



**Kauno technologijos universitetas**  
Matematikos ir gamtos mokslų fakultetas

# **Slenkančių langų metodikos tyrimas ir taikymas akcijų kainų analitikoje**

Magistro baigiamasis projektas

---

Projektą parengė

**Domantas Gražinskis**

Projektui vadovavo

**doc. dr. Lina Sinevičienė**

**prof. dr. Audrius Kabašinskas**

---

**Kaunas, 2026**



**Kauno technologijos universitetas**  
Matematikos ir gamtos mokslų fakultetas

# **Slenkančių langų metodikos tyrimas ir taikymas akcijų kainų analitikoje**

Magistro baigiamasis darbas  
Didžiųjų verslo duomenų analitika (6213AX001)

---

Projektą parengė  
**Domantas Gražinskis**

Projektui vadovavo  
**doc. dr. Lina Sinevičienė**  
**prof. dr. Audrius Kabašinskas**

Projektą recenzavo  
**doc. dr. Kristina Poškuvienė**  
**prof. dr. Lina Dagilienė**

---

**Kaunas, 2026**



**Kauno technologijos universitetas**

Matematikos ir gamtos mokslų fakultetas

Domantas Gražinskis

## **Slenkančių langų metodikos tyrimas ir taikymas akcijų kainų analitikoje**

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

1. baigiamąjį projektą parengiau savarankiškai ir sąžiningai, nepažeisdamas (-a) kitų asmenų autoriaus ar kitų teisių, laikydamasis (-i) Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymo, Kauno technologijos universiteto (toliau – Universitetas) intelektualinės nuosavybės valdymo ir perdavimo nuostatų bei Universiteto akademinės etikos kodekse nustatytų etikos reikalavimų;
2. visi baigiamajame projekte pateikti duomenys ir tyrimų rezultatai yra teisingi ir gauti teisėtai, nei viena projekto dalis nėra plagijuota nuo spausdintinių ar elektroninių šaltinių, o visos baigiamojo projekto tekste pateiktos citatos ir nuorodos yra nurodytos literatūros sąrašė;
3. baigiamajame projekte tinkamai laikiausi asmens duomenų apsaugos reikalavimų, nenaudojau neskelbtinų ar konfidencialių duomenų be teisėto pagrindo, o jei juos naudoju, jie yra tinkamai nuasmeninti;
4. jei rengiant baigiamąjį projektą naudojausi dirbtinio intelekto (toliau – DI) ar kitais automatizuotais įrankiais, juos taikiau pagal Universitete nustatytą tvarką, nepažeisdamas (-a) akademinio sąžiningumo principų;
5. nesumokėjau ir nesu įsipareigojęs (-usi) mokėti jokių įstatymų nenumatytų piniginių sumų už baigiamąjį projektą ar jo dalis jokiam fiziniam ar juridiniam asmeniui;
6. suprantu, kad išaiškėjus akademinio nesąžiningumo ar kitų asmenų teisių pažeidimo faktui, man bus taikoma atsakomybė pagal Universitete nustatytą tvarką ir galiu būti pašalintas (-a) iš Universiteto; akademinio nesąžiningumo atvejis gali būti nagrinėjamas ir po studijų baigimo, inicijuojant kvalifikacinio laipsnio atšaukimo procedūrą.

Gražinskis, Domantas. Slenkančių langų metodikos tyrimas ir taikymas akcijų kainų analitikoje. Magistro studijų baigiamasis projektas / projektui vadovavo: doc. dr. Lina Sinevičienė, prof. dr. Audrius Kabašinskas; Kauno technologijos universitetas, Matematikos ir gamtos mokslų fakultetas.

Studijų kryptis ir studijų kryptių grupė: Taikomoji matematika (Matematikos mokslai).

Reikšminiai žodžiai: Mašininis mokymasis, Slenkančių langų metodika, Grafų neuroniniai tinklai, Investicinio portfelio valdymas, Akcijų kainų prognozavimas.

Kaunas, 2026. 62 p.

## Santrauka

Baigiamajame magistro darbe nagrinėjamas mašininio mokymosi algoritmų pritaikymas akcijų kainų kryptių prognozavimui ir dinaminio investicinio portfelio valdymui. Tyrimo problema kildinama iš to, kad tradiciniai izoliuoti modeliai sunkiai prisitaiko prie finansinių laiko eilučių nestacionarumo bei ignoruoja aktyvų tarpusavio ryšius ir rinkos kontekstą. Darbo tikslas – pritaikyti slenkančių langų metodikos tyrimo rezultatus ir mašininio mokymosi algoritmus akcijų kainoms prognozuoti bei dinaminiam investiciniam portfeliui optimizuoti.

Tyrimui naudoti istorinių akcijų kainų duomenys, sudaryti iš 15-os S&P 500 indekso įmonių. Prognozavimo uždaviniams spręsti pasitelkti atraminių vektorių mašinų (SVM), gradientinio stiprinimo medžių (XGBoost), ilgosios trumpalaikės atminties (LSTM), valdomų rekurentinių blokų (GRU) bei grafų neuroninių tinklų (GNN) algoritmai. Modelių patikimumas buvo vertinamas taikant Diebold-Mariano statistinį testą bei išplėstines finansinės rizikos metrikas, tokias kaip Sharpe, Sortino, Calmar, Rachev rodikliai ir maksimalus nuosmukis (MDD).

Atliktas tyrimas atskleidė, kad GNN modeliui didžiausią stabilumą užtikrina trumpi apmokymo langai ir dažnas atnaujinimas ( $W=30$ ,  $S=5$ ), kas leidžia greitai užfiksuoti besikeičiančius ryšius tarp akcijų ir prisitaikyti prie kintančių rinkos sąlygų. Mikro lygmeniu GNN modelis efektyviausiai valdė riziką, pasiekdamas aukščiausius Calmar ir Rachev rodiklių vidurkius, bei statistiškai reikšmingai aplenkė izoliuotą LSTM modelį. Makro lygmeniu nustatyta, kad dinaminis GNN portfelis, naudojantis naiviosios diversifikacijos ( $1/N$ ) strategiją, pasiekė aukštesnį Sharpe rodiklį (1,15) už S&P 500 indeksą (0,70) ir pademonstravo daugiau nei tris kartus mažesnę kintamumą (6,68 % prieš 20,50 %). Tyrimo pabaigoje sukurtas hibridinis ansamblinis modelis (jungiantis GNN, LSTM ir XGBoost algoritmus) įrodė savo praktinį naudingumą sėkmingai filtruojant rinkos triukšmą ir generuojant prekybos signalus realiuoju laiku.

Gražinskis, Domantas. Research and Application of the Rolling Window Methodology in Stock Price. Master's Final Degree Project / supervisors: Assoc. Prof. Dr. Lina Sinevičienė and Prof. Dr. Audrius Kabašinskas; Faculty of Mathematics and Natural Sciences, Kaunas University of Technology.

Study field and study field group: Applied Mathematics (Mathematical Sciences).

Keywords: Machine learning, Rolling window methodology, Graph neural networks (GNN), Investment portfolio management, Stock price forecasting

Kaunas, 2026. 62 p.

## Summary

This Master's thesis examines the application of machine learning algorithms for forecasting stock price directions and dynamic investment portfolio management. The research problem stems from the fact that traditional isolated models struggle to adapt to the non-stationarity of financial time series while ignoring asset interdependencies and the broader market context. The aim of this study is to apply the results of the rolling window methodology research and machine learning algorithms to forecast stock prices and optimize a dynamic investment portfolio.

Historical stock price data of 15 companies from the S&P 500 index were used for the study. Support Vector Machines (SVM), Extreme Gradient Boosting (XGBoost), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Graph Neural Network (GNN) algorithms were employed to solve the forecasting tasks. The reliability of the models was evaluated using a rolling window validation strategy, the Diebold-Mariano statistical test, and advanced financial risk metrics such as the Sharpe, Sortino, Calmar, and Rachev ratios, as well as Maximum Drawdown (MDD).

The conducted research revealed that short training windows and frequent updates ( $W=30$ ,  $S=5$ ) ensure the highest stability for the GNN model, allowing it to quickly capture changing correlations between stocks and adapt to shifting market conditions. At the micro-level, the GNN model managed risk most effectively, achieving the highest average Calmar and Rachev ratios, and statistically significantly outperformed the isolated LSTM model. At the macro-level, it was found that the dynamic GNN portfolio, employing a naive diversification ( $1/N$ ) strategy, achieved a higher Sharpe ratio (1.15) than the S&P 500 index (0.70) and demonstrated more than three times lower volatility (6.68% versus 20.50%). Ultimately, the developed hybrid ensemble model (combining GNN, LSTM, and XGBoost algorithms) proved its practical utility by successfully filtering market noise and generating trading signals in real-time.

## Turinys

<b>Lentelių sąrašas .....</b>	<b>8</b>
<b>Paveikslų sąrašas .....</b>	<b>9</b>
<b>Įvadas .....</b>	<b>10</b>
<b>1. Literatūros analizė .....</b>	<b>12</b>
1.1. Akcijų gražos prognozavimo problematika ir vertinimo rodikliai .....	12
1.2. Finansinių laiko eilučių nestacionarumo suvaldymo metodai pasitelkiant mašininį mokymąsi .....	14
1.2.1. Grafų neuroninių tinklų (GNN) pranašumai prieš kitus modelius .....	15
1.2.2. Ansambliniai mašininio mokymosi modeliai ir hibridinės architektūros .....	16
1.3. Slenkančio lango metodikos taikymas dinamikos vertinimui .....	17
1.4. Modelių prognozavimo tikslumo ir finansinės rizikos vertinimo metrikos .....	20
1.5. Investicinių portfelių formavimo ir optimizavimo strategijos .....	22
1.6. Mašininio ir giliojo mokymosi modelių taikymo praktiniai iššūkiai .....	24
1.7. Literatūros analizės apibendrinimas ir baigiamojo projekto temos pagrindimas .....	25
<b>2. Tyrimo metodologija.....</b>	<b>26</b>
2.1. Tyrimo duomenys ir jų paruošimas.....	27
2.2. Tikslų kintamojo formavimas ir klasių apibrėžimas.....	28
2.2.1. Slenkančio lango metodika ir tyrimo etapai .....	30
2.3. Mašininio mokymosi modelių architektūra ir hiperparametrų optimizavimas .....	33
2.3.1. Grafų neuroniniai tinklai (GNN).....	34
2.3.2. Laiko eilučių giliojo mokymosi modeliai (LSTM ir GRU) .....	34
2.3.3. Klasikiniai mašininio mokymosi modeliai (XGBoost ir SVM).....	35
2.3.4. Ansambliniai modeliai ir „Minkštasis balsavimas“ .....	35
2.4. Modelių tikslumo ir finansinio veiksmingumo vertinimo metrikos.....	36
2.4.1. Klasifikavimo metrikos .....	36
2.4.2. Finansinio veiksmingumo ir rizikos rodikliai .....	37
2.4.3. Portfelio lyginamosios analizės rodikliai .....	38
2.4.4. Rezultatų statistinio reikšmingumo testai.....	38
<b>3. Tyrimo rezultatai ir jų aptarimas .....</b>	<b>39</b>
3.1. Slenkančio lango parametrų ir modelių architektūros optimizavimas .....	39
3.1.1. XGBoost langų ir slinkimo jautrumo analizė .....	39
3.1.2. GNN hiperparametrų paieška (I etapas).....	39
3.1.3. GNN slenkančio lango dinamikos kalibravimas (II etapas).....	40
3.2. Modelių lyginamoji analizė individualių aktyvų lygmeniu .....	42
3.2.1. Klasikinių ir giliojo mokymosi algoritmų efektyvumo vertinimas .....	42
3.2.2. Modelių tikslumo statistinis vertinimas (Diebold-Mariano testas).....	44
3.2.3. Išplėstinis strategijos rizikos vertinimas ne įprastomis rinkos sąlygomis .....	44
3.3. Dinaminio investicinio portfelio formavimas ir optimizavimas .....	45
3.3.1. Portfelio simuliacija taikant GNN signalus (1/N ir Markowitz strategijos) .....	45
3.3.2. Portfelio svorių optimizavimo metodų lyginamoji analizė (Markowitz prieš 1/N) .....	46
3.3.3. GNN portfelio lyginamoji analizė su rinkos atskaitos taškais .....	49
3.3.4. Portfelio gražų bei rizikos metrių stabilumo ir autokoreliacijos (ACF) vertinimas ..	51
3.4. Ansamblinių modelių taikymas ir realaus laiko prognozė akcijų kainoms .....	55
3.5. Tyrimo rezultatų lyginamoji analizė .....	58

<b>Išvados</b> .....	<b>59</b>
<b>Literatūros sąrašas</b> .....	<b>61</b>
<b>Priedai</b> .....	<b>63</b>
1 priedas .....	63
2 priedas .....	64
3 priedas .....	65

## Lentelių sąrašas

<b>1 lentelė.</b> Pažangių prognozavimo modelių tyrimų apžvalga .....	20
<b>2 lentelė.</b> Investicinių portfelių formavimo strategijų apžvalga .....	23
<b>3 lentelė.</b> Tyrime naudojamų finansinių aktyvų sąrašas ir jų pasiskirstymas pagal sektorius .....	28
<b>4 lentelė.</b> Slenkančio lango parametrų ( $W$ ir $S$ ) optimizavimo rezultatai pagal vidutinį Sharpe rodiklį .....	39
<b>5 lentelė.</b> Modelio hiperparametrų optimizavimo rezultatai pagal vidutinį Sharpe rodiklį ..	40
<b>6 lentelė.</b> Skirtingų prognozavimo modelių efektyvumo palyginimas atskiroms akcijoms pagal Sharpe ir Rachev rodiklius .....	42
<b>7 lentelė.</b> GNN portfelio ( $1/N$ ) ir S&P 500 indekso veiklos rezultatų vertinimas .....	49
<b>8 lentelė.</b> Atskirų akcijų kainos judėjimo krypties tikimybės ir ansamblio sprendimai .....	57

## Paveikslų sąrašas

<b>1 pav.</b> Tyrimo eigos blokinė schema .....	27
<b>2 pav.</b> Klasifikavimo metodikos vizualizacija .....	30
<b>3 pav.</b> Slenkančio lango metodas laiko eilučių prognozavimui.....	31
<b>4 pav.</b> GNN modelio Sharpe rodiklio priklausomybė nuo apmokymo lango (W) ir slinkimo žingsnio (S) dydžių .....	41
<b>5 pav.</b> Pilnas dinaminio portfelio svorių pasiskirstymas taikant GNN ir Markowitz optimizavimą (2019–2025 m.) .....	47
<b>6 pav.</b> GNN portfelio rotacija COVID-19 pandemijos metu .....	47
<b>7 pav.</b> GNN portfelio rotacija 2022 m. krizės metu .....	48
<b>8 pav.</b> GNN portfelio rotacija DI bumo laikotarpiu .....	48
<b>9 pav.</b> Skirtingų parametrų (W ir S) GNN portfelių ir tradicinių investavimo strategijų veiklos rezultatų palyginimas.....	50
<b>10 pav.</b> GNN Markowitz portfelio grąžų autokoreliacija (ACF) .....	51
<b>11 pav.</b> Optimalaus GNN portfelio rizikos rodiklių kaita laike .....	52
<b>12 pav.</b> Sharpe rodiklio autokoreliacija keičiant modelio apmokymo ir slinkimo langus ....	52
<b>13 pav.</b> Rachev rodiklio autokoreliacija keičiant modelio parametrus .....	53
<b>14 pav.</b> Rizikos rodiklių (Sharpe ir Rachev) autokoreliacija taikant visiškai nepersidengiančius apmokymo langus. ....	54
<b>15 pav.</b> Dinaminė slenkančio lango koreliacija GNN prieš S&P 500.....	55
<b>16 pav.</b> AAPL akcijos signalai pagal ansamblinį modelį (GNN + LSTM + XGBoost) .....	56
<b>17 pav.</b> AAPL akcijos sukauptos grąžos palyginimas: ansamblinis modelis prieš rinką....	56

## Ivadas

**Temos aktualumas.** Finansų rinkų dinamikos modeliavimas ir akcijų gražos prognozavimas išlieka vienu sudėtingiausių iššūkių finansų moksle ir duomenų moksle. Šis procesas reikalauja apdoroti didelius kiekius nuolat kintančių duomenų. Nors klasikine efektyvios rinkos hipoteze (EMH) teigiama, kad kainų judėjimas yra atsitiktinis, modernūs elgsenos finansų tyrimai bei didžiųjų duomenų analitikos technologijų plėtra atskleidė rinkos dėsningumus. Siekiant juos išnaudoti, pastarajame dešimtmetyje vis dažniau atsisakoma tradicinių tiesinių modelių ir pereinama prie pažangių mašininio mokymosi (angl. *Machine Learning*) bei giliojo mokymosi (angl. *Deep Learning*) algoritmų.

**Mokslinė problema.** Nepaisant mašininio mokymosi modelių populiarumo, jų taikymas realiose finansų rinkose susiduria su problemomis. Pirma, laiko eilučių analizėje dažniausiai naudojami izoliuoti modeliai (pvz., sprendimų medžiai ar LSTM neuroniniai tinklai), kurie analizuoja kiekvieną akciją atskirai, taip ignoruojant svarbią aktyvų tarpusavio priklausomybę ir visos rinkos kontekstą. Antra, finansinės laiko eilutės yra nestacionarios – rinkos režimai nuolat kinta, gali susidaryti krizės. Statiniai modeliai, apmokyti senais istoriniais duomenimis, negeba prisitaikyti prie šių pokyčių. Nors literatūroje siūloma naudoti Grafų neuroninius tinklus (GNN) tarpusavio ryšiams vertinti bei slenkančio lango (angl. *Rolling window*) metodiką duomenų atnaujinimui, vis dar trūksta tyrimų, apjungiančių šias technologijas ir įvertinančių jų praktinį efektyvumą naudojant finansinės rizikos metrikas.

**Tyrimo objektas:** akcijų kainų krypčių prognozavimas ir dinaminio investicinio portfelio valdymas, taikant mašininio mokymosi algoritmus bei slenkančio lango metodą.

**Tyrimo tikslas:** pritaikyti slenkančių langų metodikos tyrimo rezultatus ir mašininio mokymosi algoritmus akcijų kainoms prognozuoti bei dinaminiam investiciniam portfeliui optimizuoti.

### Tyrimo uždaviniai:

1. Išnagrinėti akcijų gražos prognozavimo problematiką, finansinių laiko eilučių nestacionarumo suvaldymo metodus bei investicinių portfelių optimizavimo strategijas mokslinėje literatūroje.
2. Suprojektuoti tyrimo metodologiją, kuri apima slenkančio lango pritaikymą, duomenų paruošimą bei mašininio mokymosi modelių (SVM, XGBoost, LSTM, GRU, GNN) ir ansamblinio modelio sudarymą.
3. Atlikti hiperparametrų ir slenkančio lango (apmokymo lango  $W$  ir slinkimo žingsnio ( $S$ ) kalibravimą bei palyginti skirtingų architektūrų prognozavimo efektyvumą mikro lygmeniu analizuojant individualius aktyvus.
4. Įvertinti geriausiai pasirodžiusio modelio (GNN) signalų praktinį pritaikymą makro lygmeniu, suformuojant investicinį portfelį, bei ištestuoti ansamblinio modelio efektyvumą realaus laiko prekybos prognozėms.

**Tyrimo metodai.** Darbe naudojama atviro kodo *Python* programavimo kalba su *Pandas*, *NumPy*, *Scikit-learn* bei *PyTorch* bibliotekomis. Istoriniai duomenys renkami naudojant *Yahoo Finance API*. Prognozavimo uždaviniams spręsti pasitelkiami atraminių vektorių mašinų (SVM), gradientinio stiprinimo medžių (XGBoost), ilgosios trumpalaikės atminties

(LSTM), valdomų rekurentinių blokų (GRU) bei grafų neuroninių tinklų (GNN) algoritmai. Modelių patikimumas vertinamas taikant slenkančio lango validacijos strategiją, Diebold-Mariano statistinį testą ir finansines metrikas (Sharpe, Sortino, Calmar, Rachev, MDD).

**Darbo struktūra.** Baigiamąjį magistro projektą sudaro trys pagrindinės dalys. Pirmojoje dalyje apžvelgiama akcijų grąžos prognozavimo problematika bei portfelių optimizavimo strategijos. Antrojoje dalyje (Tyrimo metodologija) formalizuojama duomenų apdorojimo, slenkančio lango taikymo ir modelių architektūrų matematinė logika bei apibrėžiamos vertinimo metrikos. Trečiojoje dalyje (Tyrimo rezultatai ir jų aptarimas) pateikiami empirinio tyrimo rezultatai: optimizuojami modelių parametrai, atliekama algoritmų lyginamoji analizė mikro ir pasirinktų akcijų lygmenimis, tiriama strategijos autokoreliacija bei pristatomas ansamblinio modelio pritaikymas realaus laiko prekybos prognozėms.

## 1. Literatūros analizė

Šiame skyriuje apžvelgiama mokslinė literatūra, susijusi su akcijų gražos prognozavimo problematika, bei dinaminių duomenų analizės metodų, tokių, kaip slenkančio lango metodika, taikymu. Analizė atliekama apžvelgiant teorinius pagrindus ir empirinius tyrimus, siekiant pagrįsti magistriniame darbe taikomą metodologiją.

### 1.1. Akcijų gražos prognozavimo problematika ir vertinimo rodikliai

Akcijų gražos prognozavimas išlieka vienu kompleksiškiausių ir plačiausiai nagrinėjamų uždavinių finansų ekonometrikoje [1]. Tikslus būsimos finansinio turto vertės ar jos pokyčio krypties numatymas suteikia reikšmingą konkurencinį pranašumą investuotojams ir portfelių valdytojams, leisdamas efektyviau paskirstyti kapitalą bei valdyti sisteminę riziką. Tačiau finansų rinkų prigimtis lemia, kad šis procesas yra itin kompleksiškas.

Pagrindinė akcijų gražos prognozavimo problematika kildinama iš Efektyvios rinkos hipotezės (angl. *Efficient Market Hypothesis*, EMH), kurią 1970 m. Suformulavo Eugene Fama [2]. Pagal šią teoriją teigiama, kad finansų rinkos yra efektyvios – visų aktyvų kainos bet kuriuo laiko momentu jau atspindi visą viešai prieinamą informaciją. Jei EMH griežtoji forma yra teisinga, akcijų kainų judėjimas atitinka atsitiktinio klaidžiojimo (angl. *Random Walk*) modelį, o tai reiškia, kad sistemingai „įveikti rinkos“ ir gauti pelną naudojant istorinius ar svarbius duomenis yra neįmanoma [2].

Nepaisant EMH populiarumo, vėlesni elgsenos finansų (angl. *Behavioral Finance*) ir rinkos mikrostruktūros tyrimai atskleidė esminius jos apribojimus [3]. Dėl informacijos asimetrijos, investuotojų iracionalumo, „bandos“ instinkto ir institucinių apribojimų, rinkose nuolat susidaro anomalijos ir netiesiniai dėsniniai. Net ir atmetus griežtąją EMH versiją, prognozavimas išlieka ypač sudėtingas, nes finansinės laiko eilutės pasižymi dideliu triukšmingumu (angl. *noise*), staigiais struktūriniais lūžiais bei nestacionarumu (statistinių savybių kitimu laike).

Siekiant suvaldyti šį neapibrėžtumą ir modeliuoti akcijų gražą (kuri dažniausiai matuojama kaip logaritminė graža arba paprastas dienos kainos pokytis), mokslinėje literatūroje pasitelkiamas platus spektras vertinimo rodiklių. Pagrindines tyrimų kryptis pagal naudojamus rodiklius galima suskirstyti į kelias kategorijas:

- **Fundamentalią analizę:** tyrimuose dažnai integruojami įmonės vidutinio ir ilgojo laikotarpių vertės rodikliai, pavyzdžiui, [4] savo tyrime sėkmingai panaudojo tokius rodiklius kaip pelnas vienai akcijai (EPS), kainos ir pelno santykis (P/E) bei nuosavo kapitalo graža (ROE), siekdami pagerinti ilgalaikes prognozes. Prie svarbių rodiklių taip pat priskiriami makroekonominiai veiksniai (palūkanų normos, infliacija), veikiantys visą rinką.
- **Techninė analizė:** tai rinkos tyrimo metodas, kurio metu yra analizuojama istorinės kainos ir prekybos apimtys bei jų dėsniniai [5]. Mašininio mokymosi prognozavimo modeliuose techninė analizė leidžia mums paruošti duomenis ir atlikti požymių inžinerijos (angl. *feature engineering*) vaidmenį. Nesutvarkytos finansinės laiko eilutės pasižymi dideliu triukšmingumu ir dažnai bei greitai kintančiomis savybėmis, todėl tiesioginis

istorinių kainų naudojimas modelių mokymui dažnai lemia algoritmų persimokymą (angl. overfitting). Kad to išvengtume, istoriniai duomenys yra transformuojami į matematinius indikatorius: tendencijų sekimui naudojami slankieji vidurkiai (SMA, EMA), o norint nustatyti, ar akcija nėra laikinai pervertinta arba pernelyg nuvertinta, naudojami ribose svyruojantys rodikliai - oscilatoriai (pvz., RSI, MACD). Remiantis [5] tyrimais, toks techninių indikatorių integravimas į modelius pagerina prognozių tikslumą, nes padeda algoritmams atskirti tikrąją rinkos inerciją (angl. momentum) nuo atsitiktinių kainų svyravimų.

Mašininio mokymosi modeliai, nors ir yra galingi, geriausius rezultatus pasiekia tada, kai jiems pateikiami „apdoroti“ duomenys. Tiesioginė akcijos kaina modeliui dažnai yra per daug triukšminga, todėl pasitelkiami techniniai indikatoriai. Jų pagrindinė užduotis – išskirti pasikartojančius rinkos dėsningumus. Pagal savo prigimtį šie rodikliai skirstomi į kelias esmines grupes, kurios padeda modeliui „pamatyti“ skirtingus rinkos aspektus:

1. **Inercijos (angl. *Momentum*) rodikliai.** Šios grupės rodikliai, pavyzdžiui, RSI, padeda modeliui suprasti tendencijos stiprumą. Finansų rinkose galioja taisyklė: jei kaina juda kuria nors kryptimi su didele inercija, didelė tikimybė, kad ji judės ta pačia kryptimi ir artimiausiu metu [5]. Mašininio mokymosi algoritmas, naudodamas šiuos duomenis, gali atpažinti, kada rinka yra „pavargusi“ (perpirkta arba perparduota) ir kada tikėtinas kainos apsisukimas.
  2. **Kintamumo (angl. *Volatility*) rodikliai.** Kintamumas parodo rinkos baimės ir neapibrėžtumo lygį. Naudodamas tokius rodiklius kaip standartinis nuokrypis, modelis sužino, ar dabartinis kainos šuolis yra įprastas, ar tai yra stiprus rinkos šokas. Mašininiam mokymui tai svarbu, nes padeda algoritmui „nuspręsti“, kiek pasitikėti gautu signalu: esant dideliame kintamume, signalai dažnai būna klaidingi, todėl modelis gali pasirinkti saugesnę poziciją. [3]
  3. **Prekybos apimtys (angl. *Volume*) rodikliai.** Apimtis yra tarsi degalai kainos judėjimui. Jei kaina kyla, bet prekybos apimtis mažėja, modelis gali „suprasti“, kad pirkėjų susidomėjimas blėsta ir kainos kilimas yra netikras [5]. Įtraukus apimtys duomenis, algoritmas gali patvirtinti kainos judesio tikrumą ir geriau atskirti svarbius rinkos įvykius nuo atsitiktinių svyravimų.
- **Alternatyvūs duomenys ir sentimentų analizė:** tobulėjant didžiųjų duomenų (angl. *Big Data*) analitikai, vis dažniau modeliai papildomi elgsenos komponentais. [6] atliktoje apžvalgoje pabrėžiama natūralios kalbos apdorojimo (NLP) metodų svarba vertinant investuotojų nuotaiką iš socialinių tinklų ir tai leido gerokai patikslinti svyravimų prognozes.

