



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

Magistro baigiamasis darbas

Kripto valiutų kainų kintamumo prognozavimo metodų analizė
Analysis of cryptocurrency volatility forecasting methods

Atliko:

Yuliya Duniak

parašas

Vadovas:

dr. Joana Katina

Vilnius
2023

Turinys

Sutartinis terminų žodynas	3
Santrauka	4
Summary	5
Ivydas	6
1. Laiko eilučių analizės samprata	8
2. Susijusių darbų apžvalga	9
3. Duomenys ir tyrimo metodologija	12
3.1. Naudojami duomenys	12
3.2. Kintamumas	12
3.3. Normalumo tikrinimas	13
3.3.1. Lilliefors testas	14
3.3.2. Shapiro-Wilk testas	14
3.3.3. Kvantilių palyginimo (QQ) diagrama	14
3.3.4. Empirinės tankio funkcijos grafikas	14
3.4. Nuostolio metrikos prognozėms įvertinti	15
3.5. Random Forest	16
3.6. SVM	17
3.7. KNN	18
3.8. LSTM	19
3.9. GRU	21
4. Laiko eilučių modelių eksperimentinis tyrimas	22
4.1. Programavimo kalba ir naudojamos bibliotekos	22
4.2. Normalumo tikrinimas	22
4.3. Modelių taikymas	26
4.3.1. SVM modelis	27
4.3.2. Random Forest modelis	29
4.3.3. KNN modelis	31
4.3.4. LSTM modelis	34
4.3.5. GRU modelis	36
4.3.6. Rezultatai	39
4.3.7. Kitos laiko eilutės	42
Išvados ir rekomendacijos	45
Ateities tyrimų planas	46
Literatūros šaltiniai	47
Priedai	50

Sutartinis terminų žodynas

- SVM - Atraminių vektorių mašinų modelis (angl. Support vector machines)
- RF - Atsitiktinio miško modelis (angl. Random forest)
- KNN - K-arčiausių kaimynų modelis (angl. K-nearest neighbors)
- LSTM - Trumpalaikės-ilgalaikės atminties neurotinis tinklas (angl. Long short term memory)
- GRU - Pralaidinis rekurentinis vienetas (angl. Gated recurrent unit)
- RMSE - Šakninė vidutinė kvadratinė paklaida (angl. Root-mean-square error)
- MAE - Vidutinė absoliuti paklaida (angl. Mean absolute error)
- MSE - Vidutinė kvadratinė paklaida (angl. Mean squared error)
- MDA - Vidutinis krypties tikslumas (angl. Mean directional accuracy)
- QQ - Kvantilių-kvantilių diagrama (angl. Quantile-quantile plot)
- R2 - Regresijos paklaida (angl. Regression error)

Santrauka

Magistro baigiamojo tikslas - susipažinti ir ištirti neuroninių tinklų prognozavimo modelius, kurių pagalba būtų įmanoma prognozuoti kriptovaliutų kainų kintamumą.

Darbe apžvelgiama kriptovaliutų kainų ir jų kintamumo laiko eilučių prognozavimo problema. Susijusių darbų analizės rezultate buvo išrinkti ne tik modeliai, kurie bus naudojami prognozavimui, bet ir kriptovaliutų kintamumo pakaitiniai kintamieji, kurie bus prognozuojami. Kiekvienas iš pasirinktų modelių buvo ištreniruotas ir ištestuotas. Tam buvo paimti dviejų, Litecoin ir Bitcoin, kriptovaliutų dienos ir valandos intervalo duomenys. Kiekvienas iš pasirinktų prognozavimo modelių buvo ištirtas, o eksperimentinio tyrimo pabaigoje modelių rezultatai buvo palyginti. Darbo pabaigoje aprašomos eksperimentinio tyrimo metų gautos išvados, taip pat sudarytas ateities planų sąrašas.

Raktiniai žodžiai: laiko eilutės, Bitcoin, Litecoin, kriptovaliutų kainų kintamumas, KNN, RF, SVM, LSTM, GRU

Summary

Analysis of cryptocurrency volatility forecasting methods

The master's thesis aims to introduce and study neural network prediction models, which could make it possible to predict the volatility of cryptocurrency prices.

The research opens with a short introduction of time series, their definition, and usage scenario. Then the work presents the problem of predicting cryptocurrency prices and their volatility. Also, it presents how neural network models such as KNN, RF, SVM, LSTM, and GRU could help to predict those values better. As a result of the analysis of related works, not only the models that will be used for forecasting in the experimental part but also cryptocurrency volatility proxies were selected. **The aim of the work** is to study and compare each of the selected models. All of them were trained and tested in different ways: Litecoin and Bitcoin cryptocurrencies daily and hourly time series were taken. Data was divided into two parts: training and testing so that models could be trained and then tested, so predicted values would be compared with the real ones. MDA, RMSE, MAE, and R2 metrics were chosen for the effectiveness evaluation of the selected models. The effectiveness of the models is compared at the end of the experimental study. At the end of the work, the findings and future plans are described.

Keywords: time series, Bitcoin, Litecoin, autocorrelation, cryptocurrency prices volatility, KNN, RF, SVM, LSTM, GRU

Ivydas

Kripto valiutos - skaitmeniniai ir tik virtualioje erdvėje egzistuojantys pinigai, kurie yra paremti blokų grandinės (angl. Blockchain) technologija įvairioms mokėjimo operacijoms atlikti be trečiųjų šalių tarpininkavimo, išlaikant finansinių operacijų anonimiškumą. Šis valiutos tipas nėra teisiškai apibrėžtas, jis neprižiūrimas jokių finansinių institucijų, o paremtas vien tarpusavio pasitikėjimu ir jo vertė nusako paklausos-pasiūlos principas: kuo daugiau žmonių nori šią kriptovaliutą įsigyti, tuo labiau kyla jos kaina. Tai lemia kriptovaliutos kainų nepastovumą: ji gali labai greitai nukristi, ar labai sparčiai išaugti.

Pirmoji pasaulyje kriptovaliuta Bitcoin buvo sukurta Satoshi Nakamoto 2008 metais, tačiau susidomėjimas ja ir augimas prasidėjo tik 2012 metais [18]. Pagal pasaulinę finansinę žiniasklaidos svetainę „CoinDesk“ kriptovaliutų rinkos kapitalizacija 2017 m. siekė 17 milijardų JAV dolerių. Galima stebėti, kaip sparčiai ji pradėjo augti nuo gruodžio 2020 iki 2021-ųjų kovo vidurio, liepos – rugsėjo ir spalio-lapkričio mėnesiais, po ko pasiekus savo didžiausią istorinę kainą pradėjo kristi. Sausio 2022 buvo užfiksuotas žemiausias Bitcoin kursas nuo 2021-ųjų birželio [23]. Nežiūrint į tokį nepastovumą, šiuo metu vien Bitcoin kapitalizacija pasiekė 850 milijardų JAV dolerių, kas rodo tik augantį susidomėjimą šia valiuta kaip investuotoju, taip ir spekuliantų tarpe.

Prekyba kriptovaliuta gali būti itin patraukli dėl didelio kainų svyravimo, iš ko prekyautojai galėtų gauti nemažai grąžos. Tačiau tai yra tiesiogiai susiję ir su aukšta rizika: ne visada kriptovaliutos kaina gali pasiekti tam tikrą numatytą skaičių. Dažnai kainos pokytį lemia socialiniai, ekonominiai, politiniai arba fundamentalūs veiksniai. Vienas tokiu buvo kai Bitcoin kainos nukrito daugiau nei 3 procentai po paslaptingo Elono Musko pranešimo Twitter platformoje, užsiminusio apie galimą išsiskyrimą su didžiausia kriptovaliuta [8]. Todėl kiekvienas investuotojas arba prekyautojas turi įvertinti visą pirkimo-pardavimo operacijų riziką prieš priimant sprendimą dėl sandorio sudarymo.

Ekonomikos ir finansų teorija teigia, kad nepastovumas atsiranda dėl realaus pasaulio sukrėtimų ir (arba) pasiūlos ir paklausos skirtumo šuolių. Tačiau 2021-2022 metų kriptovaliutų kainos rodo itin didelį nepastovumą, kurio negalima paaiškinti klasikine finansų ir ekonomikos teorija [9].

Šiais laikais kriptovaliutų kainų prognozavimas tampa viena iš populiarėjančių tyrimų sričių, kadangi žinoti, kada kriptovaliutų kainos augs, o kada kris yra neįmanoma. Tam tikslui kaip tik yra naudojama nemažai skirtingų metodų: tradiciniai statistiniai ir mašininio mokymosi metodai, tokie kaip Bajeso regresija, logistinė regresija, tiesinė regresija, autoregresiniai modeliai ir kiti [19]. Tačiau šiame darbe bus analizuojami neuroninių tinklų bei jų atmainų modeliai skirti laiko eilutėms tirti.

Tyrimo objektas – kriptovaliutų kainų grąža ir kintamumas.

Tyrimo tikslas – iširti laiko eilučių neuroninių tinklų modelius kriptovaliutų kainų kintamumo prognozavimui ir išrinkti geriausią.

Siekiant tikslo išsikelti šie **uždaviniai**:

- atlikti teorinę literatūros analizę;
- surinkti kriptovaliutų duomenis analizei;
- paruošti duomenis analizei
- iširti, sudaryti ir palyginti atrinktus neuroninių tinklų modelius juos išbandant tiek kriptovaliutų kainų kintamumui prognozuoti, tiek kitų laike kintančių eilučių kintamumo prognozavimui;

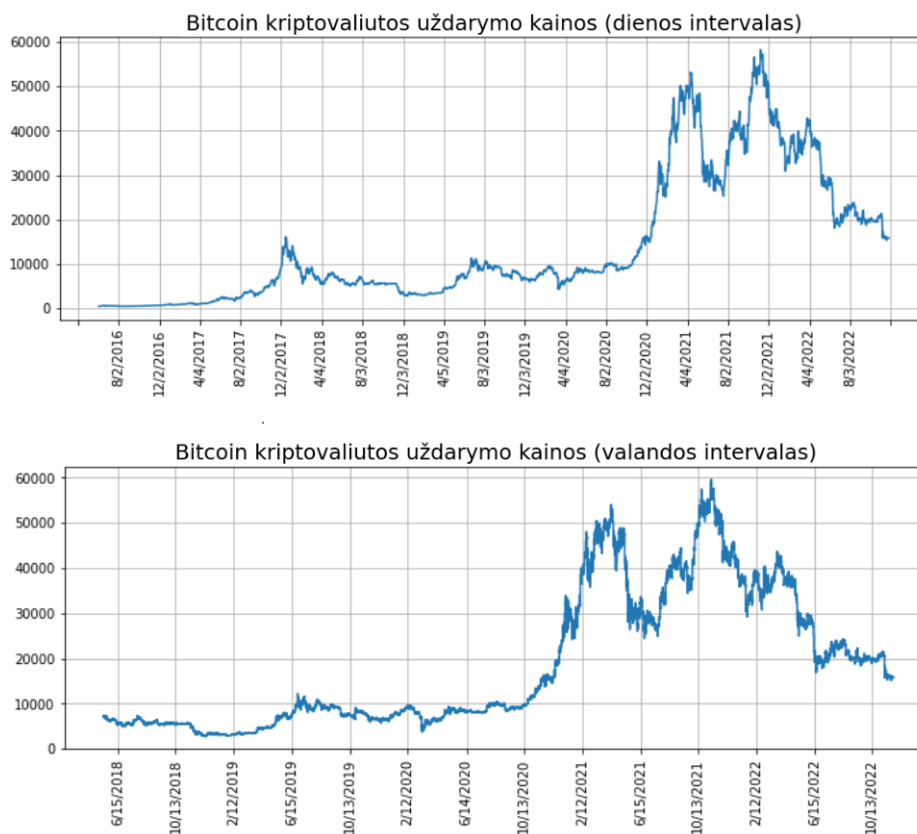
Atlikus analizę ir metodų tyrimus buvo nuspręsta šiame darbe kriptovaliutų kainų kintamumą ir gražą prognozuoti laiko eilučių prognozavimo neuroninių tinklų metodais: Random Forest, SVM, KNN, bei jų atmainomis - GRU ir LSTM. Kiekvienam iš pasirinktų tyrimui metodų buvo sudarytas prognozavimo modelis. Eksperimentams buvo paimtos kelių kriptovaliutų laiko eilutės. Kiekvienas prognozavimo modelis buvo iširtas skirtingais būdais tam, kad suprasti, ar jis tinkamas prognozuoti tiek kriptovaliutų laiko eilutes, tiek kitų laike kintančias eilutes. Tyrimo pabaigoje išrinktų modelių efektyvumai buvo palyginti ir išrinktas geriausias. Eksperimentinės dalies pabaigoje aprašomos gautos išvados ir pristatomi ateities planų darbai.

Svarbu paminėti, kad kadangi šis darbas yra tęstinis, tam tikros teksto dalys buvo panaudotos praeitame semestre ruošiant mokslo tiriamąjį darbą kriptovaliutų kainų prognozavimo metodų analizės tema. Informacija, kuri buvo naudota praeito semestro metu:

- Didžioji dalis įvado;
- Dalis susijusių darbų analizės;
- Didžioji dalis 1 skyriaus;
- Didžioji dalis 3.1 skyriaus;
- Didžioji dalis 4.1 skyriaus.

1. Laiko eilučių analizės samprata

Laiko eilutė yra tam tikra duomenų taškų seka, kuri atsiranda chronologiška tvarka per tam tikrą laikotarpį. Ji gali būti paimta iš bet kurio kintamojo, kuris keičiasi laikui bėgant. Laiko eilutės, kaip taisyklė, atsiranda išmatuojant tam tikrą rodiklį. Tai gali būti tiek techninių sistemų charakteristikos, tiek gamtos, socialinių-ekonominių reiškinių ir procesų rodikliai. Pavyzdžiui, valiutos kurso ar akcijų kainos dinamika, kurią analizuojant bandoma nustatyti pagrindinę vystymosi kryptį, t.y. tendenciją (1 pav.).



1 pav. Bitcoin kainos laiko eilutė

Laiko eilučių analizė – tai statistinių metodų rinkinys laiko eilutės komponentams nustatyti ir ją prognozuoti.

Laiko eilučių analizė gali būti naudinga norint pamatyti, kaip tam tikras turtas, vertybinis popierius ar ekonominis kintamasis keičiasi laikui bėgant. Ji taip pat gali būti naudojama norint iširti, kaip per tą patį laikotarpį su pasirinktu tašku susiję kito taško pokyčiai.

