0.85). Tai leidžia teigti, kad sumažintas diskretizacijos dažnis SVM modeliui suteikė stabilesnį ir labiau subalansuotą veikimą.

KNN klasifikatoriaus rezultatai dar aiškiau patvirtina šio duomenų apdorojimo sprendimo pagrįstumą. Naudojant sumažinto diskretizacijos dažnio signalus (128 Hz), KNN pasiekė ženkliai geresnius rezultatus visose vertinimo metrikose, ypač jautrumo srityje, kuris padidėjo nuo 0.74 iki net 0.98. Tai rodo, kad duomenų diskretizacijos dažnio sumažinimas buvo esminis veiksnys, leidęs ženkliai pagerinti KNN modelio gebėjimą aptikti epilepsijos priepuolius.

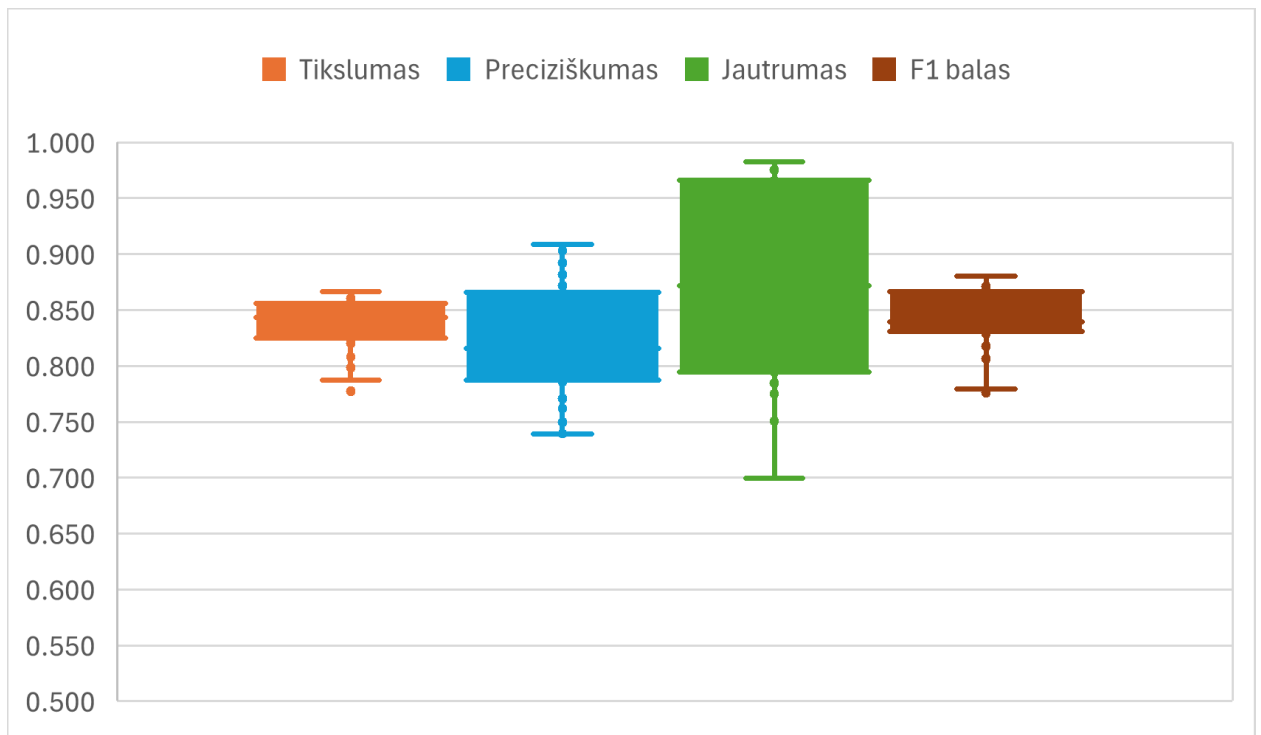
Apibendrinant galima teigti, kad tyrimo metu taikytas diskretizacijos dažnio sumažinimas buvo metodologiškai pagrįstas ir ne tik nesumažino, bet daugeliu atvejų netgi pagerino modelių veikimą, ypač KNN klasifikatoriaus atveju. Šis rezultatas pagrindžia pasirinkto duomenų paruošimo metodo tinkamumą epilepsijos priepuolių atpažinimo užduočiai.

2 lentelė. SVM ir KNN modelių rezultatai su skirtingu diskretizacijos dažniu

Modelis ir diskretizacijos dažnis	Tikslumas	Preciziškumas	Jautrumas	F1
SVM 128 Hz	0.86	0.91	0.80	0.85
SVM 256 Hz	0.83	0.79	0.90	0.84
KNN 128 Hz	0.87	0.80	0.98	0.88
KNN 256 Hz	0.78	0.81	0.74	0.78

Vilnelės pasirinkimo įtaka klasifikavimo rezultatams

Analizuojant rezultatų pasiskirstymą tarp skirtingų vilnelių šeimų ir dekompozicijos lygių, pastebima, kad vilnelės pasirinkimas turi nevienodą poveikį atskiroms vertinimo metrikoms. Skirtingų vilnelių naudojimas lemia reikšmingus metrikų reikšmių svyravimus, tačiau šių svyravimų intensyvumas nėra vienodas visoms metrikoms (žr. pav. 6).



6 pav. Vertinimo metrikų sklaida taikant skirtingas vilnelės savybių išskyrimui

Didžiausia sklaida stebima jautrumo metrikos atveju. Priklausomai nuo pasirinktos vilnelės ir dekompozicijos lygio, jautrumo reikšmės kinta labai plačiame intervale – nuo žemų reikšmių iki beveik maksimalių verčių. Tai rodo, kad būtent savybių išskyrimo strategija, paremta vilnelės parinkimu, labiausiai veikia modelio gebėjimą aptikti tikruosius teigiamus atvejus. Klinikiniu požiūriu šis aspektas yra ypač reikšmingas, nes sumažėjęs jautrumas tiesiogiai reiškia didesnį nepastebėtų epilepsijos priepuolių skaičių, o tai gali lemti neteisingą diagnozę, nepakankamą gydymo paskyrimą arba klaidingą paciento būklės įvertinimą. Tuo tarpu aukštas jautrumas užtikrina, kad didžioji dalis realių priepuolių bus aptikta, taip padedant gydytojams priimti patikimesnius sprendimus ir sumažinant riziką praleisti kliniškai reikšmingus įvykius.