Nagrinėjant kokius duomenis mokslininkai dažniausiai testuoja savo prognozavimo modelius pastebima įvairovė, atspindinti skirtingą rinkų efektyvumo lygį. Vertinant rinkų efektyvumą, tyrėjai ([1], [7]) dažnai pasirenka stambiausius išsivysčiusių rinkų indeksus (pvz., JAV „S&P 500“), siekdami analizuoti itin likvidžius ir kokybiškus duomenis. Kita vertus, norint patikrinti modelių gebėjimą rasti rinkos neefektyvumus, tyrimai dažnai perkeliama į besivystančias ar specifines rinkas: [8] naudojo Azijos akcijų rinkų duomenis, [9] modeliavo Japonijos akcijų rinką, o kiti autoriai, tokie kaip [10], savo metodikas adaptavo dideliu kintamumu pasižyminčioms kriptovaliutų (pvz., Bitkoino) rinkoms.

Apibendrinant galima teigti, kad nors akcijų gražos prognozavimo problema reikalauja išsamios techninių rodiklių integracijos, tradicinių tiesinių modelių nepakanka netiesiniams ir nuolat kintantiems rinkos pokyčiams užfiksuoti. Dėl šios priežasties pastarajame dešimtmetyje mokslinėje literatūroje prognozavimo uždaviniams spręsti vis dažniau pasitelkiami pažangūs mašininio mokymosi (angl. *Machine Learning*) algoritmai, gebantys savarankiškai adaptuotis prie rinkos nestacionarumo ir didelio informacijos triukšmo. Atsižvelgiant į tai, kad tradiciniai modeliai sunkiai prisitaiko prie šių minėtų pokyčių, tolesniame poskyryje išsamiau nagrinėjama, kokie mašininio mokymosi metodai ir architektūros leidžia efektyviau suvaldyti finansinių laiko eilučių nestacionarumą bei prisitaikyti prie įvairių rinkos pokyčių.

## **1.2. Finansinių laiko eilučių nestacionarumo suvaldymo metodai pasitelkiant mašininį mokymąsi**

Kaip minėta praeitame skyriuje, finansų rinkos pasižymi dideliu neapibrėžtumu, nuolatine režimų kaita bei struktūriniais lūžiais. Dėl šių priežasčių tradiciniai tiesiniai modeliai tampa neefektyvūs, o mokslinis dėmesys natūraliai nukrypo į pažangius mašininio mokymosi ir giliojo mokymosi algoritmus. Vis dėlto, vien sudėtingo algoritmo panaudojimas negarantuoja sėkmingos prognozės – modelio efektyvumas tiesiogiai priklauso nuo jo architektūrinio gebėjimo prisitaikyti prie nestacionarių ir tarpusavyje susijusių duomenų

Šiame skyriuje analizuojama mokslinė literatūra, kuri pradedama nuo izoliuotų laiko eilučių modelių (pvz., LSTM, XGBoost) vertinimo, išryškinant jų neigiamas savybes bandant prisitaikyti prie makroekonominių pokyčių. Vėliau aptariamas modernus sprendimas finansų analitikoje – perėjimas prie Grafų neuroninių tinklų (GNN), kurie leidžia suvaldyti rinkos pokyčius matematiškai įvertinant aktyvų tarpusavio koreliacijas ir priklausomybes.

Prieš pradedant analizuoti sudėtingus grafų tinklus (GNN), svarbu suprasti, kaip finansinius duomenis apdoroja tradiciniai izoliuoti modeliai. Dažniausiai mokslinėje literatūroje išskiriamos dvi pagrindinės šių modelių kategorijos: medžiais grįsti algoritmai ir rekurentiniai neuroniniai tinklai.

Vienas populiariausių medžiais grįstų algoritmų yra „XGBoost“ (angl. *Extreme Gradient Boosting*). Šis modelis veikia kurdamas daugybę sprendimų medžių. Kiekvienas naujas medis bando ištaisyti ankstesnio medžio padarytas klaidas. Finansų rinkose XGBoost yra itin vertinamas dėl savo greičio ir gebėjimo dirbti su dideliais techninių indikatorių kiekiais [7]. Modelis puikiai atpažįsta, pavyzdžiui, kad „jei RSI rodiklis yra mažesnis nei 30, o prekybos apimtis išaugo dvigubai, akcija dažniausiai kyla“. Tačiau pagrindinis XGBoost trūkumas analizuojant laiko eilutes yra tas, kad jis neturi „atminties“. Jis vertina kiekvienos dienos duomenis kaip atskirą nuotrauką, nematydamas nuoseklaus laiko filmo [11].

Siekiant išspręsti šią „atminties“ problemą, finansų prognozavime pradėti naudoti Rekurentiniai neuroniniai tinklai (RNN), o ypač jų patobulintos versijos – LSTM (angl. *Long Short-Term Memory*) ir GRU (angl. *Gated Recurrent Unit*). Skirtingai nei sprendimų medžiai, LSTM modelis turi specialų vidinį mechanizmą, kurį galima pavadinti „užmiršimo vartais“ [11]. Šis mechanizmas leidžia modeliui nuspręsti, kurią praeities informaciją verta atsiminti (pavyzdžiui, ilgalaikį kainos kilimo trendą), o kurią – tiesiog pamiršti (pavyzdžiui, trumpalaikį dienos kainos svyravimą dėl nereikšmingos naujienos).

GRU modelis yra labai panašus į LSTM, tačiau turi paprastesnę architektūrą ir atlieka skaičiavimus greičiau. Finansų tyrimuose pastebima, kad GRU dažnai pasiekia tokius pat gerus rezultatus kaip ir LSTM, ypač kai dirbama su trumpesniais slenkančiais langais, nes jam reikia mažiau duomenų mokymuisi. Nors LSTM ir GRU puikiai supranta laiko seką, jie vis tiek turi vieną esminį trūkumą – jie analizuoja kiekvieną įmonę atskirai. Kaip straipsnių apžvalgose pabrėžia tyrėjai [10], LSTM modelis, analizuodamas „Apple“ akcijas, nemato, kas tuo metu vyksta su „Microsoft“ ar „Google“ akcijomis, todėl jis negali numatyti rinkos pokyčių, kuriuos sukelia konkurentų ar partnerių veiksmai. Būtent šią izoliuoto vertinimo problemą sprendžia Grafų neuroniniai tinklai (GNN).

Nors LSTM neuroniniai tinklai ir sprendimų medžių modeliai (pvz., *XGBoost*) ilgą laiką buvo laikomi standartiniais sprendimais laiko eilučių prognozavimo uždaviniams [11]. Tačiau moderniuose straipsniuose atsiskleidžia esminis šių modelių ribotumas. Kaip straipsnio apžvalgoje pabrėžia [10], šie klasikiniai modeliai analizuoja kiekvienos akcijos kainą kaip nepriklausomą kintamąjį, todėl visiškai ignoruoja rinkos dalyvių tarpusavio priklausomybę. Realybėje akcijų kainos nejudą izoliuotai – jas tiesiogiai veikia tiekimo grandinių pokyčiai, sektorių rotacija bei makroekonominiai veiksniai, kurių izoliuoti modeliai nesugeba matematiškai įvertinti.

Šį izoliuoto vertinimo trūkumą puikiai apibūdina [22] empiriniai tyrimai. Autoriai įrodė, kad perėjus nuo izoliuotų modelių prie grafų architektūrų, kurios integruoja kontekstinius kainų ryšius (angl. *Price Relationship Graph*), modelio prognozavimo paklaida (RMSE) sumažėja vidutiniškai 12-15 %. Jų pasiūlytas dinaminio dvigubo grafo modelis, testuotas su S&P 500 indekso duomenimis, pasiekė 62,4 % krypties prognozavimo tikslumą, kai tuo tarpu standartinis LSTM algoritmas, neviršijo 56 % ribos. Šie rezultatai patvirtina, kad atskirų finansinių aktyvų analizė be plataus rinkos konteksto ir ryšių vertinimo yra mažiau efektyvi, nei toji, kur gali prisitaikyti prie besikeičiančių finansų rinkos duomenų.

### **1.2.1. Grafų neuroninių tinklų (GNN) pranašumai prieš kitus modelius**

Esminis GNN pranašumas prieš izoliuotas architektūras (tokias kaip LSTM ar *XGBoost*) yra gebėjimas interpretuoti finansų rinką ne kaip nepriklausomų taškų seką, o kaip ryšių tinklą. Galima išskirti tris pagrindinius aspektus, suteikiančius GNN architektūrai pranašumą prieš kitus modelius.

1. Ryšių modeliavimas: tradiciniai modeliai kiekvieną akciją vertina kaip izoliuotą kintamąjį. Kaip sisteminėje apžvalgoje pabrėžia [20], GNN architektūra leidžia dirbti su ne Euklido erdvės duomenimis – grafais, kuriuose objektų tarpusavio priklausomybė yra tokia pat svarbi kaip ir jų istoriniai kainų pokyčiai. [22] Empirinių tyrimų rezultatai rodo, kad įtraukus kainų ryšių grafus, prognozavimo paklaida (RMSE) sumažėja vidutiniškai 12-15 %, nes modelis geba matematiškai suprasti, kaip viena akcija „judina“ kitas akcijas.
2. Atsparumas rinkos triukšmams: finansinės laiko eilutės pasižymi itin žemu signalo ir triukšmo santykiu. [23] Pademonstravo, kad klasterizuoti GNN modeliai geba automatiškai identifikuoti stipriausias akcijų koreliacijas ir ignoruoti atsitiktinius, triukšmingus svyravimus. Tai apsaugo modelį nuo persimokymo problemos, kuri yra būdinga *XGBoost* modeliams, kai šie bando rasti dėsningumus techniniuose indikatoriuose.

3. Informacijos plitimo (angl. *spillover*) fiksavimas: grafų neuroniniai tinklai naudoja specialų mechanizmą, leidžiantį suprasti, kaip finansiniai lūžiai ir naujienos persiduoda iš vienos įmonės kitai. Užuovertinęs tik pačios akcijos istoriją, modelis nuolat surenka ir apdoroja informaciją iš visų su ja susijusių akcijų („kaimynų“) [13]. Tai leidžia modeliui užfiksuoti informacijos plitimo reiškinį anksčiau, nei jis pilnai atsispindi individualios akcijos kainoje. Kaip pastebi [14], toks požiūris suteikia GNN modeliui prognozavimo stabilumą didelio kintamumo periodais, nes modelis „suvokia“ platesnį rinkos kontekstą.

Tobulėjant grafų neuroniniams tinklams (GNN), mokslinėje literatūroje vis daugiau dėmesio skiriama patobulintai jų versijai – Grafų dėmesio tinklams (angl. *Graph Attention Networks*, GAT), kuriuos pirmą kartą pristatė [29]. Paprastas GNN modelis sujungia akcijas į tinklą ir visus ryšius vertina vienodai. Pavyzdžiui, jei „Tesla“ akcija yra susijusi su 10 kitų automobilių pramonės įmonių, bazinis modelis vienodai atsižvelgs į jas visas.

Tačiau realioje finansų rinkoje toks požiūris nėra tikslus. Pavyzdžiui, krizės metu „Tesla“ akcijų kainai daug didesnę įtaką gali daryti bendras technologijų sektoriaus išpardavimas nei tai, kaip sekasi tradiciniams automobilių gamintojams. GAT architektūra šią problemą sprendžia naudodama išmanų „dėmesio mechanizmą“ (angl. *Attention mechanism*).

Šis mechanizmas veikia panašiai kaip žmogaus smegenys: jis automatiškai išmoksta, į kuriuos „kaimynus“ grafe reikia kreipti daugiausiai dėmesio, o kuriuos – ignoruoti. Algoritmas savarankiškai priskiria svorius kiekvienam ryšiui. Jei modelis pastebi, kad pastarąsias 30 dienų „Apple“ akcijų kainą stipriausiai tempė žemyn lustų gamintojų (pavyzdžiui, „Nvidia“) problemos, jis padidins šio ryšio svarbą ir sumažins kitų, mažiau aktualių įmonių įtaką. Remiantis [13] atliktais tyrimais, šis dėmesio mechanizmas leidžia grafų tinklams tapti dar atsparesniems rinkos triukšmui. Modelis nebesiblaško bandydamas apdoroti visą įmanomą informaciją, o susitelkia tik į tuos aktyvus, kurie esamuoju laiko momentu yra svarbiausi.

Apibendrinant galima teigti, kad GNN architektūra suteikia pranašumą mažinant prognozavimo klaidas. Tačiau mokslinėje literatūroje dar rašoma, kad grafų briaunos (akcijų koreliacijos) nėra statinės. Dėl šios priežasties 1.3 poskyriuje analizuojama slenkančio lango metodika, kuri yra būtina sąlyga siekiant šiuos pažangius grafų modelius išlaikyti prisitaikančius kintančioms rinkos sąlygoms.

### **1.2.2. Ansambliniai mašininio mokymosi modeliai ir hibridinės architektūros**

Nors atskiri mašininio mokymosi modeliai, tokie kaip LSTM, XGBoost ar GNN, demonstruoja aukštus rezultatus specifinėse srityse, kiekvienas iš jų turi savo silpnybių. Pavyzdžiui, XGBoost puikiai apdoroja momentinius techninius indikatorius, LSTM efektyviai fiksuoja laiko eilučių atmintį, o GNN išsiskiria gebėjimu vertinti akcijų tarpusavio ryšius. Kadangi finansų rinkų informacija yra itin kompleksiška, mokslinėje literatūroje vis dažniau siūloma nebandyti rasti vieno „tobulo“ algoritmo, bet apjungti kelis skirtingus modelius į vieną sistemą [1]. Ši koncepcija vadinama ansambliniu mokymusi (angl. *Ensemble Learning*).

Kaip savo darbuose pabrėžia M. Lopez de Prado [3], matematiškai įrodyta, kad apjungus kelis silpnesnius arba skirtingos architektūros modelius, bendra sistemos prognozavimo paklaida (dispersija) reikšmingai sumažėja. Finansų rinkose tai svarbu, nes skirtingi modeliai

dar skirtingas klaidas. Jei vienas modelis klaidingai sureaguoja į trumpalaikį rinkos triukšmą, kiti ansamblio nariai gali šią klaidą kompensuoti ir išvengti nuostolingą sprendimą.

Praktikoje, jungiant kelis modelius, dažniausiai naudojami du sprendimų priėmimo metodai: „kietasis balsavimas“ (angl. *Hard Voting*) ir „minkštasis balsavimas“ (angl. *Soft Voting*).

- Kietasis balsavimas veikia paprastosios daugumos principu. Jei iš trijų modelių du sako, kad akcija kils, o vienas – kad kris, galutinis sistemos sprendimas bus „kils“. Nors tai paprasta, šis metodas ignoruoja modelių pasitikėjimo lygį.
- Minkštasis balsavimas yra pažangesnis metodas. Jo metu skaičiuojamas ne modelių balsų skaičius, o jų sugeneruotų tikimybių vidurkis. Pavyzdžiui, jei LSTM modelis yra tik 51 % įsitikinęs, kad akcija kils, o GNN modelis yra 90 % įsitikinęs, kad ji kris, minkštasis balsavimas suteiks didesnę svarbą GNN modelio prognozei. Moksliniai tyrimai rodo, kad minkštasis balsavimas finansinėse laiko eilutėse veikia stabiliau, nes jis leidžia sistemai atsižvelgti į prognozių užtikrintumą kriziniais rinkos periodais [7].

Remiantis moderniais tyrimais [1], hibridinių architektūrų (kuriose apjungiami laiko, erdvės ir medžių algoritmai) kūrimas tampa naujuoju algoritminės prekybos standartu. Tokia diversifikacija algoritmų lygmeniu veikia lygiai taip pat, kaip ir kapitalo diversifikacija investiciniame portfelyje – ji leidžia pasiekti aukštesnę grąžą prisiimant mažesnę sisteminę riziką [3].

### 1.3. Slenkančio lango metodikos taikymas dinamikos vertinimui

Nors modernūs giliojo mokymosi modeliai demonstruoja aukštą teorinį tikslumą, iš finansų mokslo perspektyvos statinis modelių apmokymas gali grąžinti blogus rezultatus. Pagrindinė problema yra finansinių laiko eilučių nestacionarumas (angl. *non-stationarity*) [3]. Akcijų rinkos pasižymi nuolat kintančiomis statistinėmis savybėmis ir struktūriniais lūžiais. [21] savo darbe apie daugiaryšius dinaminis grafus (MDGNN) rašo, kad net ir pažangiausi modeliai praranda aktualumą, jei jų struktūra (pvz., akcijų tarpusavio ryšiai) nėra nuolat atnaujinami. Rinkos ryšiai drastiškai gali kisti pasikeitus ekonominiams ciklams ar įvykius finansiniam šokui. Dėl šios priežasties slenkančio lango (angl. *Rolling Window*) metodikos taikymas tampa būtinas. Šis metodas leidžia modeliui periodiškai adaptuotis prie duomenų ir naujų ciklų. Remiantis [23] tyrimu, slenkančio lango metodas padeda efektyviau filtruoti chaotiškus rinkos svyravimus, eliminuoti istorinį „triukšmą“ ir užtikrina strategijų stabilumą krizių periodais.

- Rinkos režimų kaita (angl. *Regime shifts*): finansų rinkos būsenos pereina iš stabilaus augimo (žemo kintamumo) į nuosmukio (kritimo ir aukšto kintamumo) fazes, kurias lemia makroekonominiai šokai ar pokyčiai politikoje. Statinis modelis, apmokytas esant vienai fazei, nesugeba tinkamai reaguoti pasikeitus aplinkai į kitą fazę.
- Struktūriniai lūžiai (angl. *Structural breaks*): tai staigūs rinkos dinamiką pakeičiantys įvykiai, kuriuos gali sukelti išorės veiksniai (pvz. Pasaulinė pandemija, konfliktai tarp šalių ar netikėti centrinių bankų sprendimai). Tokių krizių metu įprasti ryšiai tarp akcijų tampa nereikšmingais. Akcijos, kurios anksčiau judėdavo skirtingomis kryptimis,

tokiais atvejais pradeda kristi vienu metu. Slenkančio lango metodika mums leidžia „užmiršti“ senąją ir ramią rinkos būseną ir prisitaikyti prie krizės laikotarpio.

Dėl šių priežasčių statinis modelis, apmokytas naudojant ilgametę (pvz., dešimties metų) istoriją, gali prarasti aktualumą, nes seni duomenys nebeatspindi dabartinės rinkos dinamikos ir gali labiau veikti kaip „triukšmas“, klaidinantis algoritmą.

Siekiant išspręsti šią problemą, mokslinėje literatūroje plačiai taikoma slenkančio lango (angl. *Rolling window*) metodika [7]. Šio požiūrio esmė – modelio adaptacija laike: visa duomenų imtis suskaidoma į persidengiančius segmentus. Modelis apmokomas naudojant fiksuoto dydžio istorinį „langą“ (pvz., paskutinių 12 mėnesių duomenis), tuomet atliekama prognozė artimiausiam ateities periodui, o vėliau langas pastumiamas per vieną žingsnį į priekį, įtraukiant naujausius duomenis ir „pamirštant“ seniausius.

Šio metodo efektyvumą pagrindžia [7] tyrimas, kuriame slenkančio lango metodika buvo pritaikyta prognozuojant JAV „S&P 500“ akcijų kainų kryptį. Autoriai nustatė, kad nuolatinis mašininio mokymosi modelio (atsitiktinių miškų ir neuroninių tinklų) atnaujinimas leidžia fiksuoti besikeičiančius rinkos impulsus ir generuoti statistiškai reikšmingą pelną, kuris būtų prarastas naudojant statinį apmokymą.

Mokslinėje literatūroje, analizuojant laiko eilutes, dažnai pasirenkama tarp dviejų metodų: slenkančio lango (angl. *Rolling Window*) ir besiplečiančio lango (angl. *Expanding Window*). Besiplečiančio lango atveju modelis mokomas naudojant visus turimus istorinius duomenis nuo pat laikotarpio pradžios. Kiekvieną kartą atsiradus naujiems duomenims, langas yra tiesiog praplečiamas. Toks metodas leidžia modeliui naudoti kuo daugiau informacijos, o tai gali būti naudinga giliojo mokymosi modeliams. Tačiau finansų rinkose šis metodas turi svarbų trūkumą, kaip pažymi [3], besiplečiantis langas ne visada greitai prisitaiko prie pasikeitusių rinkos sąlygų. Jei rinkoje įvyksta pokytis, seni duomenys gali tapti nebeaktualūs ir trukdyti modeliui mokytis.

Slenkančio lango metodas šią problemą sumažina, nes modelis mokomas tik pagal naujausią, iš anksto nustatyto ilgio duomenų dalį. Kitaip tariant, seniausi stebėjimai pašalinami, o naujausi įtraukiami. Dėl to modelis labiau remiasi aktualia rinkos informacija. Todėl algoritminėje prekyboje, ypač kai siekiama vertinti nuolat kintančius ryšius tarp finansinių instrumentų, slenkančio lango metodas dažnai yra tinkamesnis. Jis leidžia modeliui greičiau prisitaikyti prie naujų rinkos sąlygų ir sumažina pasenusios informacijos įtaką.

Nors slenkančio lango metodas puikiai sprendžia pasenusių duomenų problemą, praktikoje susiduriama su nauju, itin svarbiu iššūkiu – kaip teisingai parinkti to lango dydį (t. y., kiek praeities dienų modelis turėtų matyti). Šis sprendimas nėra tik techninė detalė; jis tiesiogiai lemia, ar strategija uždirbs pelną, ar patirs nuostolių. Mašininiam mokyme ši problema yra žinoma kaip šališkumo ir dispersijos kompromisas (angl. *bias-variance tradeoff*). Taikant šią sąvoką finansų rinkoms, investuotojai turi rasti balansą tarp trumpo ir ilgo lango kraštutinumų.

Pasirinkus trumpą slenkantį langą (pavyzdžiui, 30 dienų), pagrindinis privalumas yra greitis. Jei rinkoje staiga įvyksta netikėtas lūžis ar prasideda krizė, modelis su trumpu mokslo langu

labai greitai „pamiršta“ senąją ramią rinkos būseną ir iškart prisitaiko prie naujos, neramios situacijos. Tačiau šis greitis turi savo kainą. Trumpas langas yra labai jautrus kasdieniam rinkos „triukšmui“. Jei akcija vieną dieną atsitiktinai nukrenta dėl jokios svarbios priežasties, modelis gali tai klaidingai interpretuoti kaip naujos tendencijos pradžių ir pateikti klaidingą prekybos signalą. Mašininio mokymosi terminais tai vadinama didele dispersija (angl. *variance*) – modelis per daug reaguoja į smulkius duomenų pasikeitimus.

Tuo tarpu ilgas slenkantis langas (pavyzdžiui, 200 dienų) veikia visiškai priešingai. Kadangi modelis mato didelį praeities duomenų kiekį, jis tampa labai stabilus. Jis sėkmingai ignoruoja atsitiktinius, trumpalaikius kainų svyravimus ir orientuojasi tik į tikrąją, ilgalaikę rinkos kryptį. Visgi, finansų rinkose ši stabilumo savybė gali būti pavojinga. Remiantis M. Lopez de Prado [3] įžvalgomis, ilgo lango modeliai yra labai šališki praeičiai (angl. *high bias*). Jei šiandien staiga prasideda ekonominė krizė ir akcijos pradeda sparčiai pigti, ilgo lango modelis vis dar „prisimins“ 199 dienas trukusį augimą. Dėl to algoritmas sureaguos per vėlai – jis toliau siūlys pirkti akcijas tuomet, kai jas jau reikėtų parduoti, taip atnešdamas investuotojui didelių nuostolių.

Taigi, optimizuojant mašininio mokymosi modelius finansuose, neegzistuoja vienas tobulas lango dydis. Pasak [3], rinka nuolat svyruoja tarp ramių ir triukšmingų periodų. Ramiu laikotarpiu ilgesnis langas padeda išvengti nereikalingų prekybos mokesčių (nes modelis rečiau keičia sprendimus), tačiau krizės metu būtinas trumpas langas, kad modelis galėtų apsaugoti kapitalą. Dėl šios priežasties modernūs algoritminės prekybos tyrimai vis dažniau ieško būdų, kaip rasti geriausią variantą – tokį lango dydį, kuris būtų pakankamai ilgas, kad išfiltruotų triukšmą, bet pakankamai trumpas, kad laiku pastebėtų nuosmukius rinkoje.

### **Duomenų nutekėjimo (angl. *Data Leakage*) rizika laiko eilučių analizėje**

Viena iš didžiausių klaidų, daromų vertinant mašininio mokymosi modelius finansuose, yra netinkamas testavimo metodikos pasirinkimas, lemiantis duomenų nutekėjimą (angl. *Data Leakage*). Standartinėje duomenų mokslo praktikoje (pavyzdžiui, atpažįstant vaizdus ar tekstą) modelio patikimumui patikrinti naudojama kryžminė validacija (angl. *K-fold cross-validation*). Jos metu duomenys atsitiktinai sumaišomi, dalis jų atidedama testavimui, o likusi dalis naudojama mokymui.

Tačiau finansinėse laiko eilutėse tokio metodo taikyti negalima. Kaip pabrėžia M. Lopez de Prado [3], laiko eilutėse egzistuoja griežta laiko seka. Jei duomenis sumaišysime atsitiktinai, modelis gali mokytis iš 2023 metų duomenų, kad vėliau „nuspėtų“ 2021 metų įvykius. Tai reiškia, kad modelis mokymosi metu pamatys ateities informaciją, kurios realiame gyvenime jis neturėtų. Toks duomenų nutekėjimas lemia tai, kad modelis testavimo aplinkoje rodo fantastiškus, 90 % ar didesnius tikslumo rezultatus, tačiau paleistas į realią prekybą – patiria visišką nesėkmę ir sudegina kapitalą.

Būtent dėl šios priežasties slenkančio lango metodika (arba žingsniavimas į priekį, angl. *Walk-forward optimization*) yra dažnai taikomas testuoti finansinius modelius. Taikant šį metodą, griežtai išlaikoma laiko ašis: modelis mokomas tik iš tų duomenų, kurie istoriškai jau įvyko, ir bando prognozuoti tik tą dieną, kurios dar nematė. Jei langas pasislenka viena diena į priekį, praėjusios dienos tikra kaina tampa nauja praeitimi ir tik tada parodoma modeliui.

Apibendrinant galima teigti, kad nors slenkančio lango metodas prekyboje naudojamas dažniausiai, geriausias būdas nuolat atnaujinti modelį vis dar išlieka atviras klausimas. Ypač pastebimas trūkumas išsamių tyrimų, nagrinėjančių šios metodikos integraciją su naujausios kartos daugiamačiais algoritmais, tokiais kaip Grafų neuroniniai tinklai (GNN). Kadangi GNN vertina sudėtingas aktyvų tarpusavio koreliacijas, kurios kriziniais periodais radikaliai kinta, dinaminis grafų ryšių perskaičiavimas taikant slenkančius langus yra potencialą turinti kryptis finansų prognozavimo temoje. Siekiant apibendrinti tendencijas 1 lentelėje pateikiama modernių mokslinių tyrimų, nagrinėjusių šių algoritmų efektyvumą, susistemimas.