Vienas pagrindinių laiko eilučių analizės tikslas yra prognozavimas, kuris yra pagrįstas praeities duomenimis ir žiniomis. Laiko eilučių prognozavimas naudoja informaciją apie istorines reikšmes ir susijusius modelius, kad būtų galima numatyti būsimą rezultatą. Dažniausiai tai susiję su tendencijų analize, ciklinių svyravimų analize ir sezoniškumo problemomis. Kaip ir naudojant visus prognozavimo metodus, sėkmė nėra garantuota.

Egzistuoja pakankamai daug metodų, kurie skirti prognozuoti įvairias laiko eilutes, tačiau šiame darbe bus apžvelgiami neuroninių tinklų RF, SVM, KNN ir jų atmainų LSTM ir GRU modeliai.

2. Susijusių darbų apžvalga

Kintamumas (angl. volatility) - viena iš įdomiausių Bitcoin ypatybių. Jo modeliavimas ir prognozavimas yra labai svarbūs bitkoinų investuotojų sprendimų priėmimo analizei ir rizikos valdymui. Tačiau dauguma ankstesnių bitkoino kintamumo tyrimų buvo pagrįsti ekonometriniais modeliais.

Ekonomikos ir finansų teorija teigia, kad kintamumas atsiranda dėl realaus pasaulio sukrėtimų ir (arba) pasiūlos ir paklausos skirtumo šuolių. Tačiau 2021 - 2022 metų kriptovaliutų kainos rodo itin didelį kintamumą, kurio negalima paaiškinti klasikine finansų ir ekonomikos teorija. Nepaisant to, naujausioje laiko eilučių ekonometrijos literatūroje jau egzistuoja pakankamai mokslinių tyrimų, kur sėkmingai sumodeliuojamas ir suprognozuojamas toks nepaaiškinamas kintamumas [11].

Egzistuoja pakankamai darbų, parašytų remiantis autoregresiniais modeliais. Daugumoje nagrinėjamų darbų buvo pastebėta, kad ARIMA, SARIMA ir GARCH modeliai ganėtinai prastai prognozuoja skirtingų kriptovaliutų kainas, tačiau įvairių papildomų tyrimų eigoje GARCH autoregresinis modelis parodė neblogus rezultatus dirbant su kriptovaliutų kintamumo (nepastovumo) prognozavimu.

Bitcoin kintamumo prognozavimo, naudojant mašininio mokymosi algoritmus, tyrimų vis dar yra nedaug, o pats kintamumas laikomas latentiniu kintamuoju, tad tokiuose tyrimuose autoriai dažnai nagrinėja įvairius pakaitinius kintamuosius (angl. proxy)[21]. Labai dažnai pasitaiko darbai, kur kintamumas išreikštas per absoliučią grąžą arba realizuotąją grąžą. Pavyzdžiui, Lorella Fatone, Francesca Mariani ir Francesco Zirilli savo konferencijos darbo tyrimams panaudojo logaritminę grąžą [12], o Elie Bouri, Konstantinos Gkillas, Rangan Gupta ir Christian Pierdzioch - realizuotąją grąžą [7].

Autoriai Rahmat Albariqi ir Edi Winarko tyrė Bitcoin kriptovaliutos kainas rekurentinio ir daugiasluoksnio neuroninių tinklų metodais. Jų tyrimo objektas - Bitcoin kriptovaliutos kainos pokytis. Tyrimo rezultatai parodė, kad abu metodai turėjo geresnį tikslumą prognozuojant į ilgesnį laiko tarpą. Lyginant abu modelius, daugiasluoksnis neuroninių tinklų metodas pasirodė geriausiai su 81.3 % tikslumu [4].

Sean McNally, Jason Roche ir Simon Caton savo darbe analizavo Bitcoin kainų prognozavimo algoritmus ir lygino tradicinį autoregresinį modelį ARIMA su LSTM ir RNN modeliais. Šio tyrimo rezultate LSTM buvo tiksliausias (tikslumas - 52.78 %), o RNN turėjo mažiausia RMSE paklaidą (RMSE- 5.45 %) [24].

Tarptautiniame "Rizikos ir finansų valdymo" žurnale buvo pristatytas Ze Shen, Qing Wan ir David J. Leatham mokslinis tyrimas. Jo tikslas – palyginti jų rezultatus prognozuojant tikslumą ir rizikos valdymo efektyvumą. Autoriai priėjo išvadą, kad mašininio mokymosi modeliai prognozuojant laiko eilutes yra pranašesni už tradicinius ekonometrinius modelius. Rezultatai rodo, kad RNN lenkia GARCH ir EWMA vidutinius prognozavimo rezultatus. Tačiau jis yra mažiau efektyvus fiksuojant ekstremalius bitkoinų rinkos įvykius. Be to, RNN rodo prastus rizikos vertės prognozavimo rezultatus, o tai rodo, kad jis negali gerai veikti kaip ekonometriniai modeliai aiškinant ekstremalų kintamumą [32].

Matthew Chen, Neha Narwal and Mila Schultz taip pat išstudijavo ARIMA modelį ir lygino jo rezultatus su LR, SVM, RF, RNN, NN ir NB modeliais analizuojant Ethereum pokyčio laiko eilutes. ARIMA modelis pralenkė kitus modelius pagal savo tikslumą (61.17 %), tačiau labai neatsiliko ir SVM bei LR modeliai (atitinkamai, 55.99 % ir 56.94 %). Taip pat svarbu paminėti, kad buvo paimti valandos tarpo duomenys už 2015-2017 metus, ko paprasčiausiai galėjo būti per

mažai neuroninių tinklų modeliams [10].

Samiha Nanjiba, Amitabha Chakrabarty ir Shaily Roy siekiant iširti įvairius modelius kintamumui prognozuoti, tyrė ir darė prognozes Bitcoin kriptovaliutos kintamumui pasinaudojant HARRV, MLP, LSTM, GRU, CNN, SVM ir Ridge Regression. Atlikę analizę, jie nustatė, kad iš visų tiriamų modelių ARIMA modelis parodė geriausias rezultatus, jo našumas buvo 90.31 % [30].

Tuo tarpu autoriai Hakan Pabuçcu, Serdar Ongan and Ayse Ongan bandė suprasti, kuris mašininio mokymosi algoritmas rodo didesnę Bitcoin kriptovaliutos pokyčio prognozavimo našumą: SVM, ANN, NB ar RF palyginus su LR modelio našumu. Atlikę tyrimą su duomenimis ir palyginę gautus rezultatus autoriai priėjo išvada, kad RF turi didžiausią prognozavimo našumą su 0.884 F testo rezultatu, o NB - mažiausią su 0.717. Neblogai pasirodė ir SVM ir ANN modeliai. Taip pat buvo pastebėta, kad išankstinis duomenų apdorojimas padidina prognozavimo našumą, nors jau yra plačiai pripažįstama, jog išankstinis duomenų apdorojimas nėra būtinas, kai naudojami mašininio mokymosi algoritmai [27].

Panašias išvadas padarė ir Yiqing Hua. Savo darbe jis palygino ilgalaikės-trumpalaikės atminties metodą (LSTM) ir ARIMA modelį. ARIMA modelis pasižymėjo savo efektyvumu prognozuojant duomenis per trumpą laiko intervalą, tačiau LSTM modelis davė geresnius rezultatus [16].

Analizuojant susijusius darbus, buvo pastebėta, kad dauguma autorių savo tyrimams naudoja realizuotąjį kintamumą. Pavyzdžiui, R. Miura, L. Pichl ir T. Kaizoji agregavo minutinės Bitcoin gražos realizuotąjį kintamumą trijų valandų intervale. Prognozavimui jie pasirinko MLP, GRU, LSTM, SVM ir Ridge Regression modelius. Rezultate GRU modelis geriau susitvarkė su validavimo duomenų rinkiniu (angl. validation data set), o Ridge Regression parodė geriausias rezultatus su testiniais duomenimis [26].

Mimoza Mjoska, Blagoj Ristevski, Snezana Savoska ir Vladimir Trajkovik taip pat darė Bitcoin kriptovaliutos kintamumo tyrimus pasitelkiant Random Forest algoritmu. Tyrimo objektu buvo realizuotas Bitcoin kriptovaliutos kintamumas. O tyrimo pabaigoje buvo padaryta išvada, kad RF yra vienas geriausių mašininio mokymosi algoritmų dėl prognozavimo didelio tikslumo. MAE paklaida mokymosi metu buvo 0,006, o testavimo metu - 0,015 [25].

Z. Shen, Q. Wan ir D. J. Leatham domėjosi Bitcoin kintamumo prognoze: jie atliko lyginamąjį GARCH, EWMA ir RNN tyrimą. Autoriai pasirinko du kintamumo pakaitinius kintamuosius (angl. proxy): kvadratinę Bitcoin gražą ir Garman-Klass matą. Modelių našumui naudojo RMSE ir MAE metrikas. Tyrimo rezultatai parodė, kad RNN turi geriausią našumą pagal MAE vertinimo metriką, tačiau atsilieka pagal RMSE. Taip pat jie pastebėjo, kad RNN modelis geriau fiksuoja kintamumo tendencijas, tačiau neįvertina pačio kintamumo. Antrasis Garman-Klass kintamasis jų tyrimuose pasirodė mažiau nepastovus, nei pirmasis. Taigi, anot autorių, jis neveikia taip gerai, kaip kvadratinė Bitcoin graža [32].

Pagal bitkoinų ir kriptovaliutų naujienų portalą „NewsBTC“, „Sentix“ ir „Bloomberg“ duomenys rodo, kad nuo 2010 m. liepos mėnesio Bitcoin geriausi mėnesio vidurkiai buvo balandžio, gegužės, spalio ir lapkričio mėnesiais. Kita vertus, pirmojo ketvirčio gruodį ir ypač vasarą rinkos buvo gana ramios. Tokias pat išvadas pagal "CoinDesk" svetainės duomenis galima daryti ir apie 2021 m. Bitcoin ir Ethereum kriptovaliutos rinkos kainas [23]. „NewsBTC“ taip pat atkreipė dėmesį ir į aiškią prekybos tendenciją savaitės ir valandos laikotarpiu: vidutiniškai pirmadienis ir penktadienis buvo aktyviausios dienos per visą savaitę, o savaitgalį vidutinė prekybos apimtis buvo žymiai mažesnė nei darbo dienomis [17].

Ganėtinai mažai darbų egzistuoja, kur būtų išnagrinėtas KNN modelis, kuris gali būti naudojamas tiek klasifikavimo, tiek regresijos problemoms spręsti kriptovaliutų kontekste. Tačiau Khalid

Alkhatib, Hassan Najadat, Ismail Hmeidi ir Mohammed K. Ali Shatnawi pasinaudojo jo pagalba tiriant akcijų kainas. Savo tyrimų metu pastebėjo, kad šis modelis visai neblogai susitvarko su akcijų kainų prognozėmis [5]. Jų tyrimo rezultatai parodė, kad KNN algoritmas buvo stabilus ir patikimas, su nedidele paklaida, todėl rezultatai gavosi racionali ir pagrįsti. Taip pat jie pastebėjo, kad ir prognozavimo rezultatai buvo gan artimi faktinių akcijų kainų.

Taip pat M.S.K.P.Sri Prada, K.Sudhakar, A.Sukeerthi savo darbe išnagrinėjo RF, SVM, LR ir KNN algoritmus prognozuojant akcijų grąžą. Jų tyrimo metu KNN parodė geriausius rezultatus [29]. Tokią pat išvadą padarė ir R.S. Latha su savo bendraautoriais. Jų tyrimo metu buvo pastebėta, kad prognozuojant akcijų kainų pokyčius, KNN yra labai galingas, kadangi padarytų eksperimentų metu jo tikslumas buvo didesnis nei 70 % [22].

Susijusių darbų analizės metu taip pat buvo pastebėta, kad skirtingų modelių našumui įvertinti, prognozuojant tiek kriptovaliutų kainas, tiek kintamumą, buvo naudojami dažniausiai tokie kriterijai ir metrikos, kaip MAPE, RMSE, MSE, MAE. Taip pat keleta autorių pasinaudojo ir determinacijos koeficiento pagalba [34][3].

Autoriai Emmanuel Pintelas, Ioannis E. Livieris, Stavros Stavroyiannis, Theodore Kotsilieris ir Panagiotis Pintelas analizavo ir lygino CNN, LSTM, BiLSTM modelių našumą. Tyrimo metu naudojo tik RMSE metriką iš visų išvardintų aukščiau. Toks sprendimas buvo pagrįstas tuo, kad anot jų MAE kriterijus elgiasi beveik taip pat, kaip ir RMSE. Įdomu, kad jiems pavyko apskaičiuoti ir kainų judėjimo krypties prognozavimo tikslumą (jei kaina padidės ar mažės). Tam panaudojo dar dvi papildomas metrikas: tikslumą (Acc) ir F1-balas (F1) [28]. Tačiau nežiūrint į tai, kad šį kartą metrikos buvo panaudotos kiek kitokios, tyrimo rezultatai buvo gana prasti. Autoriai priėjo išvadą, visi tirti modeliai yra neefektyvūs ir nepatikimi kriptovaliutų kainų prognozei ir negali veiksmingai išspręsti problemos.

Atlikus susijusių darbų analizę galima teigti, kad tikrasis kriptovaliutų kintamumas yra latentinis kintamasis, t.y. nepastebimas, o pats terminas ganėtinai abstraktus, todėl, kaip ir kitų autorių darbuose, kintamumas bus išreikštas pakaitiniais kintamaisiais (angl. proxy), tokiais kaip kvadratine grąža arba realizuotąja grąža. Norint gauti patikimesnį prognozavimo modelių našumo palyginimo rezultata, bus atsižvelgta į Z. Shen, Q. Wan ir D. J. Leatham pastabas [32], ir pasirinktas antrasis kintamumą išreiškiantis kintamasis. Atsižvelgiant į autorių pastabas, galima manyti, kad Garman-Klass kintamasis pasirodė ne toks našus todėl, kad jis ignoruoja naktinius šuolius arba pervertina tikrąją dispersiją. Todėl antruoju kintamuoju bus laikoma 2000 metais Yang-Zhang sukurta kintamumo priemonė, kuri, pasak J. Shu ir J. E. Zhang, neignoruoatų šuolių ir yra nepriklausoma nuo dreifo, tad gali pateikti tikslų įvertinimą [33]. Taip pat tolimesniam tyrimui buvo pasirinkti šie modeliai: Random Forest, SVM, KNN ir jų atmainų LSTM ir GRU modeliai. O šių modelių našumui įvertinti buvo pasirinktos RMSE, MAE, R2 ir MDA metrikos.