Be to, didelė jautrumo reikšmių sklaida rodo, kad netinkamai parinkta vilnelė ar dekompozicijos lygis gali ženkliai sumažinti visos sistemos klinikinį patikimumą, net jei kitos metrikos išlieka santykinai aukštos. Tai patvirtina, jog savybių išskyrimo sprendimai turi būti vertinami ne tik pagal bendrą klasifikavimo tikslumą, bet ir pagal jų poveikį klinicinei diagnostinei vertei.

Tuo tarpu tikslumo, preciziškumo ir F1 balo metrikos pasižymi gerokai mažesne sklaida. Jų reikšmės išlieka santykinai stabilios nepriklausomai nuo vilnelės tipo, o tai leidžia teigti, kad bendras klasifikavimo veikimas yra mažiau jautrus vilnelės pasirinkimo pokyčiams nei atskirų klasių atpažinimo charakteristikos.

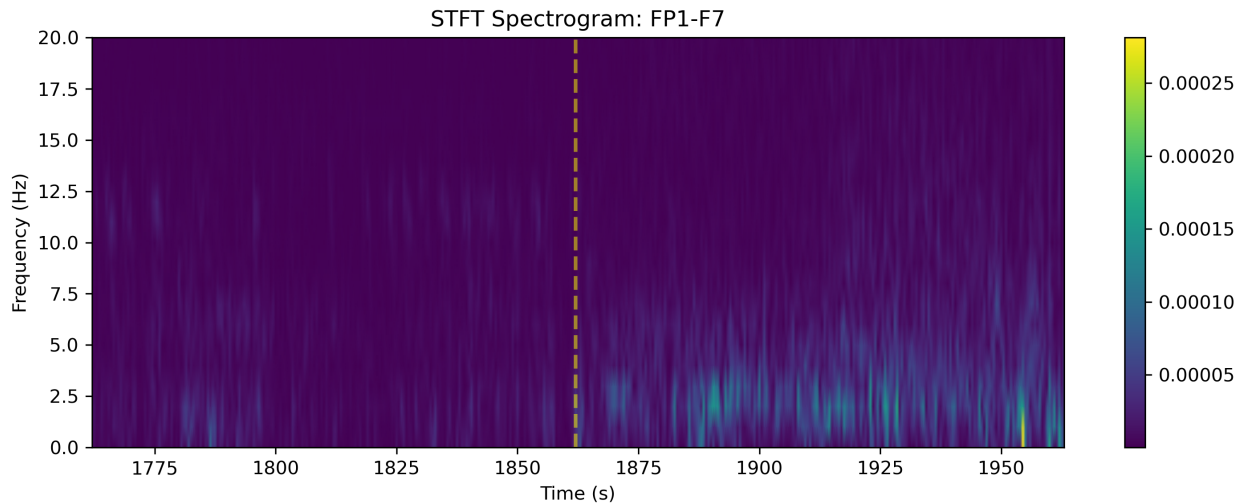
Apibendrinant, galima teigti, kad vilnelės pasirinkimas pirmiausia veikia ne pačią vidutinę modelio veikimo kokybę, o rezultatų stabilumą ir pasiskirstymą, ypač jautrumo metrikos atžvilgiu. Tai patvirtina, jog savybių išskyrimo metodo pasirinkimas yra svarbus ne tik rezultatų lygiui, bet ir jų patikimumui.

Antras modelis: STFT spektrogramos + CNN

Antrą modelį sudaro savybių išskyrimas naudojant trumpojo lango Furjė transformaciją (STFT) ir klasifikavimas taikant skirtingas konvoliucinių neuroninių tinklų (CNN) architektūras. Šiame

etape buvo lyginamas originalus (nuo nulio apmokytas) CNN modelis bei iš anksto apmokyti giliaji modeliai: *ResNet50* ir *EfficientNetB0*. Taip pat buvo tiriama, kaip atskirų CNN architektūros komponentų pridėjimas veikia klasifikavimo rezultatus.

Šiame tyrime EEG signalų laiko domeno duomenys buvo transformuoti į dažnio–laiko reprezentaciją taikant STFT metodą. Spektrogramos buvo generuojamos naudojant `scipy.signal.stft` funkciją su šiais parametrais: mėginių dažnis $f_s = 128$ Hz, lango ilgis $nperseg = 64$, persidengimas $noverlap = 32$. Tokia parametrų konfigūracija užtikrina kompromisą tarp laiko ir dažnio raiškos, kuris yra ypač svarbus nestacionariems EEG signalams. Kiekvienam EEG kanalui buvo sugeneruota atskira spektrograma, atspindinti spektrinės energijos pokyčius laike.



7 pav. Originali FP1-F7 kanalo, kuriam pritaikyta STFT, spektrograma, vertikali geltona linija žymi epilepsijos priepuolio pradžią

Duomenys buvo padalinti santykiu 70 %, 15 % ir 15 % į mokymo, validavimo ir testavimo rinkinius. Pirmosios 1433 epochos be priepuolio ir su priepuoliu, buvo skirtos mokymui, sekantios 307 epochos validavimui ir likusios 3070 – testavimui, taip sumažinant tų pačių pacientų pasikartojimą tarp rinkinių. STFT pagrįstos savybės buvo normalizuotos taikant kanalų z-score normalizaciją. Vidurkis ir standartinis nuokrypis buvo apskaičiuoti tik iš mokymo rinkinio, apjungiant visus mėginius, dažnių ir laiko intervalus, o vėliau tie patys parametrai pritaikyti validavimo ir testavimo duomenims. Tokia strategija užtikrina nuoseklų mastelį ir leidžia išvengti duomenų nutekėjimo (angl. *data leakage*).