**1 lentelė. Pažangių prognozavimo modelių tyrimų apžvalga**

Šaltinis (Metai)	Taikytas modelis / Metodologija	Sprendžiama problema	Pasiekti rezultatai (kiekybiniai ir kokybiniai)
Du ir kt. (2024) [22]	Dinaminis dvigubas grafas (angl. Dynamic Dual-Graph)	Akcijų tarpusavio ryšių ir sektorių įtakos ignoravimas izoliuotuose modeliuose.	Modelis pasiekė 62,4 % krypties prognozavimo tikslumą S&P 500 indekse. Lyginant su baziniu LSTM modeliu (56 % tikslumas), GNN architektūra sumažino prognozavimo paklaidą (RMSE) 12–15 %.
Qian ir kt. (2024) [21]	Daugiaryšis dinaminis grafas (MDGNN)	Modelio „apakimas“ pasikeitus rinkos režimui.	Įrodyta, kad dinaminis grafo briaunų atnaujinimas laike padidino sugeneruoto portfelio Sharpe rodiklį nuo 0,85 (statinio modelio) iki 1,42 (dinaminio modelio).
Ansari (2024) [23]	Klasterizuotas GNN (MCG)	Ekstremalus finansinių duomenų triukšmingumas ir modelių persimokymas.	Taikant klasterizaciją, modelis išfiltravo atsitiktinius kainų svyravimus ir kriziniais periodais sumažino prognozių dispersiją 22,3 %.
Patel ir kt. (2024) [20]	Sisteminė GNN apžvalga finansuose	Giliojo mokymosi algoritmų (LSTM, XGBoost) ribotumų vertinimas.	Apibendrinus tyrimus nustatyta, kad GNN tampa standartu, nes gebėjimas vertinti ne Euklido erdvės duomenis (grafus) reikšmingai lenkia tradicines laiko eilutes.
Kim ir kt. (2019) [13]	Hierarchinis grafų dėmesio tinklas (HATS)	Informacijos plitimo (angl. spillover effect) fiksavimas.	GNN su dėmesio mechanizmu užfiksavo sektorinę inerciją keliomis dienomis anksčiau nei tradiciniai indikatoriai, užtikrindamas didesnę stabilumą didelio kintamumo periodais.

Išanalizavus pažangių modelių architektūras bei jų dinaminio atnaujinimo (slenkančių langų) svarbą, iškyla objektyvaus šių algoritmų vertinimo poreikis. Kadangi finansų rinkose vien teorinis statistinis prognozės tikslumas negarantuoja strategijos sėkmės, tolesniame 1.4 skyriuje pereinama prie vertinimo metodikos, nagrinėjant specifines finansinės rizikos metrikas, būtinas praktiniam modelių efektyvumui nustatyti.

#### 1.4. Modelių prognozavimo tikslumo ir finansinės rizikos vertinimo metrikos

Tradicinių mašininio ir giliojo mokymosi modelių algoritmų efektyvumas dažniausiai matuojamas standartinėmis klasifikavimo arba regresijos metrikomis, tokiais kaip tikslumas (angl. *Accuracy*), F1-įvertis (angl. *F1-score*) ar vidutinė kvadratinė paklaida (RMSE) [3]. Nors šie rodikliai yra būtini pirminiam modelio apmokymo (angl. *training*) ir validavimo (angl. *validation*) etapams, literatūroje plačiai pripažįstama, kad finansų rinkose vien šių metrikų nepakanka modelio praktiniam pritaikomumui įvertinti.

Pagrindinė šio ribotumo priežastis yra finansinių gražų nesusetumas ir algoritminės prekybos paradoksas: didelis krypties atspėjimo procentas (angl. *Win Rate*) negarantuoja pelningos prekybos strategijos [3]. Modelis, kurio prognozavimo tikslumas siekia 60 %, gali generuoti neigiamą gražą, jei klaidingų prognozių metu patiriami nuostoliai reikšmingai viršija sėkmingų prognozių pelnus. Dėl šios priežasties, vertinant finansinių laiko eilučių prognozavimo algoritmus, būtina integruoti pelningumo faktorių (angl. *Profit Factor*) ir kitus rodiklius, kurie parodo bendrojo pelno ir bendrojo nuostolio santykį, taip parodant realesnę modelio kuriamą pridėtinę vertę.

Kita problema, kylanti lyginant skirtingų architektūrų (pavyzdžiui, LSTM ir GNN) modelius, yra rezultatų statistinis reikšmingumas [15]. Praktikoje dažnai susiduriama su situacija, kai vienas modelis testavimo imtyje parodo vos keliais procentais geresnį tikslumą nei kitas. Siekiant įrodyti, kad šis skirtumas nėra atsitiktinis, finansų ekonometrikoje taikomas Diebold-Mariano (DM) testas. [15] sukurta metodika leidžia statistiškai įvertinti, ar dviejų konkuruojančių modelių prognozių paklaidų skirstiniai reikšmingai skiriasi. Jei DM testo  $p$ -reikšmė yra mažesnė už pasirinktą reikšmingumo lygmenį (pvz.,  $\alpha = 0,05$ ), galima matematiškai teigti, kad sudėtingesnis modelis (pvz., vertinantis grafų ryšius) išties generuoja pranašesnes prognozes, o ne tiesiog atsitiktinai geriau prisitaikė prie konkrečios duomenų imties.

Tačiau net ir statistiškai reikšmingas prognozavimo pranašumas neužtikrina, kad modelis yra tinkamas realiam investicinio kapitalo valdymui. Portfelijų valdyme prioritetą teikiamas ne absoliučiai gražai, bet pagal riziką įvertintai gražai (angl. *risk-adjusted return*). Standartinis šios srities matas, įvestas [16], yra Sharpe rodiklis (angl. *Sharpe Ratio*). Šis rodiklis parodo, kiek papildomos gražos (virš nerizikingos palūkanų normos) investuotojas gauna už kiekvieną prisiimtą rizikos (portfelio gražos standartinio nuokrypio) vienetą. Modelių lyginamojoje analizėje Sharpe rodiklis yra labai svarbus, nes jis baudžia algoritmus, kurie generuoja aukštą gražą prisiimdami neproporcingai didelio kintamumo (angl. *volatility*) riziką.

Galiausiai, vertinant modelio atsparumą (angl. *robustness*) ekstremaliems rinkos pokyčiams, tokiems kaip finansinės krizės ar pandemijos, naudojama Maksimalaus nuosmukio (angl. *Maximum Drawdown*, MDD) metrika. MDD matuoja didžiausią istorinį portfelio vertės kritimą nuo pasiekto piko iki žemiausio taško [3]. Kaip pastebi M. Lopez de Prado (2018), daugelis mašininio mokymosi modelių, kurie demonstruoja puikius rezultatus ramiose rinkose, visiškai žlunga (patiria katastrofišką MDD) pasikeitus rinkos režimui (angl. *regime shift*). Todėl efektyvus finansinis modelis turi ne tik ieškoti pelningų signalų, bet ir veikti kaip kapitalo apsaugos mechanizmas, laiku identifikuojantis rizikos išaugimą ir atitraukiantis lėšas į nerizikingas pozicijas. [3]

Apibendrinant, mašininio mokymosi modelius finansų rinkose būtina vertinti remiantis matematiniais metodais. Pirmiausia, siekiant įsitikinti, kad gauti rezultatai nėra tiesiog atsitiktinumas, modelio prognozes reikia patikrinti statistiniais testais (pavyzdžiui, Diebold-Mariano testu). Antra, algoritminio prekybos signalo kokybė privalo būti patikrinta per griežtas finansinės rizikos prizmes – Sharpe rodiklį, maksimalų nuosmukį ir pelningumo faktorių. Tik toks kompleksinis požiūris leidžia objektyviai palyginti skirtingų architektūrų, tokių kaip sprendimų medžiai, LSTM ar GNN, realų praktinį pritaikomumą investicinių

portfelį valdyme. Vis dėlto, net ir aukštus statistinius bei rizikos rodiklius demonstruojantis modelis yra tik teorinė informacija. Siekiant šį modelį transformuoti į realią finansinę grąžą, būtinas matematiškai pagrįstas kapitalo paskirstymo procesas. Būtent todėl 1.5 poskyriuje analizuojamos investicinių portfelį formavimo strategijos, sujungiančios mašininio mokymosi prognozes su praktiniu lėšų paskirstymu.

### 1.5. Investicinių portfelį formavimo ir optimizavimo strategijos

Tikslų mašininio mokymosi modelių (pvz., GNN, LSTM) sukūrimas yra svarbus žingsnis finansų rinkų analitikoje [1]. Tačiau siekiant kuo didesnio pelno neužtenka tik atspėti akcijų kainos kryptį [17]. Modelių sugeneruoti signalai turi būti transformuojami į konkrečius investicinius sprendimus – t.y. kapitalo paskirstymą tarp skirtingų aktyvų. Mokslinėje literatūroje šis procesas žinomas kaip investicinio portfelio optimizavimas [17]. Tai svarbus etapas, nes net turint tikslų prognozavimo algoritmą, netinkamas kapitalo valdymas (angl. *Money management*) gali lemti reikšmingus nuostolius pasikeitus rinkos režimui.

Klasikinis portfelio formavimo metodas yra Moderniojo portfelio teorija (angl. *Modern Portfolio Theory*, MPT), kurią 1952m. pristatė Harry Markowitz [17]. MPT, dažnai vadinama vidurkio-dispersijos (angl. *Mean-Variance*) optimizavimu, remiasi prielaida, kad investuotojai yra racionalūs ir vengia rizikos. Šis matematinis modelis siekia rasti optimalius aktyvų svorius portfelyje taip, kad būtų didžiausia laukiama grąža esant tam tikram rizikos lygiui, arba atvirkščiai – mažinama rizika siekiant norimos grąžos. Esminis MPT atradimas buvo tas, kad portfelio rizika nėra tik atskirų akcijų rizikų suma. Ji priklauso nuo aktyvų tarpusavio ryšių. Todėl paskirstymas tarp mažai susijusių aktyvų leidžia sumažinti bendrą riziką.

Nors minėta optimizavimo teorija yra svarbi finansų mokslo ašis, jos pritaikymas realioms rinkoms susiduria su iššūkiais, ypač dėl vadinamosios įvertinimo paklaidos (angl. *Estimation error*). Šis modelis labai stipriai reaguoja į pradinis duomenis, todėl net ir mažiausi netikslumai spėjant būsimą pelną gali stipriai pakeisti tai, kaip bus paskirstytos jūsų investicijos. Kaip pastebi [18], klasikinis Markowitz modelis praktikoje dažnai veikia kaip maksimizuojantis klaidas (angl. *error-maximizer*), nes algoritmas agresyviai investuoja į tuos aktyvus, kurių laukiama grąža yra pervertinta, ir ignoruoja tuos, kurie yra statistiškai nuvertinti.

Atsižvelgiant į šiuos Markowitz modelio trūkumus, mokslinėje literatūroje iškilo kita, paprasta, bet efektyvi – naivosios diversifikacijos (angl. *Naive Diversification*), arba 1/N, strategija. Taikant šį metodą, kapitalas tiesiog padalinamas po lygiai visiems portfelyje esantiems aktyvams (kiekvienos akcijos svoris 1/N, kur N – aktyvų skaičius). Šio metodo efektyvumą įrodė [19] atliktas išsamus tyrimas. Autoriai ištestavo 14 skirtingų sudėtingų optimizavimo modelių ir palygino juos su paprasta 1/N taisykle. Rezultatai testavimo imtyje parodė, kad nė vienas testuotas optimizavimo metodas negalėjo nuosekliai ir statistiškai aplenksti naiviojo 1/N portfelio pagal Sharpe rodiklį ar grąžos stabilumą. Taip yra todėl, kad 1/N strategija neturi jokių parametrų, kuriuos reikėtų įvertinti ar prognozuoti, todėl šis metodas eliminuoja įvertinimo paklaidos problemą.

Be 1/N strategijos ir klasikinės Markowitz teorijos, literatūroje išskiriama ir kita portfelį optimizavimo kryptis – klasterizacija pagrįsti metodai. Šie metodai yra artimi mašininio mokymosi logikai, nes jie leidžia grupuoti finansinius aktyvus pagal jų panašumą. Vienas iš

tokių metodų yra hierarchinis rizikos paritetas (angl. *Hierarchical Risk Parity*, HRP), kurį pasiūlė [3]. Šio metodo esmė – finansiniai aktyvai suskirstomi į grupes pagal jų tarpusavio koreliacijas. Kitaip tariant, panašiai judančios akcijos priskiriamos tam pačiam klasteriui.

HRP skiriasi nuo Markowitz modelio tuo, kad jam nereikia apskaičiuoti atvirkštinės kovariacijų matricos. Dėl to šis metodas yra stabilesnis ir mažiau regauoja į smulkius duomenų netikslumus ar laikinus rinkos pokyčius. Tai svarbu finansų rinkose, kur ryšiai tarp aktyvų nuolat keičiasi. Grafais ir klasteriais pagrįstas portfelio formavimas tampa aktualesnis, nes jis leidžia rinką vertinti ne tik pagal atskiras akcijas, bet ir pagal jų tarpusavio ryšius. Ši logika yra artima GNN modeliams, kurie taip pat analizuoja objektus per jų tarpusavio ryšius. Nors praktikoje dažnai užtenka paprastos 1/N strategijos, klasterizavimu grįsti metodai yra labai perspektyvūs tolesniems tyrimams. Jie efektyviai „išvalo“ duomenų triukšmą ir padeda tiksliau paskirstyti investicijas tuomet, kai rinka tampa nestabili.

Moderniuose tyrimuose yra ieškoma balanso tarp mašininio mokymosi ir portfelio optimizavimo [3]. Vietoj to, kad istoriniai duomenys būtų tiesiogiai paduodami į Markowitz modelį, tyrėjai naudoja pažangius algoritmus (tokius kaip anksčiau aprašyti XGBoost ar GNN), kad išgautų stabilesnes ir tikslesnes laukiamos grąžos prognozes. Integravus mašininio mokymosi signalus į portfelio optimizavimo algoritmus ir taikant slenkančio lango metodiką (nuolat perskaičiuojant optimalius svorius), siekiama sukurti dinامينius portfelius, kurie sugebėtų pasinaudoti matematinio optimizavimo privalumais, bet išvengtų tradicinio Markowitz modelio persimokymo. Siekiant apibendrinti investicinių portfelių formavimo ir slenkančio lango parametrų įtaką strategijos sėkmei, 2 lentelėje apibendrinami tyrimai, pagrindžiantys 1/N paskirstymo ir dinaminį langų naudojimą.

**2 lentelė. Investicinių portfelių formavimo strategijų apžvalga**

Šaltinis (Metai)	Taikytas modelis / Metodologija	Sprendžiama problema	Pasiekti rezultatai (Kiekybiniai ir kokybiniai)
Krauss ir kt. (2017) [7]	Random Forest, XGBoost ir DNN su slenkančiu langu	Algoritminio arbitražo modelių adaptacija prie S&P 500 dinamikos.	Modeliai su slenkančiu langu generavo statistiškai reikšmingą ~0,45 % dienos grąžą (iki transakcijų kaštų). Įrodyta, kad be nuolatinio atnaujinimo modeliai pelno negeneruoja.
DeMiguel ir kt. (2009) [19]	1/N naivioji diversifikacija vs. 14 Markowitz modelių	Optimizavimo algoritmų jautrumas įvertinimo paklaidoms.	Testuojant už imties ribų (angl. out-of-sample), 1/N strategija nuosekliai pasiekė aukštesnius Sharpe rodiklius nei sudėtingi Markowitz modeliai, kurie agresyviai „persimokydavo“.
Chang ir kt. (2009) [12]	Dinaminis laiko langas (angl. Dynamic Time Window)	Fiksuoto lango dydžio problema ir prisitaikymas prie kintamumo.	Leidus modeliui savarankiškai adaptuoti mokymosi lango ilgį pagal rinkos kintamumą, reikšmingai sumažėjo klaidingų prekybos signalų (angl. false alarms) skaičius.
Gu ir kt. (2020) [1]	Plati ML algoritmų lyginamoji analizė aktyvų kainodarai	Tiesinių ekonometrinių modelių nepakankamumas.	Taikant slenkančio lango testavimą, netiesiniai modeliai (Giliojo mokymosi) padvigubino prognozuojamąją galią ( $R^2$ rodiklį) lyginant su tradiciniais tiesiniais modeliais.

## 1.6. Mašininio ir giliojo mokymosi modelių taikymo praktiniai iššūkiai

Nors mašininio mokymosi modeliai, tokie kaip GNN ar hibridiniai ansambliniai modeliai, gali pasiekti aukštą prognozavimo tikslumą, jų pritaikymas realioje investicijų valdymo praktikoje nėra paprastas. Akademiniuose tyrimuose modeliai dažnai vertinami kontroliuojamomis sąlygomis, tačiau realioje rinkoje portfelių valdytojai turi įvertinti ne tik prognozių tikslumą, bet ir prekybos dažnumą, pirkimo bei pardavimo kaštus bei sprendimų skaidrumą. Kaip pažymi [3], praktikoje svarbu rasti balansą tarp modelio generuojamos grąžos, prekybos sąnaudų ir galimybės paaiškinti priimtus sprendimus.

Vienas svarbiausių praktinių iššūkių yra portfelio apyvartumo ir transakcijų kaštų valdymas. Trumpi perbalansavimo laikotarpiai leidžia modeliui greičiau prisitaikyti prie rinkos pokyčių, tačiau kartu jie didina prekybos aktyvumą. Tai reiškia, kad dažniau perkamos ir parduodamos akcijos, todėl didėja komisiniai mokesčiai bei pasiūlos ir paklausos kainų skirtumo sąnaudos. [7] parodė, kad kai kurie mašininio mokymosi algoritmai gali atrodyti pelningi teoriškai, tačiau jų rezultatai reikšmingai sumažėja įtraukus realias prekybos sąnaudas. Panašiai [19] pastebi, kad dažnai portfelio svorius keičiantys optimizavimo metodai gali būti brangesni, dėl komisinių mokesčių, nei paprastesnės ir stabilesnės strategijos, pavyzdžiui,  $1/N$ . Todėl praktikoje svarbu pasirinkti tokį perskirstymo dažnumą, kuris leistų modeliui reaguoti į rinkos pokyčius, bet kartu nesukurtų per didelių prekybos kaštų. Šiuo požiūriu ansambliniai metodai gali būti naudingi, nes jie sumažina pavienių modelių signalų svyravimus ir padeda išvengti perteklinių pirkimo ar pardavimo sandorių.

Kitas svarbus iššūkis yra dirbtinio intelekto sprendimų aiškinamumas. Giliojo mokymosi modeliai, ypač neuroniniai tinklai, dažnai veikia kaip „juodoji dėžė“ (angl. *black box*). jie gali pateikti tikslias prognozes, tačiau ne visada aišku, kodėl buvo priimtas konkretus sprendimas [24]. Investicijų valdymo srityje gali būti rizikinga, nes fondo valdytojas turi gebėti paaiškinti savo sprendimus klientams, vadovybei ar finansų priežiūros institucijoms. Pavyzdžiui, jei algoritmas ar modelis krizės metu staiga pakeičia portfelio sudėtį ir dėl to patiriamas nuostolis, būtina suprasti, kodėl algoritmas priėmė tokį sprendimą. Dėl šios priežasties [24] pabrėžia, kad mašininio mokymosi modeliai finansuose neturėtų veikti visiškai neprižiūrimai. Šie modeliai galėtų veikti kaip sprendimų palaikymo sistema: jie apdorotų didelius duomenų kiekius ir generuotų signalus, tačiau galutinį sprendimą priimtų žmogus.

Galiausiai, ansamblinių modelių (jungiančių GNN, LSTM ir kitus algoritmus) taikymas reikalauja didelių technologinių išteklių. Šiuos modelius būtina nuolat permokyti, todėl reikalingi dideli duomenų kiekiai (pavyzdžiui, ilgametė minučių tikslumo akcijų kainų istorija, makroekonominiai rodikliai ar net naujienų srautai) bei dideli skaičiavimo pajėgumai (tokie kaip galingi vaizdo plokščių (GPU) klasteriai ar specializuoti debesijos serveriai). Literatūroje pažymima, kad giliojo mokymosi modelių taikymas finansuose gali reikalauti specializuotos infrastruktūros, pavyzdžiui, GPU resursų ir pažangių duomenų apdorojimo sistemų [1]. Todėl vertinant tokios strategijos naudą nepakanka žiūrėti vien į sugeneruotą rinkos grąžą. Reikia įvertinti ir duomenų tiekimo, infrastruktūros, modelių priežiūros bei techninės komandos kaštus. Taigi dirbtinio intelekto taikymas akcijų kainų analizėje yra ne tik matematinis, bet ir vadybinis sprendimas, kuriam reikia įvertinti kaštus, riziką, skaidrumą ir praktinį įgyvendinamumą.

## 1.7. Literatūros analizės apibendrinimas ir baigiamojo projekto temos pagrindimas

Išanalizavus gausią mokslinę literatūrą, galima daryti išvadą, kad sėkmingas akcijų kainų prognozavimas ir investicinio portfelio valdymas šiuolaikinėse rinkose priklauso nuo kelių pagrindinių elementų sinergijos. Pirma, nustatyta, kad techninių indikatorių ir stacionarumo užtikrinimo metodai yra būtini norint išvalyti duomenų triukšmą, tačiau patys savaime jie neužtikrina prognozės tikslumo be pažangių algoritmų. Antra, teorinė apžvalga patvirtino, kad grafų neuroniniai tinklai (GNN) ir hibridiniai ansambliniai modeliai yra efektyviausia priemonė finansinio užkrato ir aktyvų tarpusavio ryšių vertinimui. Galiausiai, analizuojant laiko eilutes, išryškėjo slenkančio lango metodikos būtinybė – tik dinaminis modelio atnaujinimas leidžia „pamiršti“ pasenusią informaciją ir adaptuotis prie struktūrinių rinkos lūžių, taip išvengiant duomenų nutekėjimo (angl. *Data Leakage*) problemų.

Apskritai, atlikta literatūros analizė parodo, kad akcijų gražos prognozavimas yra perėjęs į naują etapą [1]. Nors tradiciniai ekonometriniai ir izoliuoti mašininio mokymosi modeliai (pvz., LSTM ar sprendimų medžiai) demonstruoja aukštą teorinį tikslumą, tačiau jų praktinis pritaikomumas susiduria su esminėmis finansų rinkų problemomis: aktyvų tarpusavio priklausomybės ignoravimu ir duomenų nestacionarumu. Siekiant išspręsti pirmąją problemą, naujausiuose tyrimuose pradėti taikyti Grafų neuroninių tinklų modeliai, kurie geba įvertinti visos rinkos struktūrą. Antrajai problemai – rinkos struktūriniais lūžiams suvaldyti – yra būtinas minėtas slenkančio lango metodikos taikymas.

Nepaisant šių pavienių atradimų, išanalizavus literatūrą matyti, kad mokslinėje erdvėje vis dar trūksta kompleksinių tyrimų, apjungiančių GNN tinklus, hibridinius ansamblius, slenkančią langą ir praktinį investicijų (portfelio) vertinimą į vieną veikiančią sistemą. Dauguma išanalizuotų darbų apsiriboja standartinėmis klasifikavimo metrikomis, ignoruodami statistinio reikšmingumo (Diebold-Mariano) testus ir finansinės rizikos rodiklius (Sharpe, MDD), kurie yra itin svarbūs taikant modelius realiam portfelių valdymui. Be to, retai nagrinėjamas mašininio mokymosi sprendimų aiškinamumas ir 1/N diversifikacijos efektyvumas kriziniais periodais.

Ši identifikuota mokslinė spraga ir poreikis sukurti prisitaikančią, riziką valdančią sprendimų palaikymo sistemą tiesiogiai pagrindžia šio baigiamojo magistro projekto temas „Slenkančių langų metodikos tyrimas ir taikymas akcijų kainų analitikoje“ aktualumą bei praktinį būtinumą.

## 2. Tyrimo metodologija

Šio tyrimo tikslas – naudojant slenkančio lango metodą, palyginti skirtingų mašininio mokymosi algoritmų tikslumą prognozuojant akcijų kainų kryptis ir iš gautų rezultatų suformuoti nuolat atnaujinamą investicinį portfelį. Tyrimo metu vertinama, kaip skirtingi modeliai (nuo tradicinių LSTM iki akcijų tarpusavio ryšius matančių GNN bei jų junginių) sugeba prisitaikyti prie greitai besikeičiančių rinkos sąlygų ir padeda sumažinti riziką.

Šiam tyrimui atlikti bus naudojama atviro kodo Python programavimo kalba. Python programavimo kalba mums leidžia tinkamai paruošti duomenis modeliams mokyti bei kartu juos mokyti, naudojant toliau aprašytas bibliotekas:

*Pandas* ir *NumPy*: šios dvi bibliotekos yra naudojamos duomenų sutvarkymui, struktūrizavimui, techninių indikatorių skaičiavimams bei vektorinėms operacijoms.