3. Duomenys ir tyrimo metodologija

3.1. Naudojami duomenys

Šiame darbe naudojami duomenys buvo gauti iš svetainės www.cryptodatadownload.com. Svetainė teikia plačią informaciją apie tokias kriptovaliutas kaip Bitcoin, Ethereum, Ripple, Litecoin ir kitas. Svetainėje pateikiami duomenys nuolat atnaujinami. Tuo labiau, šis šaltinis laikomas patikimu, kadangi juo buvo remtasi ne viename moksliniame darbe [1].

Darbui buvo paimti šie duomenys:

- Bitcoin dienos intervalo kainų istorija nuo 2016 m. birželio 4 d. iki 2022 m. lapkričio 27 d.
- Bitcoin valandos intervalo kainų istorija nuo 2018 m. gegužės 15 d. iki 2022 m. lapkričio 27 d.
- Litecoin dienos intervalo kainų istorija nuo 2017 m. birželio 25 d. iki 2022 m. lapkričio 27 d.
- Litecoin valandos intervalo kainų istorija nuo 2018 m. gegužės 15 d. iki 2022 m. lapkričio 27 d.

Duomenų rinkiniai sudaryti iš 9 stulpelių: "unix", "date", "symbol", "open", "high", "low", "close", "Volume BTC" ir "Volume EUR", kur:

- "unix" - Unix laiko žyma,
- "date" - data,
- "symbol" - finansinės operacijos valiutos kodas,
- "open" - atidarymo kaina,
- "high" - dienos aukščiausia kaina,
- "low" - dienos žemiausia kaina,
- "close" - uždarymo kaina,
- "Volume BTC" - dienos prekybos apimtys BTC,
- "Volume EUR" - dienos prekybos apimtys EUR.

Eksportuoti iš www.cryptodatadownload.com svetainės duomenys yra CSV formato. Taip pat dėl patogesnio duomenų nuskaitymo ir išrinkimo kintamumo reikšmei išsaugoti pridėtas stulpelis "vol".

3.2. Kintamumas

Šiame darbe nagrinėjamos Bitcoin ir Litecoin kriptovaliutų kainų kintamumo laiko eilutės. Kintamumui prognozuoti bus panaudotas esamos kainos ir ankstesnės kainos santykis, kitaip tariant graža. Gražai apskaičiuoti bus naudojama ši formulė:

$$r_t = \log(p_t/p_{t-1}), \quad (3.1)$$

kur bus apskaičiuojamas p_t kriptovaliutos uždarymo kainos t dieną ir p_{t-1} kriptovaliutos uždarymo kainos $t - 1$ dieną santykio logaritmas.

Norint gauti patikimesnį prognozavimo modelių našumo palyginimo rezultatą, bus atsižvelgta į Z. Shen, Q. Wan ir D. J. Leatham pastabas [32], ir pasirinktas antrasis kintamumą išreiškiantis kintamasis. Atsižvelgiant į autorių pastabas, galima manyti, kad Garman-Klass kintamasis pasirodė ne toks našus todėl, kad jis ignoruoja naktinius šuolius arba pervertina tikrąją dispersiją. Todėl antruoju kintamuoju bus laikoma 2000 metais Yang-Zhang sukurta kintamumo priemonė, kuri, pasak J. Shu ir J. E. Zhang, neignoruotų šuolių ir yra nepriklausoma nuo dreifo, tad gali pateikti tikslių įvertinimą [33]. Toks kintamumas apskaičiuojamas pagal formulę žemiau:

$$\sigma_{Yang-Zhang} = \sqrt{\sigma_o^2 + k\sigma_c^2 + (1 - k)\sigma_{rs}^2}, \quad (3.2)$$

kur

$$k = \frac{\alpha - 1}{\alpha + \frac{T+1}{T-1}} \quad (3.3)$$

$$\sigma_o^2 = \frac{1}{T-1} \sum_{t=1}^T \left(\ln\left(\frac{o_t}{c_{t-1}}\right) - Avg\ln\left(\frac{o_t}{c_{t-1}}\right) \right)^2 - \text{naktinis kintamumas}, \quad (3.4)$$

$$\sigma_c^2 = \frac{1}{T-1} \sum_{t=1}^T \left(\ln\left(\frac{c_t}{o_t}\right) - Avg\ln\left(\frac{c_t}{o_t}\right) \right)^2 - \text{kintamumas nuo atidarymo iki uždarymo}, \quad (3.5)$$

$$\sigma_{rs} = \sqrt{\frac{1}{T} \sum_{t=1}^T \left(\ln\left(\frac{h_t}{c_t}\right) * \ln\left(\frac{h_t}{o_t}\right) + \ln\left(\frac{l_t}{c_t}\right) * \ln\left(\frac{l_t}{o_t}\right) \right)} - \text{Rogers-Satchell kintamumas}, \quad (3.6)$$

T – dienų kiekis, o_t – kriptovaliutos atidarymo kaina t dieną, c_t – kriptovaliutos uždarymo kaina t dieną, h_t – aukščiausia kriptovaliutos kaina t dieną, l_t – žemiausia kriptovaliutos kaina t dieną.

3.3. Normalumo tikrinimas

Duomenų pasiskirstymas pagal Gauso normalųjį skirstinį statistikoje vadinamas normalumu. Normalumo prielaida statistikoje vadinama reikalavimu, kad duomenys skirstytųsi normaliai [15]. Normalumo prielaidai tikrinti yra keletas būdų, pavyzdžiui, kvantilių palyginimo (QQ) diagramą, Lilliefors ir Shapiro-Wilk testai.

Normalumo prielaidai patikrinti yra formuluojamos dvi hipotezės: H_0 , kuri nusako, kad pasiskirstymas yra normalus ir nuokrypio nuo normalumo nėra, ir H_1 , kuri, atvirkščiai, nusako, kad pasiskirstymas nėra normalus ir yra nuokrypis [15].

Tam, kad patvirtinti arba paneigti hipotezes pasirenkamas nuokrypio nuo normalumo matas p ir reikšmingumo lygmuo, kuri kitaip vadinamas riba. Dažniausiai – reikšmingumo lygmuo lygus 0.05. Tokiu būdu, jei $p \geq 0.05$, nulinė hipotezė gali būti priimta, o duomenys yra pasiskirstę pagal normalųjį skirstinį arba pasiskirstymo nuokrypis yra nereikšmingas. Normalumo prielaida tenkinama. Tuo atveju, jei $p < 0.05$, tada pasiskirstymo nuokrypis yra ganėtinai reikšmingas, normalumo prielaida pažeidžiama, o hipotezė atmetama [15].

3.3.1. Lilliefors testas

Lilliefors testas - statistinis testas, pavadintas Džordžo Vašingtono universiteto statistikos profesorius Huberto Lillieforso vardu. Šis testas yra Kolmogorovo-Smirnovo testo modifikacija.

Testas naudojamas norint patikrinti nulinę hipotezę, kad imtis pasiskirsto pagal normalųjį dėsnį tuo atveju, kai normaliojo skirstinio parametrai (vidurkis ir dispersija) yra nežinomi. Jei p reikšmė mažesnė už 0.05, tada nulinė hipotezė, kad imtis yra normalaus skirstinio, gali būti atmesta.

3.3.2. Shapiro-Wilk testas

Shapiro-Wilk testas - dar vienas dažnumo statistikos normalumo testas. Jis 1965 metais buvo pristatytas dvejais statistikais - Samueliu Sanfordu Shapiro ir Martinu Wilku [31]. Apskaičiuota testo reikšmė gaunama pagal šią formulę:

$$W = \frac{\left(\sum_{i=1}^n a_i x_{(i)}\right)^2}{\sum_{i=1}^n (x_i - \bar{x})^2}, \quad (3.7)$$

kur \bar{x} - imties vidurkis, a_i - konstantos, gautos iš kovariacijų, dispersijų ir imties vidurkio, o $x_{(i)}$ žymi i-osios eilės statistiką, t.y. i-tą mažiausią skaičių imtyje.

Kaip ir dauguma statistinio patikimumo testų, jei duomenų yra pakankamai daug, šis testas gali aptikti net nereikšmingus nukrypimus nuo nulinės hipotezės, todėl patartina atlikti papildomą patikrinimą, pavyzdžiui, su kvantilių palyginimo (QQ) diagrama [13].

3.3.3. Kvantilių palyginimo (QQ) diagrama

Kvantilių palyginimo (QQ) diagrama dar vadinama kvantilių-kvantilių diagrama (angl. quantile-quantile plot, QQ plot). Ją 1968 metais pristatė statistikai M.B. Wilk ir R. Gnanadesikan [14].

QQ diagrama yra puikus grafinis būdas parodyti, ar du duomenų rinkiniai yra iš to paties skirstinio, kitaip tariant ji padeda vizualiai patikrinti turimų duomenų normalumą.

Diagramos veikimo principas yra gana paprastas: tarp nurodytos imties ir normalaus skirstinio nubrėžiama koreliacija. Taip pat nubrėžiama 45 laipsnių atskaitos linija. Normalusis skirstinys yra simetriškas, todėl neturi pakreipimo (vidurkis lygus medianai) ir pačioje QQ diagramoje paprastai paskirstyti duomenys atrodo kaip maždaug tiesi linija.

3.3.4. Empirinės tankio funkcijos grafikas

Empirinės tankio funkcija taip pat naudojama duomenims analizuoti, o jos grafikas dažnai vadinamas histograma [6]. Tankio funkcija aprašoma taip:

$$f_n(x) = \begin{cases} 0, & \text{kai } x < c_0 \\ \frac{f_j}{nh}, & \text{kai } c_{j-1} \leq x < c_j \\ 0, & \text{kai } x \geq c_k \end{cases}, \quad (3.8)$$

kur $\frac{f_j}{nh}$ - braižomo stačiakampio aukštis nurodytame intervale.

O pats grafikas sudaromas atliekant šiuos žingsnius:

1. Abscisių ašyje atidedami intervalai;

2. Kiekviename intervale braižomas stačiakampis, kurio aukštis proporcingas pakliuvusiam į intervalą santykiniam dažniui $\frac{f_j}{nh}$, kur f_j - reikšmės dažnis [6].

Svarbu paminėti, kad visų stačiakampių plotų suma turėtų būti lygi 1. Tuomet histograma turėtų tikimybinę interpretaciją [6].

3.4. Nuostolio metrikos prognozėms įvertinti

Atlikus susijusių darbų analizę paaiškėjo, kad dažniausiai naudojamos metrikos įvairiems modeliams įvertinti prognozuojant kriptovaliutų kainas ir jų pokyčius yra RMSE, MSE, MAE. MAE kaip geras rodiklis vidutinio modelio našumo, o RMSE dirba gerai su nuokrypiais, labiau bausdamas už dideles paklaidas. Tačiau ganėtinai dažnai autoriai naudoja ir determinacijos koeficientą R^2 , ypač kai tyrimo objektas - kintamumas [34][3].

Vidutinė absoliuti paklaida (MAE) yra absoliučių skirtumų tarp faktinių ir prognozuojamų reikšmių suma. Ši metrika neatsižvelgia į tendenciją arba kryptį, t.y. teigiamą ar neigiamą. Metrika aprašoma tokia formule:

$$\text{MAE} = \frac{1}{n} \sum_{t=1}^n |(r_t - \hat{r}_t)|, \quad (3.9)$$

kur (r_t) - aktuali reikšmė, \hat{r}_t - prognozės reikšmė.

Vidutinė kvadratinė paklaida (MSE) įvertina paklaidų kvadratų vidurkį, t.y. vidutinį kvadratinį skirtumą tarp faktinių ir prognozuojamų reikšmių. Remiantis Emmanuelio Pintelaso ir jo bendraautorių išvalgomis šiame darbe MSE bus naudojama tik RMSE reikšmei apskaičiuoti [28]. MSE aprašoma taip:

$$\text{MSE} = \frac{1}{n} \sum_{t=1}^n (r_t - \hat{r}_t)^2 \quad (3.10)$$

kur (r_t) - aktuali reikšmė, \hat{r}_t - prognozės reikšmė.

Šakninė vidutinė kvadratinė paklaida (RMSE) yra vidutinės kvadratinės paklaidos kvadratinė šaknis. Ši metrika padeda įvertinti standartinį nuokrypį. Galima apskaičiuoti taip:

$$\text{RMSE} = \sqrt{\text{MSE}} \quad (3.11)$$

Šakninė vidutinė kvadratinė paklaida (RMSE) ir vidutinė absoliuti paklaida (MAE) parodo, kiek tikslus yra modelis ir koks yra jo suprognozuotų reikšmių nukrypimas nuo faktinių reikšmių. Kuo RMSE ir MAE reikšmės yra mažesnės, tuo geresnį rezultatą rodo modelis. Gerai sudarytame modelyje RMSE rezultatas visada bus didesnis arba lygus MAE.

Determinacijos koeficientas (R^2) - tai atsitiktinių dydžių tiesinio ryšio matas. Šio koeficiento reikšmė gali svyruoti nuo 0 iki 1, ir kuo ji yra aukštesnė, tuo geresnis ir tinkamesnis modelis yra. R^2 formule yra tokia:

$$R^2 = 1 - \frac{\sum (r_t - \hat{r}_t)}{\sum (r_t - \bar{r}_t)}, \quad (3.12)$$

kur (r_t) - aktuali reikšmė, \hat{r}_t - prognozės reikšmė.

Norint įvertinti modelio prognozavimo efektyvumą iš skirtingų perspektyvų bus panaudotas dar vienas vertinimo kriterijus - MDA, atsakantys už krypties tikslumą. Jis lygina prognozuojamą kryptį (aukšтын arba žemyn) su faktine realizuota kryptimi ir yra apskaičiuojamas taip:

$$MDA = \frac{1}{n} \sum_{t=1}^n a(t) \times 100\%, a(t) = \begin{cases} 1, & \text{kai } (r_{t+1} - r_t)(\hat{r}_{t+1} - r_t) \geq 0 \\ 0, & \text{kai } (r_{t+1} - r_t)(\hat{r}_{t+1} - r_t) < 0 \end{cases}, \quad (3.13)$$

kur (r_t) - aktuali reikšmė, \hat{r}_t - prognozės reikšmė.