Siekiant kompensuoti galimą priepuolių ir nepriepuolių klasių disbalansą, buvo naudojami klasės svoriai, apskaičiuoti pagal atvirkštinį klasės dažnį:

$$w_c = \frac{N}{2N_c},$$

kur N – bendras mokymo pavyzdžių skaičius, o N_c – pavyzdžių skaičius klasėje c . Šie svoriai buvo integruoti į kryžminės entropijos nuostolio funkciją, taip siekiant sumažinti mažumos klasės klaidingą klasifikavimą ir pagerinant jautrumą.

CNN modelio kūrimas ir testavimas

Remiantis Chhabra ir Goyal [2] pateikta metodika, šiame darbe buvo sukurta bazinė konvoliucinio neuroninio tinklo (CNN) architektūra. Autoriai išskiria keturis pagrindinius CNN struktūrinius

komponentus: įvesties sluoksnį, konvoliucinį sluoksnį, jungimo sluoksnį ir visiškai sujungtą sluoksnį. Šio darbo modelio įvestis sudaryta iš anksčiau apskaičiuotų STFT spektrogramų. Kiekvienas EEG kanalas transformuojamas į atskirą laiko–dažnio atvaizdą, o visos vienai epohai priklausančios 23 spektrogramos kartu sudaro vieną daugiakanalį įvesties pavyzdį.

Konvoliucinis sluoksnis realizuotas naudojant PyTorch bibliotekos `torch.nn.Conv2d` klasę, kuri įgyvendina dvimatę konvoliucijos operaciją ir leidžia išgauti erdvinius požymius iš kelių įvesties plokštumų. Išvesties kanalų skaičius (`out_channels`) buvo parinktas lygus 32, remiantis [50] pateiktais praktiniais CNN architektūrų projektavimo principais. Toks kanalų skaičius laikomas tinkamu kompromisu tarp modelio reprezentacinės galios ir skaičiavimo sudėtingumo, nes leidžia modeliui išmokti pakankamai sudėtingus požymius, kartu ribojant perteklinį parametrų skaičiaus augimą bei persimokymo riziką.

Konvoliucijos branduolio dydis buvo parinktas 3×3 , vadovaujantis standartine CNN projektavimo praktika, kuri užtikrina efektyvų lokalių struktūrų išgavimą iš įvesties duomenų, išlaikant stabilų mokymo procesą ir nedidelį parametrų skaičių. Siekiant išsaugoti kraštų informaciją ir išvengti pernelyg greito savybių žemėlapių erdvinio susitraukimo, buvo taikytas papildymas (angl. *padding*), kuris užtikrina, kad reikšmingi požymiai, esantys spektrogramų ribose, išliktų modelio mokymosi proceso dalimi.

Jungimo sluoksnis realizuotas taikant didžiausios reikšmės jungimą (angl. *max pooling*), kurio taikymas pagrįstas [29], kuriame nustatyta, jog daugumoje CNN architektūrų, naudojamų EEG požymių išskyrimui, taikomas būtent šis jungimo metodas.

Visiškai sujungtas sluoksnis suformuotas naudojant linijinės transformacijos (Linear) ir nelineinės aktyvacijos funkcijos (ReLU) derinį iš PyTorch bibliotekos. Šių sluoksnių parametrų parinkimas atliktas remiantis [50], pabrėžiančiu tinkamos sluoksnių konfigūracijos svarbą modelio konvergencijai ir klasifikavimo tikslumui. Apjungus visus šiuos sluoksnius, gauta architektūra:

- Features
 - Conv2d (23, 32, kernel=3, padding=1)
 - MaxPool2d (2)
- Flatten
- Classifier
 - Linear(1024, 128)
 - ReLU
 - Linear(128, 2)

Šis modelis pasiekė neblogus tikslumo, preciziškumo, jautrumo ir f1 balo rezultatus: 0.741, 0.664, 0.977 ir 0.791 atitinkamai. Rezultatai parodė, kad net ir paprasta architektūra geba išmokti pagrindinius spektrinius požymius, tačiau modelio gebėjimas apibendrinti sudėtingesnius raštus yra ribotas.

Modelio gerinimas

Siekiant pagerinti CNN modelio veikimą, buvo taikomi papildomi architektūriniai ir mokymo stabilizavimo metodai: ReLU aktyvacijos funkcija, grupės normalizacija bei išmetimo (angl. *dropout*) metodika.

Paprastai po kiekvienos konvoliucijos operacijos taikoma ReLU (Rectified Linear Unit) aktyvacijos funkcija. Ši funkcija suteikia modeliui netiesiškumą, kuris leidžia išmokti sudėtingus ryšius tarp įvesties duomenų požymių. Be to, ReLU padeda sumažinti nykstančio gradiento problemą, kuri dažnai pasireiškia gilesniuose tinkluose. ReLU veikimo principas yra paprastas – teigiamos įvesties reikšmės perduodamos be pakeitimų, o neigiamos reikšmės nustatomos lygios nuliui. Matematiškai ReLU funkcija apibrėžiama taip:

$$ReLU(x) = \max(0, x) \quad (4.5)$$

Normalizavimo sluoksniai buvo taikomi siekiant pagerinti mokymo stabilumą ir sumažinti vidinių aktyvacijų pasiskirstymo pokyčius tarp sluoksnių. Normalizavimas leidžia kiekvieno sluoksnio įvestims turėti nulinį vidurkį ir vienetinį standartinį nuokrypį, todėl tinklas tampa mažiau jautrus pradinių svorių pasirinkimui. Tai padeda greičiau pasiekti konvergenciją, leidžia naudoti didesnius mokymosi tempus ir sumažina mokymo proceso nestabilumo riziką. Be to, normalizavimas turi ir silpną regularizacinį efektą, kuris gali padėti sumažinti persimokymą.

Išmetimo metodas buvo taikomas kaip papildoma regularizacijos priemonė. Šio metodo esmė – mokymo metu atsitiktinai „išjungti“ dalį neuronų, taip neleidžiant tinklui pernelyg prisitaikyti prie konkrečių požymių ar neuronų kombinacijų. Tai skatina tinklą mokytis labiau apibendrintų savybių ir pagerina jo gebėjimą generalizuoti naujus, nematytus duomenis.