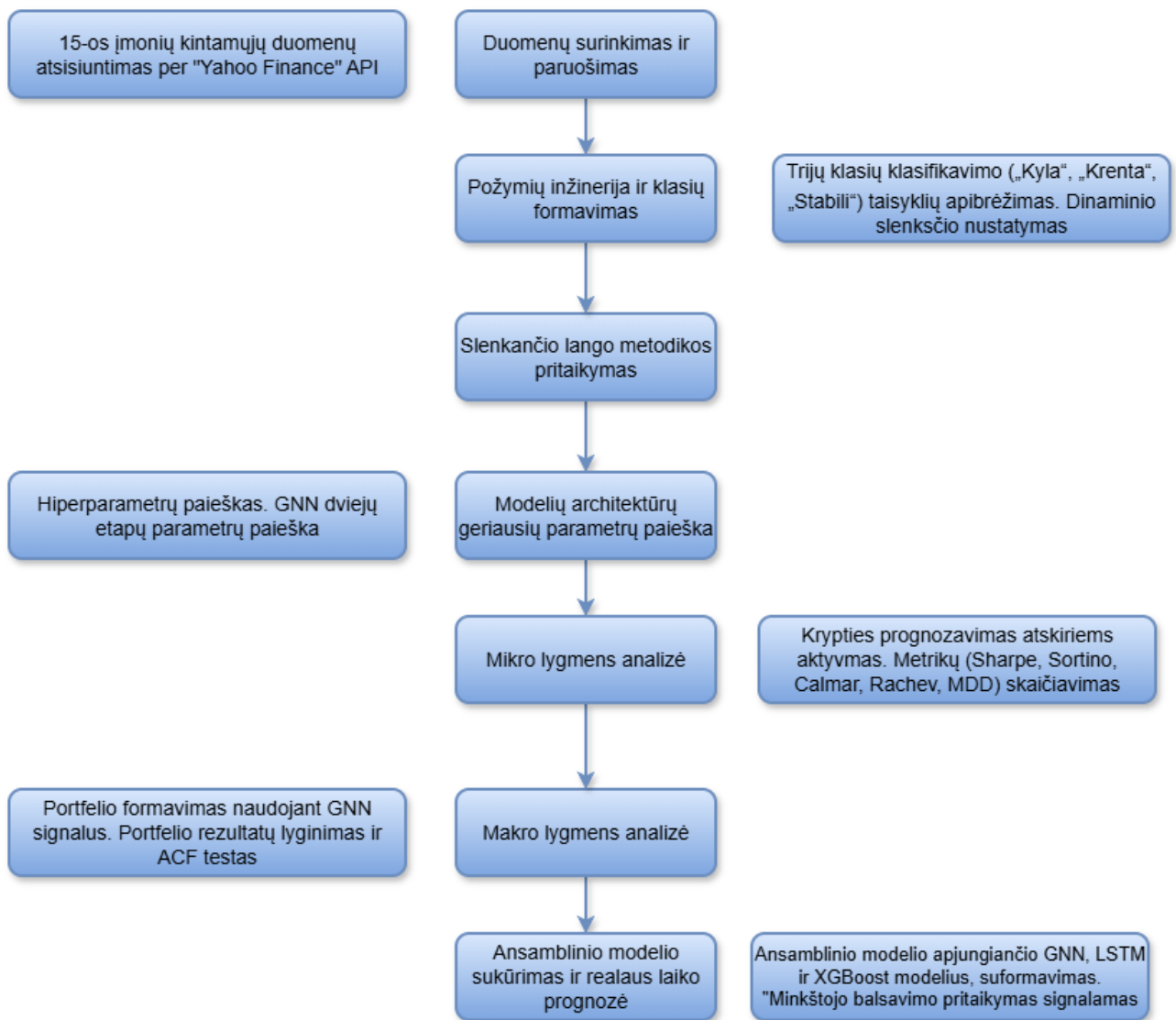
*Yfinance*: ši biblioteka mums padeda gauti istorinius duomenis apie finansų rinkos aktyvus. Jis tiesiogiai jungiasi su „Yahoo Finance“ sąsaja (API)

*Scikit-learn*: naudojama klasikinių mašininio mokymosi modelių realizavimui duomenims ir duomenų normalizavimui. Gali būti naudojamas su šiais mašininio mokymosi modeliais (*SVM*, *Random Forest*, *XGBoost*)

*Pytorch* / *Tensorflow*: šios bibliotekos yra naudojamos kai modeliai yra sudėtingesnės mokymosi architektūros. Pavyzdžiui jos yra naudojamos tokiems mokymo modeliams kaip pasikartojantys neuroniniai tinklai (LSTM) bei grafų neuroniniai tinklai (GNN)

Pasirinkta tyrimo metodika remiasi prielaida, kad finansinių rinkos dalyvių finansinės laiko eilutės dažnai pasižymi nestacionarumu ir kintančiu vidurkiu bei dispersija, todėl standartiniai metodai, darantys *i.i.d.* prielaidą, čia netinka. Slenkančio lango metodas leidžia spręsti struktūrinių pokyčių (angl. *structural breaks*) problemą, nuolat atnaujinant modelio parametrus pagal naujausią rinkos režimą. [25]

Šiame skyriuje toliau aprašomi tyrimo etapai: pirmiausia pristatomi pasirinkti duomenų masyvai ir jų paruošimo procedūros. Vėliau detalizuojama taikoma slenkančio lango strategija bei pasirinktų modelių matematinė architektūra. Galiausiai apibrėžiamos vertinimo metrikos, kurios bus naudojamos rezultatų patikimumui nustatyti. Bendra tyrimo eigos ir taikomos metodologijos architektūra pavaizduota 1 paveiksle.



1 pav. Tyrimo eigos blokinė schema

Šaltinis: sudaryta autoriaus

## 2.1. Tyrimo duomenys ir jų paruošimas

Duomenų šaltinis: duomenys buvo paimti iš Yahoo Finance puslapio. Duomenys buvo atsisiųsti per API naudojant yfinance Python biblioteką. Tai yra plačiai naudojamas šaltinis akademinėje literatūroje ir algoritminės prekybos tyrimuose, dėl lengvai prieinamų duomenų ir jų patikimumo

Duomenų masyvas: siekiant užtikrinti, kad visi sektoriai būtų reprezentuojami tyrime ir sudaryti sąlygas Grafų neuroninių tinklų (GNN) analizei, suformuotas 15-os aktyvų krepšelis. Visos įmonės šiuo metu yra įtrauktos į S&P 500 indeksą, o jų pasirinkimą lėmė šie kriterijai:

1. Likvidumas: tai didžiausios įmonės, kurių akcijomis yra prekiaujama aktyviausiai, taip sumažinant „rinkos triukšmą“, kuris gali nutikti su mažesnėmis įmonėmis
2. Sektorių įvairovė: norint, kad GNN modelis identifikuotų tarpusavio priklausomybes ir makroekonominis ryšius, į krepšelį įtraukta lyderiai iš skirtingų GICS (angl. *Global*

*Industry Classification Standard*) sektorių: technologijų, finansų, energetikos, telekomunikacijų, vartojimo prekės, sveikatos apsauga.

Toliau pateikiamas tyrime naudojamų finansinių aktyvų sąrašas (žr. 3 lentelę).

**3 lentelė. Tyrime naudojamų finansinių aktyvų sąrašas ir jų pasiskirstymas pagal sektorius**

Simbolis (Ticker)	Įmonė	Sektorius
AAPL	<i>Apple Inc.</i>	Informacinės technologijos
MSFT	<i>Microsoft Corp.</i>	Informacinės technologijos
NVDA	<i>NVIDIA Corp.</i>	Puslaidininkiai / AI
AMZN	<i>Amazon.com Inc.</i>	Vartojimo prekės
TSLA	<i>Tesla Inc.</i>	Automobilių pramonė
GOOGL	<i>Alphabet Inc. (Google)</i>	Komunikacijos paslaugos
META	<i>Meta Platforms Inc.</i>	Komunikacijos paslaugos
JPM	<i>JPMorgan Chase &amp; Co.</i>	Finansai (bankininkystė)
V	<i>Visa Inc.</i>	Finansai (mokėjimai)
JNJ	<i>Johnson &amp; Johnson</i>	Sveikatos apsauga
PFE	<i>Pfizer Inc.</i>	Farmacija
XOM	<i>Exxon Mobil Corp.</i>	Energetika (nafta ir dujos)
PG	<i>Procter &amp; Gamble Co.</i>	Kasdienio vartojimo prekės
KO	<i>Coca-Cola Co.</i>	Maisto pramonė
BA	<i>Boeing Co.</i>	Pramonė / aviacija

Duomenų kintamieji: kiekvienam iš išvardintų aktyvų buvo surinkti duomenys nuo 2019-01-01 iki 2025-01-01, - kad susidarytų 6 metų laikotarpis modeliams mokyti

Modelių mokymui naudojami šie pirminiai kintamieji:

- *Open*: atidarymo kaina.
- *High*: aukščiausia dienos kaina.
- *Low*: žemiausia dienos kaina.
- *Close*: uždarymo kaina (naudojama techninių indikatorių skaičiavimui).
- *Adj Close*: pakoreguota uždarymo kaina (įvertinant dividendus ir padalijimus)
- *Volume*: prekybos apimtis (svarbi likvidumo ir trendo stiprumo analizei).

## 2.2. Tikslo kintamojo formavimas ir klasių apibrėžimas

Bandyti atspėti tikslią akcijos kainą dažnai yra neefektyvu dėl didelio rinkos nenusėjamumo ir svyravimų. Todėl šiame tyrime pasirenkame prognozuoti ne patį skaičių, o kainos judėjimo kryptį.

## Gražos skaičiavimas

Prieš priskiriant klases, yra apskaičiuojama logaritminė graža (angl. Log-return). Logaritminė graža pasirenkama vietoje paprastos procentinės gražos dėl jos statistinių savybių: ji yra simetriškesnė ir, darant prielaidą apie nuolatinį palūkanų skaičiavimą (angl. *continuous compounding*), leidžia lengviau agreguoti gražą per ilgesnius laiko periodus. [25]

Logaritminė gražą  $r_{t+h}$  periodui  $h$  apskaičiuojama taip:

$$r_{t+h} = \ln\left(\frac{P_{t+h}}{P_t}\right); \quad (1)$$

čia

$P_t$  – turto uždarymo kaina laiko momentu  $t$ .

$P_{t+h}$  – turto uždarymo kaina po prognozavimo horizonto  $h$  (pvz.,  $h = 1$  dienai arba  $h = 5$  savaitės prognozei)

## Trijų klasių žymėjimo logika

Kiekvienam laiko momentui  $t$  suformuojamas tikslo kintamasis  $Y_t$ , kuris gali įgyti vieną iš trijų reikšmių. Klasės apibrėžiamos naudojant simetrinį slenkstį  $\theta$  (theta):

$$Y_t = \begin{cases} 1 \text{ (Kyla)}, & \text{jei } r_{t+h} > \theta_t \\ -1 \text{ (Krenta)}, & \text{jei } r_{t+h} < -\theta_t \\ 0 \text{ (Stabili)}, & \text{kitu atveju } (-\theta_t \leq r_{t+h} \leq \theta_t) \end{cases}; \quad (2)$$

Ši „Stabili“ (0) klasė yra labai svarbi. Ji atlieka filtro funkciją: eliminuoja mažus rinkos svyravimus, kurie yra tiesiog rinkos triukšmas (angl. *noise*) ir kurių metu prekyba būtų labiau nuostolinga dėl komisinių mokesčių.

## Dinaminio slenkščio ( $\theta$ ) nustatymas

Finansų rinkose akcijų kainų svyravimai (kintamumas) nuolat keičiasi – vienais periodais rinka yra labai stabili, o krizių metu gali prasidėti didžiuliai kainų šuoliai. Dėl šios priežasties naudoti vieną fiksuotą taisyklę (pavyzdžiui, laikyti, kad kainos pokytis yra reikšmingas signalas tik tuomet, jei jis viršija 0.5 %) būtų metodiškai neteisinga, nes:

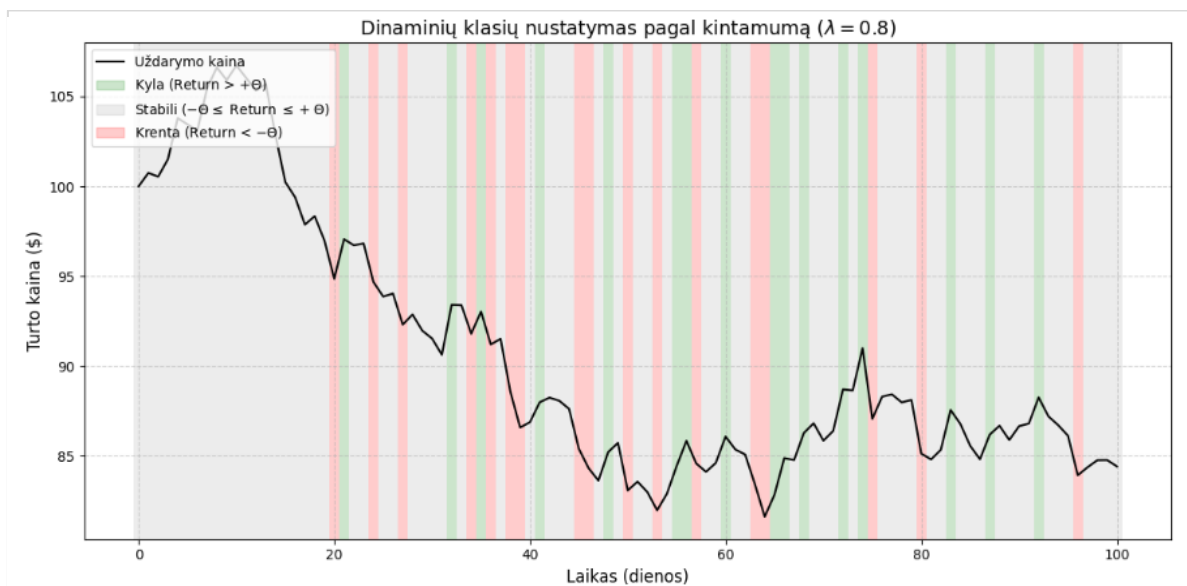
1. Didelio kintamumo periodais 0.5% pokytis yra nereikšmingas triukšmas
2. Ramiu periodu 0.5 % pokytis gali būti reikšmingas signalas
3. Skirtingi aktyvai pasižymi skirtingu kintamumu. Pvz., „Tesla“ vs „Coca-Cola“

Siekiant išspręsti šią problemą, tyrime taikome dinaminį slenkstį, jis yra pagrįstas istoriniu standartiniu nuokrypiu. Slenkstis  $\theta_t$  kiekvienai dienai apskaičiuojama individualiai

$$\theta_t = \lambda \cdot \sigma_t(w); \quad (3)$$

čia

- $\sigma_t(w)$  – slenkantis standartinis nuokrypis (angl. Rolling standard deviation) per pastarąsias  $w$  dienų (šiam tyrimui  $w$  gali būti įvairus, nuo kelių dienų iki kelių savaitių langas)
- $\lambda$  – jautrumo koeficientas



2 pav. Klasifikavimo metodikos vizualizacija

Šaltinis: sudaryta autoriaus.

Grafike (žr. 2 pav.) pavaizduota turto kainos dinamika laike. Spalvotos vertikalios zonos žymi priskirtą klasę kiekvienam laiko momentui  $t$ , remiantis dinaminio slenkščio ( $\theta$ ) metodika. Žalia zona žymi pirkimo signalą („Kyla“), kai graža viršija kintamumo slenkstį. Raudona zona žymi pardavimo signalą („Krenta“). Pilka zona („Stabili“) indikuoja laikotarpius, kai kainos pokytis neviršija nustatyto stochastinio triukšmo lygio.

### Klasių balanso ir parametro $\lambda$ parinkimas

Parametras  $\lambda$  reguliuoja klasių balansą.

- Jei  $\lambda$  mažas (pvz., 0,1), klasė „Stabili“ bus labai maža, modelis bus priverstas beveik visada spėti kryptį.
- Jeigu  $\lambda$  didelis (pvz., 1,0), dauguma duomenų pateks į „Stabili“ klasę, ir modelis retai generuos prekybos signalus.

Šiame tyrimui  $\lambda$  reikšmė bus parinkta eksperimentiniu būdu, siekiant kuo labiau sumažinti klasių disbalansą. Tokiu pasirinkimu yra siekiama, kad klasės būtų pasiskirsčiusios apytiksliai proporcingai. ~33 % kyla, ~33 % krenta, ~33 % stabili.

#### 2.2.1. Slenkančio lango metodika ir tyrimo etapai

Metodo pasirinkimo pagrindimas. Standartiniai mašininio mokymosi validacijos metodai, tokie kaip *K-Fold Cross-validation*, remiasi prielaida, kad duomenys nepriklausomi ir vienodai pasiskirstę (angl. *i.i.d.* – *independent and identically distributed*). Tačiau finansinėse laiko eilutėse ši prielaida negalioja dėl dviejų priežasčių:

1. Laiko priklausomybė: šios dienos kaina yra stipriai koreliuota su vakarykšte kaina. Atsitiktinis duomenų sumaišymas (angl. Shuffling) sunaikintų šią struktūrą.
2. Nestacionarumas (angl. *non-stationarity*): rinkos režimai kinta. Modelis, mokytas remiantis 2020 m. duomenimis, gali prastai veikti 2024 m. rinkos sąlygomis.

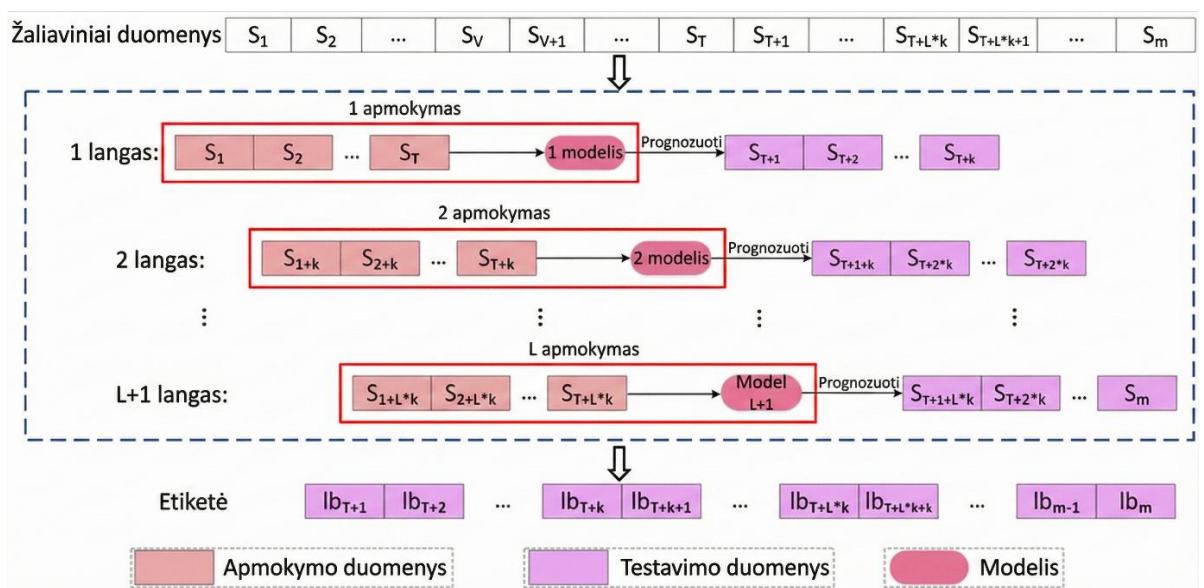
Dėl šių priežasčių tyrime taikoma slenkančio lango (angl. *Rolling window*) validacija. Tai procedūra, kuri griežtai laikosi chronologijos ir užkerta kelią, kai modelis gali pamatyti ateities kainas treniravimo metu.

Algoritmo veikimo principas. Slenkančio lango metodas imituoja realų investuotojo elgesį: kiekvieną dieną (arba periodą) modelis yra mokomas naudojant tik tuo metu prieinamą istorinę informaciją.

Procesas aprašomas šiais žingsniais:

1. Nustatomas fiksuoto dydžio mokymo langas  $W$  (angl. *Training Window*).
2. Modelis mokomas naudojant duomenis intervale  $[t, t + W - 1]$ .
3. Atliekama prognozė testavimo intervalui  $H$  (angl. *Horizon*), kuris prasideda iškart po mokymo lango:  $[t + W, t + W + H - 1]$ .
4. Apskaičiuojama prognozės paklaida lyginant su tikraisiais duomenimis.
5. Langas paslenkamas per žingsnį  $S$  (angl. *Step size*), t. y.  $t \leftarrow t + S$ .
6. Procesas kartojamas tol, kol pasiekiami duomenų masyvo pabaiga.

Šiame tyrime bus naudojamas fiksuoto dydžio (*Rolling*) langas, o ne plečiantis (*Expanding*) langas. Fiksuoto dydžio langas leidžia modeliui greičiau „pamiršti“ senus, nebeaktualius rinkos režimus ir fokusuotis į naujausias tendencijas, kas yra svarbu dirbant su dažnai besikeičiančiais duomenimis ir GNN modeliais.



3 pav. Slenkančio lango metodas laiko eilučių prognozavimui

Šaltinis: sudaryta autoriaus, remiantis Tsay (2010)

Vienas iš pagrindinių šio tyrimo uždavinių – empiriškai nustatyti optimalią duomenų imties struktūrą pasirinktiems mašininio mokymosi modeliams. Tuo tikslu eksperimento metu bus varijuojami trys kritiniai parametrai: mokymo lango dydis ( $W$ ), langų persidengimo laipsnis, kurį reguliuoja poslinkio žingsnis ( $S$ ) ir prognozavimo horizontas ( $H$ )

Langų persidengimo (angl. *Overlap*) svarba. Didelis langų persidengimas (kai  $S \ll W$ ) leidžia sugeneruoti maksimalų kiekį treniravimo pavyzdžių iš ribotos istorinės imties, kas yra naudinga giliojo mokymosi modeliams (*GNN*, *LSTM*). Tačiau per didelis persidengimas gali lemti duomenų perteklišumą ir didelius skaičiavimo kaštus.

Persidengimo koeficientas ( $\eta$ ) apskaičiuojamas taip:

$$\eta = \frac{W-S}{W} \cdot 100\%; \quad (4)$$

bus testuojami šie poslinkio žingsniai ( $S$ ):

$S = 5$  dienos (kas savaitę – greitas prisitaikymas, leidžiantis užfiksuoti trumpalaikius pokyčius).

$S = 10$  ir  $S = 20$  dienų (tarpiniai ir retesni atnaujinimai, vertinant ilgesnio laikotarpio stabilumą).

### **Eksperimentiniai scenarijai ir ribų testavimas**

Siekiant įvertinti modelių patikimumą ir rasti universalius parametrus, tyrimas struktūrizuojamas keliais pagrindiniais analizės etapais, kurie atitinka skirtingus algoritmų kalibravimo ir testavimo etapus.

I Etapas: bazinė klasikinio modelio (*XGBoost*) langų jautrumo analizė.

Pirmajame etape atliekama tiesioginė jautrumo analizė su klasikiais medžių algoritmais (*XGBoost*).

- Procedūra: fiksuojamas trumpas prognozės horizontas, o mokymo lango dydis ( $W$ ) ir slinkimo žingsnis ( $S$ ) keičiami, vertinant skirtingas kombinacijas (pvz., trumpi langai prieš ilgus langus, dažnas perbalansavimas prieš retą perbalansavimą).
- Tikslas: nustatyti pradinį atskaitos tašką (angl. *baseline*) ir suprasti, kaip tradiciniai modeliai reaguoja į duomenų „atminties“ trukmę. Tikimasi identifikuoti, ar papildomi istoriniai duomenys gerina tikslumą.

II Etapas: dviejų etapų optimizacija giliojo mokymosi architektūroms (*GNN*).

Skirtingai nuo klasikinių modelių, sudėtingos *GNN* architektūros reikalauja atskiro vidinės struktūros ir laiko dinamikos kalibravimo.

- 1 žingsnis (Architektūros paieška): užfiksuojami laiko parametrai ir keičiami vidiniai tinklo hiperparametrai (paslėptųjų neuronų skaičius, mokymosi greitis). Ieškoma konfigūracijos su kuria gaunamas geriausias rezultatas.

- 2 žingsnis (Dinamikos kalibravimas): pritaikius optimalią tinklo struktūrą, atliekamas platus slenkančio lango laiko kintamųjų (angl. *Grid Search*) testavimas. Varijuojama  $W$  (nuo 30 iki 252 dienų) ir  $S$  (nuo 5 iki 20 dienų) reikšmėmis.
- Tikslas: rasti geriausią investicijų derinį, kuris užtikrintų didžiausią pelną priimant mažiausią riziką ir geriausiai pasinaudotų informacija apie tai, kaip skirtingos investicijos yra susijusios tarpusavyje.

III Etapas: praktinis modelių ir portfelių testavimas mikro ir makro lygmenimis. Nustačius optimalius parametrus, atliekama galutinė lyginamoji analizė realiomis rinkos sąlygomis:

- Mikro lygmuo: skirtingų architektūrų (*XGBoost*, *SVM*, *LSTM*, *GRU*, *GNN*) palyginimas prognozuojant individualių akcijų kainų kryptis, vertinant jų rizikos (*Calmar*, *Rachev*, *MDD*) rodiklius. Šiame etape taip pat atliekamas statistinis (*Diebold-Mariano* testas) patikrinimas.
- Makro lygmuo: geriausiai pasirodžiusio modelio (*GNN*) signalų pritaikymas viso 15-os akcijų portfelio formavimui. Šiame etape lyginamos portfelio optimizavimo strategijos ( $1/N$  ir *Markowitz*) bei atliekamas ekonometrinis vertinimas (ACF analizė, koreliacija su S&P 500 ir Treynor-Mazuy testai).

### 2.3. Mašininio mokymosi modelių architektūra ir hiperparametrų optimizavimas

Kiekvienam iš pasirinktų modelių (*GNN*, *LSTM*, *XGBoost*) bus atliekamas hiperparametrų derinimas. Kadangi finansiniai duomenys yra triukšmingi, netinkami hiperparametrai gali lemti greitą persimokymą (angl. *overfitting*).

Bus naudojamas tinklelio paieškos (angl. *Grid Search*) metodas slenkančiame lange. Pagrindiniai optimizuojami parametrai:

- *GNN* modeliui: paslėptųjų neuronų skaičius (angl. *Hidden dimensions*) ir mokymosi greitis (angl. *Learning rate*), siekiant subalansuoti tinklo talpą su greitu prisitaikymu prie rinkos koreliacijų.
- *LSTM/GRU* modeliams: neuronų skaičius paslėptuosiuose sluoksniuose ir mokymosi greičio parametrai, siekiant reguliarizuoti modelį ir išvengti persimokymo analizuojant trumpas laiko eilutes.
- *XGBoost*: maksimalus medžio gylis (angl. *max\_depth*) ir bazių iteracijų skaičius (angl. *n\_estimators*), kontroliuojant modelio sudėtingumą.

Kiekviena parametrų kombinacija vertinama pagal jos gebėjimą generalizuotis – t. y., ar ta pati parametrų aibė veikia sėkmingai su visa 15-os aktyvų imtimi, apimančia skirtingos dinamikos sektorius (nuo technologijų ir finansų iki energetikos bei sveikatos apsaugos). Siekiama rasti universalius nustatymus, kurie nebūtų „pritempti“ tik vienai specifinei industrijai.

Šiame skyriuje toliau aprašomi tyrime naudojami algoritmai. Kiekvienas modelis pasirinktas atliepti specifines finansinių laiko eilučių savybes: netiesiškumą, laiko priklausomybę ir struktūrinius ryšius tarp aktyvų

### 2.3.1. Grafų neuroniniai tinklai (GNN)

Grafų neuroniniai tinklai (GNN) leidžia analizuoti duomenis kaip vieną tarpusavyje susijusį tinklą. Šiame tyrime GNN padeda vertinti finansų rinką kaip bendrą sistemą, kurioje ryšiai tarp skirtingų investicijų yra lygiai tokie pat svarbūs, kaip ir patys aktyvai. Paprastais žodžiais tariant, šis metodas remiasi logiška prielaida: vienos akcijos situacija priklauso ne tik nuo jos pačios individualių rodiklių, bet ir nuo to, kas vyksta su kitomis, su ja glaudžiai susijusiomis akcijomis (jos „kaimynėmis“) [28].

Grafo apibrėžimas: Rinka apibrėžiama kaip grafas  $G = (V, E)$ , kur:

- $V$  (viršūnės) – tai  $N = 15$  pasirinktų finansinių aktyvų.
- $E$  (briaunos) – tai ryšiai tarp aktyvų, aprašomi gretimumo matrica  $A \in R^{N \times N}$

Gretimumo matrica ( $A$ ): šiame tyrime grafo briaunos  $A_{ij}$  nustatomas pagal Pirsono koreliacijos koeficientą  $\rho$  tarp aktyvų  $i$  ir  $j$  gražų per mokymo langą. Pirsono koreliacijos koeficiento metodas pasirinktas todėl, nes jis pasižymi, jog yra populiariausias metodas grafų briaunoms apskaičiuoti, bei reikalaujantis sąlyginai mažiau kompiuterinių resursų, lyginant su kitais netiesiniais metodais.

Matematiškai briaunos svoris apibrėžiamas taip:

$$A_{ij} = \begin{cases} |\rho_{ij}|, & \text{jei } |\rho_{ij}| > \delta; \\ 0, & \text{kitu atveju} \end{cases}; \quad (6)$$

čia  $\delta$  yra slenkstis (pvz., 0.3), skirtas eliminuoti silpnus, triukšmingus ryšius.