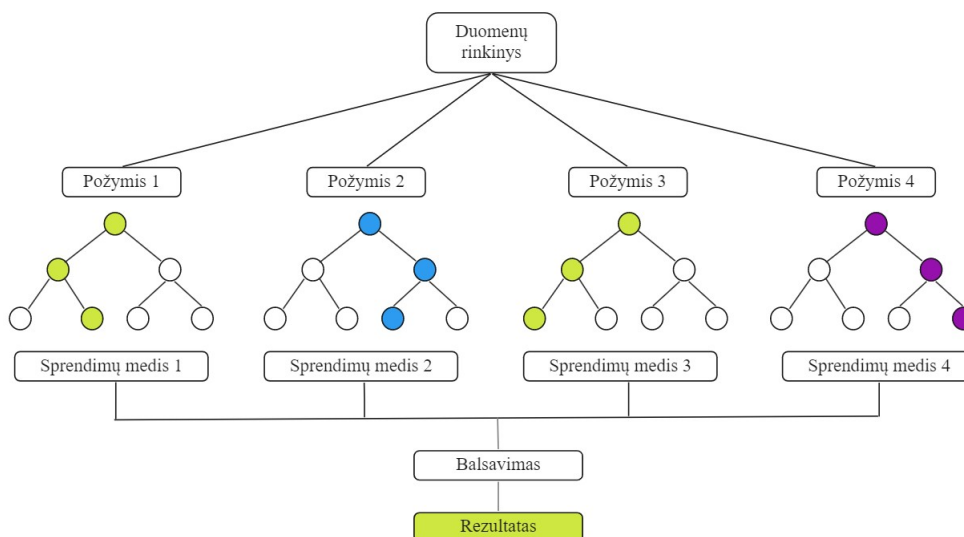
Svarbu paminėti, kad naudojamų pradinių duomenų diapazonas taip pat yra svarbus nustatant, ar nurodyta RMSE reikšmė yra „žema“, ar ne. Pavyzdžiui, Bitcoin kriptovaliutų atveju RMSE rodikliai gali būti žymiai didesni, nei Litecoinin kriptovaliutų atveju ir tai bus visiškai normalu.

3.5. Random Forest

Atsitiktinio miško (angl. Random Forest) algoritmas yra universalus mašininio mokymosi algoritmas, kuris buvo pristatytas Leo Breimanu ir Adele Cutler. Jis sujungia dvi pagrindines idėjas: Breimano maišymo metodą (angl. bagging) ir atsitiktinį funkcijų pasirinkimo metodą, kurį pasiūlė Tin Kam Ho. Algoritmas labai dažnai naudojamas klasifikavimo, regresijos ir klasterizacijos problemoms įvairiose srityse spręsti: nuo farmacijos iki akcijų biržos.

Algoritmo esmė – naudoti sprendimų medžių ansamblį. Ansambliu įprastai vadina kelių mokymosi algoritmų derinį. Tai paaiškina ir pats algoritmo pavadinimas, kurį galima traktuoti kaip sprendimo medžių mišką. Ansambliai sujungia kelias hipotezes, vėliau suformuojant vieną geriausią iš jų (2 pav.).

Atsitiktiniu mišku galima vadinti klasifikatorių, kuriame yra kelių sprendimų medžių, sudarytų iš atsitiktiniu būdu parinktų įrašų ir kintamųjų, vidurkiai. Kitaip tariant, šio klasifikatoriaus sudarymo procesas ir yra sprendimo medžių maišymas (angl. bagging) su atsitiktiniu požymių parinkimu.



2 pav. RF veikimo principas

Medžių yra tiek, kiek yra pogrupių (arba klasių). Jie sudaro sprendimo medžių mišką, o galutinės prognozės gali būti sudarytos apskaičiuojant visų sprendimų medžių prognozių vidurkį:

$$a(x) = \frac{1}{n} \sum_{t=1}^n a_t(x), \quad (3.14)$$

kur $a_i(x)$ - sprendimo medis, sudarytas kiekvienam iš pogrupių, n - medžių kiekis.

Kiekvienas sprendimų medis suformuojamas naudojant tokius indikatorius, kaip prieaugio santykį arba kiekvieno požymio Gini indeksą. Visi jie padeda apskaičiuoti kiekvieno požymio svarbą.

Gini indeksas, dar žinomas kaip Gini priemaiša, apskaičiuoja tikimybę, kuri nurodo, ar atsitiktinai parinktas požymis yra parinktas neteisingai. Jis apskaičiuojamas atėmus kiekvieno požymio tikimybių kvadrato sumą iš vieneto:

$$G = 1 - \sum_{i=1}^n (p_i)^2, \quad (3.15)$$

kur n - požymių kiekis, $p(i)$ - tikimybė pasirinkti duomenų tašką su požymiu i .

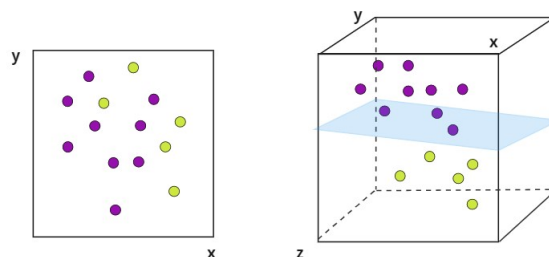
Kuriant sprendimų medį pirmenybė teikiama požymiams, turintiems mažiausią Gini indekso vertę.

3.6. SVM

Dar vienas prižiūrimas mašininio mokymosi algoritmas, tinkantis spręsti tiek regresijos, tiek klasifikavimo problemas yra atraminių vektorių mašinų algoritmas, sutrumpintai SVM.

SVM algoritmas perkelia apmokymo duomenų rinkinį į didesnę dimensiją ir stengiasi surasti hiperplokštumą, kuri galėtų atskirti duomenų rinkinio taškus taip, kad atstumas tarp dviejų klasių (teigiamos ir neigiamos) būtų kiek įmanoma didesnis, o klaidingo klasifikavimo atveju - kiek įmanoma mažiau. Artimiausi hiperplokštumos taškai vadinami atraminiais vektoriais. Tad galima teigti, jog algoritmo tikslas - surasti didžiausią atstumą tarp šių atraminių vektorių ir hiperplokštumos.

Perkelimas į didesnę dimensiją priklauso nuo to, ar duomenų rinkinys yra tiesiškai atskiriamas ar ne. Jei duomenys nėra tiesiškai atskiriami, tuomet jie bus perkeltami į vis didesnę dimensiją tol, kol juos bus galima atskirti tiesia linija (3 pav.).



3 pav. Duomenys prieš perkėlimą į didesnę dimensiją (kairėje), ir po perkėlimo (dešinėje).

Tam, kad surasti hiperplokštumą, duomenys turi būti suskirstyti į kategorijas, o vėliau patransformuojami. Transformacijai naudojama matematinė funkcija, žinoma kaip branduolio funkcija arba branduolio gudrybė (angl. kernel trick). Ji dažniausiai aprašoma taip:

$$K(x,y) = (\Phi(x) \times \Phi(y)), \quad (3.16)$$

ir nusako, kad kiekvienai transformuoto vektoriaus $\Phi(x)$ koordinatei yra atitikimas x žemesnėje dimensijoje.

Egzistuoja įvairių branduolio funkcijų. Populiariausi yra polinominio ir radialinės bazės funkcijos (angl. radial basis function) branduoliai:

$$K(x, y)_{\text{polinominis}} = (1 + x * y)^b, \quad (3.17)$$

kur d yra polinominis laipsnis, o x ir y - vektoriai.

$$K(x, y)_{RBF} = e^{(-\gamma \|x-y\|^2)}, \quad (3.18)$$

kur x ir y - vektoriai, γ - dispersija.

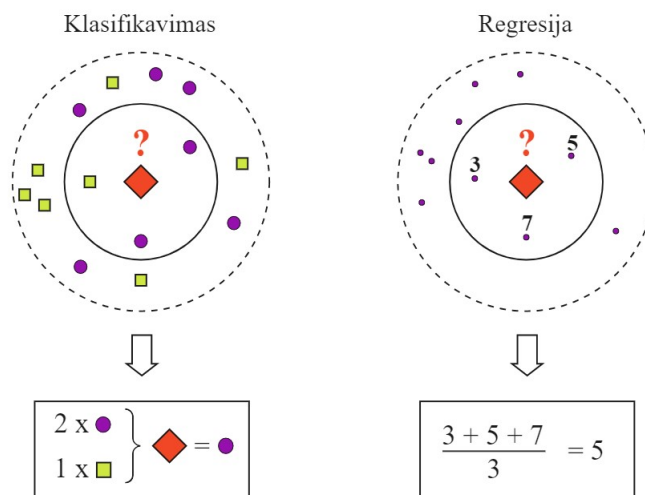
Šios funkcijos gali būti naudojamos duomenų transformavimui, kad duomenys būtų perkelti į kitą, didesnę, dimensiją.

SVM yra dažniausiai naudojamas įvairiose srityse: nuo rašysenos atpažinimo iki genų klasifikavimo. Tačiau taip pat laikomas daug žadančia alternatyva prognozavimui finansų srityje [20].

3.7. KNN

K-arčiausių kaimynų modelis, taip pat žinomas kaip KNN arba k-NN, yra neparаметrinis, prižiūrimas mokymosi metodas, kuris naudoja atstumą atliekant klasifikavimo arba prognozavimo užduotį. KNN metodas veikia pagal prielaidą, kad panašūs objektai yra arti vienas kito.

Naudojant šį metodą klasifikavimo užduočiai spręsti, objektas bus priskiriamas tai klasei, kuri yra labiausiai paplitusi tarp šio elemento kaimynų, kurių klasės jau žinomos. Tuomet jei KNN naudojamas regresijos užduočiai spręsti, objektui bus priskiriama vidutinė objektų, esančių arčiausiai jo, reikšmė, kurių reikšmės jau žinomos (4 pav.).



4 pav. KNN klasifikavimo ir regresijos pavyzdžiai.

Atstumui tarp dviejų objektų nustatyti dažniausiai naudojamas Euklido algoritmas, metrika, rodanti atstumą tarp dviejų taškų Euklido erdvėje, apskaičiuotą pagal Pitagoro teoremą. Euklido atstumui apskaičiuoti naudojama ši formulė:

$$d(x, y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2}, \quad (3.19)$$

kur x ir y - taškai, tarp kurių reikia rasti atstumą.

Dar vienas svarbesnių žingsnių implementuojant KNN modelį yra optimaliausios k reikšmės nustatymas. k reikšmė apibrėžia, kiek kaimynų bus patikrinta ir panaudota prognozavimui. Ją galima parinkti intuityviai arba kryžminio patikrinimo pagalba. Tačiau verta atkreipti dėmesį, kad kuo daugiau duomenų, tuo didesnis triukšmas gali būti, ir tuo didesnė k reikšmė rekomenduojama.

Apibendrinant, KNN algoritmo veikimo principas yra ganėtinai paprastas:

1. parinkti k reikšmę pagal turimą duomenų rinkinį;
2. kiekvienam duomenų taškui apskaičiuoti atstumą, o rezultata išsaugoti į naują indeksuotą duomenų rinkinį;
3. surūšiuoti naujo rinkinio atstumus didėjančia tvarka;
4. apskaičiuoti vidurkį pirmų naujojo rinkinio k reikšmių.

3.8. LSTM

Trumpalaikės-ilgalaikės atminties neurotinis tinklas (angl. Long Short Term Memory) arba sutrumpintai LSTM, yra nuolat tobulinamas grįžtamojo ryšio neuroninis tinklas, kuris buvo sukurtas siekiant išspręsti ilgalaikes priklausomybės problemas.

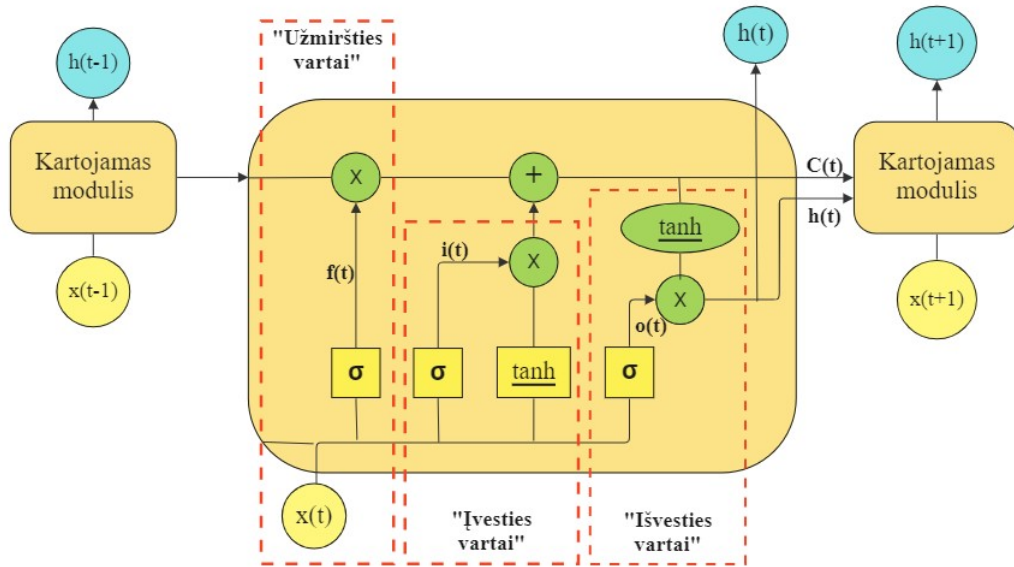
LSTM sąvokai suprasti reikia susipažinti su tokiomis sąvokomis kaip dirbtinis neuroninis tinklas ir pasikartojantis neuroninis tinklas.

Dirbtinis neuroninis tinklas yra daugiasluksnė sujungtų neuronų struktūra, įkvėpta biologinių neuroninių tinklų. Tai ne vienas algoritmas, o skirtingų algoritmų derinys, leidžiantis atlikti sudėtingas operacijas su duomenimis.

Pasikartojantis neuroninis tinklas - neuroninių tinklų klasė, skirta dirbti su laiko duomenimis. RNN neuronai turi ląstelės būseną / atmintį, o įvestis apdorojama pagal šią vidinę būseną, kuri pasiekama per kilpas neuroniniame tinkle. RNN turi pasikartojančius „tanh“ sluoksnio modulius, leidžiančius saugoti informaciją, tačiau neilgam, tad kitais atvejais ir yra reikalingas LSTM modelis.

LSTM vadina išskirtiniu pasikartojančiu neuroniniu tinklu, kuris gali išmokti ilgalaikes duomenų priklausomybes. Tai pasiekama dėl to, kad pasikartojančiame modelio modulyje yra keturi sluoksniai, sąveikaujantys vienas su kitu.