Siekiant surasti optimaliausią modelio architektūrą, kiekvieno įvardinto metodo reikšmė buvo įvertinta individualiai bei bendrai. Gauti rezultatai (3 lentelė)

3 lentelė. CNN modelio rezultai su skirtingomis architektūromis (paryškinta geriausias atitinkamo rodiklio rezultatas)

Modelio architektūra	Tikslumas	Preciziškumas	Jautrumas	F1
Conv + MaxPool	0.741	0.664	0.977	0.791
Conv + ReLU + MaxPool	0.645	0.587	0.981	0.734
Conv + MaxPool + Dropout	0.645	0.587	0.981	0.734
Conv + BatchNorm + MaxPool	0.827	0.764	0.948	0.846
Conv + ReLU + MaxPool + Dropout	0.744	0.667	0.977	0.793
Conv + BatchNorm + ReLU + MaxPool	0.787	0.709	0.974	0.820
Conv + BatchNorm + ReLU + MaxPool + Dropout	0.844	0.771	0.977	0.862

Įtraukus ReLU aktyvacijos funkciją (Conv + ReLU + MaxPool), jautrumas išlieka aukštas (0.981), tačiau pastebimai sumažėja bendras modelio tikslumas (64.5 %) ir F1 rodiklis (0.734). Panašūs rezultatai gauti ir pridėjus Dropout sluoksnį be papildomos normalizacijos, kas leidžia daryti prielaidą, jog šių dviejų komponentų nepakanka užtikrinti stabilų mokymąsi be papildomos architektūrinės optimizacijos.

Ženklus našumo pagerėjimas pasiektas naudojant Batch Normalization sluoksnį. Modelis Conv + BatchNorm + MaxPool pasiekė 82.7 % tikslumą ir 0.846 F1 rodiklį, kartu išlaikydamas aukštą jautrumą. Tai patvirtina, kad Batch Normalization efektyviai stabilizuoja mokymosi procesą ir pagerina modelio generalizacijos savybes.

Toliau plėtojant architektūrą ir sujungiant kelis optimizavimo metodus, gauti geriausi rezultatai naudojant pilną architektūrą – Conv + BatchNorm + ReLU + MaxPool + Dropout, kuri pasiekė

aukščiausią tikslumą, preciziškumą ir F1 balą, išlaikant aukštą jautrumą. Gauti rezultatai rodo, kad galutinis CNN modelis labai tiksliai aptinka epilepsijos priepuolius, ką patvirtina aukštas jautrumo rodiklis (0.977). Tuo pačiu metu preciziškumo reikšmė (0.771) leidžia teigti, kad modelis daro tam tikrą skaičių klaidų klasifikuodamas ne priepuolių būsenas kaip priepuolius, tačiau bendras rodiklių balansas rodo gerą modelio klasifikavimo patikimumą.

Taip pat buvo tiriama ar gilesnis modelis pasiektų geresnius rezultatus. Tam prie bazinio CNN modelio buvo pridetas dar vienas konvoliucinis sluoksnis. Atlikus šiuos pakeitimus, dabartinio modelio savybių išgavimo sluoksnio architektūra buvo:

- Conv2d (23, 32, kernel=3, padding=1)
- MaxPool2d (2)
- Conv2d(32, 64, kernel=3, padding=1)
- MaxPool2d (2)

Atlikus tokias pačias architektūros manipuliacijas, kaip prieš tai aptartame vieno konvoliucinio sluoksnio modelyje, gauti rezultatai:

4 lentelė. CNN modelio, su dviem konvoliuciniais sluoksniais, rezultai su skirtingomis architektūromis (paryškinta geriausias atitinkamo rodiklio rezultatas)

Modelio architektūra	Tikslumas	Preciziškumas	Jautrumas	F1
(Conv + MaxPool) ×2	0.640	0.584	0.974	0.730
(Conv + ReLU + MaxPool) ×2	0.809	0.733	0.974	0.836
(Conv + MaxPool) ×2 + Dropout	0.774	0.698	0.964	0.810
(Conv + BatchNorm + MaxPool) ×2	0.707	0.634	0.981	0.770
(Conv + ReLU + MaxPool) ×2 + Dropout	0.770	0.691	0.977	0.810
(Conv + BatchNorm + ReLU + MaxPool) ×2	0.679	0.611	0.987	0.755
(Conv + + BatchNorm + ReLU + MaxPool) ×2 + Dropout	0.774	0.696	0.971	0.811

Modelio gilinimas ne tik nepagerino rezultatų, bet daugeliu atvejų juos pablogino. Dvieju sluoksnių modeliai pasižymėjo didesniu nestabilumu: kai kurie variantai pasiekė ekstremaliai aukštą jautrumą (iki 0.987), tačiau kartu demonstravo ženkliai mažesnę preciziškumą ir tikslumą. Tai rodo didesnę klaidingai teigiamų sprendimų skaičių ir silpnesnę generalizaciją.

Šie rezultatai leidžia daryti išvadą, kad esamo duomenų rinkinio ir architektūrinių sprendimų kontekste vieno konvoliucinio sluoksnio CNN modelis yra optimalus, o papildomas gylys nesuteikia pranašumo ir gali net apsunkinti mokymosi procesą dėl per didelio modelio sudėtingumo.

Nustačius optimaliausią architektūrą buvo patikrinta, ar duomenų apdorojimas, konkrečiai, diskretizacijos dažnio sumažinimas, turi įtakos modelio rezultatams. Ši teorija buvo patikrinta pateikus STFT originalau diskretizacijos dažnio (256 Hz) duomenis. Modelis, apmokytas su 128 Hz duomenimis, pasiekė geresnius klasifikavimo rezultatus (5 lentelė). Tai galima paaiškinti tuo, kad mažesnis mėginių dažnis sumažina aukšto dažnio triukšmo įtaką ir leidžia modeliui sutelkti dėmesį į kliniškai reikšmingus žemesnio dažnio EEG komponentus.