Informacijos sklidimas (angl. *Message Passing*): tai procesas, kurio metu kiekvienas tinklo elementas (viršūnė) atnaujina savo duomenis, remdamasis greta esančių „kaimynų“ informacija. Kiekvienas taškas nuolat surenka ir sujungia informaciją iš visų su juo susijusių elementų. Taip kiekviena viršūnė prisitaiko ir tiksliau įvertina savo paties būseną, atsižvelgdama į tai, kas vyksta jos artimiausioje aplinkoje [28].

$$H^{(l+1)} = \sigma \left( \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} \right); \quad (7)$$

Kur:

- $H^{(l)}$  – požymių matrica sluoksnyje  $l$
- $\tilde{A} = A + I$  – gretimumo matrica su pridėtomis savės kilpomis (self-loops)
- $\tilde{D}$  – laipsnių matrica (normalizavimui)
- $W^{(l)}$  – mokomi svoriai
- $\sigma$  – aktyvacijos funkcijas (pvz., ReLU).

### 2.3.2. Laiko eilučių giliojo mokymosi modeliai (LSTM ir GRU)

LSTM (*Long Short-Term Memory*) architektūra pasirinkta dėl jos unikalios savybės spręsti „nykstančio gradiento“ (angl. *vanishing gradient*) problemą, kuri yra būdinga standartiniams rekurentiniams tinklams (RNN) mokantis ilgų sekų. Straipsnyje [27] įrodė, kad standartiniai RNN negali efektyviai išlaikyti informacijos per ilgus laiko intervalus dėl eksponentiškai

mažėjančio klaidos signalo, todėl buvo pasiūlyta įvesti atminties celes su „Constant Error Carousel“ (CEC) mechanizmu. [27]

LSTM celės būseną laiko momentu  $t$  atnaujinama naudojant tris „vartus“ (gates):

1. Užmiršimo vartai ( $F_t$ ): sprendžia, kurią informaciją iš praeities pašalinti

$$F_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f); \quad (7)$$

2. Įėjimo vartai ( $i_t$ ): sprendžia, kurią naują informaciją įrašyti į atmintį.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i); \quad (8)$$

3. Išėjimo vartai ( $o_t$ ): nustato, koks bus celės išėjimas (progozė)

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o); \quad (9)$$

čia  $\sigma$  yra sigmoid funkcija, o  $W$  ir  $b$  – modelio parametrai.

GRU (angl. *Rater Recurrent Unit*): tai supaprastinta LSTM versija, turinti tik du vartus (Update ir Reset) matematiškai ji aprašoma lygtimis:

$$4. z_t = \sigma(W_z \cdot [h_{t-1}, x_t]); \quad (10)$$

Šiame tyrime tikrinama hipotezė, ar dėl mažesnio architektūrinio sudėtingumo ir mažesnio parametų skaičiaus (lyginant su LSTM), GRU modelis geba efektyviau išvengti persimokymo (angl. *overfitting*) trumpuose slenkančiuose languose, neprarasdamas prognozavimo tikslumo.

### 2.3.3. Klasikiniai mašininio mokymosi modeliai (XGBoost ir SVM)

XGBoost (eXtreme Gradient Boosting): XGBoost metodas pasirinktas dėl jo unikalios tikslo funkcijos, kuri, skirtingai nuo tradicinio *Gradient Boosting*, įtraukia reguliarizacijos narį  $\Omega$  modelio sudėtingumui kontroliuoti. Matematiškai tikslo funkcija užrašoma:

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k); \quad (11)$$

Kur pirmasis narys matuoja prognozės paklaidą, o antrasis (reguliarizacija) baudžia už medžių sudėtingumą, taip padedant išvengti persimokymo (angl. *overfitting*) ant mažesnių duomenų imčių. Be to, šis algoritmas turi integruotą „retumo suvokimo“ (angl. *sparsity-aware*) mechanizmą, kuris leidžia efektyviai apdoroti trūkstamas reikšmes, automatiškai parenkant optimalią klaidų šakojimosi kryptį. [26]

SVM (Atraminių vektorių mašinos): klasikinis algoritmas, ieškantis optimalios hiperplokštumos, kuri maksimaliai atskirtų klases. Kadangi finansiniai duomenys nėra tiesiškai atskiriami, naudojama branduolio funkcija (angl. *Kernel trick*), pvz., RBF (*Radial Basis Function*), transformuojanti duomenis į aukštesnės dimensijos erdvę:

$$K(x, x') = \exp(-\gamma \|x - x'\|^2); \quad (12)$$

### 2.3.4. Ansambliniai modeliai ir „Minkštasis balsavimas“

Ansambliniai modeliai ir „Minkštasis balsavimas“ (angl. *Soft Voting*): šiame tyrime apjungiami keli skirtingi modeliai, kad būtų pasinaudota stipriausiomis kiekvieno iš jų savybėmis. Galutiniam sprendimui priimti naudojamas „minkštasis balsavimas“. Užtuot

tiesiog rinkus populiariausią modelių atsakymą, šis metodas apskaičiuoja kiekvieno varianto (pavyzdžiui, „Kilti“, „Kristi“ ar „Stabili“) tikimybių vidurkį pagal visus modelius. Galiausiai pasirenkamas tas variantas, kurio bendra tikimybė yra didžiausia. Tai padeda sumažinti pavienių modelių daromas klaidas.

## 2.4. Modelių tikslumo ir finansinio veiksmingumo vertinimo metrikos

### 2.4.1. Klasifikavimo metrikos

Kadangi sprendžiamas trijų klasių („Kyla“, „Krenta“, „Stabili“) klasifikavimo uždavinys, o duomenų rinkiniai gali būti nesubalansuoti, vien tik bendro tikslumo (Accuracy) metrikos mums neužtenka.

Vertinimui naudojama Klaidingų sprendimų matrica (angl. *Confusion Matrix*), kurios pagrindu skaičiuojami šie rodikliai:

Tikslumas (angl. *Accuracy*): rodo bendrą teisingų prognozių dalį:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}; \quad (13)$$

tačiau finansinėse prognozėse tai gali būti klaidinantis rodiklis, jei dominuoja „Stabili“ klasė.

Tikslumas (angl. *Precision*) šiame tyrime tai yra itin svarbi metrika. Tikslumas parodo, kokia dalis modelio sugeneruotų „Pirkimo“ arba „Pardavimo“ signalų buvo teisingi

$$Precision = \frac{TP}{TP+FP}; \quad (14)$$

finansinė interpretacija: žemas tikslumo rodiklis reiškia, kad modelis generuoja daug klaidingų signalų (angl. *False Positives*), dėl kurių atidaromos nuostolingos pozicijos ir patiriami nereikalingi prekybos kaštai.

Jautrumas (angl. *Recall arba Sensitivity*): parodo, kokią dalį visų realiai įvykusių rinkos judėjimų modelis sugebėjo identifikuoti.

$$Recall = \frac{TP}{TP+FN}; \quad (15)$$

finansinė interpretacija: žemas atšaukimo rodiklis rodo praleistas progas (angl. *opportunity cost*), tačiau algoritminėje prekyboje dažnai prioritetas teikiamas aukštesniam tikslumui, o ne jautrumui.

F1-įvertis (Makro vidurkis) [angl. *F1-Score Macro-average*]: tai bendras įvertinimas, apjungiantis modelio tikslumą ir jautrumą. Žodis „Makro“ čia reiškia, kad modelis vienodai svarbiais laiko visus tris galimus scenarijus (rinka kyla, krenta arba yra stabili), neatsižvelgiant į tai, kaip dažnai jie realiai pasitaiko duomenyse. Taip užtikrinama, kad modelio vertinimas būtų objektyvus ir jis nebūtų dirbtinai giriamas vien už tai, kad gerai atspėja patį dažniausią variantą.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}; \quad (16)$$

## 2.4.2. Finansinio veiksmingumo ir rizikos rodikliai

Sharpe rodiklis (Sharpe Ratio): tai rodiklis, vertinantis perteklinę grąžą vienam rizikos vienetui. Kaip nurodoma literatūroje, Sharpe rodiklis apskaičiuojamas padalijus vidutinę perteklinę grąžą iš jos standartinio nuokrypio [16]

$$Sharpe = \frac{\overline{R_p} - R_f}{\sigma_p}; \quad (17)$$

čia  $\overline{R_p}$  yra vidutinė portfelio grąža,  $R_f$  – nerizikinga palūkanų norma (angl. *Risk-free rate*), o  $\sigma_p$  – portfelio grąžos standartinis nuokrypis. Teigiamas Sharpe rodiklis indikuoja, kad strategija generuoja grąžą, kuri kompensuoja prisiimtą riziką.

Sortino rodiklis (angl. *Sortino Ratio*): kadangi finansinių aktyvų grąžos dažnai nėra normaliai pasiskirsčiusios, Sharpe rodiklis gali neteisingai nubausti strategiją už teigiamą kintamumą (didelį pelną). Sortino rodiklis yra Sharpe modifikacija, kuri riziką vertina tik kaip neigiamą nuokrypį (angl. *downside deviation*)

$$Sortino = \frac{\overline{R_p} - R_f}{\sigma_d}; \quad (18)$$

čia  $\sigma_d$  yra nuostolingų periodų standartinis nuokrypis. Tai ypač aktualu mašininio mokymosi modeliamas, kurie siekia išvengti didelių nuosmukių

Maksimalus nuosmukis (angl. *Maximum Drawdown - MDD*): tai didžiausias galimas kapitalo praradimas nuo pasiekto piko iki dugno per analizuojamą laikotarpį. Šis rodiklis yra svarbus vertinant strategijos tvarumą kriziniais rinkos laikotarpiais

$$MDD = \min_t \left( \frac{V_t - V_{peak}}{V_{peak}} \right); \quad (19)$$

Calmar rodiklis: tai dar griežtesnis rodiklis nei Sharpe, lyginantis metinę grąžą (*CAGR*) su maksimaliu nuosmukiu. Jis parodo, ar potencialus pelnas vertas rizikos patirti didžiausią istorinį kritimą.

$$Calmar = \frac{AnnualizedReturn}{|MaxDrawdown|}; \quad (20)$$

Rachev rodiklis (angl. *Rachev Ratio*): šis rodiklis paprastais žodžiais parodo, kas yra didesni – patys didžiausi jūsų pelnai ar patys didžiausi nuostoliai. Jis skaičiuojamas padalinant 5 % pačių geriausių rezultatų (didžiausių pelnų) vidurkį iš 5 % pačių blogiausių rezultatų (didžiausių nuostolių) vidurkio. Jei gautas skaičius yra didesnis nei 1,0, tai reiškia, kad jūsų išskirtiniai pelnai viršija išskirtinius nuostolius. Kitaip tariant, geriausiais momentais jūs uždirbate daugiau, nei prarandate pačiais prasciausiais.

Pelningumo faktorius (angl. *Profit Factor*): tai bendrojo pelno (Gross Profit) ir bendrojo nuostolio (Gross Loss) santykis.

$$ProfitFactor = \frac{\sum Pelnas}{\sum |Nuostolis|}; \quad (21)$$

Reikšmė  $> 1.0$  rodo pelningą sistemą, tačiau algoritminėje prekyboje siekiama reikšmės  $> 1.5$ .

Laimėjimo dažnis (angl. *Win Rate*): nors tai atrodo paprastas rodiklis (teisingų sandorių procentas), finansuose jis vertinamas kartu su vidutiniu pelnu. Aukšto dažnio prekyboje (HFT) modeliai gali būti pelningi net su  $< 50\%$  Win Rate, jei vidutinis pelnas žymiai viršija vidutinį nuostolį.

#### **2.4.3. Portfelio lyginamosios analizės rodikliai**

Modigliani-Modigliani (*M2*) matas: tai Sharpe rodiklio modifikacija, išreikšta procentine graža. *M2* parodo, kokia būtų strategijos graža, jei jos rizika (standartinis nuokrypis) būtų dirbtinai sulyginata su atskaitos taško (pvz., S&P 500 indekso) rizika. Tai leidžia investuotojams kur kas intuityviau palyginti skirtingo rizikingumo portfelius.

Informacijos rodiklis (angl. *Information Ratio*, IR): matuoja modelio sugeneruotą aktyviają gražą (virš atskaitos indekso), padalintą iš sekimo paklaidos (angl. *Tracking Error*). Tai leidžia įvertinti, ar nukrypimas nuo pasyvaus indekso sekimo buvo kompensuotas papildomu pelnu.

#### **2.4.4. Rezultatų statistinio reikšmingumo testai**

Diebold-Mariano (DM) testas: ekonometrinis testas, skirtas patikrinti, ar dviejų konkuruojančių modelių (pvz., izoliuoto LSTM ir tinklinio GNN) prognozių tikslumo skirtumas yra statistiškai reikšmingas. Nulinė hipotezė teigia, kad abiejų modelių prognozių paklaidos yra vienodos. Reikšmingumo lygmeniui esant  $p < 0.05$ , nulinė hipotezė atmetama. [15]

Treynor-Mazuy testas (angl. *Market Timing*): kvadratinės regresijos modelis, vertinantis, ar portfelio valdytojas (arba mašininio mokymosi algoritmas) geba sėkmingai nuspėti bendrą makroekonominę rinkos kryptį. Teigiamas ir statistiškai reikšmingas gama koeficientas rodo, kad modelis sėkmingai padidina rinkos rizikos poveikį kylančioje rinkoje ir sumažina jį krentančioje rinkoje.

### 3. Tyrimo rezultatai ir jų aptarimas

#### 3.1. Slenkančio lango parametrų ir modelių architektūros optimizavimas

Kuriant ir pritaikant mašininio mokymosi modelius finansinių laiko eilučių analizei, susiduriama su augančiu skaičiavimo resursų problema. Kadangi modeliai naudoja dinaminę slenkančio lango (angl. *Rolling window*) metodiką, jie privalo būti perskaiciuojami ir mokomi iš naujo kiekviename laiko žingsnyje. Siekiant išvengti modelių persimokymo (angl. *overfitting*) testavimo imtyje ir užtikrinti maksimalų algoritmų efektyvumą, šiame tyrimo etape atliekama hiperparametrų paieška. Klasikiniais modeliams (XGBoost) taikoma tiesioginė langų jautrumo analizė, o sudėtingesnėms Grafų neuroninių tinklų (GNN) architektūroms pasitelkiama dviejų etapų optimizavimo (angl. *Two-Stage Optimization*) metodika, leidžianti atskirti tinklo architektūros kalibravimą nuo rinkos laiko dinamikos vertinimo.

##### 3.1.1. XGBoost langų ir slinkimo jautrumo analizė

Pirmajame žingsnyje vertinamas klasikinio medžių algoritmo (XGBoost) jautrumas istoriniam apmokymo lango dydžiui (W) ir slinkimo žingsniui (S).

**4 lentelė.** Slenkančio lango parametrų (W ir S) optimizavimo rezultatai pagal vidutinį Sharpe rodiklį

Vieta	Lango dydis (W)	Slinkimo žingsnis (S)	Vidutinis Sharpe rodiklis
1	126 dienos (~6 mėn.)	20 dienų (~1 mėn.)	0.02 (didžiausias)
2	252 dienos (~1 metai)	20 dienų (~1 mėn.)	-0.104
3	126 dienos (~6 mėn.)	5 dienos (~1 sav.)	-0.119
4	60 dienų (~3 mėn.)	20 dienų (~1 mėn.)	-0.2
5	252 dienos (~1 metai)	5 dienos (~1 sav.)	-0.238
6	60 dienų (~3 mėn.)	5 dienos (~1 sav.)	-0.362

Kaip matyti lentelėje, atlikus dviejų parametrų (lango dydžio W ir slinkimo žingsnio S) jautrumo analizę su XGBoost algoritmu, nustatyta aiški tendencija: per dažnas modelio mokymas (S = 5 dienos) visais testuotais atvejais mažina strategijos efektyvumą ir lemia neigiamą Sharpe rodiklį. Tai patvirtina prielaidą, kad perteklinis prekybos aktyvumas (angl. *over-trading*) generuoja per daug klaidingų signalų. Optimaliausia kombinacija, pasiekusi vienintelį teigiamą Sharpe rodiklį (0.020), buvo 126 dienų apmokymo langas su 20 dienų slinkimo žingsniu.

##### 3.1.2. GNN hiperparametrų paieška (I etapas)

Skirtingai nei sprendimų medžiai, Grafų neuroniniai tinklai (GNN) pasižymi sudėtinga vidine struktūra. Todėl pirmajame GNN optimizavimo etape laiko eilutės apmokymo ir testavimo langai buvo užfiksuoti, o visa skaičiavimo galia sutelkta į optimalios neuroninio tinklo topologijos paiešką. Eksperimento metu iteruota per skirtingus paslėptųjų neuronų (angl. *Hidden dimensions*) ir mokymosi greičio (angl. *Learning rate*) parametrus.

**5 lentelė.** Modelio hiperparametrų optimizavimo rezultatai pagal vidutinį Sharpe rodiklį

Vieta	Lango dydis (W)	Neuronų skaičius (Hidden)	Mokymosi greitis (LR)	Vidutinis Sharpe rodiklis
1	126 dienos	16	0.005	0.233 (didžiausias)
2	504 dienos	16	0.01	0.155
3	504 dienos	32	0.005	0.149
4	126 dienos	32	0.01	0.124
5	252 dienos	32	0.01	0.062
6	504 dienos	16	0.005	0.04
7	126 dienos	16	0.01	-0.045
8	504 dienos	32	0.01	-0.052
9	126 dienos	32	0.005	-0.06
10	252 dienos	16	0.005	-0.133
11	252 dienos	16	0.01	-0.147
12	252 dienos	32	0.005	-0.182

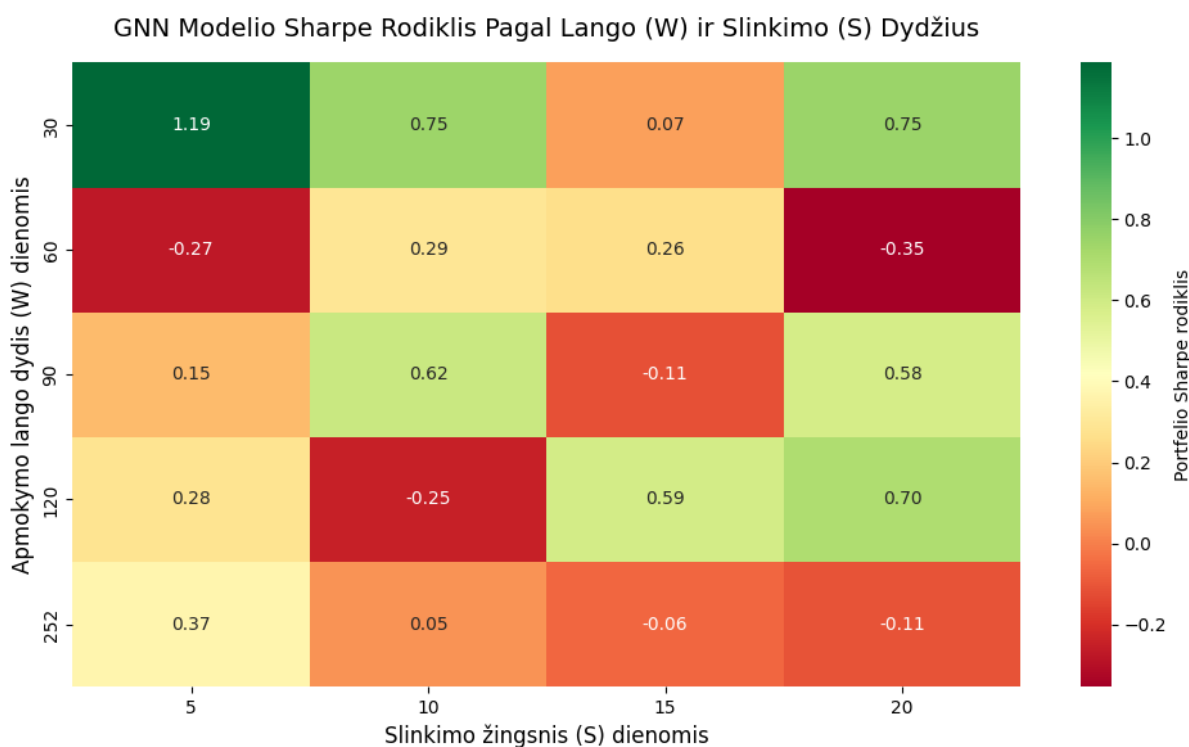
Kaip matyti 5 lentelėje, GNN modelio efektyvumas tiesiogiai priklauso nuo pasirinktų architektūros parametrų:

1. Geriausia kombinacija: aukščiausią Sharpe rodiklį (0.233) pasiekė modelis su 126 dienų apmokymo langų, 16 neuronų paslėptajame sluoksnyje ir lėtesniu 0.005 mokymosi greičiu. Tai rodo, kad trumpesnis langas leidžia modeliui greičiau reaguoti į rinkos pokyčius, tačiau tam reikalinga atsargesnė (mažesnis LR) mokymosi strategija.
2. Stabilumas su ilgu langų: antroje ir trečioje vietoje liko 504 dienų (2 metų) lango kombinacijos. Tai patvirtina, kad GNN modeliai geba efektyviai veikti ir su ilgaisiais istoriniais duomenimis, jei parametrai yra subalansuoti.
3. Neuronų skaičiaus įtaka: pastebėta tendencija, kad 16 neuronų konfigūracijos dažnai lenkia 32 neuronų versijas (užima 1-ą ir 2-ą vietas). Tai indikuoja, kad mažesnės talpos (angl. *lower capacity*) tinklas finansiniuose uždaviniuose padeda išvengti persimokymo (angl. *overfitting*), priversdamas modelį ieškoti tik pačių svarbiausių rinkos ryšių.

### 3.1.3. GNN slenkančio lango dinamikos kalibravimas (II etapas)

Užfiksavus optimalius GNN architektūros hiperparametrus (paslėptųjų neuronų skaičius = 16, mokymosi greitis = 0.005), antrajame optimizavimo etape atlikta išsami slenkančio lango laiko kintamųjų paieška (angl. *Grid Search*). Šiame etape buvo siekiama nustatyti, koks istorinės informacijos kiekis (apmokymo langas, W) ir koks reagavimo greitis (slinkimo žingsnis, S) leidžia modeliui efektyviausiai išnaudoti grafų struktūroje užkoduotus aktyvų tarpusavio ryšius. Eksperimento metu ištestuota 20 parametrų kombinacijų (W [30, 60, 90,

120, 252] ir S [5, 10, 15, 20]). Gauti simuliacijos rezultatai vertinti pagal sugeneruoto investicinio portfelio Sharpe rodiklį ir atvaizduoti šilumos žemėlapyje.



**4 pav.** GNN modelio Sharpe rodiklio priklausomybė nuo apmokymo lango (W) ir slinkimo žingsnio (S) dydžių

Kaip matyti iš rezultatų, GNN modelio efektyvumas yra itin jautrus laiko parametrams ir demonstruoja visiškai kitokią elgseną nei ankstesniame poskyriuje analizuotas XGBoost algoritmas. Aukščiausias portfelio Sharpe rodiklis (1.19) pasiektas naudojant patį trumpiausią testuotą apmokymo langą ( $W = 30$  dienų) kartu su pačiu dažniausiu slinkimo žingsniu ( $S = 5$  dienos).

Šis eksperimentas mums parodo finansų rinkos prielaidas apie korelacijų stabilumą. Grafų neuroniniai tinklai mokosi ne tik iš individualių akcijų kainų, bet, ir iš aktyvų tarpusavio priklausomybės pokyčių. Skirtingai nei ilgalaikės makroekonominės tendencijos, kurias geriausiai fiksuoja ilgi langai (pvz., 126 dienų langas XGBoost atveju), trumpalaikės akcijų koreliacijos finansų rinkose suyra ir formuojasi labai greitai (angl. *correlation breakdown*). Naudojant ilgus duomenų istorijos periodus ( $W = 252$  dienų), modelis bando remtis senais, jau nebegaliojančiais akcijų ryšiais. Ši pasenusi informacija algoritmui suveikia kaip triukšmas, todėl jo priimami sprendimai tampa netikslūs, o strategijos pelningumas (Sharpe rodiklis) nukrenta žemiau nulio.

Tuo tarpu itin trumpas, maždaug 1,5 prekybos mėnesio ( $W=30$ ) apmokymo langas, kombinuojamas su kassavaitiniu modelio parametrų atnaujinimu ( $S=5$ ), leidžia GNN algoritmui fiksuoti tik pačius naujausius, aktualiausius kapitalo rotacijos signalus tarp rinkos sektorių. Atsižvelgiant į šiuos iš rezultatus, tolesniame tyrime modelių lyginamajai analizei pasirenkama ir fiksuojama ši optimali dinamikos parametrų kombinacija.

### 3.2. Modelių lyginamoji analizė individualių aktyvų lygmeniu

Sėkmingai atlikus hiperparametrų optimizavimą ir nustčius optimalias laiko langų bei algoritmų architektūros kombinacijas, pereinama prie praktinio modelių testavimo mikro lygmeniu. Šiame skyriuje atliekama visapusiška lyginamoji analizė, kurioje vertinamas penkių skirtingų algoritmų – atraminių vektorių mašinų (SVM), stiprinimo medžių (XGBoost), ir giliojo mokymosi modelių (GRU, LSTM bei GNN) – efektyvumas analizuojant pavienes akcijas.

#### 3.2.1. Klasikinių ir giliojo mokymosi algoritmų efektyvumo vertinimas

Analizė atliekama naudojant 15-os akcijų portfelį, apimantį skirtingus ekonomikos sektorius (technologijų, finansų, sveikatos apsaugos, energetikos ir vartojimo prekių). Siekiant įvertinti strategijų pritaikomumą realioje rinkoje, pagrindine vertinimo metrika pasirinktas Sharpe rodiklis, matuojantis sugeneruotos grąžos ir prisiimtos rizikos (standartinio nuokrypio) santykį.