5 paveikslėlyje pavaizduoti keturi neuroninio tinklo sluoksniai geltonais langeliais, taškų operatoriai – žaliais apskritimais, įvestis – geltonais apskritimais, o ląstelės būseną – mėlynais apskritimais (5 pav.). LSTM modulis turi langelio būseną ir tris vartus, kurie suteikia jiems galimybę pasirinktinai mokytis, išjungti arba saugoti informaciją iš kiekvieno modulio. Ląstelės būseną LSTM padeda informacijai pereiti per vienetus be pokyčių, leidžiant tik keletą linijinių sąveikų. Kiekvienas įrenginys turi įvesties, išvesties ir užmiršimo vartus, kurie gali pridėti arba pašalinti informaciją prie ląstelės būsenos.



5 pav. LSTM skirtingų sluoksnių sąveikos diagrama.

Pirmas žingsnis yra nuspręsti, kurią informaciją iš ankstesnės ląstelės būsenos reikia pamiršti. Tai atlieka užmiršimo vartai sigmoidinės σ funkcijos pagalba. Jie žiūri į prieš tai buvusios ląstelės išvestį h_{t-1} ir dabartinę įvestį x_t ir gražina 0, jei informaciją reikia pašalinti, ir 1 - jei palikti:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (3.20)$$

Kitame žingsnyje sprendžiama, kuri nauja informacija saugosis esamoje ląstelės būsenoje. Tą kontroliuoja įvesties vartai, naudodami dviejų sluoksnių filtrą. Pirmas sluoksnis sprendžia, kurias reikšmes reikia atnaujinti:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \quad (3.21)$$

antras sukuria naujų \tilde{C}_t reikšmių vektorių, kurias galima pridėti prie ląstelės būsenos:

$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C), \quad (3.22)$$

Kai jau yra žinoma, kokia informacija turi būti pašalinta, o kokia pridėta, reikia pakeisti seną ląstelės būseną nauja. Tam reikia seną ląstelės būseną C_{t-1} padauginti iš to, ką reikia pamiršti f_t , vėliau pridėdant naujus duomenis $i_t \tilde{C}_t$:

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t, \quad (3.23)$$

Galiausiai, išvesties vartai nusprendžia, kokia informacija turi būti perduota į kitą paslėptą būseną. Tai atliekama dviem žingsniais. Pirmas, sigmoidinio sluoksnio pagalba bus nuspręsta, kuri informacija turi būti išvesta:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \quad (3.24)$$

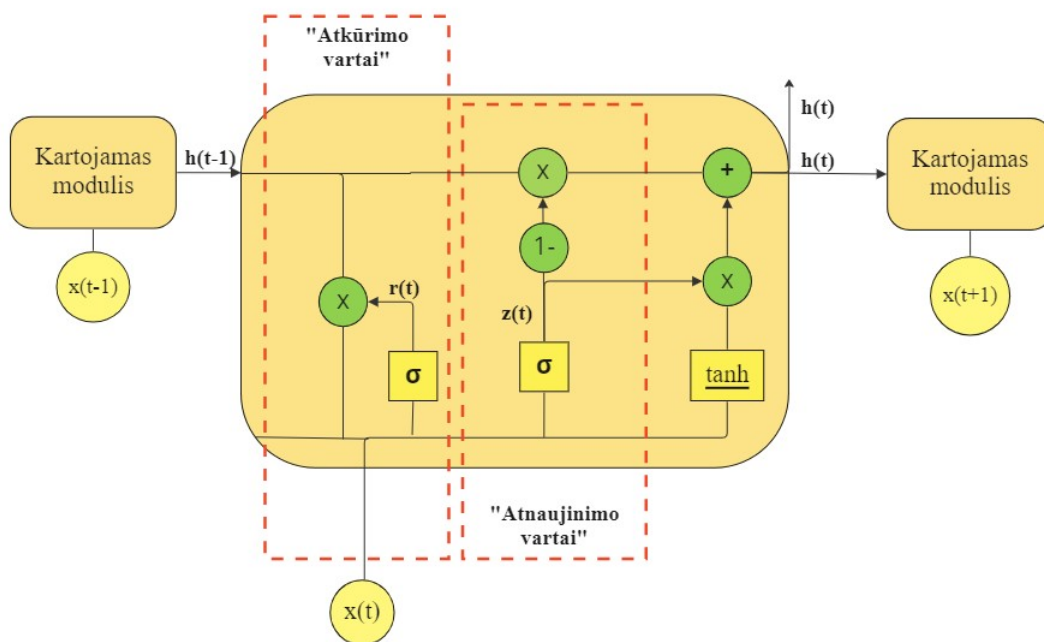
Toliau, ląstelės būsenos reikšmės praeina per kitą sluoksnį, kuris ir padeda išvesti tik reikalingą informaciją:

$$h_t = o_t \tanh(C_t), \quad (3.25)$$

3.9. GRU

Dar vienas algoritmas naudojantis vartų mechanizmą išiminimo procesui valdyti yra GRU. GRU (angl. gated recurrent unit) yra naujesnės kartos pasikartojantys neuroninių tinklų modelis, kuris dažnai pristatomas kaip supaprastintas LSTM variantas.

GRU nebeturi ląstelės būsenos, o taip pat turi ne tris, kaip LSTM, o du vartus - atkūrimo vartus (angl. reset gate) ir atnaujinimo vartus (angl. update gate) (6 pav.). Šie vartai nusprendžia, kokia informacija turi būti perduodama į išvestį. Ypatinga jų savybė yra ta, kad jie gali būti išmokyti saugoti seną informaciją, neprarandant jos per laiką arba nepašalinant informacijos, kuri nėra svarbi prognozei.



6 pav. GRU skirtingų sluoksnių sąveikos diagrama.

Atnaujinimo vartai yra labai panašūs į LSTM modelio užmiršties ir įvesties vartus. Jie nusprendžia, kokia informacija turi būti perduota tolimesniam algoritmo vykdymui.

Atkūrimo vartai - vartai, nusprendžiantys, kokia ir kiek senos informacijos turi būti pašalinta arba praignoruota.

Tai gi pirma GRU algoritmo žingsnis yra nustatyti, kokia sena informacija bus reikalinga kitiems žingsniams sigmoidinės funkcijos pagalba. Čia dirba atkūrimo vartai. Jie žiūri į seną išvestį h_{t-1} ir dabartinę įvestį x_t ir grąžina 0, jei informaciją reikia pašalinti, ir 1 - jei palikti:

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \quad (3.26)$$

Toliau, tos pačios funkcijos pagalba yra nustatoma, kiek informacijos turi būti pamiršta arba praignoruota. Pati formulė yra tokia pati:

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r). \quad (3.27)$$

Taip pat reikia nustatyti, kas bus siunčiama į išvestį. Tad tangento aktyvacijos funkcija nusprendžia, kokia informacija kandidatuoja perdavimui į išvesti. Kitaip sakant, šis žingsnis atsakingas už einamosios ląstelės atminties sukūrimą:

$$\tilde{h}_t = \tanh(W_C[h_{t-1}, x_t] + b_C), \quad (3.28)$$

Ir toliau jau apskaičiuojamas išvesties rezultatas, t.y. einamosios atminties vektorius:

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t, \quad (3.29)$$

4. Laiko eilučių modelių eksperimentinis tyrimas

Šiame tiriamojo darbo skyriuje bus aprašomas kriptovaliutų kintamumo laiko eilučių modelių eksperimentinio tyrimas. Šiame skyriuje bus įvertintas kiekvienos laiko eilutės normalumas: bus atlikti Lilliefors ir Shapiro-Wilk testai, o taip pat sudaryti kvantilių, empirinės tankio funkcijos ir autokoreliacijos grafikai. Vėliau, kiekviena iš laiko eilučių bus ištestuota ir panaudota sudarant prognozavimo modelius. Tyrimo eigoje gauti rezultatai bus palyginami.

4.1. Programavimo kalba ir naudojamos bibliotekos

Šiame poskyryje bus aprašomos visos pasirinktos bibliotekos, jų nauda bei pasirinkimo priežastys.

Python – tai galinga programavimo kalba, kurios kodas gana lengvai skaitomas ir rašomas. Python pasižymi padidintu produktyvumu: kadangi nėra kompiliavimo žingsnio, redagavimo-testavimo ciklas yra neįtikėtinai greitas, todėl ši programavimo kalba suteikia galimybę susitelkti į sprendžiamą problemą, o ne kodo tvarkymą. Python palaiko daugelį modulių ir paketų, žymiai palengvinančių darbą, nes dažnai jau egzistuojančiuose paketuose esančios funkcijos padeda ne tik sutaupyti daug laiko ir pastangų, bet ir gauti kokybišką rezultatą. Žemiau išvardintos Python kalbai skirtos bibliotekos puikiai padėtų ištirti kriptovaliutų kainų laiko eilučių prognozavimą.

„**Pandas**“ – tai atviro kodo Python programavimo kalbai skirta biblioteka. Pagrindinė paskirtis yra darbas su lentelėmis, Excel ir CSV failais. „Pandas“ turi galingų funkcijų, kurios supaprastina darbą su duomenų rinkiniais.

„**Statsmodels**“ yra Python modulis, kuriame pateikiamos klasės ir funkcijos, skirtos daugelio skirtingų statistinių modelių įvertinimui, taip pat statistiniams testams atlikti ir statistinių duomenų tyrinėjimui.

„**Sklearn**“ yra Python biblioteka, kurioje pateikiamos klasės ir funkcijos, suteikiančios galimybę pasinaudoti RF, KNN ir SVM modeliais, juos ištirti ir palyginti.

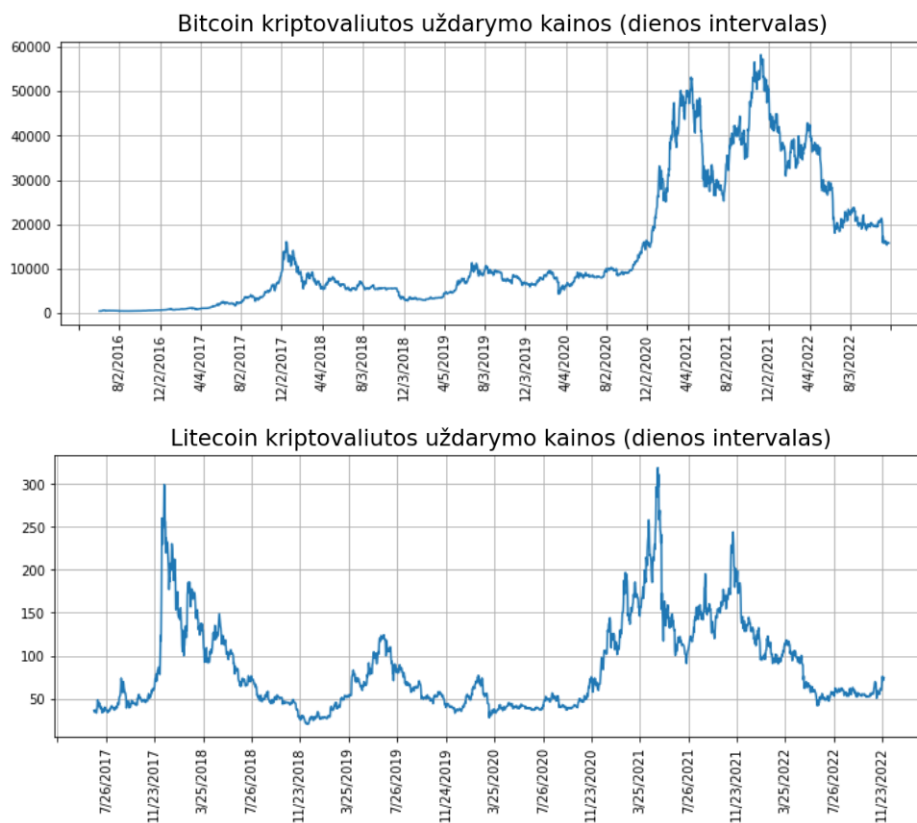
„**Tensorflow**“ yra mašininiam mokymuisi skirtas Python modulis, suteikiantis galimybę pasinaudoti LSTM ir GRU modeliais, juos ištirti ir palyginti.

„**NumPy**“ – tai mokslinių skaičiavimų įrankių biblioteka. Buvo naudojama vidutinio kvadratinio nuokrypio skaičiavimui, diferencijavimui, darbui su masyvais ir matricomis.

„**Matplotlib**“ yra išsami biblioteka, skirta kurti statines, animacines ir interaktyvias vizualizacijas. Šiame darbe ši biblioteka buvo naudojama darbui su grafikai.

4.2. Normalumo tikrinimas

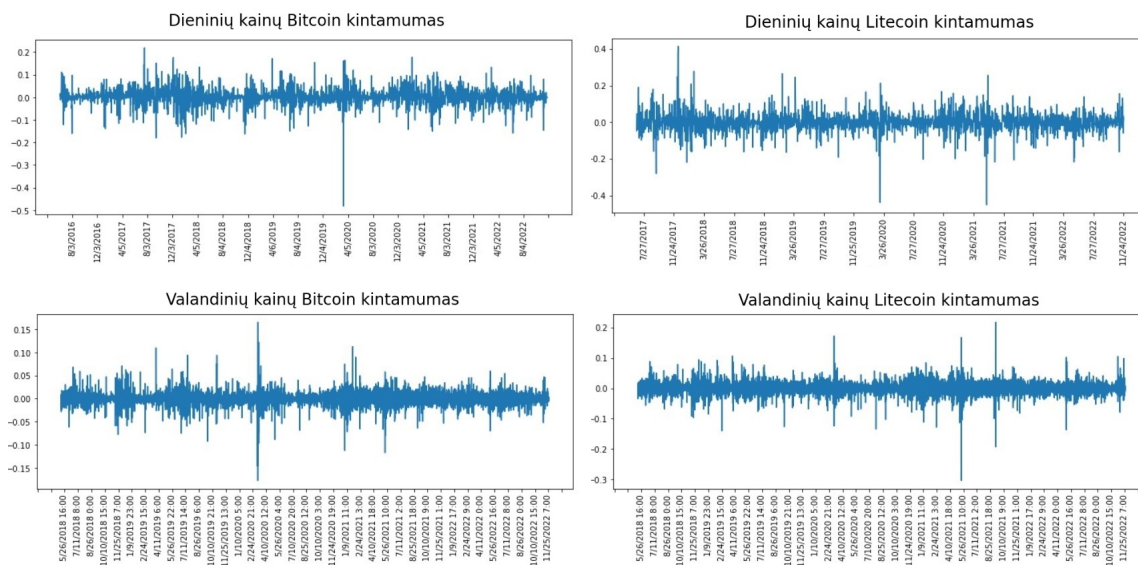
Kriptovaliutų rinka išgyvena gana sudėtingą nestabilų laikotarpį. Tą galima matyti atvaizdavirus šiame darbe nagrinėjamų kriptovaliutų laiko eilutes (7 pav.).