5 lentelė. CNN modelio rezultatai su skirtingu diskretizacijos dažniu

Diskretizacijos dažnis	Tikslumas	Preciziškumas	Jautrumas	F1
128 Hz	0.844	0.771	0.977	0.862
256 Hz	0.710	0.635	0.987	0.773

Apmokytų CNN modelių testavimas

Siekiant nustatyti sukurto modelio kokybę, jis buvo palygintas su jau apmokytomis modeliais: ResNet50, ResNet50V2, EfficientNetB0 ir EfficientNetV2B0. Šie modeliai buvo pasirinkti dėl jų plataus paplitimo ir lengvo pritaikymo, kadangi „TensorFlow Keras“ biblioteka pateikia jų paruoštas aplikacijas, leidžiančias greitai ir patikimai taikyti perkėlimo mokymą (transfer learning). Visi modeliai buvo inicializuoti su ImageNet duomenimis apmokytomis svoriais, o jų konvoliucinė dalis naudota kaip savybių išgavėjas. Klasifikavimo galva buvo pritaikyta dvejetainės klasifikacijos užduočiai (ne priepuolis / priepuolis). Siekiant užtikrinti vienodas mokymo sąlygas ir sąžiningą palyginimą, visi modeliai buvo mokomi ne ilgiau nei 5 epochas, taikant ankstyvojo stabdymo (angl. *Early Stopping*) ir mokymosi greičio mažinimo (ReduceLROnPlateau) mechanizmus.

Geriausius rezultatus pasiekė ResNet50V2 modelis, kuris demonstravo 68.73 tikslumą, 0.7017 preciziškumą, 0.6873 jautrumą ir 0.6816 F1 rodiklį. Šie rezultatai rodo, kad ResNet50V2 architektūra užtikrina geriausią balansą tarp klaidingų teigiamų ir klaidingų neigiamų sprendimų bei pasižymi geriausia bendrąja klasifikavimo kokybe tarp tirtų modelių.

EfficientNetB0 modelis pasiekė 63.36 tikslumą, kartu demonstruodamas aukštą preciziškumą (0.7885), tačiau jo F1 rodiklis buvo tik 0.5767, kas leidžia daryti prielaidą apie nesubalansuotą modelio elgseną: nors modelis pakankamai patikimai identifikuoja teigiamus pavyzdžius, jis praleidžia dalį teigiamų atvejų, ką patvirtina santykinai mažas jautrumas (0.6336).

EfficientNetV2B0 architektūra parodė nuosaikius rezultatus: 59.45 tikslumą ir 0.5907 F1 rodiklį, su gana subalansuotais preciziškumo (0.5981) ir jautrumo (0.5945) rodikliais, tačiau bendras našumas išliko prastesnis nei EfficientNetB0 ir ResNet50V2.

Silpniausius rezultatus pateikė ResNet50 modelis, kuris pasiekė tik 51.14 tikslumą ir itin žemą 0.3736 F1 rodiklį, kas rodo labai ribotą šios architektūros tinkamumą sprendžiamam uždaviniui, nepaisant vidutinio preciziškumo (0.595).

6 lentelė. Iš anksto apmokytų CNN modelių rezultatai

Modelis	Tikslumas	Preciziškumas	Jautrumas	F1
ResNet50	0.511	0.595	0.511	0.374
ResNet50V2	0.687	0.702	0.687	0.682
EfficientNetB0	0.634	0.789	0.634	0.577
EfficientNetV2B0	0.595	0.598	0.595	0.591

Išvados ir rekomendacijos

Šiame darbe buvo siekiama sukurti modelį, gebantį, iš epilepsija sergančių pacientų gautų elektroencefalografijos duomenų, klasifikuoti duomenis į grupes: be epilepsijos priepuolio ir su epilepsijos priepuoliu. Kad pasiekti šį tikslą, pirmiausia buvo aprašyta epilepsijos apibrėžimas, klasifikacija ir diagnostika (žr. 1 skyrių) Taip pat aprašyta elektroencefalografija (žr 2 skyrių), tarptautinė 10 – 20 (žr. 2.1 skyrių) EEG sistema ir epilepsijos priepuolių EEG charakteristika (žr. 2.2. skyrių) ir kituose moksliniuose straipsniuose naudotos epilepsijos aptikimo sistemos (žr. 3 skyrių). Toliau darbe naudojant realius, epilepsija sergančių pacientų, duomenis (žr. 4.1 skyrių), buvo testuojami skirtingi epilepsijos aptikimo modeliai. Šiame darbe buvo gauti tokie rezultatai bei išvados:

- Taikant diskrečiąją vilnelių transformaciją (DWT) savybių išskyrimui, SVM klasifikatorius pasiekė 0.86 tikslumą, 0.91 preciziškumą, 0.80 jautrumą ir 0.85 F1 balą, o KNN klasifikatorius – atitinkamai 0.87, 0.80, 0.98 ir 0.88.
- Nustatyta, kad diskretizacijos dažnio sumažinimas nuo 256 Hz iki 128 Hz turėjo teigiamą poveikį tiek klasikinių mašininio mokymosi modelių (SVM, KNN), tiek neuroninių tinklų mokymui, pagerindamas jų stabilumą ir bendrus klasifikavimo rezultatus.
- Parodyta, kad vilnelės tipo ir dekompozicijos lygio parinkimas yra esminis veiksnys, turintis didelę įtaką klasifikavimo metrikų pasiskirstymui, ypač jautrumo reikšmei. Tai reiškia, jog savybių išskyrimo strategija tiesiogiai veikia sistemos klinikinį patikimumą.
- Originalus bazinis CNN modelis, apmokytas naudojant STFT, pasiekė 0.74 tikslumą, 0.67 preciziškumą, 0.98 jautrumą ir 0.80 F1 balą.
- Parodyta, kad papildomų CNN technikų taikymas savybių išskyrimo sluoksnyje — grupinė normalizacija, ReLU aktyvacijos funkcija ir išmetimo metodas — reikšmingai pagerino CNN modelio klasifikavimo rezultatus, padidindamas tikslumą iki 0.84, preciziškumą iki 0.77 ir F1 balą iki 0.86, išlaikant itin aukštą jautrumą (0.98). Tai patvirtina, jog tinkamas architektūrinių sprendimų parinkimas turi lemiamą įtaką CNN modelio veikimui.
- Nustatyta, kad CNN architektūros gilinimas ne tik nepagerino rezultatų, bet kai kuriais atvejais pablogino klasifikavimo veikimą, kas rodo, jog efektyvus architektūros projektavimas yra svarbesnis už jos sudėtingumo didinimą.
- Tarp iš anksto apmokytų konvoliucinių neuroninių tinklų geriausią bendrą veikimą parodė ResNet50V2, pasiekęs 0.69 tikslumą, 0.69 jautrumą ir 0.68 F1 balą, tačiau šie rezultatai vis tiek nusileido klasikiniams modeliams su tinkamai parinktomis savybėmis.
- Klinikinės diagnostikos požiūriu KNN modelis su DWT yra tinkamiausias scenarijams, kuriuose prioritetas teikiamas maksimaliam epilepsijos priepuolių aptikimui, kadangi jo jautrumas (0.98) užtikrina, jog beveik visi realūs priepuoliai bus aptikti. Tuo tarpu SVM modelis labiau tinka situacijoms, kuriose būtina sumažinti klaidingų pavojaus signalų skaičių, nes jo aukštas preciziškumas (0.91) leidžia patikimiau atskirti EEG segmentus su priepuoliai nuo be priepuolių.
- Apibendrinant galima teigti, kad labiausiai subalansuotas tikslumo, preciziškumo, jautrumo ir F1 balo santykis buvo pasiektas taikant KNN klasifikatorių kartu su diskrečiosios vilnelių

transformacijos pagrindu išgautomis savybėmis, todėl šis sprendimas gali būti laikomas perspektyviausiu praktiniam epilepsijos priepuolių atpažinimo taikymui.

Ateities tyrimų planai

Ateityje, naudojant tuos pačius duomenis, būtų tikslinga išplėsti tyrimą, analizuojant platesnį vilnelių šeimų ir dekompozicijos parametrų spektrą, įtraukiant daugiau vilnelių tipų.

Taip pat rekomenduojama į tyrimą įtraukti naujesnes iš anksto apmokytas konvoliucinių neuroninių tinklų architektūras, siekiant įvertinti, ar modernesni modeliai gali dar labiau pagerinti klasifikavimo rezultatus.

Galiausiai, tyrimą būtų tikslinga plėsti epilepsijos priepuolių aptikimo taikymo kryptimi klinikinėje diagnostikoje, siekiant įvertinti sukurto metodo praktinį naudingumą realiose medicininėse sąlygose.

Literatūra

- [1] A. T. Tzallas, M. G. Tsipouras, and D. I. Fotiadis, “Epileptic seizure detection in EEGs using time-frequency analysis,” *IEEE transactions on information technology in biomedicine: a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 13, pp. 703–710, Sept. 2009.
- [2] P. Chhabra and D. S. Goyal, “A Thorough Review on Deep Learning Neural Network,” in *2023 International Conference on Artificial Intelligence and Smart Communication (AISC)*, pp. 220–226, Jan. 2023.
- [3] S. Toraman, “Preictal and Interictal Recognition for Epileptic Seizure Prediction Using Pre-trained 2DCNN Models,” *Traitement du Signal*, vol. 37, pp. 1045–1054, Dec. 2020. Publisher: International Information & Engineering Technology Association (IIETA).
- [4] R. Ramos-Aguilar, J. A. Olvera-López, I. Olmos-Pineda, and S. Sánchez-Urrieta, “Feature extraction from EEG spectrograms for epileptic seizure detection,” *Pattern Recognition Letters*, vol. 133, pp. 202–209, May 2020.
- [5] R. S. Fisher, C. Acevedo, A. Arzimanoglou, A. Bogacz, J. H. Cross, C. E. Elger, J. Engel, L. Forsgren, J. A. French, M. Glynn, D. C. Hesdorffer, B. Lee, G. W. Mathern, S. L. Moshé, E. Perucca, I. E. Scheffer, T. Tomson, M. Watanabe, and S. Wiebe, “ILAE Official Report: A practical clinical definition of epilepsy,” *Epilepsia*, vol. 55, pp. 475–482, Apr. 2014.
- [6] W. H. Organization, “Epilepsy,” Feb. 2024.
- [7] S. T. Sarmast, A. M. Abdullahi, and N. Jahan, “Current Classification of Seizures and Epilepsies: Scope, Limitations and Recommendations for Future Action,” *Cureus*, vol. 12, p. e10549, Oct. 2020.
- [8] N. Moghim and D. W. Corne, “Predicting Epileptic Seizures in Advance,” *PLoS ONE*, vol. 9, p. e99334, June 2014.
- [9] S. Shorvon, R. Guerrini, M. Cook, and S. Lhatoo, *Oxford Textbook of Epilepsy and Epileptic Seizures*. OUP Oxford, Dec. 2012. Google-Books-ID: dpdpAgAAQBAJ.
- [10] S. Murakami and Y. Okada, “Contributions of principal neocortical neurons to magnetoencephalography and electroencephalography signals,” *The Journal of Physiology*, vol. 575, no. 3, pp. 925–936, 2006. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1113/jphysiol.2006.105379>.
- [11] H. Shen and Y. Yu, “Robust Evaluation and Comparison of EEG Source Localization Algorithms for Accurate Reconstruction of Deep Cortical Activity,” *Mathematics*, vol. 11, p. 2450, Jan. 2023. Number: 11 Publisher: Multidisciplinary Digital Publishing Institute.
- [12] S. Næss, G. Halnes, E. Hagen, D. J. Hagler, A. M. Dale, G. T. Einevoll, and T. V. Ness, “Biophysically detailed forward modeling of the neural origin of EEG and MEG signals,” *NeuroImage*, vol. 225, p. 117467, Jan. 2021.
- [13] A. Rayi and N. Murr, “Electroencephalogram,” in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2024.