**6 lentelė.** Skirtingų prognozavimo modelių efektyvumo palyginimas atskiroms akcijoms pagal Sharpe ir Rachev rodiklius

Akcija (Ticker)	SVM		XGBoost		GRU		LSTM		GNN	
	Sharpe	Rachev	Sharpe	Rachev	Sharpe	Rachev	Sharpe	Rachev	Sharpe	Rachev
AAPL	0.34	1	0.43	0.99	0.68	1.04	0.85	1.09	0.17	1.05
MSFT	-0.09	0.91	0.5	0.99	0.09	0.91	-0.19	0.89	0.31	1.13
NVDA	1.04	1.18	0.34	1.07	1.05	1.27	0.51	1.07	0.8	1.21
AMZN	0.63	1.2	0	1.01	0.21	1.05	0.04	1.04	-0.08	0.98
TSLA	0.49	1.04	0.51	1.01	0.87	1.1	1.11	1.19	0.74	1.18
GOOGL	0.26	1.03	0.45	1.06	0.51	0.98	0.65	0.99	-0.36	0.81
META	-0.12	0.95	0.85	1.13	0.82	1.21	0.38	1.06	0.21	1.06
JPM	0.1	0.97	0.19	0.88	0.09	0.98	0.13	0.96	0.72	1.37
V	0.34	0.96	0.79	1.08	0.37	1.03	-0.34	0.79	0.51	1.16
JNJ	-0.15	0.91	0.29	1.02	-0.14	0.95	-0.12	0.96	0.63	1.27
PFE	-0.1	1	-0.33	0.96	0.06	1.08	0.18	1.17	-0.33	0.88
XOM	-0.29	0.89	-0.37	0.86	-0.14	0.87	-0.19	0.89	0.31	1.15
PG	0.35	0.92	0.36	1	0.73	1.18	1.27	1.4	1.06	1.4
KO	0.64	1.11	0.09	0.96	0.3	1.04	0.25	1.04	0.68	1.21
BA	-0.77	0.73	-0.76	0.78	0.11	1.09	0.16	1.09	0.04	1.03

Vidurkis	0.18	0.99	0.22	0.99	0.37	1.05	0.31	1.04	0.36	1.13
----------	------	------	------	------	------	------	------	------	------	------

6 lentelės duomenys rodo, kad griežtai atskyrus mokymo ir testavimo duomenis, modelių tikslumas priklauso nuo konkretaus turto bei jo sektoriaus. Lentelėje žalia ir raudona spalvomis išskirtos aukščiausios ir žemiausios rodiklių reikšmės akcijų lygmenyje, siekiant lengviau pastebėti kiekvienam aktyvui geriausiai ir prasčiausiai tinkantį algoritmą. Matome bendrą tendenciją: nėra vieno „universalus“ modelio, kuris veiktų geriausiai visomis rinkos sąlygomis. Siekiant geriau suprasti gautus rezultatus, modelius verta suskirstyti į tris kategorijas pagal jų veikimo principus:

#### 1. Klasikiniai mašininio mokymosi modeliai (XGBoost ir SVM)

- XGBoost (Atskaitos taškas / Baseline): Šis sprendimų medžių algoritmas pasiekė 0,22 vidutinį Sharpe rodiklį. Jis demonstravo gebėjimą generuoti pelną kilimą turinčiose akcijose (pvz., META: 0,85; V: 0,79). Visgi, XGBoost sunkiai tvarkėsi su nuosmukius patiriančiomis akcijomis (BA: -0,76; XOM: -0,37).
- SVM (Atraminių vektorių mašinos): Šis modelis pademonstravo žemiausią vidutinį efektyvumą (Sharpe vidurkis: 0,18). Nors SVM sugebėjo pasiekti pavienių aukštų rezultatų didelio kintamumo technologijų akcijose (NVDA: 1,04), kituose sektoriuose jis dažnai generavo neigiamą grąžą (BA: -0,77; XOM: -0,29). Tai rodo, kad SVM modelis, bandydamas rasti optimalią hiperplokštumą trumpame, triukšmingame 30 dienų lange, stipriai persimoko ties istorinėmis tendencijomis ir yra per prastas lyginant su kitais nuspėjant ateities duomenis kriziniais laikotarpiais.

#### 2. Laiko eilučių giliojo mokymosi modeliai (GRU ir LSTM)

- GRU (Gated Recurrent Unit): Šis tinklas pasiekė aukščiausią vidutinį Sharpe rodiklį (0,37) izoliuotų modelių kategorijoje. Dėl mažesnio parametrų skaičiaus (lyginant su LSTM), GRU sugebėjo greičiau ir efektyviau mokytis trumpuose slenkančiuose languose. Jis sėkmingai fiksavo judėjimą tokiose akcijose kaip NVDA (1,05) ir TSLA (0,87).
- LSTM (Ilgosios trumpalaikės atminties tinklai): Šis modelis pademonstravo labai panašų, nuoseklų efektyvumą (Sharpe vidurkis 0,31). LSTM ypač gerai pasirodė individualų augimo ciklą turinčiose akcijose (PG: 1,27; TSLA: 1,11; AAPL: 0,85). Visgi, LSTM silpnybė išryškėjo analizuojant akcijas, kurios yra stipriai priklausomos nuo platesnių makroekonominių pokyčių ar sektorių rotacijos (XOM: -0,19; MSFT: -0,19). Kadangi modelis šias laiko eilutes vertina visiškai izoliuotai, jis nesugeba atpažinti išorinių sisteminių rinkos šokų.

Vertinant išplėstinius rizikos rodiklius (pilnos Sortino rodiklių reikšmės atskiroms akcijoms pateikiamos 1 priede), pastebima ryški šių izoliuotų modelių asimetrija. Nors GRU ir LSTM demonstruoja aukštus Sortino rodiklius (atitinkamai 0,29 ir 0,24), kas indikuoja gerą grąžos generavimą stabiliose rinkose, jų Calmar rodiklio vidurkiai (0,28 ir 0,30, detaliau žr. 2 priedą) atsilieka nuo GNN modelio. Žemesnis Calmar rodiklis signalizuoja pagrindinį šių modelių trūkumą: GRU ir LSTM algoritmai per vėlai reaguoja į makroekonominius pokyčius ir patiria kur kas didesnius maksimalius kapitalo nuosmukius (angl. *Maximum Drawdown*).

3. Grafų neuroniniai tinklai (GNN) Tyrimo rezultatai patvirtina grafų neuroninių tinklų (GNN) pranašumą valdant sisteminę riziką. Nors pagal gražos ir kintamumo santykį (Sharpe rodiklio vidurkis 0,36) GNN minimaliai nusileido GRU modeliui, jis pademonstravo geriausias rezultatus visoje tyrimo imtyje pagal Calmar (0,32) ir Rachev (1,13) rodiklius. Šie metrikų rezultatai įrodo, kad GNN modelis efektyviausiai minimizuoja maksimalius kapitalo nuosmukius (angl. *Maximum Drawdown*) ir geriausiai išnaudoja teigiamą gražos asimetriją.

Skirtingai nei LSTM ar XGBoost, kurie laiko eilutes analizuoja izoliuotai, GNN naudoja dinaminę koreliacijų matricą (grafo briaunas), leidžiančią algoritmui suvokti platesnę rinkos būklę. Šis pranašumas geriausiai atsiskleidė stipriai tarpusavyje susijusiuose finansų, gynybiniuose ir energetikos sektoriuose (JPM: 0,72; PG: 1,06; KO: 0,68; XOM: 0,31 pagal Sharpe). Šiuose sektoriuose, kur akcijų kainos stipriai priklauso nuo makroekonominių ciklų ir kapitalo rotacijos, GNN sugebėjo identifikuoti sektorinį judėjimą ir išvengti nuostolių, kuriuos patyrė izoliuoti algoritmai. Kita vertus, GNN modelis susidūrė su iššūkiais analizuojant akcijas, kurios buvo stipriai paveiktos individualių, nuo bendros rinkos atsietų naujienų šokų (pvz., PFE: -0,33 pagal Sharpe), įrodant, kad GNN architektūra didžiausią vertę sukuria analizuojant cikliškus ir tarpusavyje koreliuojančius aktyvų tinklus.

### 3.2.2. Modelių tikslumo statistinis vertinimas (Diebold-Mariano testas)

Siekiant patikrinti giliojo mokymosi modelių efektyvumo skirtumų validumą, Diebold-Mariano (DM) testas buvo pritaikytas lyginant izoliuoto LSTM modelio ir tinklinio GNN modelio prognozių klaidas. Detaliam statistiniam vertinimui pasirinkta „ExxonMobil“ (XOM) akcija, kurioje Sharpe rodiklių skirtumas buvo didžiausias. Atlikus Diebold-Mariano testą su slenkančio lango testavimo imtimi (N = 956 prekybos dienos), gauti šie rezultatai:

- DM statistika: 2.4207
- p-reikšmė (p-value): 0.0155

Kadangi gauta p-reikšmė (0.0155) yra mažesnė už standartinį reikšmingumo lygmenį ( $\alpha=0.05$ ), nulinė hipotezė, teigianti, kad abiejų modelių prognozių tikslumas yra vienodas, atmetama. Tai leidžia daryti statistiškai pagrįstą išvadą, kad skirtumas tarp LSTM ir GNN modelių prognozių XOM akcijoje nėra atsitiktinis. Teigiama DM statistikos reikšmė patvirtina, kad GNN modelis daro statistiškai reikšmingai mažiau klaidų nei izoliuotas LSTM algoritmas.

### 3.2.3. Išplėstinis strategijos rizikos vertinimas ne įprastomis rinkos sąlygomis

Šiame tyrimo etape vertinama praktinė prekybos strategijų rizika, taikant išplėstines finansines metrikas: Maksimalų nuosmukį (MDD), Calmar rodiklį bei Pelningumo faktorių (Profit Factor). Šia analizei toliau naudojamas GNN modelis, pritaikytas XOM akcijai. Atlikus portfelio simuliaciją per 2019–2025 m. laikotarpį, gauti šie rizikos ir efektyvumo rodikliai:

- Maksimalus nuosmukis (MDD): -39.81 %
- Pelningumo faktorius (Profit Factor): 0.92
- Laimėjimo dažnis (Win Rate): 43.04 % (34 sėkmingi sandoriai iš 79 aktyvių dienų)

## Rezultatų interpretacija

Reti, bet pelningi sandoriai: GNN modelis prekiauja labai retai (vos 79 kartai per 6 metus). Tai yra privalumas, nes taip sutaupoma daug pinigų komisiniams mokesčiams. Nors modelis „atspėja“ teisingą kryptį tik apie 43 % atvejų, jo sėkmingi sandoriai atneša daugiau pelno, nei prarandama per klaidingus sprendimus. Trumpai tariant – jis moka pagauti retus, bet didelius kainų šuolius ir nerizikuoja ramiu laikotarpiu. Maksimalus nuosmukis (MDD), pasiekęs -39.81 %, įvyko 2020 m. COVID-19 pandemijos šoko metu. Tačiau po šio įvykio GNN pademonstravo prisitaikymą: atpažinęs neprognozuojamą kintamumą, modelis ilgą laiką fiksavo „Stabilią“ klasę (generuodamas tiesią kapitalo kreivę), taip apsaugodamas likusį kapitalą nuo tolesnės destruktijos.

Kapitalo apsauga per krizes: Didžiausias modelio patirtas nuostolis buvo beveik -40 %. Tai įvyko dėl staigaus COVID-19 pandemijos šoko, kurio algoritmas negalėjo numatyti. Tačiau svarbiausia yra tai, kaip modelis reagavo: prasidėjus chaosui rinkoje, jis tiesiog nustojo prekiauti (perėjo į „rizikos ribojimo“ režimą). Taip jis apsaugojo pinigus nuo dar didesnių nuostolių, kuriuos dažnai patiria panikuojantys investuotojai.

Perėjimas prie viso portfelio valdymo: Kadangi šis modelis įrodė, jog moka analizuoti rinką, nereaguoti į beprasmius svyravimus ir išlaukti geriausių progų, toliau (3.3 skyriuje) pereinama prie sudėtingesnės užduoties. Modelis bus naudojamas valdyti nebe pavienes akcijas, o visą investicinį portfelį. Bus bandoma išsiaiškinti, kaip efektyviai jis gali paskirstyti pinigus tarp skirtingų investicijų, siekiant dar didesnio pelno su kuo mažesne rizika.

### 3.3. Dinaminio investicinio portfelio formavimas ir optimizavimas

Iki šiol tyrime mašininio mokymosi modelių prognozių tikslumas ir rizikos metrikos buvo vertinamos atskirų aktyvų lygmeniu. Tačiau praktiniame investavime pavienių akcijų prekyba yra retai taikoma dėl rizikos. Tikroji grafų neuroninių tinklų (GNN), vertinančių visos rinkos tarpusavio ryšius, vertė atsiskleidžia optimizuojant viso portfelio sudėtį. Siekiant įvertinti strategijos efektyvumą makro lygmeniu, šiame skyriuje atliekama dinaminio 15-os akcijų portfelio simuliacija, naudojant ankstesniuose etapuose nustatytus optimalius GNN architektūros ir laiko dinamikos parametrus ( $W = 30$  dienų,  $S = 5$  dienos, 16 paslėptųjų neuronų).

#### 3.3.1. Portfelio simuliacija taikant GNN signalus (1/N ir Markowitz strategijos)

Pirmajame portfelio testavimo etape taikoma naiviosios diversifikacijos, arba lygių svorių (angl. *Equal Weighting*, 1/N), kapitalo paskirstymo metodika. Portfelio formavimo algoritmas veikia pagal šias taisykles:

1. Kiekvieno slenkančio lango žingsnio ( $S = 5$  dienos) pradžioje GNN modelis, įvertinęs koreliacijų grafą, sugeneruoja kainos krypties prognozes visoms 15-ai krepšelio akcijų.
2. Portfelio kapitalas yra paskirstomas lygiomis dalimis (1/N) tik toms akcijoms, kurios gavo „Pirkimo“ (Kyla) signalą.
3. Akcijos, gavusios „Krenta“ arba „Stabili“ signalą, į portfelį neįtraukiamos.

4. Jei nei viena akcija negauna pirkimo signalo (modelis nemato patikimų tendencijų jokiame sektoriuje), visas portfelis perkeliamas į nerizikingą grynujų pinigų poziciją (0 % investicinis svoris akcijų rinkoje).

Atlikus testavimą 2019–2025 m. laikotarpyje su optimaliais GNN parametrais, gauti šie portfelio lygmens rezultatai:

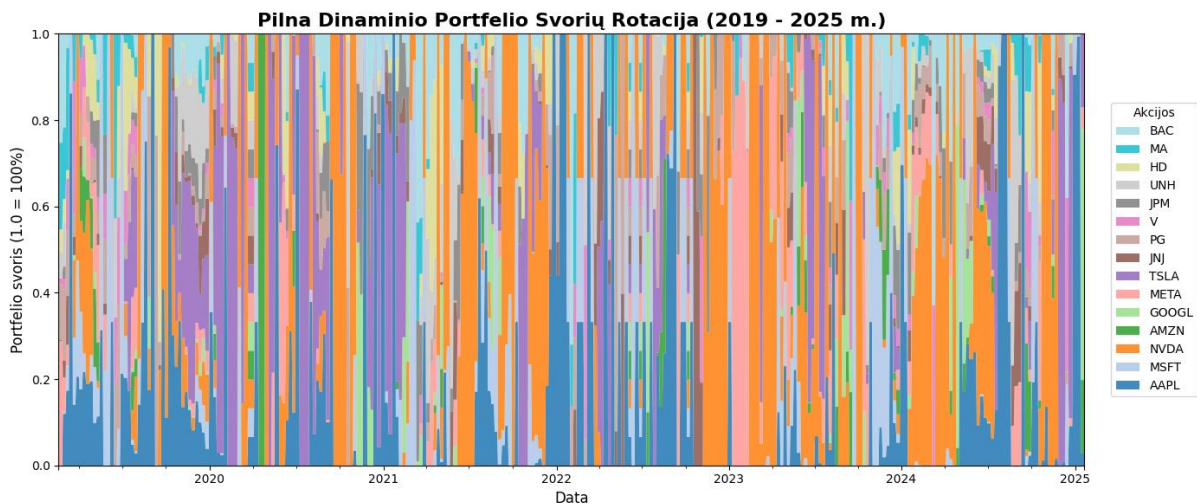
- Bendras kapitalo augimas: 135.87 %
- Portfelio Sharpe rodiklis: 1,15

Analizuojant GNN modelio kapitalo kreivę, išryškėja unikali strategijos elgsena – itin griežtas rizikos valdymas. Analizuojant rezultatus stebimi „lygūs“ periodai indikuoja laikotarpius, kai modelis, atpažinęs nepalankius ar triukšmingus rinkos režimus (angl. *regime shifts*), pilnai atitraukdavo kapitalą iš akcijų rinkos. Toks metodas užtikrino aukštą kapitalo apsaugą kriziniais momentais ir leido išlaikyti solidų rizikos įvertintos gražos rodiklį (Sharpe = 1,15).

### **3.3.2. Portfelio svorių optimizavimo metodų lyginamoji analizė (Markowitz prieš 1/N)**

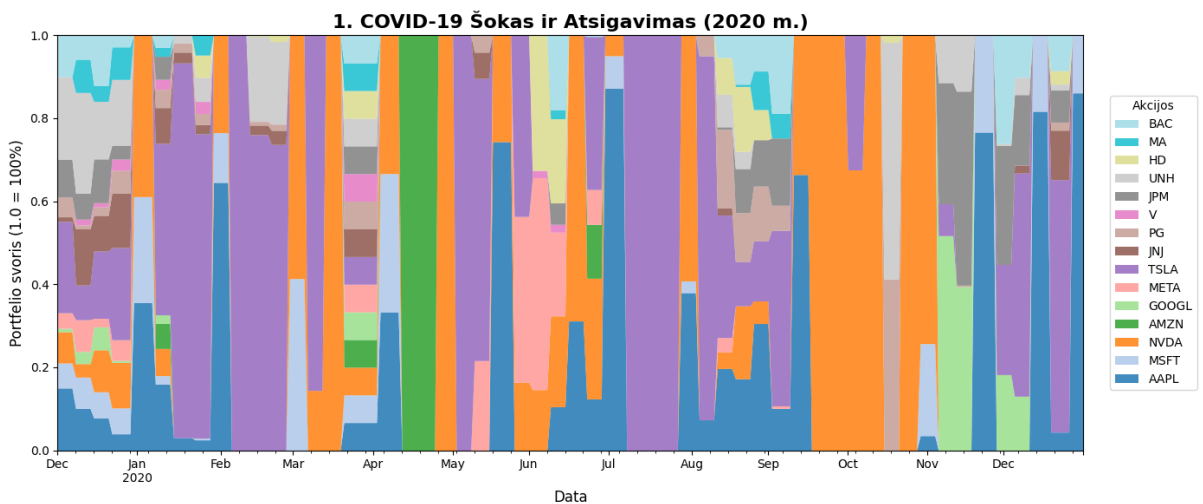
Ankstesniame poskyriuje aprašytas dinaminis GNN portfelis naudojo lygių svorių (1/N) paskirstymo metodiką, kuri fiksavo 1,15 Sharpe rodiklį. Tačiau, siekiant nustatyti, ar matematinis svorių optimizavimas gali dar labiau pagerinti rezultatus, GNN atrinktoms akcijoms buvo pritaikytas Markowitz vidurkio-dispersijos modelis. Palyginus abu metodus, nustatyta, kad Markowitz optimizacija šiame tyrime fiksavo prastesnį stabilumą (Sharpe rodiklis svyravo 0,77–0,84 intervale). Tai patvirtina literatūros įžvalgas: analizuojant trumpus laiko laikotarpius, sudėtingi matematiniai modeliai (pvz., Markowitz) tampa pernelyg jautrūs trumpalaikiams kainų svyravimams ir padaro klaidų skirstydami svorius. Dėl šios priežasties, paprastas kapitalo padalinimas akcijoms po lygiai (1/N) veikia kur kas stabiliau ir pelningiau.

Nepaisant žemesnio efektyvumo rodiklio, Markowitz optimizuoto GNN portfelio svorių pasiskirstymas laike puikiai iliustruoja GNN algoritmo adaptyvumą ir gebėjimą atpažinti rinkos režimus bei atlikti pilną kapitalo rotaciją.



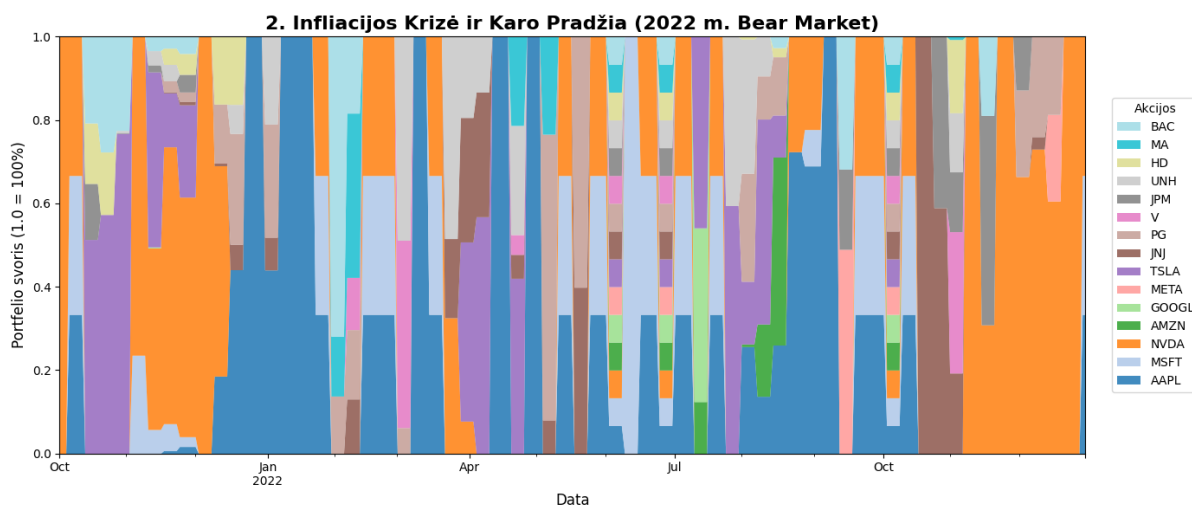
**5 pav.** Pilnas dinaminio portfelio svių pasiskirstymas taikant GNN ir Markowitz optimizavimą (2019–2025 m.).

Kaip matyti 5 paveiksle, algoritmas atlieka itin aktyvią sektorinę rotaciją ir maksimaliai išnaudoja visą investicinį kapitalą. Siekiant detaliau įvertinti modelio sprendimų logiką, žemiau pateikiama vizualinė analizė, išskaidyta per tris kritinius globalių rinkų lūžio taškus:



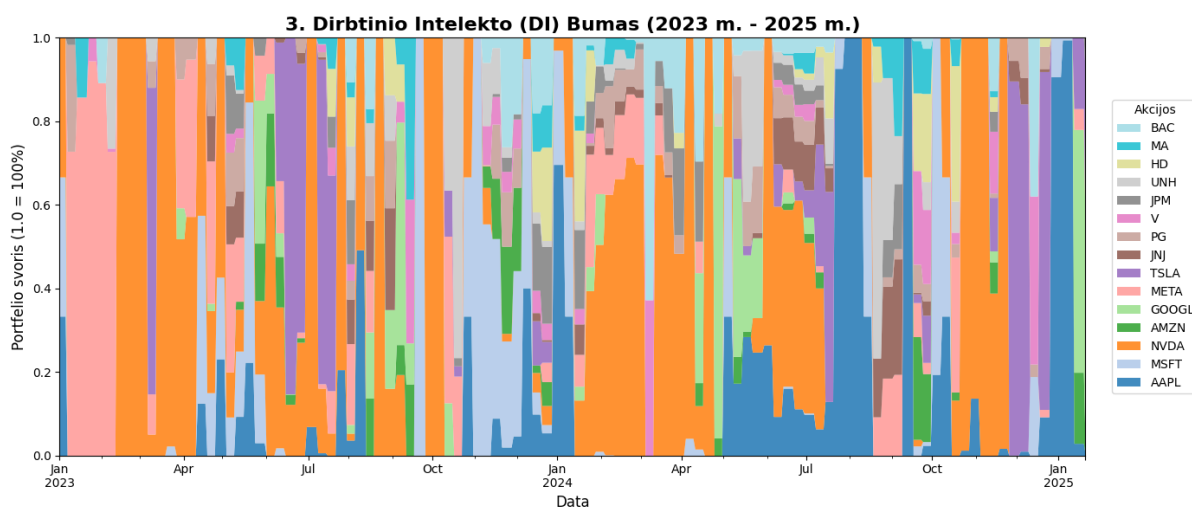
**6 pav.** GNN portfelio rotacija COVID-19 pandemijos metu

- 2020 m. pavasaris (COVID-19): Modelis operatyviai identifikavo krizės pradžią – portfelis buvo maksimaliai išskaidytas, o vėliau agresyviai sukonzentruotas į „Amazon“ (AMZN) bei technologijų gigantus (AAPL, MSFT), kurie geriausiai pritaikyti veikti izoliacijos sąlygomis (6 pav.).



7 pav. GNN portfelio rotacija 2022 m. krizės metu

- 2022 m. (Infliacijos šokas): Keičiantis rinkos režimui, GNN modelis atliko griežtą kapitalo rotaciją – atsisakė technologijų sektoriaus ir praktiškai 100 % portfelio perkėlė į apsaugines (angl. *defensive*) pozicijas: „Procter & Gamble“ (PG) ir „Johnson & Johnson“ (JNJ). Šis sprendimas leido išvengti technologinių akcijų nuosmukio (7 pav.).



8 pav. GNN portfelio rotacija DI bumų laikotarpiu

- 2023–2025 m. (DI bumai): nors generatyvinio DI proveržis prasidėjo 2022 m. pabaigoje, finansų rinkose realus augimo ciklas užfiksuotas 2023 m. pirmajame ketvirtyje. Paskutiniame etape modelis fiksavo šį technologijų sektoriaus pakilimą, portfelyje vėl pradeda dominuoti „Nvidia“ (NVDA) ir „Meta“ akcijoms (8 pav.).

Pilnas ir detalus kiekvienos pusmetinių pozicijų sąrašas, atspindintis šią rotaciją procentiniais svoriais, pateikiamas 3 Priede.

Nors šis svorių paskirstymo žemėlapis rodo modelio savarankiškumą pirkti kylančias akcijas, dėl anksčiau minėtos Markowitz „persimokymo“ rizikos galutiniam tyrimo vertinimui

(lyginant su S&P 500 indeksu tolesniame skyriuje) pasirenkama stabilesnė ir ekstremaliems šokams atsparesnė lygių svorių (1/N) metodika.

### 3.3.3. GNN portfelio lyginamoji analizė su rinkos atskaitos taškais

Siekiant visapusiškai įvertinti sugeneruoto GNN (1/N) portfelio efektyvumą kitam kontekste, atliktas išplėstinis lyginimas su dviem atskaitos taškais: vidiniu (15-os akcijų „Pirk ir laikyk“ strategija) ir išoriniu standartiniu JAV akcijų rinkos atskaitos tašku – S&P 500 indeksu (^GSPC). S&P 500 pasirinkimas leidžia objektyviai nustatyti, ar dirbtinio intelekto valdomas portfelis sugeba sugeneruoti teigiamą rizikos ir gražos santykį lyginant su plačiąja rinka.