7 pav. Bitcoin ir Litecoin kriptovaliutų uždarymo kainų laiko eilutės.

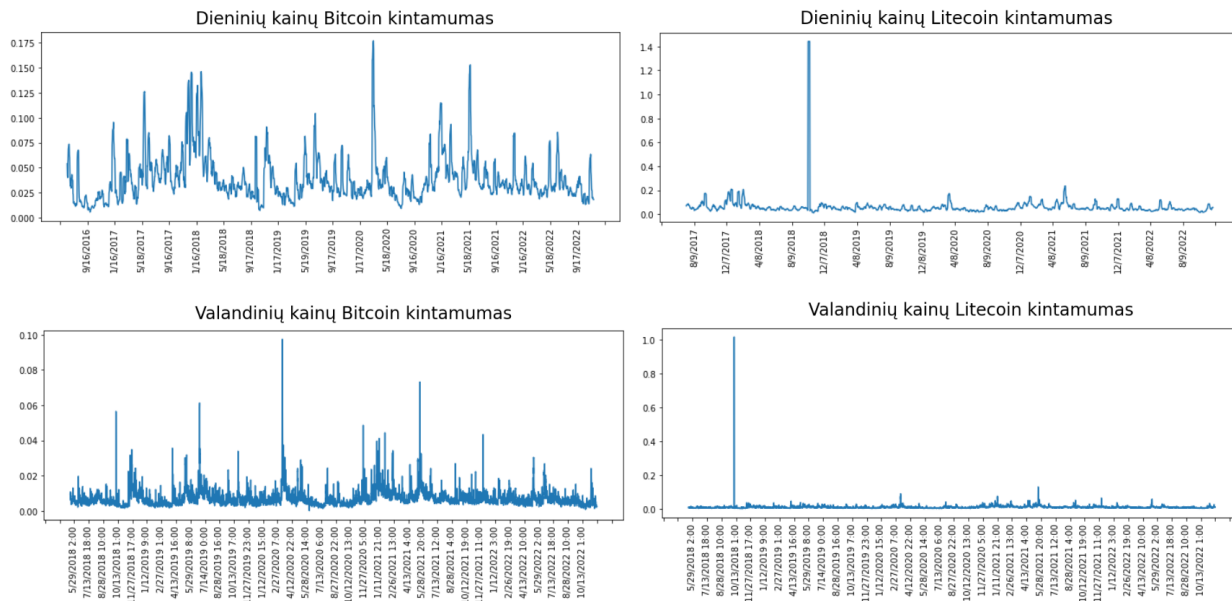
Iš atvaizduotų laiko eilučių grafikų galima pastebėti, kad abi kriptovaliutos 2020-2021 metų laikotarpyje pradėjo smarkiai augti, tačiau vėliau pasiekus piką 2021 metais, pradėjo taip pat staigiai kristi. Taip pat, nors 2017-2018 metais Litecoin kriptovaliuta turėjo aiškiai matomų šuolių, galima sakyti, kad tiek Bitcoin, tiek Litecoin laiko eilučių kreivės yra ganėtinai panašios - krentančios ir kylančios tendencijos sutampa.

Vienas iš pakaitinių kintamųjų kintamumui prognozuoti - esamos ir ankstesnės kainos santykio logaritmas pagal 3.1 formulę (8 pav.).



8 pav. Bitcoin ir Litecoin kriptovaliutų logaritmimo kintamumo laiko eilutės.

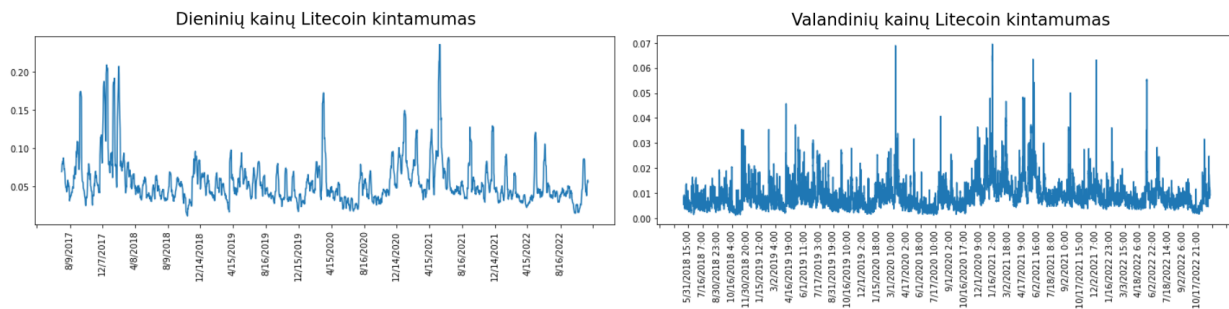
Antras kintamasis - kintamumo indeksas pagal Yang Zhang pagal 3.2 formulę. Pasak J. Shu ir J. E. Zhang, jis geras tuo, kad neignoruotų šuolių ir yra nepriklausomas nuo dreifo, tad gali pateikti tikslų įvertinimą [33].



9 pav. Bitcoin ir Litecoin kriptovaliutų Yang Zhang kintamumo laiko eilutės.

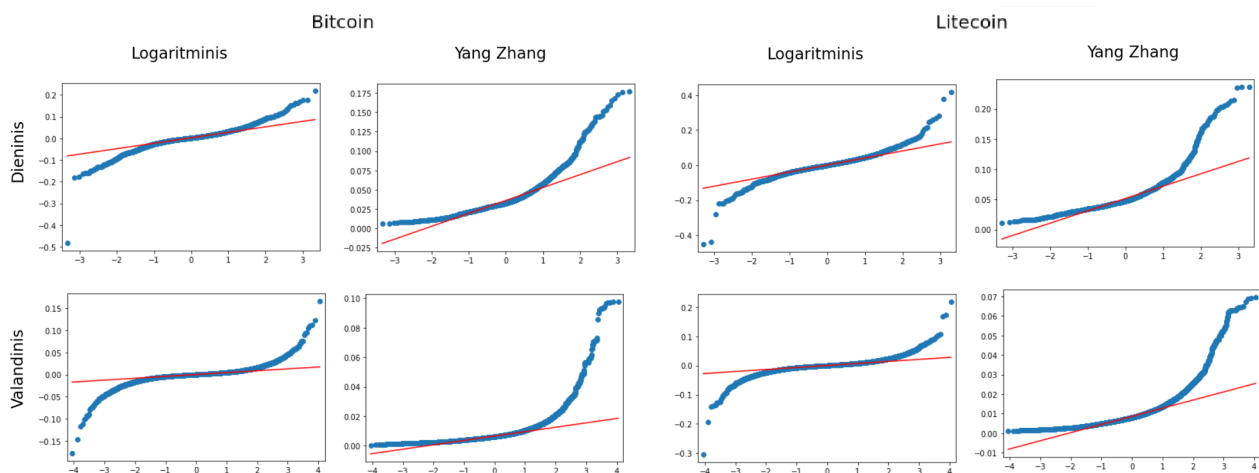
Iš žemiau sudarytų kriptovaliutų Yang Zhang kintamumo grafikų, ypatingai Litecoin atveju vizualiai matosi, jog yra anomalinių riktų (angl. outliers), kurie gali būti pašalinti tam, kad duomenų rinkiniai būtų kokybiškesni (9 pav.).

Pašalinus riktas, Litecoin kintamumo eilutės atrodo taip (10 pav.):



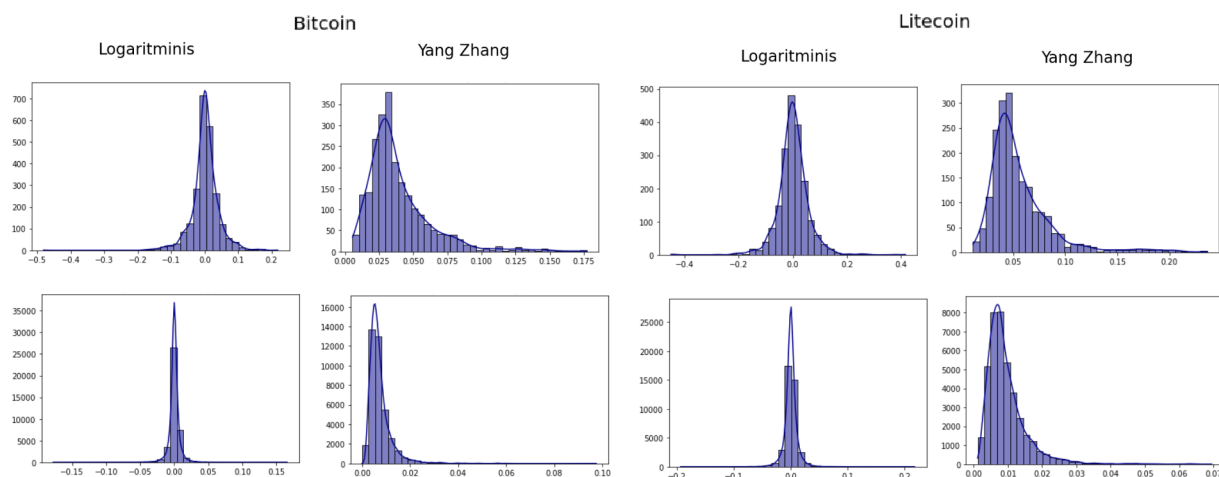
10 pav. Litecoin kriptovaliutų Yang Zhang kintamumo laiko eilutės po riktų šalinimo.

Normalumui įvertinti buvo sudaryti kvantilių grafikai. Jie pavaizduoti žemiau (11 pav.). Galiama matyti, kad vizualiai išskirčių yra žymiai daugiau, nei normaliojo skirstinio atveju. Todėl galima daryti prielaidą, kad kriptovaliutų kainų kintamumas nėra pasiskirstęs pagal normalųjį skirstinį.



11 pav. Bitcoin ir Litecoin kriptovaliutų QQ grafikai.

Tą patvirtina ir empirinės tankio funkcijos grafikai (12 pav.). Vizualiai matosi, kad kairioji ir dešinioji skirstinių „uodegos“ yra labai stiprios, kas nusako, jog kriptovaliutų graža yra toli nuo normaliai pasiskirstytų reikšmių.



12 pav. Bitcoin ir Litecoin kriptovaliutų empirinės tankio funkcijos grafikai.

Buvo atliktas ir tiesioginis šių duomenų patikrinimas Lilliefors normalumo testu. Lentelės p stulpelio esančios reikšmės nurodo p-reikšmes. Pagal Lilliefors testą, jei reikšmingumo lygis yra 0,05 ir p-reikšmė mažesnė už šį lygį – nulinė hipotezė, kad imtis yra normalaus skirstinio, gali būti atmesta.

Kriptovaliuta	Dažnumas	Kintamumas	p-reikšmė
Bitcoin	Dienos	Logaritminis	0.0009999999999998899
		Yang Zhang	0.0009999999999998899
	Valandos	Logaritminis	0.0009999999999998899
		Yang Zhang	0.0009999999999998899
Litecoin	Dienos	Logaritminis	0.0009999999999998899
		Yang Zhang	0.0009999999999998899
	Valandos	Logaritminis	0.0009999999999998899
		Yang Zhang	0.0009999999999998899

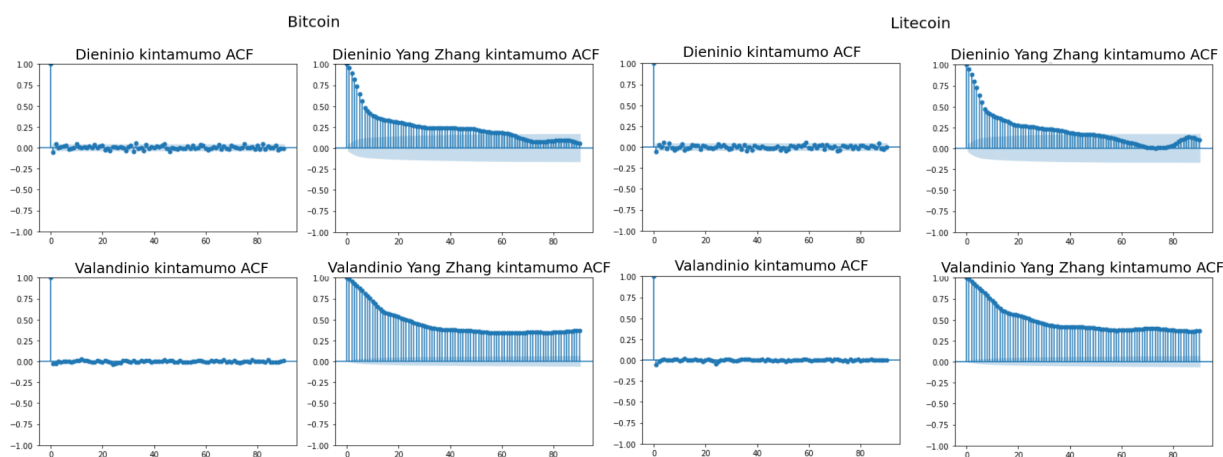
1 lentelė. Skirtingų Bitcoin ir Litecoin kriptovaliutų laiko eilučių Lillfors testo rezultatai

Tokias pat išvadas galima padaryti ir po duomenų patikrinimo Shapiro-Wilk testu: p-reikšmė yra mažiau nei 0.05, tad nulinė hipotezė, kad imtis yra normalaus skirstinio, gali būti atmesta.

Kriptovaliuta	Dažnumas	Kintamumas	p-reikšmė
Bitcoin	Dienos	Logaritminis	1.6269552392531982e-34
		Yang Zhang	0.0
	Valandos	Logaritminis	1.1210387714598537e-44
		Yang Zhang	0.0
Litecoin	Dienos	Logaritminis	1.8070833277789877e-30
		Yang Zhang	0.0
	Valandos	Logaritminis	2.802596928649634e-45
		Yang Zhang	0.0

2 lentelė. Skirtingų Bitcoin ir Litecoin kriptovaliutų laiko eilučių Shapiro-Wilk testo rezultatai

Papildomai buvo sudaryti ir autokoreliacijos grafikai, kurie yra pristatyti žemiau (13 pav.).



13 pav. Bitcoin ir Litecoin kriptovaliutų laiko eilučių ACF grafikai.