- [14] T. C. Ferree, M. T. Clay, and D. M. Tucker, "The spatial resolution of scalp EEG," *Neurocomputing*, vol. 38-40, pp. 1209–1216, June 2001.
- [15] G. Rojas, C. Alvarez, C. Montoya Moya, M. de la Iglesia Vaya, J. Cisternas, and M. Gálvez, "Study of Resting-State Functional Connectivity Networks Using EEG Electrodes Position As Seed," *Frontiers in Neuroscience*, vol. 12, Mar. 2018.
- [16] A. Quintero-Rincon, C. D’Giano, and H. Batatia, "Artefacts Detection in EEG Signals," in *Advances in Signal Processing: Reviews Book Series, Volume 2*, pp. 413–441, Mar. 2021.
- [17] P. D. Emmady, R. M. D. Asuncion, and A. C. Anilkumar, "EEG Abnormal Waveforms," in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2025.
- [18] E. K. S. Louis, L. C. Frey, J. W. Britton, L. C. Frey, J. L. Hopp, P. Korb, M. Z. Koubeissi, W. E. Lievens, E. M. Pestana-Knight, and E. K. S. Louis, "Appendix 4. Common Artifacts During EEG Recording," in *Electroencephalography (EEG): An Introductory Text and Atlas of Normal and Abnormal Findings in Adults, Children, and Infants [Internet]*, American Epilepsy Society, 2016.
- [19] I. Kaya, "A Brief Summary of EEG Artifact Handling," July 2021.
- [20] S. Coelli, A. Calcagno, C. M. Cassani, F. Temporiti, P. Reali, R. Gatti, M. Galli, and A. M. Bianchi, "Selecting methods for a modular EEG pre-processing pipeline: An objective comparison," *Biomedical Signal Processing and Control*, vol. 90, p. 105830, Apr. 2024.
- [21] H. Khorrami and M. Moavenian, "A comparative study of DWT, CWT and DCT transformations in ECG arrhythmias classification," *Expert Systems with Applications*, vol. 37, pp. 5751–5757, Aug. 2010.
- [22] D. Chen, S. Wan, J. Xiang, and F. S. Bao, "A high-performance seizure detection algorithm based on Discrete Wavelet Transform (DWT) and EEG," *PLoS ONE*, vol. 12, p. e0173138, Mar. 2017.
- [23] M. Shen, P. Wen, B. Song, and Y. Li, "An EEG based real-time epilepsy seizure detection approach using discrete wavelet transform and machine learning methods," *Biomedical Signal Processing and Control*, vol. 77, p. 103820, Aug. 2022.
- [24] A. Bussalb, F. L. Gac, G. Jubien, M. Rahmouni, R. G. Bettinardi, P. M. R. d. Oliveira, P. Derambure, N. Gaspard, J. Jonas, L. Maillard, L. Vercueil, H. Vespignani, P. Laval, L. Koessler, and U. Gimenez, "Automatic detection of abnormal clinical EEG: comparison of a finetuned foundation model with two deep learning models," May 2025. arXiv:2505.21507 [q-bio].
- [25] I. B. Slimen, L. Boubchir, and H. Seddik, "Epileptic seizure prediction based on EEG spikes detection of ictal-preictal states," *Journal of Biomedical Research*, vol. 34, pp. 162–169, May 2020.
- [26] R. Nath Bairagi, M. Maniruzzaman, S. Pervin, and A. Sarker, "Epileptic seizure identification in EEG signals using DWT, ANN and sequential window algorithm," *Soft Computing Letters*, vol. 3, p. 100026, Dec. 2021.

- [27] S. E. Sánchez-Hernández, R. A. Salido-Ruiz, S. Torres-Ramos, and I. Román-Godínez, “Evaluation of Feature Selection Methods for Classification of Epileptic Seizure EEG Signals,” *Sensors (Basel, Switzerland)*, vol. 22, p. 3066, Apr. 2022.
- [28] J. Prasanna, M. S. P. Subathra, M. A. Mohammed, R. Damaševičius, N. J. Sairamya, and S. T. George, “Automated Epileptic Seizure Detection in Pediatric Subjects of CHB-MIT EEG Database—A Survey,” *Journal of Personalized Medicine*, vol. 11, p. 1028, Oct. 2021.
- [29] I. Rakhmatulin, M.-S. Dao, A. Nassibi, and D. Mandic, “Exploring Convolutional Neural Network Architectures for EEG Feature Extraction,” *Sensors (Basel, Switzerland)*, vol. 24, p. 877, Jan. 2024.
- [30] S. Talebi, “The Wavelet Transform. An Introduction and Example | by Shaw Talebi | . . .,” 2020.
- [31] Aayesha, M. B. Qureshi, M. Afzaal, M. S. Qureshi, and M. Fayaz, “Machine learning-based EEG signals classification model for epileptic seizure detection,” *Multimedia Tools and Applications*, vol. 80, pp. 17849–17877, May 2021.
- [32] N. R. Choudhury, S. S. Roy, A. Pal, S. Chatterjee, and R. Bose, “Epileptic Seizure Detection Employing Cross-Hyperbolic Stockwell Transform,” in *2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pp. 70–74, Nov. 2018.
- [33] “Chang: 2018 IEEE 9th international conference on... - Google” mokslinčius.”
- [34] S. Raghu, N. Sriraam, Y. Temel, S. V. Rao, A. S. Hegde, and P. L. Kubben, “Performance evaluation of DWT based sigmoid entropy in time and frequency domains for automated detection of epileptic seizures using SVM classifier,” *Computers in Biology and Medicine*, vol. 110, pp. 127–143, July 2019.
- [35] O. Karabiber Cura, S. Kocaaslan Atli, H. S. Türe, and A. Akan, “Epileptic seizure classifications using empirical mode decomposition and its derivative,” *BioMedical Engineering OnLine*, vol. 19, p. 10, Feb. 2020.
- [36] S. M. Vihar Kurama, “Deep Learning Architectures Explained: ResNet, InceptionV3, SqueezeNet | DigitalOcean,” Mar. 2025.
- [37] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” Sept. 2020. arXiv:1905.11946 [cs].
- [38] D.-K. Kim, “EfficientNet: Smarter, Not Just Bigger, Neural Networks,” Mar. 2025.
- [39] A. Shoeb, “CHB-MIT Scalp EEG Database,” 2010.
- [40] A. H. Shoeb, *Application of machine learning to epileptic seizure onset detection and treatment*. Thesis, Massachusetts Institute of Technology, 2009. Accepted: 2010-04-28T17:17:43Z.
- [41] B. Deepa and K. Ramesh, “Epileptic seizure detection using deep learning through min max scaler normalization,” *International journal of health sciences*, vol. 6, pp. 10981–10996, May 2022.