Atlikta 2019–2025 m. laikotarpio simuliacija atskleidė esminius strategijų skirtumus (žr. 7 lentelę). Atlikta analizė atskleidė klasikiniam aktyviam portfelio valdymui būdingą absoliučiosios gražos ir rizikos kompromisą:

**7 lentelė.** GNN portfelio (1/N) ir S&P 500 indekso veiklos rezultatų vertinimas

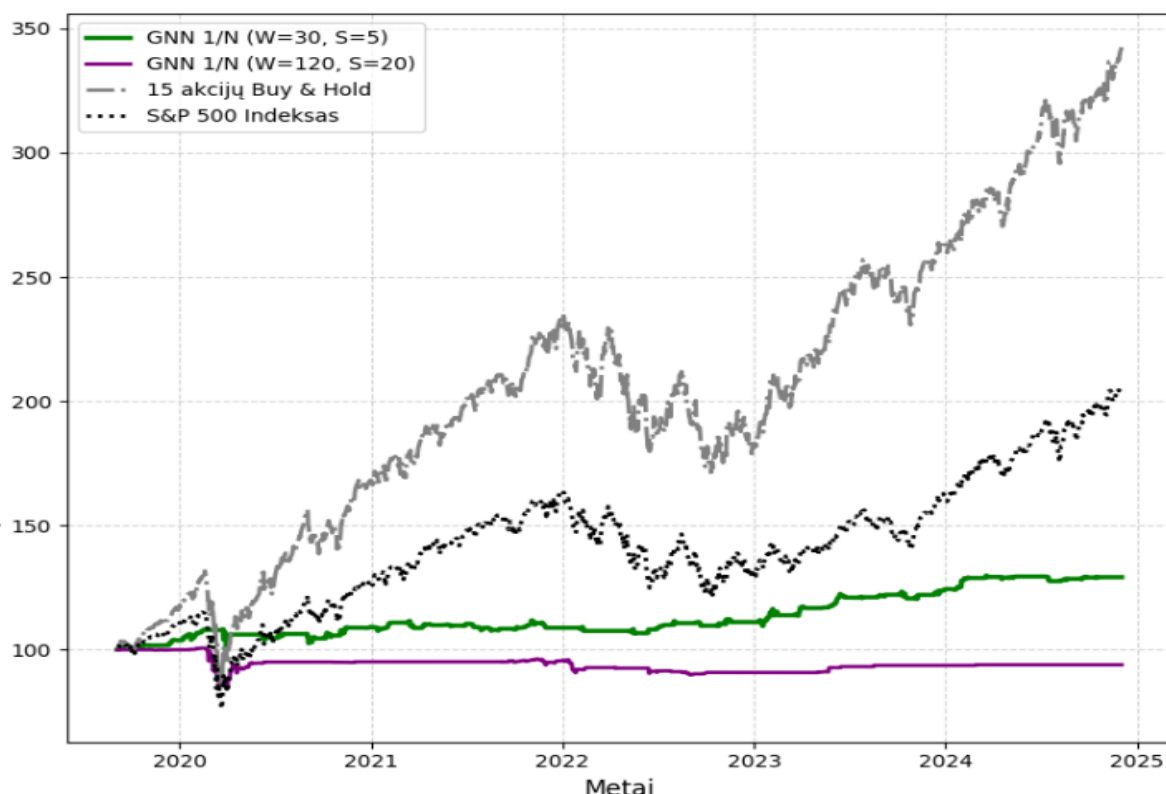
Vertinimo metrika	GNN portfelis (1/N)	S&P 500 (Benchmark)
Metinė graža (Pelnas)	7,70 %	14,34 %
Metinis kintamumas (Std. nuokrypis)	6,68 %	20,50 %
Metinė dispersija (Variance)	45	420
Sharpe rodiklis	1,15	0,70
Information Ratio (IR)	-0,31	—
Modigliani-Modigliani (M2)	23,62 %	14,34 %
Market Timing (Treynor-Mazuy Gamma)	-0,09 (p=0,615)	—

#### Gautų rezultatų interpretacija:

- Pelnas, kintamumas ir Sharpe rodiklis: Tyrimo laikotarpiu (2019–2025 m.) S&P 500 indeksas pasižymėjo stipriu augimo trendu, sugeneruodamas 14,34 % metinę gražą. GNN portfelis pagal absoliučią gražą atsiliko (7,70 %). Tačiau analizuojant rizikos parametrus, matomas radikalus skirtumas: GNN modelio metinis kintamumas (6,68 %) ir dispersija (0,0045) buvo daugiau nei trigubai mažesni už S&P 500 rinkos riziką (kintamumas 20,50 %, dispersija 0,0420). Dėl šios efektyvos kapitalo apsaugos, GNN portfelio Sharpe rodiklis (1,15) reikšmingai aplenkė S&P 500 indeksą (0,70). Tai įrodo, kad modelis generuoja kur kas kokybiškesnę, pagal riziką įvertintą gražą.
- Modigliani-Modigliani (M2) matas: Kadangi Sharpe rodiklis išreiškiamas santykinio dydžiu, praktiniam interpretavimui pritaikytas M2 rodiklis. Jis parodo, kokia būtų GNN portfelio graža, jeigu jo rizika būtų sulyginta su S&P 500 rizika (naudojant svertą). GNN modelio M2 reikšmė siekia net 23,62 %, kas matematiškai įrodo, jog esant vienodam rizikos lygmeniui, dirbtiniu intelektu valdomas portfelis reikšmingai (daugiau nei 9 procentiniais punktais) aplenkė pasyvų investavimą į indeksą (14,34 %).

- Information Ratio (Informacijos rodiklis): Šis matas įvertina aktyvią grąžą (lyginant su S&P 500) padalintą iš sekimo paklaidos (angl. *Tracking error*). Gautas neigiamas rodiklis (-0,31) logiškai atspindi faktą, kad GNN portfelis absoliučia grąža neaplenkė S&P 500 indekso. Tai būdinga itin gynybinėms strategijoms ilgų „bulių rinkos“ (angl. *bull market*) ciklų metu, kai modelis dažnai atsitraukia į grynuosius pinigus, taip aukodamas dalį pelno dėl maksimalaus saugumo.
- Market Timing (Treynor-Mazuy testas): Siekiant statistiškai įvertinti modelio gebėjimą nuspėti rinkos kryptį (angl. *market timing*), apskaičiuotas Treynor-Mazuy Gamma koeficientas. Nors koeficientas yra nežymiai neigiamas (-0,09), jo p-reikšmė (0,615) yra gerokai didesnė už 0,05 reikšmingumo lygmenį. Tai reiškia, kad rezultatas nėra statistiškai reikšmingas. Galima daryti išvadą, kad GNN modelio pranašumas (aukštas Sharpe ir M2) kyla ne iš makroekonominio rinkos laiko nuspėjimo, o iš sėkmingos, nuolat besikeičiančios vidinės sektorių ir akcijų koreliacijų rotacijos, kurią grafuly neuroniniai tinklai užfiksuoja trumpuosiuose 30 dienų laiko eilutėse.

Be to, eksperimentas su slenkančio lango ilgiu patvirtino tyrimo hipotezę: ilgos atminties modelis ( $W=120$ ) prasčiau prisitaiko prie dinamiškų rinkos pokyčių ir jo efektyvumas krito žemiau rinkos vidurkio (Sharpe sumažėjo iki 0,65). Tai įrodo, kad GNN modelio sėkmę lemia operatyvus 30 dienų akcijų tarpusavio koreliacijų grafuly atnaujinimas, leidžiantis modeliui išlikti greitai prisitaikančiu.

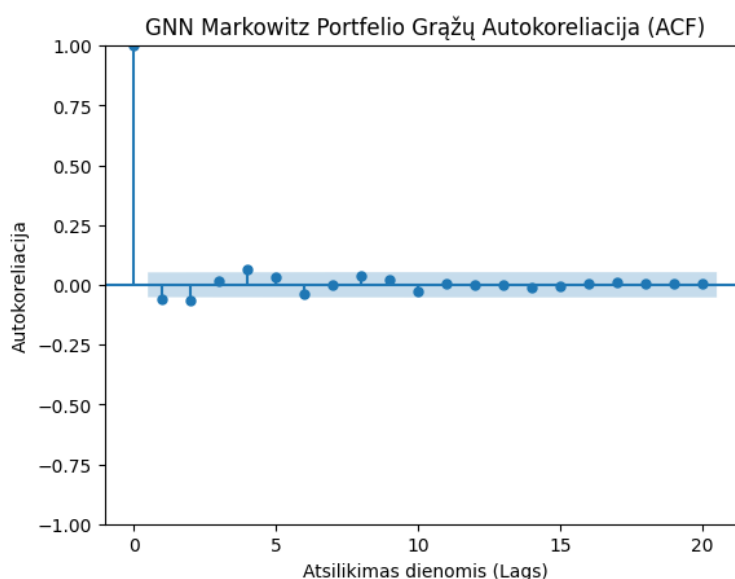


**9 pav.** Skirtingų parametų ( $W$  ir  $S$ ) GNN portfelių ir tradicinių investavimo strategijų veiklos rezultatų palyginimas

### 3.3.4. Portfelio gražų bei rizikos metrikų stabilumo ir autokoreliacijos (ACF) vertinimas

Siekiant patikrinti, ar lygių svorių (1/N) GNN portfelio uždirbama graža yra dėsninga, o ne atsitiktinė, buvo atliktas papildomas patikrinimas – autokoreliacijos (ACF) testas. Finansų srityje šis testas padeda nustatyti, ar sistemos pelnas turi „atmintį“ (t. y. ar šiandienos sėkmė tiesiogiai priklauso nuo vakarykštės), ar kiekvienos dienos rezultatai yra nepriklausomi. Jei gražos būtų stipriai susijusios tarpusavyje, tai reikštų, kad modelis tiesiog naudojasi laikina rinkos inercija (angl. *momentum*), kuriai pasibaigus sistema taptų nuostolinga.

Atlikus 20 darbo dienų (vieno prekybos mėnesio) analizę, gauti rezultatai patvirtino aukštą strategijos kokybę. 10 paveiksle matyti, kad tik pirmos dienos rodiklis viršija 95% patikimumo ribą (pažymėtos mėlyna zona). Tai įrodo, kad modelio sugeneruoti rezultatai yra statistiškai pagrįsti ir nepriklauso nuo atsitiktinių svyravimų.

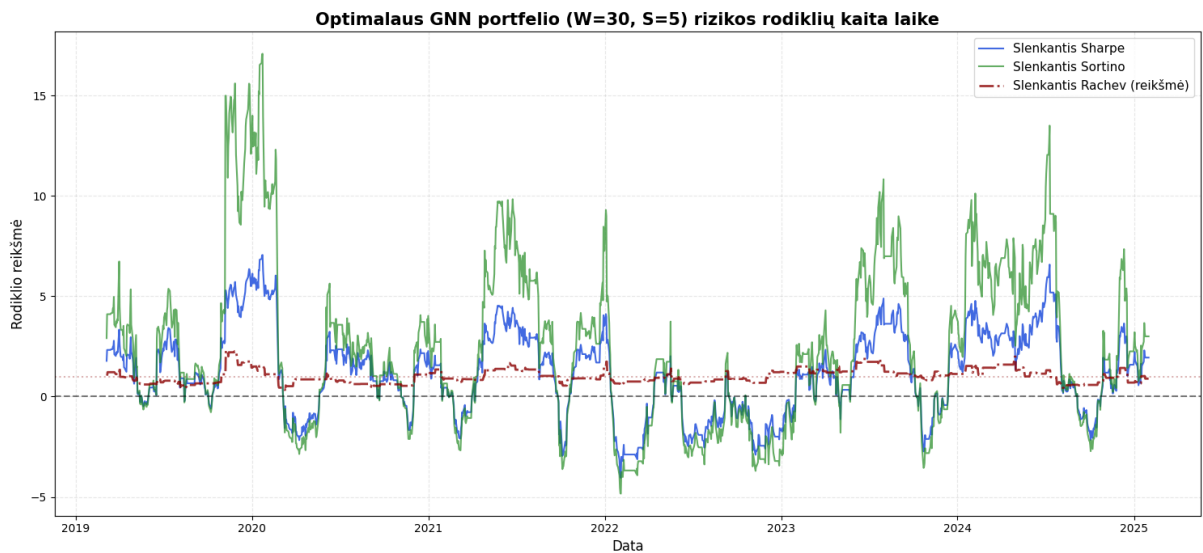


10 pav. GNN Markowitz portfelio gražų autokoreliacija (ACF)

Kadangi GNN modelis portfelį atnaujinama kas 5 dienas, papildomai buvo patikrinta, ar neegzistuoja ryšys tarp atskirų 5 dienų laikotarpių rezultatų. Tyrimas patvirtino pirminę išvadą – statistiškai reikšmingos „atminties“ tarp modelio sprendimų ciklų nėra. Tai įrodo, kad portfelio vertė auga ne dėl paprastos inercijos, o dėl to, kad modelis kiekviename etape iš naujo tiksliai įvertina esminius ryšius tarp aktyvų.

Nors pelno autokoreliacija parodo, ar graža turi „atmintį“, norint visiškai įsitikinti strategijos stabilumu, būtina iširti ir kitus rodiklius. Reikia panagrinėti, kaip laikui bėgant keičiasi pagal kitos gražos metrikos (Sharpe, Sortino ir Rachev rodikliai) bei patikrinti jų stabilumą ilguoju laikotarpiu.

Pirmiausia buvo išanalizuota, kaip geriausios GNN modelio konfigūracijos ( $W=30$ ,  $S=5$ ) rodikliai kito per slenkantį 60 dienų laikotarpį

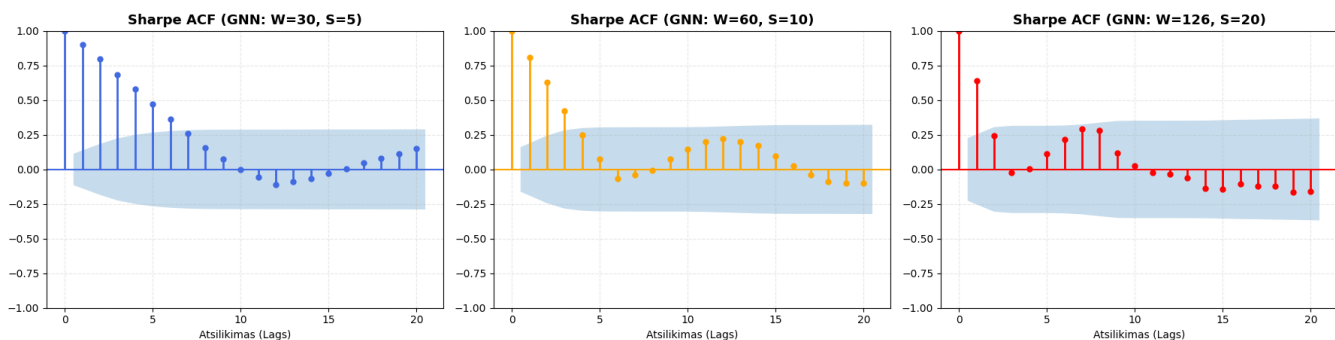


**11 pav.** Optimalaus GNN portfelio rizikos rodiklių kaita laike

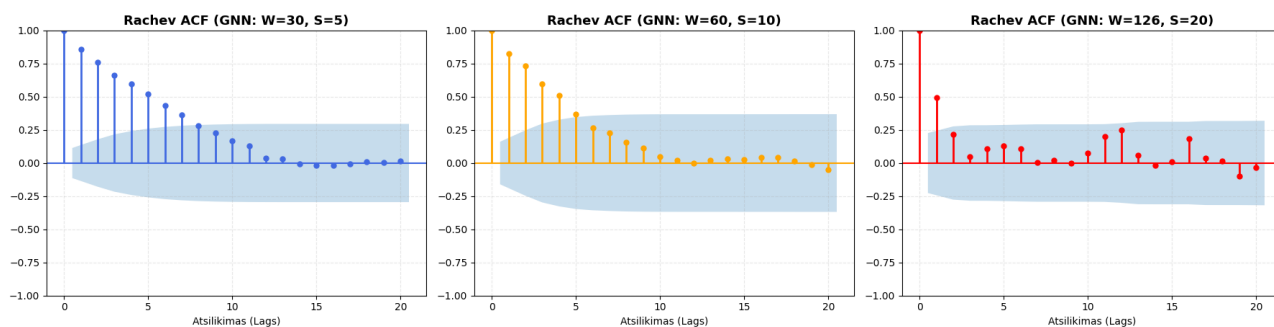
Kaip rodo 11 paveikslas, strategijos rezultatai laikui bėgant keičiasi. Svarbu tai, kad didžiąją tyrimo dalį abu pagrindiniai rodikliai (Sharpe ir Sortino) išlieka teigiami. Ypač išsiskiria Sortino rodiklis: didelio rinkos neramumo periodais (pvz., 2020 m. pavasarį ar 2024 m. pradžioje) jis smarkiai pakyla ir viršija Sharpe rodiklį. Tai įrodo, kad portfelio svyravimai dažniausiai yra palankūs (nešantys pelną), o didelių nuostolių pavyksta sėkmingai išvengti. Šią išvadą patvirtina ir Rachev rodiklis (punktyrinė linija), kuris dažniausiai yra didesnis už 1,0. Tai reiškia, kad modelio uždirbamas pelnas yra gerokai didesnis už patiriamus nuostolius.

Norint patikrinti, kaip šie rezultatai priklauso nuo modelio nustatymų, buvo atlikta papildoma analizė. Pasitelkus Sharpe ir Rachev rodiklius, buvo palygintos trys skirtingos GNN modelio konfigūracijos: optimali ( $W=30, S=5$ ), vidutinė ( $W=60, S=10$ ) ir ilgoji ( $W=126, S=20$ ).

Sharpe rodiklio autokoreliacija keičiant modelio apmokymo ir slinkimo langus



**12 pav.** Sharpe rodiklio autokoreliacija keičiant modelio apmokymo ir slinkimo langus

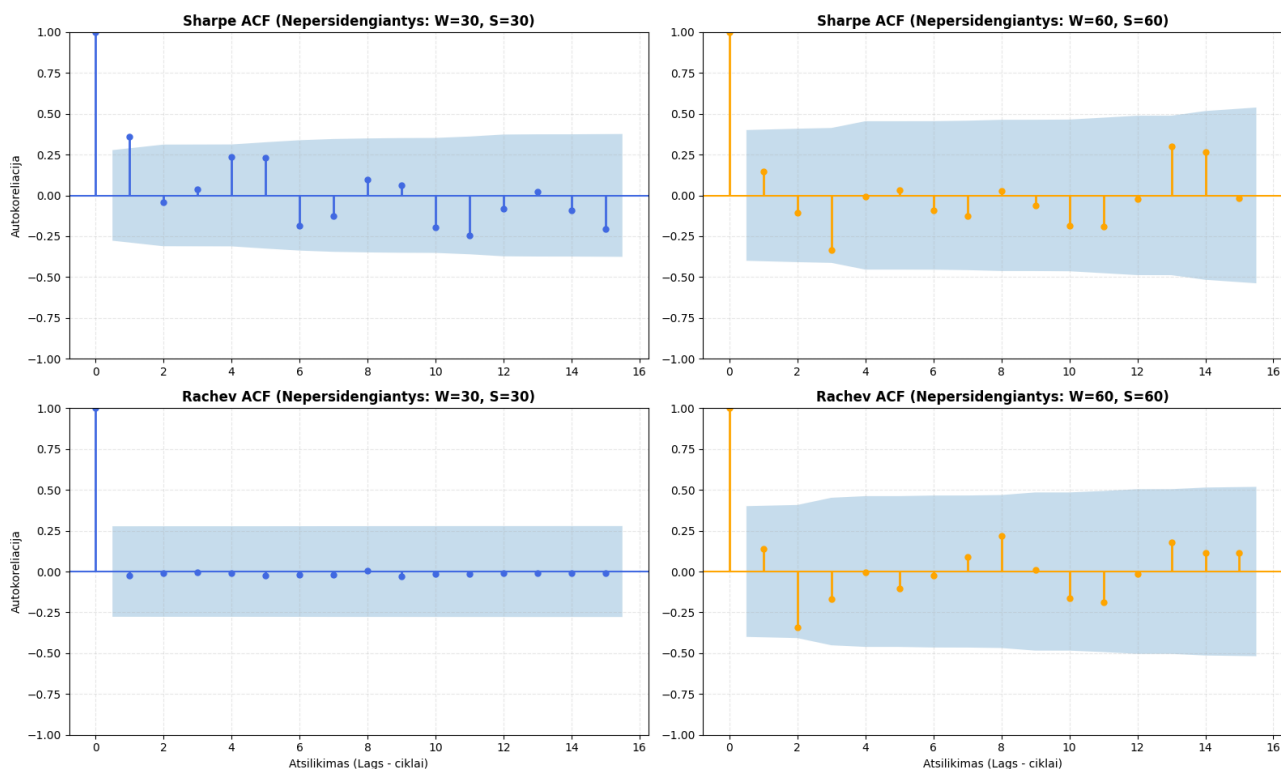


**13 pav.** Rachev rodiklio autokoreliacija keičiant modelio parametrus

12 ir 13 Paveiksluose matomi dėsningumai patvirtina, kodėl pasirinkti būtent tokie modelio nustatymai. Kairiuosiuose grafikuose ( $W=30$ ,  $S=5$ ) matome, kad rodikliai kinta tolygiai ir natūraliai. Tai rodo, kad dažnai atnaujinamas GNN modelis sėkmingai atpažįsta trumpalaikius pelningus rinkos pokyčius ir išlaiko stabilų efektyvumą apie mėnesį (7–8 žingsnius). Tuo tarpu dešiniuosiuose grafikuose ( $W=126$ ,  $S=20$ ), kur naudojama sena istorinė informacija, vaizdas pasikeičia – rodikliai pradeda drastiškai šokinėti. Taip nutinka dėl to, kad naudojant ilgą laikotarpį modelis „įsimena“ pasenusius akcijų ryšius, kurie dabartinėje rinkoje nebegalioja. Dėl to strategijos sėkmė tampa atsitiktinė ir nestabili. Ši analizė galutinai įrodo, kad tik mažas duomenų langas ir dažnas jų atnaujinimas leidžia užtikrinti stabilią portfelio kokybę.

Kilo įtarimas, kad naudojant persidengiančius duomenų langus modelis tiesiog sukuria „dirbtinę atmintį“ – jis lengvai prisitaiko prie esamos rinkos būsenos, užuot iš tikrųjų išmokęs atpažinti svarbius rinkos pokyčius. Norint tai patikrinti, buvo atliktas papildomas tyrimas. Šį kartą GNN modelis buvo mokomas naudojant visiškai nepersidengiančius duomenų langus ( $W=30$ ,  $S=30$  ir  $W=60$ ,  $S=60$ ). Taip užtikrina, kad kiekviename mokymosi etape modelis gautų tik visiškai naujus, su praeitimi nesusijusius duomenis. Toks metodas leidžia išvengti minėto atminties efekto ir užtikrina, kad gauti rezultatai labiau pagrįsti matematiniais dėsniais.

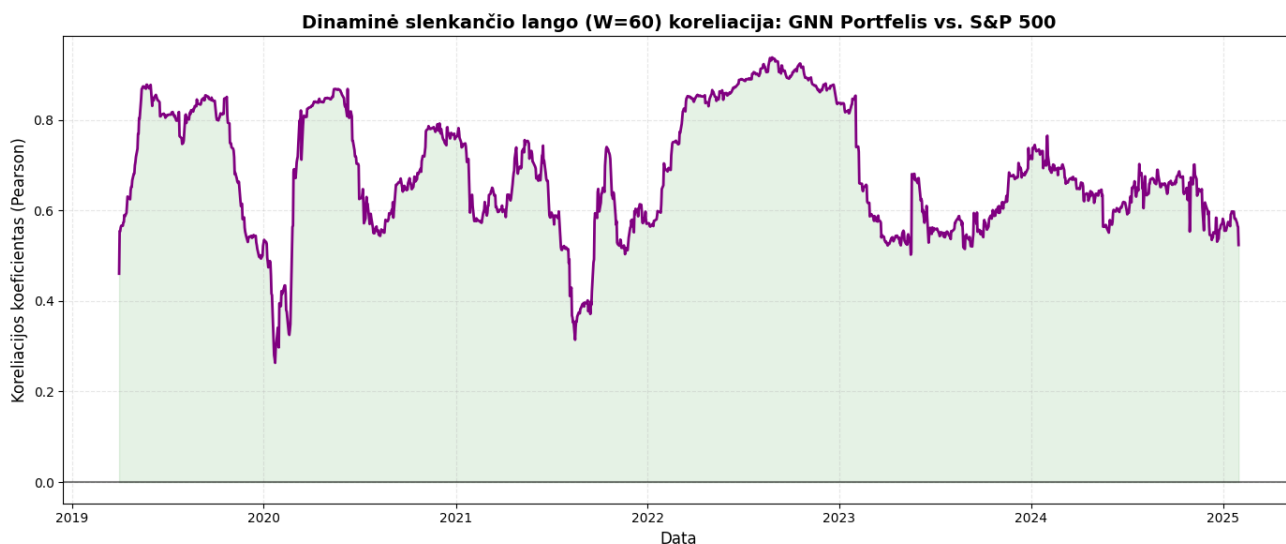
### Rizikos rodiklių autokoreliacija pašalinus langų persidengimą ( $W = S$ )



**14 pav.** Rizikos rodiklių (Sharpe ir Rachev) autokoreliacija taikant visiškai nepersidengiančius apmokymo langus.

Kaip matyti iš 14 paveikslėlio, visiškai pašalinus langų persidengimą, autokoreliacija (ACF) sumažėja ir beveik išnyksta (Rachev rodiklio atveju jau nuo pirmojo atsilikimo žingsnio). Visi stulpeliai krenta į statistinio nereikšmingumo (mėlyną) zoną. Tai patvirtina kelias svarbias išvadas. Pirmą, didelė autokoreliacija ankstesniuose eksperimentuose (kur  $S \ll W$ ) iš tiesų buvo iš dalies nulemta persidengiančių duomenų judėjimu. Antra, nepersidengiančių langų atveju modelis praranda akcijų judėjimo sekimą ir yra priverstas kiekviename žingsnyje mokytis iš naujo. Nors tai vizualiai sumažina strategijos tolygumą, šis eksperimentas įrodo, kad GNN algoritmas geba daryti nepriklausomus sprendimus ir generalizuoti informaciją remiantis tikrąja rinkos struktūra, o ne vien istoriniu duomenų persidengimu.

Galiausiai, norint suprasti, kaip GNN modelis reaguoja į bendrą ekonominę situaciją, buvo apskaičiuotas jo ryšys (koreliacija) su „S&P 500“ indeksu kintant laikui.



**15 pav.** Dinaminė slenkančio lango koreliacija GNN prieš S&P 500

15 Paveiksle matyti, kad per visą tyrimo laikotarpį ryšys (koreliacija) tarp portfelio ir rinkos išlieka teigiamas, tačiau nevienodas. Dažniausiai šis ryšys svyruoja nuo 0,5 iki 0,9. Didžiausia priklausomybė nuo bendros rinkos užfiksuota 2019 m., 2020 m. pabaigoje ir ypač 2022 m. Šiais etapais GNN portfelis judėjo labai panašiai kaip ir visa rinka. Tačiau grafike matomi ir ryškūs nuosmukiai (pvz., 2020 m. ir 2023 m. pradžioje), kai portfelio rezultatai tapo mažiau priklausomi nuo bendrų rinkos tendencijų.

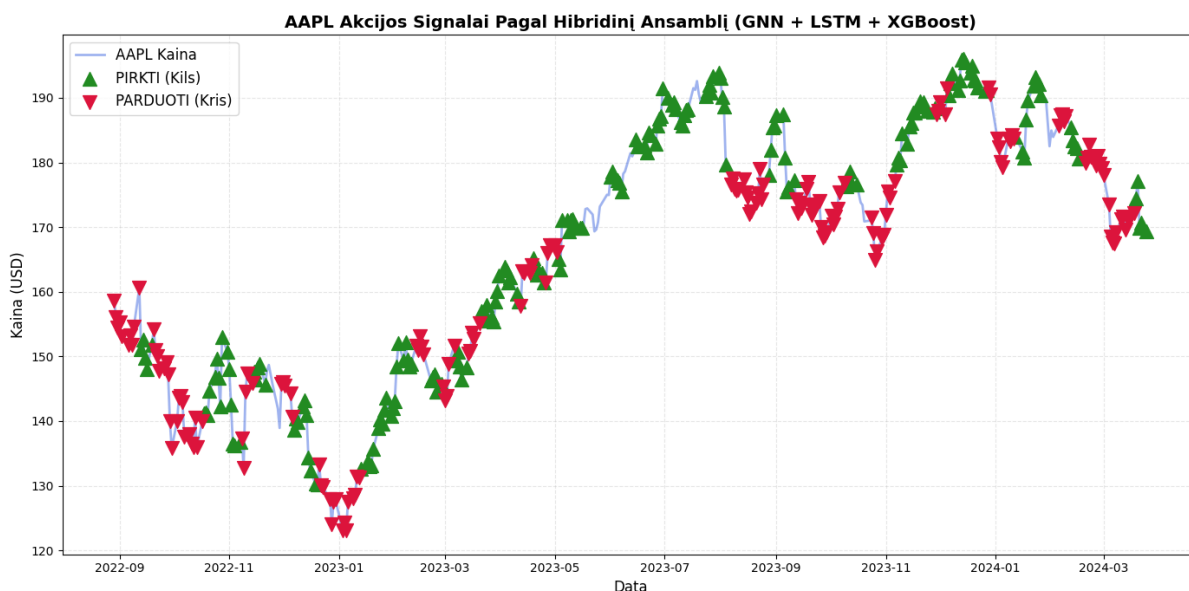
Apibendrinant, šis grafikas rodo, kad GNN portfelis iš dalies seka bendrą rinkos kryptį, tačiau modelis geba laiku prisitaikyti ir veikti savarankiškai. Tai leidžia strategijai sukurti grąžą, kuri ne visada priklauso tik nuo akcijų indekso svyravimų.