Grafikai rodo, kad pirmojo kintamojo, t.y. logaritminio kintamumo, duomenys turi itin mažą autokoreliaciją. Todėl šių eilučių prognozavimo rezultatai gali būti ganėtinai prasti. Tačiau antrojo kintamojo, t.y. Yang Zhang kintamumo, duomenys, atvirkščiai, turi didelę autokoreliacinę priklausomybę. Todėl prognozavimo modeliai, sukurti remiantis ankstesnėmis šio kintamojo duomenimis, gali turėti didesnę prognozavimo galią.

Taip pat duomenų stacionarumui patikrinti buvo panaudotas Dickey–Fuller testas, kuris parodė, kad abiejų kriptovaliutų kintamumo laiko eilutės yra stacionarios.

4.3. Modelių taikymas

Kiekvienas prognozavimui naudojamas modelis turi būti apmokytas. Šiame darbe bus bandoma atsižvelgti į praeitas 7, dienos dažnumo eilučių atveju, arba 12, valandos dažnumo eilučių atveju, reikšmes, kurios vėliau bus palygintos su 8 arba 13 reikšmėmis. Praeitose reikšmės dar vadinamos prediktoriais (angl. predictors), o tos, su kuriomis bus palyginta, - atsakais (angl. responses). Vėliau, duomenų rinkiniai bus suskirstyti į treniravimosi ir testavimo rinkinius 70/30 santykiu.

Dauguma nagrinėjamų modelių turi hiperparametrus, kurie gali stipriai įtakoti prognozuojamą rezultatą. Todėl "tinklelio paieškos" metodo ir k-dalių kryžminės patikros (angl. k-fold cross-validation) pagalba yra gaunamos visos įmanomos modelių hiperparametrų kombinacijos, vėliau įvertinamas kiekvieno modelio kiekvienos kombinacijos efektyvumas ir pagal tai išrenkama geriausia modelio hiperparametrų kombinacija, su kuria ir bus dirbama toliau.

4.3.1. SVM modelis

SVM modelis buvo sudarytas naudojantis sklearn biblioteka. Šis modelis turi 2 reikšmingus hiperparametrus: reguliavimo parametą C ir nejausrios netekties funkcijos parametą ϵ . Kryžmeninės validacijos metodo pagalba bus išrinkta geriausia C ir ϵ hiperparametrų kombinacija. Kai kombinacija bus rasta, modelis bus treniruojamas ir pritaikomas $\text{fit}()$ metodo pagalba.

Kiekvienai iš laiko eilučių bus ieškomas atskiras geriausiai tinkantis modelis. Tokiu būdu kryžminio patikrinimo pagalba žemiau pristatomos geriausios kombinacijos kiekvienai iš turimų laiko eilučių:

Kripto valiuta	Dažnumas	Kintamumas	Kombinacija
Bitcoin	Dienos	Logaritminis	$C = 0.01$, $\epsilon = 0.001$
		Yang Zhang	$C = 1.0$, $\epsilon = 0.001$
	Valandos	Logaritminis	$C = 0.01$, $\epsilon = 0.001$
		Yang Zhang	$C = 0.01$, $\epsilon = 0.001$
Litecoin	Dienos	Logaritminis	$C = 0.01$, $\epsilon = 0.01$
		Yang Zhang	$C = 0.01$, $\epsilon = 0.01$
	Valandos	Logaritminis	$C = 1.0$, $\epsilon = 0.01$
		Yang Zhang	$C = 0.01$, $\epsilon = 0.001$

3 lentelė. SVM modelio skirtingų Bitcoin ir Litecoin kripto valiutų laiko eilučių geriausios kombinacijos

Gavus geriausias kombinacijas modeliai buvo sudaryti, o treniravimosi ir prognozavimo rezultatai palyginti. Rezultatai atvaizduojami dviejose lentelėse žemiau.

Dažniausiai pagal treniravimosi rezultatus galima nustatyti, ar modelis prognozuos gerai. SVM modelio atveju, galima matyti, kad geriausi rezultatai pasiekti prognozuojant Yang Zhang kintamumo eilutes. Pagal gautas metrikas galima daryti išvadas, kad Yang Zhang eilučių atveju modelis buvo sudarytas gerai. R^2 ir MDA metrikų rezultatai yra arti vienetui, o tai reiškia, kad suprognozuoti duomenys gali neblogai atitikti realius duomenis ir judėjimo kryptį.

Kripto valiuta	Dažnumas	Kintamumas	MAE	R^2	RMSE	MDA
Bitcoin	Dienos	Logaritminis	0.026	0.006	0.040	0.524
		Yang Zhang	0.003	0.916	0.007	0.929
	Valandos	Logaritminis	0.005	-0.001	0.008	0.527
		Yang Zhang	0.000	0.966	0.001	0.963
Litecoin	Dienos	Logaritminis	0.038	0.007	0.056	0.551
		Yang Zhang	0.005	0.924	0.009	0.913
	Valandos	Logaritminis	0.007	0.003	0.011	0.531
		Yang Zhang	0.000	0.971	0.001	0.957

4 lentelė. SVM modelio skirtingų Bitcoin ir Litecoin kriptovaliutų laiko eilučių treniravimosi rezultatai

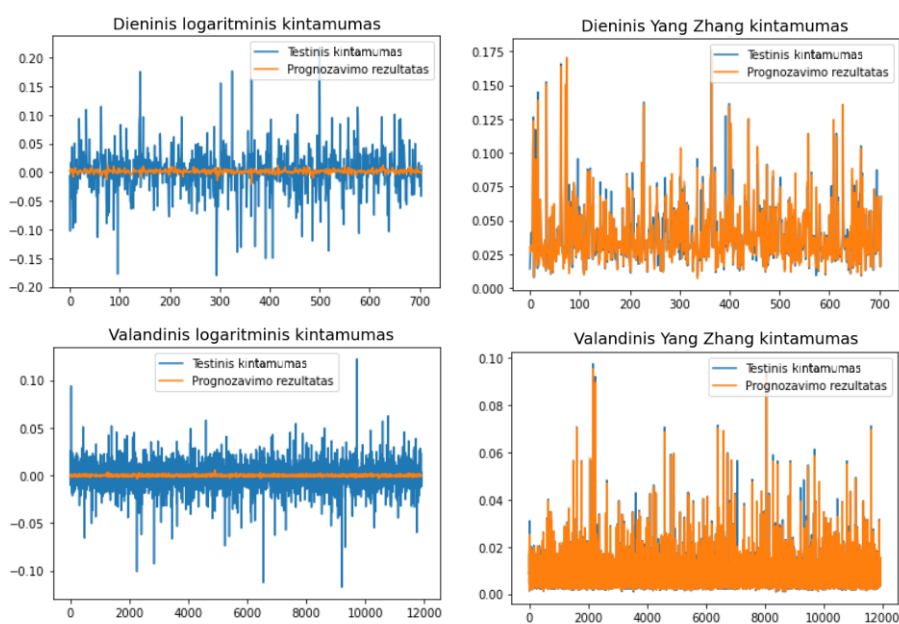
Suprognozavus duomenis, išvados apie Yang Zhang kintamumo eilutes pasitvirtino: jų rezultatai yra žymiai geresni už logaritminio kintamumo eilučių rezultatus. Tą kuo puikiai rodo R2 ir MDA metrikos. Taip pat reikia atkreipti dėmesį į MAE kriterijų: abiejų kriptovaliutų kainų valandos intervalo Yang Zhang kintamumo eilučių prognozavimo rezultatų MAE buvo lygus nuliui, t.y. neturėjo paklaidų.

Kriptovaliuta	Dažnumas	Kintamumas	MAE	R2	RMSE	MDA
Bitcoin	Dienos	Logaritminis	0.028	-0.009	0.041	0.480
		Yang Zhang	0.003	0.931	0.006	0.930
	Valandos	Logaritminis	0.005	-0.005	0.008	0.525
		Yang Zhang	0.000	0.966	0.001	0.961
Litecoin	Dienos	Logaritminis	0.039	-0.031	0.058	0.534
		Yang Zhang	0.005	0.885	0.010	0.916
	Valandos	Logaritminis	0.007	0.002	0.011	0.527
		Yang Zhang	0.000	0.959	0.001	0.956

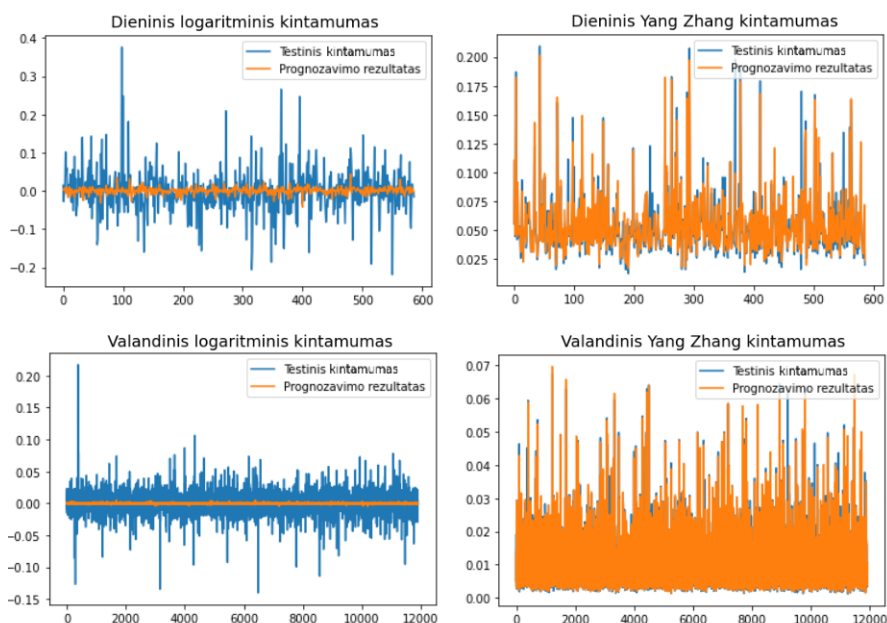
5 lentelė. SVM modelio skirtingų Bitcoin ir Litecoin kriptovaliutų laiko eilučių prognozavimo rezultatai

Galima pastebėti, kad prognozuojant dienos intervalo abiejų kriptovaliutų kainų logaritminio kintamumo laiko eilutes, RMSE paklaida yra kiek didesnė, nei kitur. Čia gali turėti įtakos duomenų rinkinio dydis, kadangi dienos intervalo laiko eilutės turi žymiai mažesnę imtį. Taip pat buvo pastebėta, kad ir kitos metrikos geresnės ten, kur buvo remiamasi didesniu duomenų rinkiniu.

Vizualizavus treniravimosi ir prognozavimo rezultatus, taip pat matosi, kad tiek prognozuojant Bitcoin (14 pav.), tiek Litecoin (15 pav.) kriptovaliutų kainų kintamumo eilutes, Yang Zhang kintamumo kintamasis prognozuojamas žymiai geriau.



14 pav. Bitcoin kriptovaliutos kainų kintamumo prognozavimo rezultatai SVM metodu.



15 pav. Litecoin kriptovaliutos kainų kintamumo prognozavimo rezultatai SVM metodu.

4.3.2. Random Forest modelis

Dar vienas modelis, kuris buvo sudarytas naudojantis sklearn biblioteka yra atsitiktinių miškų modelis. Šis modelis turi vieną reguliuojamą hiperparametrą - medžių skaičių miške $n_estimators$. Kryžmeninės validacijos metodo pagalba bus išrinkta geriausia $n_estimators$ hiperparametro reikšmė. Kai ji bus rasta, modelis bus treniruojamas ir pritaikomas fit() metodo pagalba.

Kiekvienai iš laiko eilučių bus ieškomas atskiras geriausiai tinkantis modelis. Tokiu būdu kryžminio patikrinimo pagalba žemiau pristatoma geriausia hiperparametro reikšmė kiekvienai iš turimų laiko eilučių:

Kriptovaliuta	Dažnumas	Kintamumas	Kombinacija
Bitcoin	Dienos	Logaritminis	$n_estimators = 200$
		Yang Zhang	$n_estimators = 100$
	Valandos	Logaritminis	$n_estimators = 10$
		Yang Zhang	$n_estimators = 10$
Litecoin	Dienos	Logaritminis	$n_estimators = 100$
		Yang Zhang	$n_estimators = 200$
	Valandos	Logaritminis	$n_estimators = 10$
		Yang Zhang	$n_estimators = 10$

6 lentelė. RF modelio skirtingų Bitcoin ir Litecoin kriptovaliutų laiko eilučių geriausia hiperparametro reikšmė

Gavus geriausias kombinacijos modeliai bus pritaikomi, o treniravimosi ir prognozavimo rezultatai palyginami. Rezultatai pristatomi dviejose lentelėse žemiau.

Ištreniravus RF modelį galima padaryti tokias pat išvadas, kaip ir SVM modelio atveju: geriausi rezultatai pasiekti prognozuojant Yang Zhang kintamumo eilutes. Tuo pačiu reikia atkreipti dėmesį, kad visų laiko eilučių R2 ir MDA metrikų rezultatai yra arti vieneto, o tai reiškia, kad suprognozuoti duomenys gali neblogai atitikti realius duomenis ir judėjimo kryptį. Tačiau skirtumas

tarp dienos ir valandos intervalų laiko eilučių prognozavimo rezultatu visgi yra. Mažesnis imties eilutės prognozuojamos prasčiau.