- [42] W. McKinney, “Data Structures for Statistical Computing in Python,” *SciPy 2010*, May 2010.
- [43] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, “Array programming with NumPy,” *Nature*, vol. 585, pp. 357–362, Sept. 2020. Publisher: Nature Publishing Group.
- [44] G. R. Lee, R. Gommers, F. Waselewski, K. Wohlfahrt, and A. O’Leary, “PyWavelets: A Python package for wavelet analysis,” *Journal of Open Source Software*, vol. 4, p. 1237, Apr. 2019.
- [45] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, and P. van Mulbregt, “SciPy 1.0: fundamental algorithms for scientific computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, Mar. 2020. Publisher: Nature Publishing Group.
- [46] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Duchesnay, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830, 2011.
- [47] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, and X. Zheng, “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,” Mar. 2016.
- [48] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimeshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” Dec. 2019. arXiv:1912.01703 [cs].
- [49] T. Alotaiby, F. E. A. El-Samie, S. A. Alshebeili, and I. Ahmad, “A review of channel selection algorithms for EEG signal processing,” *EURASIP Journal on Advances in Signal Processing*, vol. 2015, p. 66, Aug. 2015.
- [50] S. Mukherjee, “Writing CNNs from Scratch in PyTorch | DigitalOcean,” Aug. 2025.

Priedai

Dokumentą sudaro vienas priedas: lentelė, kurioje pateikta klasifikavimo našumas naudojant skirtingus bangų tipus, dekompozicijos lygius ir klasifikatorius. Duomenys lentelėje pateikti mažėjančia tvarka pagal modelio kokybę (nuo geriausio iki prasčiausio).

7 lentelė. Klasifikavimo našumas naudojant skirtingus bangų tipus, dekompozicijos lygius ir klasifikatorius

vilnelė	lygis	klasifikatorius	tikslumas	preciziškumas	jautrumas	F1 balas
bior6.8	2	k-NN	0.867	0.799	0.980	0.880
haar	5	k-NN	0.866	0.892	0.831	0.861
coif3	3	k-NN	0.866	0.820	0.936	0.874
coif3	2	k-NN	0.863	0.801	0.966	0.876
dmey	2	k-NN	0.863	0.796	0.976	0.877
haar	1	k-NN	0.862	0.904	0.809	0.854
rbio3.5	1	k-NN	0.862	0.805	0.956	0.874
haar	5	SVM	0.861	0.909	0.802	0.852
sym5	4	k-NN	0.858	0.818	0.922	0.867
sym5	2	k-NN	0.858	0.811	0.934	0.868
db10	2	k-NN	0.858	0.789	0.978	0.873
rbio3.5	3	k-NN	0.857	0.790	0.973	0.872
sym5	1	k-NN	0.856	0.816	0.919	0.864
rbio3.5	2	k-NN	0.856	0.788	0.973	0.871
haar	2	k-NN	0.853	0.816	0.912	0.861
bior6.8	1	k-NN	0.852	0.786	0.968	0.868
db10	3	k-NN	0.852	0.779	0.983	0.869
dmey	2	SVM	0.850	0.779	0.976	0.866
db10	1	k-NN	0.847	0.882	0.802	0.840
haar	4	k-NN	0.847	0.805	0.917	0.857
coif3	1	SVM	0.846	0.903	0.775	0.834
haar	3	k-NN	0.846	0.894	0.785	0.836
coif3	2	SVM	0.846	0.877	0.804	0.839
coif3	3	SVM	0.846	0.877	0.804	0.839
rbio3.5	4	k-NN	0.844	0.887	0.787	0.834
haar	6	SVM	0.844	0.801	0.914	0.854
rbio3.5	1	SVM	0.840	0.874	0.795	0.832
sym5	1	SVM	0.840	0.862	0.809	0.835
haar	2	SVM	0.840	0.858	0.814	0.836
bior6.8	1	SVM	0.837	0.863	0.802	0.831
db10	2	SVM	0.836	0.883	0.775	0.826
bior6.8	3	k-NN	0.836	0.767	0.966	0.855
haar	3	SVM	0.835	0.861	0.800	0.829
bior6.8	2	SVM	0.835	0.763	0.973	0.855
sym5	4	SVM	0.830	0.855	0.795	0.824
db10	3	SVM	0.828	0.864	0.778	0.819

vilnelė	lygis	klasifikatorius	tikslumas	preciziškumas	jautrumas	F1 balas
coif3	1	k-NN	0.828	0.754	0.973	0.850
haar	1	SVM	0.825	0.856	0.782	0.817
sym5	3	k-NN	0.825	0.762	0.946	0.844
sym5	3	SVM	0.820	0.843	0.787	0.814
sym5	2	SVM	0.815	0.826	0.800	0.812
dmey	1	k-NN	0.815	0.743	0.963	0.839
db10	1	SVM	0.814	0.827	0.795	0.811
haar	4	SVM	0.814	0.750	0.944	0.836
dmey	1	SVM	0.813	0.739	0.968	0.838
rbio3.5	2	SVM	0.808	0.813	0.800	0.806
rbio3.5	4	SVM	0.808	0.740	0.951	0.832
haar	6	k-NN	0.798	0.872	0.699	0.776
rbio3.5	3	SVM	0.787	0.810	0.751	0.779
bior6.8	3	SVM	0.778	0.771	0.790	0.780