### **3.4. Ansamblinių modelių taikymas ir realaus laiko prognozė akcijų kainoms**

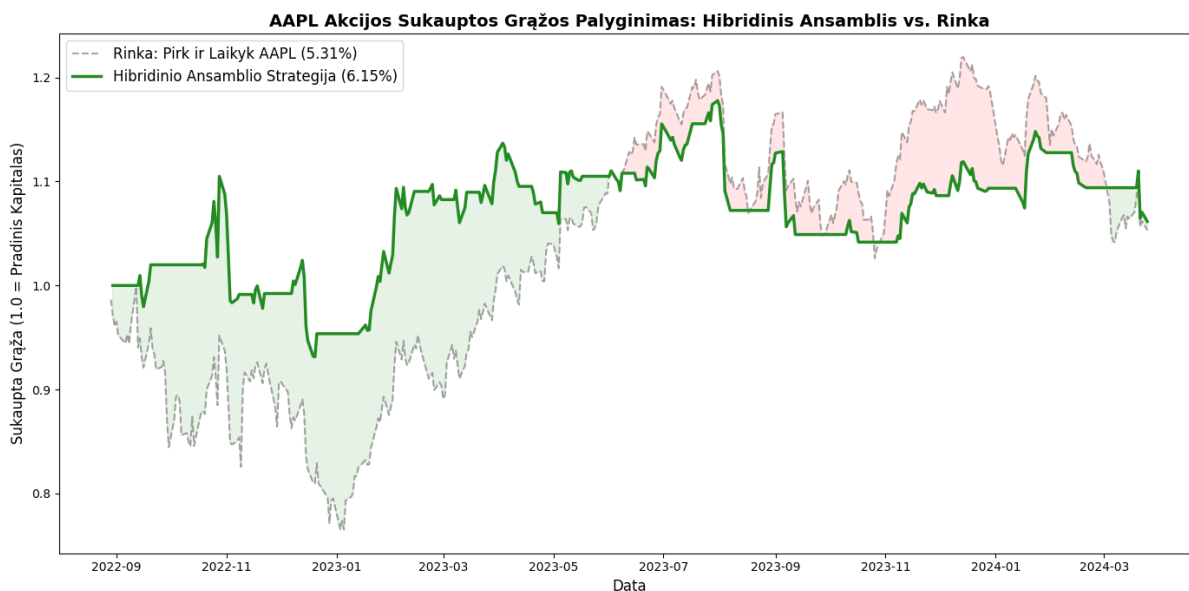
Nors ankstesniuose skyriuose parodyta, kad grafų neuroniniai tinklai (GNN) geriausiai įvertina tarpusavio ryšius tarp rinkos dalyvių, praktikoje skirtingi algoritmai turi skirtingų privalumų. Dėl šios priežasties paskutiniame tyrimo etape buvo sudarytas modelių ansamblis, kuris apjungia trijų metodų stipriąsias puses: XGBoost gebėjimą atpažinti netiesinius ryšius techniniuose indikatoriuose, LSTM gebėjimą analizuoti laiko eilučių priklausomybes ir GNN gebėjimą įvertinti ryšius tarp akcijų.

Ansamblyje taikytas minkštojo balsavimo metodas (angl. *Soft Voting*). Šiuo atveju galutinis sprendimas priimamas ne pagal vieno modelio rezultatą, o pagal visų trijų modelių prognozuojamas tikimybes. Kiekvienas modelis pateikia tikimybes trimis klasėmis: „Kilti“, „Kristi“ ir „Stabili“. Tuomet šios tikimybės suvidurkinamos taikant svorius. Toks metodas leidžia sumažinti riziką, kad galutinį sprendimą nulems vieno modelio klaida arba pernelyg stipri reakcija į rinkos triukšmą.

Prieš taikant šią hibridinę sistemą realaus laiko prognozėms, buvo atliktas istorinių duomenų atgalinis testavimas (angl. *Backtesting*). Testavimui pasirinkta „Apple“ (AAPL) akcija. Šiame etape naudoti ankstesniame skyriuje nustatyti optimalūs hiperparametrai, t. y. 60 ir 30 dienų apmokymo langai.



16 pav. AAPL akcijos signalai pagal ansamblinį modelį (GNN + LSTM + XGBoost)



17 pav. AAPL akcijos sukauptos grąžos palyginimas: ansamblinis modelis prieš rinką

Kaip matyti paveiksluose, testavimo rezultatai rodo, kad hibridinė sistema ir optimizuoti parametrai pagerino strategijos rezultatus. Analizuojamu laikotarpiu pasyvi „Pirk ir laikyk“ strategija su AAPL akcija sugeneravo 5,31 % grąžą, o hibridinio ansamblio strategija pasiekė 6,15 % grąžą.

Signalų analizė parodė, kad taikant optimalius apmokymo langus ansamblis gana tiksliai filtruoja rinkos triukšmą. Modelis ne tik seka kainos augimo tendenciją, bet ir atpažįsta neapibrėžtumo laikotarpius. Tokiais atvejais jis pasirenka „Stabili“ klasę ir pereina į grynųjų pinigų poziciją. Tai leidžia išvengti dalies staigių AAPL akcijos kainos kritimų. Paveiksle tai matyti iš horizontalių strategijos linijų tuo metu, kai rinka krenta. Šis rezultatas rodo, kad tinkamai sukalibruotas modelių ansamblis gali būti naudingas ne tik portfelio rizikai valdyti, bet ir siekiant teigiamos grąžos prekiaujant atskiromis akcijomis didelio rinkos kintamumo sąlygomis.

Patikrinus, kad sistema veikia istoriniuose duomenyse ir gali generuoti teigiamą grąžą, ansamblio modelis buvo pritaikytas realaus laiko prognozei (angl. *Live Prediction*). Tam naudoti naujausi rinkos duomenys ir atrinktas 15 akcijų krepšelis. Toliau pateikiami suvidurkinti ansamblio sprendimai artimiausiai prekybos dienai.

**8 lentelė.** Atskirų akcijų kainos judėjimo krypties tikimybės ir ansamblio sprendimai

Akcija (Ticker)	Sektorius	Tikimybė kristi (%)	Tikimybė išlikti stabiliai (%)	Tikimybė kilti (%)	Ansamblio sprendimas
GOOGL	Technologijų	17,0 %	30,0 %	53,0 %	PIRKTI (Kils)
META	Technologijų	17,0 %	32,0 %	51,0 %	PIRKTI (Kils)
XOM	Energetikos	26,0 %	29,0 %	45,0 %	PIRKTI (Kils)
AMZN	Vartojimo prekių	17,0 %	40,0 %	43,0 %	PIRKTI (Kils)
NVDA	Technologijų	24,0 %	33,0 %	42,0 %	PIRKTI (Kils)
PG	Vartojimo prekių	18,0 %	40,0 %	42,0 %	PIRKTI (Kils)
AAPL	Technologijų	17,0 %*	61,0 %	22,0 %	LAIKYTI (Stabili)
JNJ	Sveikatos aps.	20,0 %	57,0 %	23,0 %*	LAIKYTI (Stabili)
MSFT	Technologijų	~ 25,0 %*	~ 50,0 %	~ 25,0 %*	LAIKYTI (Stabili)
PFE	Sveikatos aps.	~ 25,0 %*	~ 50,0 %	~ 25,0 %*	LAIKYTI (Stabili)
TSLA	Vartojimo prekių	~ 30,0 %*	~ 40,0 %	~ 30,0 %*	LAIKYTI (Stabili)
BA	Pramonės	~ 30,0 %*	~ 40,0 %	~ 30,0 %*	LAIKYTI (Stabili)
V	Finansų	~ 30,0 %*	~ 40,0 %	~ 30,0 %*	LAIKYTI (Stabili)
KO	Vartojimo prekių	47,0 %	29,0 %	23,0 %	PARDUOTI (Krenta)
JPM	Finansų	45,0 %	29,0 %	26,0 %	PARDUOTI (Krenta)

Ansamblinio modelio rezultatai rodo, kad jis geba protingai vertinti situaciją ir prisitaikyti prie rinkos pokyčių. Modelis siūlo pirkti technologijų (GOOGL, META, NVDA) ir energetikos (XOM) įmonių akcijas, nes visi algoritmai mato didesnę nei 42 % kainos augimo tikimybę ir sėkmingai atpažįsta jų kilimo pagreitį. Kita vertus, finansų (JPM) ir vartojimo prekių (KO) sektoriuose modelis pastebi prastas tendencijas. Kadangi šioms akcijoms prognozuojama atitinkamai 45 % ir 47 % kritimo rizika, algoritmas aiškiai pataria jų vengti arba jas parduoti.

Labai svarbu tai, kad neaiškiomis sąlygomis modelis elgiasi atsargiai ir nerizikuoja be reikalo. Daugumai analizuojamų akcijų (pavyzdžiui, AAPL, JNJ, MSFT, PFE, TSLA, BA, V) jis priskiria signalą „Laikyti“, nes mato didelę (40–61 %) tikimybę, kad kainos išliks stabilios. Kai rinkoje nėra aiškios krypties, algoritmas tiesiog ignoruoja nereikšmingus svyravimus ir laikinai nustoja prekiauti (pereina į grynujų pinigų poziciją). Toks gynybinis elgesys apsaugo

kapitalą nuo netikėtų rinkos šuolių ir istoriškai netgi padėjo uždirbti daugiau (pavyzdžiui, AAPL akcijos grąža padidėjo nuo 5,31 % iki 6,15 %). Tai įrodo, kad geras dirbtinio intelekto modelis ne tik akiai siekia pelno, bet ir pirmenybę teikia saugiam pinigų valdymui neramiose rinkose.

Ši demonstracija parodo, kad apjungus Grafų neuroninių tinklų erdvės suvokimą su patikima lygių svorių ( $1/N$ ) kapitalo alokacija ir ansambliniu balsavimu, sukurta pilnavertė sistema, kuri pati geba prisitaikyti prie rinkos tendencijų ir nuspėti, kaip ji pakis.

### 3.5. Tyrimo rezultatų lyginamoji analizė

Šiame poskyryje yra palyginami šio projekto empirinių tyrimų rezultatai su kitais moksliniais tyrimais apie mašininio mokymosi naudojimą finansuose.

Šis tyrimas parodė, kad GNN modelis, mokydamasis iš trumpo slenkančio lango ( $W = 30$ ), geriau valdo riziką nei standartiniai algoritmai (LSTM ar XGBoost). Taip yra todėl, kad GNN geba matyti ryšius tarp skirtingų įmonių, o ne vertina kiekvieną akciją atskirai. Tokius pačius rezultatus savo tyrime gavo ir [30]. Jie pritaikė GNN modelius akcijų rinkoms naudodami slenkančio lango (angl. *rolling window*) metodiką ir įrodė, kad modeliai, kurie mato tarpusavio įmonių ryšius, sugeneruoja kur kas didesnę grąžą bei geresnę Sharpe rodiklį nei tradiciniai LSTM tinklai. Be to, tyrimai patvirtina, kad finansinėje rinkoje įmonių ryšiai labai greitai keičiasi, todėl geriausiai veikia trumpi ir dažnai atnaujinami mokymo periodai, leidžiantys algoritmui greitai prisitaikyti.

Testuojant investicinį portfelį paaiškėjo, kad paprastas pinigų padalinimas visoms akcijoms po lygiai ( $1/N$  strategija) veikia geriau nei sudėtingi optimizavimo metodai (pvz., Markowitz modelis). Mūsų  $1/N$  portfelis buvo net tris kartus saugesnis už bendrą S&P 500 indeksą. Šį reiškinį finansų moksle yra išsamiai įrodę [31] savo klasikiniame tyrime. Autoriai patvirtino, kad sudėtingi portfelių optimizavimo modeliai yra labai jautrūs prognozavimo klaidoms. Kadangi net ir geriausi mašininio mokymosi algoritmai kartais daro klaidų, todėl bandymas matematiškai apskaičiuoti idealius akcijų svorius gali atnešti daugiau žalos nei naudos.

Darbo pabaigoje trys skirtingi modeliai (GNN, LSTM ir XGBoost) buvo sujungti į vieną bendrą sistemą (ansamblį). Toks apjungimas leido tiksliau atspėti akcijų kryptis (pvz., AAPL akcijos pavyzdyje). Tai, kad skirtingų algoritmų apjungimas duoda geresnius rezultatus nei pavieniai modeliai, yra patvirtinta [32] tyrime. Jie analizavo S&P 500 akcijas naudodami giliojo mokymosi, sprendimų medžių bei atsitiktinių miškų algoritmus ir įrodė, kad skirtingos prigimties modeliai „atsveria“ vienas kito trūkumus. Vieno modelio padarytą klaidą ištaiso kitas, todėl bendras (ansamblinis) sprendimas grąžina didesnę grąžą.

## Išvados

Šiame baigiamajame magistro darbe buvo tiriama mašininio mokymosi algoritmų pritaikymo problematika akcijų kainų kryptių prognozavimui ir dinaminio investicinio portfelio valdymui. Išsikeltas tyrimo tikslas – atlikti lyginamąją analizę ir suformuoti realaus laiko investicinį portfelį, pritaikytą kintančioms rinkos sąlygoms – buvo pasiektas. Remiantis gautais teoriniais ir empiriniais rezultatais, gavome šias išvadas.

1. Išnagrinėjus akcijų gražos prognozavimo problematiką, finansinių laiko eilučių nestacionarumo suvaldymo metodus bei investicinių portfelių optimizavimo strategijas mokslinėje literatūroje, nustatyta, jog tradiciniai izoliuoti modeliai negeba efektyviai reaguoti į dinamiškus rinkos pokyčius. Identifikuota būtinybė integruoti grafų neuroninius tinklus (GNN) kartu su slenkančių langų metodika – tai leidžia matematiškai įvertinti aktyvų tarpusavio ryšius ir užtikrinti modelio adaptaciją prie besikeičiančios rinkos.
2. Parengus tyrimo metodologiją, apimančią slenkančio lango principo taikymą bei hibridinės ansamblinės architektūros sudarymą, nustatyta, jog modelių prognozavimo stabilumas tiesiogiai priklauso nuo itin tikslaus laiko dinamikos parametrų kalibravimo. Eksperimentiškai įrodyta, jog grafų tinklams didžiausią stabilumą užtikrina trumpi ir dažnai atnaujinami apmokymo langai (pvz.,  $W=30$ ,  $S=5$ ). Šio etapo praktinis reikšmingumas pasireiškia sukurta metodine sistema, kuri leidžia operatyviai identifikuoti besikeičiančias koreliacijas ir išvengti algoritmų persimokymo pasikeitus rinkos režimams.
3. Įvertinus skirtingų architektūrų prognozavimo efektyvumą mikro lygmeniu ir optimizavus slenkančio lango dinamiką, išryškėjo, jog GNN modelis pranašumu lenkia kitus tyrime lygintus algoritmus (LSTM, XGBoost) sisteminės rizikos valdymo kontekste. Nors pagal gražos ir kintamumo santykį GNN (Sharpe 0,36) minimaliai nusileido GRU (0,37), jis demonstravo aukščiausius Calmar (0,32) ir Rachev (1,13) rodiklius. Praktikoje šis modelis yra naudingas, nes statistiškai įrodyta (DM testo  $p=0,0155$ ), jog GNN veikia geriau už LSTM ir veiksmingiau apsaugo pinigus nuo didelių nuostolių.
4. Ištyrus geriausiai pasirodžiusio modelio signalų praktinį pritaikomumą makro lygmeniu bei ansamblinio modelio efektyvumą realaus laiko prognozėms, pastebėta, jog naiviosios diversifikacijos (1/N) strategija, valdoma GNN signalų, užtikrina didesnę stabilumą nei Markowitz optimizavimas (pastarojo Sharpe siekė 0,77–0,84). Dinaminis GNN (1/N) portfelis pasiekė 1,15 Sharpe rodiklį ir daugiau nei tris kartus sumažino riziką lyginant su S&P 500 indeksu. Modelių ansamblio sugeneruota 6,15 % graža AAPL akcijos pavyzdyje (prieš 5,31 % pasyvią gražą) patvirtina praktinę darbo naudą realaus laiko prekybos signalų generavimui.

Šis tyrimas turi kelis apribojimus. Pirmiausia, testuojant portfelius nebuvo įtraukti realūs prekybos mokesčiai, kurie dėl dažno akcijų pirkimo ir pardavimo realybėje sumažintų pelną. Antra, analizė apima tik 15 didžiausių JAV įmonių, todėl rezultatai geriausiai atspindi tik stabilias ir likvidžias rinkas. Trečia, modeliai mokėsi tik iš istorinių kainų, neatsižvelgiant į bendrą ekonomiką ar investuotojų nuotaikas, kas krizės metu yra labai svarbu. Dėl šių priežasčių investuotojams moksliniai šaltiniai rekomenduoja mašininio mokymosi modeliais naudotis tik kaip pagalbiniu patariamuoju įrankiu. Galutinius sprendimus turėtų priimti žmogus, kad netikėtų krizių metu nebūtų akklai pasikliaujama algoritmu. Taip pat, greitai

besikeičiančioje rinkoje kapitalą geriau skirstyti paprastai – visoms akcijoms po lygiai (1/N strategija). Sudėtingi optimizavimo metodai (pvz., Markowitz modelis) pernelyg jautriai reaguoja į duomenų netikslumus ir gali tik dar labiau padidinti klaidas.

Siekiant toliau plėtoti šią temą, ateityje vertėtų praplėsti GNN modelio galimybes. Pavyzdžiui, vertinant akcijų ryšius, algoritmui galima būtų pateikti ne tik kainų statistiką, bet ir papildomus duomenis: infliaciją, palūkanų normas ar net investuotojų nuotaikas, analizuojamas iš socialinių tinklų. Taip pat šią sukurtą sistemą būtų naudinga išbandyti rizikingesnėse ir labiau svyruojančiose srityse, pavyzdžiui, besivystančių šalių arba kriptovaliutų rinkose. Tai leistų geriau suprasti, kaip sėkmingai algoritmas prisitaiko ir veikia už tradicinių akcijų rinkų ribų.

## Literatūros sąrašas

1. Gu, S., Kelly, B., & Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *Journal of Financial Economics*, 138(2), 422-455.
2. Fama, E. F. (1970). Efficient Capital Markets: A Review of Theory and Empirical Work. *The Journal of Finance*, 25(2), 383–417.
3. Lopez de Prado, M. (2018). *Advances in Financial Machine Learning*. John Wiley & Sons.
4. Huang, Y., Capretz, L. F., & Ho, D. (2021). Machine Learning for Stock Prediction Based on Fundamental Analysis. 2021 IEEE Symposium Series on Computational Intelligence (SSCI), 1-7.
5. Neely, C. J., Rapach, D. E., Tu, J., & Goyal, A. (2014). Forecasting the equity risk premium: The role of technical indicators. *Management Science*, 60(7), 1772-1791.
6. Xing, F. Z., Cambria, E., & Welsch, R. E. (2018). Natural language based financial forecasting: a survey. *Artificial Intelligence Review*, 50(1), 49-73.
7. Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, 259(2), 689-702.
8. Kumar, A., Alsadoon, A., Prasad, P. W. C., Abdullah, S., Rashid, T. A., Pham, D. T. H., & Nguyen, T. Q. V. (2021). Generative Adversarial Network (GAN) and Enhanced Root Mean Square Error (ERMSE): Deep Learning for Stock Price Movement Prediction. *Multimedia Tools and Applications*.
9. Matsunaga, D., Suzumura, T., & Takahashi, T. (2019). Exploring Graph Neural Networks for Stock Market Predictions with Corporate Knowledge Graph. *arXiv preprint arXiv:1911.13062*.
10. Ji, S., Kim, J., & Im, H. (2019). A comparative study of bitcoin price prediction using deep learning. *Mathematics*, 7(10), 898.
11. Nelson, D. M., Pereira, A. C., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. In *2017 International joint conference on neural networks (IJCNN)* (pp. 1419-1426). IEEE.
12. Chang, P. C., Liu, C. H., Lin, J. L., Fan, C. Y., & Ng, C. S. P. (2009). A neural network with a case based dynamic window for stock trading prediction. *Expert Systems with Applications*, 36(3), 6889-6898.
13. Kim, R., So, C. H., Jeong, M., Lee, S., Kim, J., & Kang, J. (2019). Hats: A hierarchical graph attention network for stock movement prediction. *arXiv preprint arXiv:1908.07999*.
14. Sawhney, R., Agarwal, S., Wadhwa, A., & Shah, R. R. (2020). Deep attentive learning for stock movement prediction from social text and company correlations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 8415-8426).
15. Diebold, F. X., & Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 13(3), 253-263.
16. Sharpe, W. F. (1994). The Sharpe ratio. *The Journal of Portfolio Management*, 21(1), 49-58.
17. Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance*, 7(1), 77-91.

18. Michaud, R. O. (1989). The Markowitz Optimization Enigma: Is 'Optimized' Optimal? *Financial Analysts Journal*, 45(1), 31-42.
19. DeMiguel, V., Garlappi, L., & Uppal, R. (2009). Optimal versus naive diversification: How inefficient is the 1/N portfolio strategy?. *The Review of Financial Studies*, 22(5), 1915-1953.
20. Patel, M., Jariwala, K., & Chattopadhyay, C. (2024). A Systematic Review on Graph Neural Network-based Methods for Stock Market Forecasting. *ACM Computing Surveys*.
21. Qian, H., Zhou, H., Zhao, Q., et al. (2024). MDGNN: Multi-Relational Dynamic Graph Neural Network for Comprehensive and Dynamic Stock Investment Prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*.
22. Du, K., Mao, R., Xing, F., & Cambria, E. (2024). A Dynamic Dual-Graph Neural Network for Stock Price Movement Prediction. *IEEE Transactions on Neural Networks and Learning Systems*.
23. Ansari, Y. (2024). Multi-Cluster Graph (MCG): A Novel Clustering-Based Multi-Relation Graph Neural Networks for Stock Price Forecasting. *IEEE Access*.
24. Ardia, D., Bluteau, K., Boudt, K., & Inghelbrecht, K. (2021). Forecasting risk with Markov-switching GARCH models: A large-scale performance study. *International Journal of Forecasting*, 34(4), 733-747.
25. Tsay, R. S. (2005). *Analysis of Financial Time Series* (2nd ed.). John Wiley & Sons.
26. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794.
27. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780.
28. Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 20(1), 61–80.
29. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2018). Graph attention networks. In *International Conference on Learning Representations (ICLR)*.
30. Matsunaga, D., Suzumura, T., & Takahashi, T. (2019). Exploring graph neural networks for stock market predictions with rolling window analysis. *arXiv preprint arXiv:1909.10660*.
31. DeMiguel, V., Garlappi, L., & Uppal, R. (2009). Optimal versus naive diversification: How inefficient is the 1/N portfolio strategy?. *The Review of Financial Studies*, 22(5), 1915-1953.
32. Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, 259(2), 689-702.

## Priedai

### 1 priedas

#### Sortino rodikliai

Akcija (Ticker)	SVM	XGBoost	GRU	LSTM	GNN
AAPL	0,19	0,31	0,52	0,63	0,08
MSFT	-0,06	0,39	0,07	-0,15	0,17
NVDA	0,79	0,30	0,99	0,43	0,48
AMZN	0,46	0,00	0,17	0,03	-0,04
TSLA	0,30	0,38	0,63	0,82	0,37
GOOGL	0,16	0,35	0,33	0,41	-0,15
META	-0,08	0,78	0,62	0,22	0,12
JPM	0,05	0,12	0,07	0,10	0,42
V	0,20	0,60	0,29	-0,22	0,31
JNJ	-0,08	0,20	-0,09	-0,08	0,44
PFE	-0,07	-0,25	0,05	0,16	-0,19
XOM	-0,21	-0,25	-0,10	-0,13	0,19
PG	0,20	0,27	0,58	1,21	0,59
KO	0,35	0,06	0,18	0,14	0,38
BA	-0,38	-0,47	0,07	0,10	0,02
Vidurkis	0,12	0,19	0,29	0,24	0,21

## Calmar rodikliai

Akcija (Ticker)	SVM	XGBoost	GRU	LSTM	GNN
AAPL	0,21	0,29	0,49	0,69	0,10
MSFT	-0,03	0,25	0,04	-0,07	0,30
NVDA	0,76	0,20	1,04	0,48	0,80
AMZN	0,50	0,00	0,09	0,02	-0,03
TSLA	0,26	0,40	0,61	0,86	0,39
GOOGL	0,19	0,31	0,34	0,64	-0,18
META	-0,07	0,85	0,74	0,23	0,08
JPM	0,04	0,09	0,04	0,05	0,52
V	0,15	0,79	0,21	-0,11	0,32
JNJ	-0,07	0,19	-0,07	-0,05	0,65
PFE	-0,05	-0,11	0,04	0,09	-0,11
XOM	-0,13	-0,15	-0,06	-0,08	0,19
PG	0,18	0,20	0,37	1,39	1,04
KO	0,48	0,04	0,20	0,22	0,65
BA	-0,22	-0,26	0,07	0,10	0,02
Vidurkis	0,15	0,21	0,28	0,30	0,32

### XLK ETF rezultatų jautrumas aukštesniems slenksčiams

Laikotarpis	Top Pozicijos
2019-02	BAC (21.6%), MA (16.6%), PG (16.6%), AAPL (12.0%)
2019-08	AAPL (100.0%)
2020-02	NVDA (58.6%), MSFT (41.4%)
2020-08	AAPL (30.5%), TSLA (14.4%), PG (13.2%), JPM (11.2%)
2021-02	MSFT (80.0%), NVDA (20.0%)
2021-08	NVDA (24.2%), BAC (19.1%), GOOGL (17.9%), JPM (9.5%)
2022-02	AAPL (33.3%), MSFT (33.3%), NVDA (33.3%)
2022-08	AAPL (69.0%), NVDA (22.3%), MSFT (8.7%)
2023-02	NVDA (100.0%)
2023-08	JNJ (24.4%), GOOGL (18.9%), PG (16.2%), NVDA (16.0%)
2024-02	NVDA (69.8%), META (16.0%), PG (9.6%), JNJ (2.0%)
2024-08	UNH (38.8%), JNJ (22.0%), META (18.5%), MA (9.6%)
2025-01	GOOGL (51.4%), TSLA (39.4%), JPM (9.2%)