Kripto valiuta	Dažnumas	Kintamumas	MAE	R2	RMSE	MDA
Bitcoin	Dienos	Logaritminis	0.010	0.857	0.015	0.942
		Yang Zhang	0.001	0.988	0.003	0.966
	Valandos	Logaritminis	0.002	0.795	0.004	0.866
		Yang Zhang	0.000	0.992	0.000	0.981
Litecoin	Dienos	Logaritminis	0.015	0.852	0.022	0.945
		Yang Zhang	0.002	0.988	0.003	0.966
	Valandos	Logaritminis	0.003	0.795	0.005	0.868
		Yang Zhang	0.000	0.995	0.000	0.983

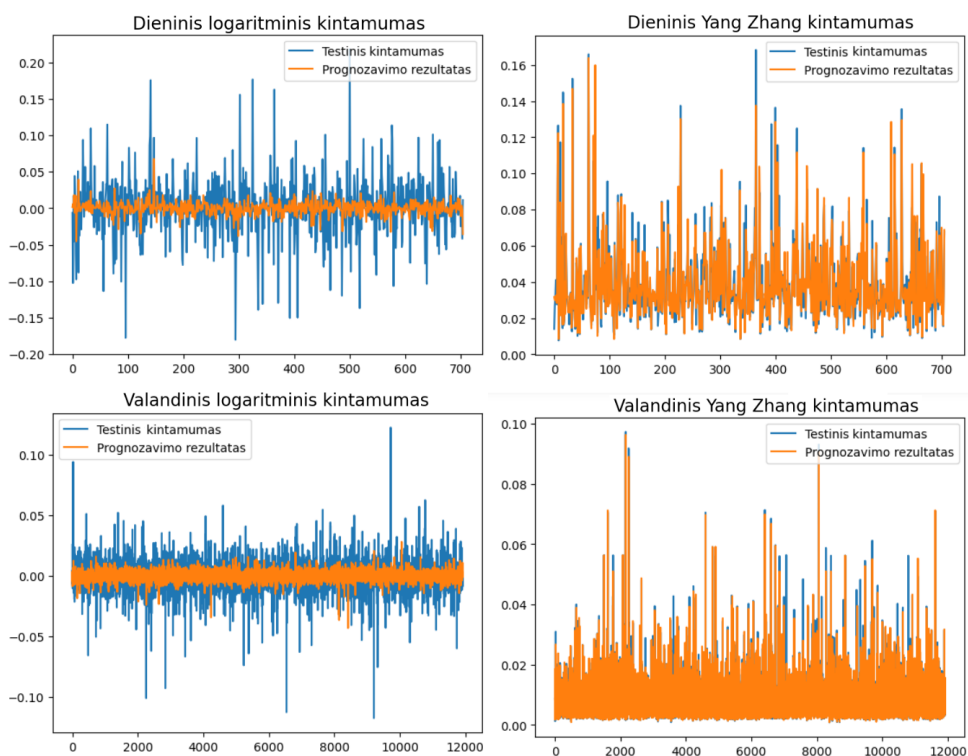
7 lentelė. RF modelio skirtingų Bitcoin ir Litecoin kriptovaliutų laiko eilučių treniravimosi rezultatai

Nežiūrint į neblogus, ir net gi geresnius, nei SVM modelio, treniravimosi rezultatus, prognozavimo rezultatai yra analogiški pirmojo tirta modelio rezultatams. Yang Zhang kintamumo prognozavimo rezultatai yra žymiai geresni už logaritminio kintamumo eilučių prognozavimo rezultatus. Taip pat buvo pastebėta, kad ir visų metrikų įverčiai yra geresni ten, kur buvo naudojamas didesnis duomenų rinkinys.

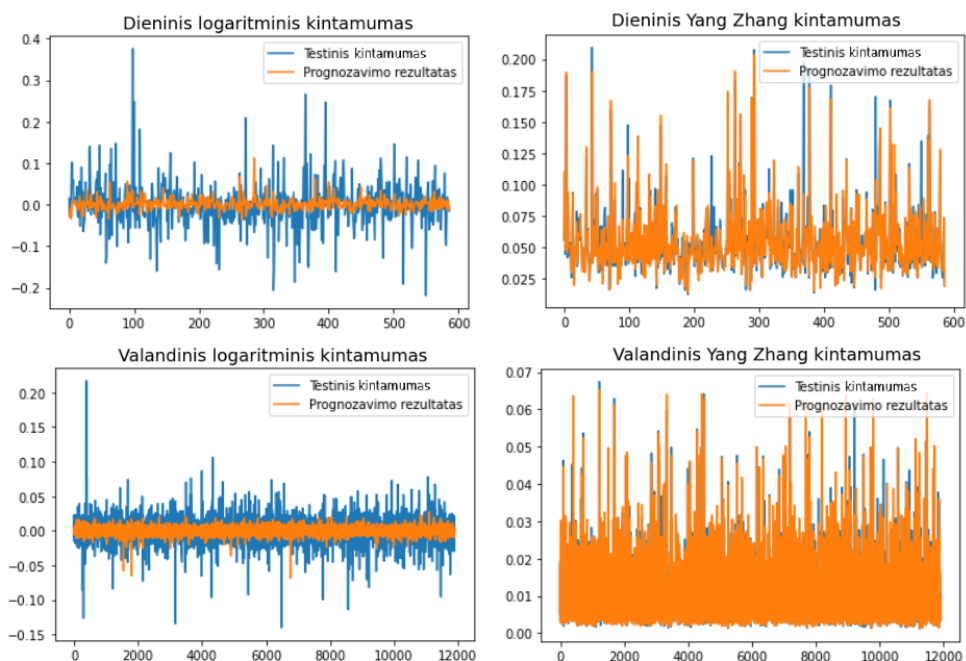
Kripto valiuta	Dažnumas	Kintamumas	MAE	R2	RMSE	MDA
Bitcoin	Dienos	Logaritminis	0.029	-0.017	0.042	0.507
		Yang Zhang	0.004	0.918	0.007	0.920
	Valandos	Logaritminis	0.005	-0.122	0.008	0.500
		Yang Zhang	0.000	0.955	0.001	0.951
Litecoin	Dienos	Logaritminis	0.040	-0.033	0.058	0.491
		Yang Zhang	0.005	0.886	0.010	0.922
	Valandos	Logaritminis	0.008	-0.126	0.012	0.513
		Yang Zhang	0.001	0.956	0.001	0.952

8 lentelė. RF modelio skirtingų Bitcoin ir Litecoin kriptovaliutų laiko eilučių prognozavimo rezultatai

Padarytos išvados pasitvirtina ir po prognozavimo rezultatų vizualizavimo. Matosi, kad tiek prognozuojant Bitcoin (16 pav.), tiek Litecoin (17 pav.) kriptovaliutų kainų kintamumo eilutes, Yang Zhang kintamumo kintamasis prognozuojamas žymiai geriau.



16 pav. Bitcoin kriptovaliutos kainų kintamumo prognozavimo rezultatai RF metodu.



17 pav. Litecoin kriptovaliutos kainų kintamumo prognozavimo rezultatai RF metodu.

4.3.3. KNN modelis

KNN modelis taip pat buvo sudarytas naudojantis sklearn biblioteka. Šiam modeliui taip pat bus parenkamas tik vienas hiperparametras - kaimynų skaičius miške $n_neighbors$. Kryžmeninės validacijos metodo pagalba bus išrinkta geriausia $n_neighbors$ hiperparametro reikšmė. Kai ji bus rasta, modelis bus treniruojamas ir pritaikomas `fit()` metodo pagalba.

Kiekvienai iš laiko eilučių bus ieškomas atskiras geriausiai tinkantis modelis. Tokiu būdu kryžminio patikrinimo pagalba žemiau pristatoma geriausia hiperparametro reikšmė kiekvienai iš turimų laiko eilučių. Pagal lentelę esančią žemiau, galima matyti, kad visoms laiko eilutėms geriausiai pasižymėjo modelis su tokiu pat hiperparametru $n_neighbors$:

Kripto valiuta	Dažnumas	Kintamumas	Kombinacija
Bitcoin	Dienos	Logaritminis	$n_neighbors = 5$
		Yang Zhang	$n_neighbors = 5$
	Valandos	Logaritminis	$n_neighbors = 5$
		Yang Zhang	$n_neighbors = 5$
Litecoin	Dienos	Logaritminis	$n_neighbors = 5$
		Yang Zhang	$n_neighbors = 5$
	Valandos	Logaritminis	$n_neighbors = 5$
		Yang Zhang	$n_neighbors = 5$

9 lentelė. KNN modelio skirtingų Bitcoin ir Litecoin kripto valiutų laiko eilučių geriausios kombinacijos

Gavus geriausias kombinacijos modeliai bus pritaikomi, o treniravimosi ir prognozavimo rezultatai palyginami. Rezultatai atvaizduojami dviejose lentelėse žemiau.

Analogiškai, kaip ir su dviem modeliais tirtais aukščiau, SVM ir RF, ištreniravus KNN modelį buvo padarytos tokios pat išvados: geriausi rezultatai pasiekti prognozuojant Yang Zhang kintamumo eilutes. Taip pat, KNN treniravimosi rezultatai parodė kiek mažesnę krypties tikslumo paklaidą, lyginant su SVM modeliu. Taip pat jaučiasi ir skirtumas tarp dienos ir valandos intervalų laiko eilučių prognozavimo rezultatu: kaip buvo pastebėta ir su SVM ir RF modeliais, mažesnės imties eilutės prognozuojamos prasčiau.

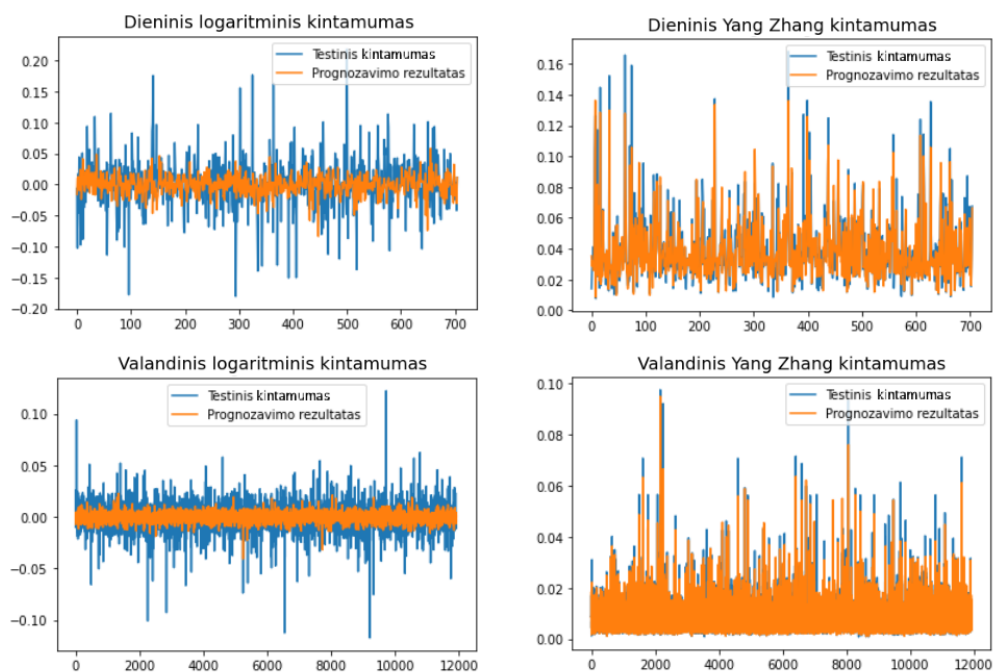
Kripto valiuta	Dažnumas	Kintamumas	MAE	R2	RMSE	MDA
Bitcoin	Dienos	Logaritminis	0.024	0.222	0.035	0.648
		Yang Zhang	0.004	0.925	0.007	0.914
	Valandos	Logaritminis	0.004	0.245	0.007	0.660
		Yang Zhang	0.005	0.877	0.008	0.911
Litecoin	Dienos	Logaritminis	0.035	0.248	0.049	0.655
		Yang Zhang	0.005	0.941	0.008	0.917
	Valandos	Logaritminis	0.006	0.241	0.010	0.663
		Yang Zhang	0.001	0.970	0.001	0.951

10 lentelė. KNN modelio skirtingų Bitcoin ir Litecoin kripto valiutų laiko eilučių treniravimosi rezultatai

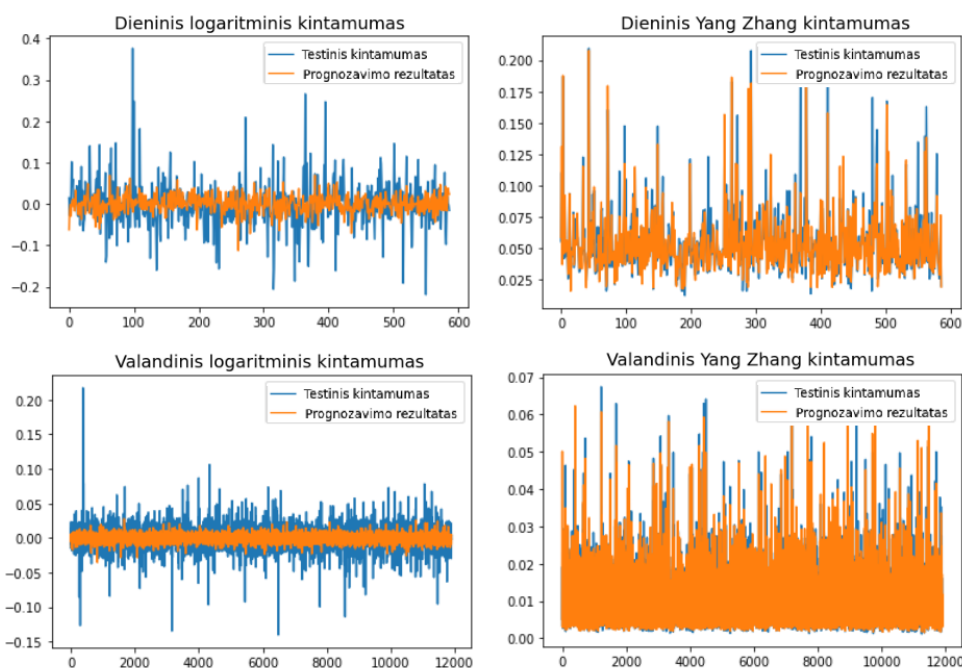
Nežiūrint į neblogus, ir netgi geresnius, nei SVM modelio, treniravimosi rezultatus, prognozavimo rezultatai yra analogiški pirmojo tirta modelio rezultatams. Yang Zhang kintamumo prognozavimo rezultatai yra žymiai geresni už logaritminio kintamumo eilučių prognozavimo rezultatus. Taip pat buvo pastebėta, kad ir visų metrikų įverčiai yra geresni ten, kur buvo naudojamas didesnis duomenų rinkinys. Rezultatai atvaizduojami paveikslukuose žemiau (18 pav., 19 pav.).

Kripto valiuta	Dažnumas	Kintamumas	MAE	R2	RMSE	MDA
Bitcoin	Dienos	Logaritminis	0.031	-0.123	0.044	0.518
		Yang Zhang	0.000	0.963	0.001	0.954
	Valandos	Logaritminis	0.005	-0.132	0.008	0.509
		Yang Zhang	0.001	0.939	0.001	0.944
Litecoin	Dienos	Logaritminis	0.043	-0.121	0.061	0.491
		Yang Zhang	0.006	0.860	0.011	0.899
	Valandos	Logaritminis	0.008	-0.122	0.012	0.510
		Yang Zhang	0.001	0.940	0.001	0.939

11 lentelė. KNN modelio skirtingų Bitcoin ir Litecoin kriptovaliutų laiko eilučių prognozavimo rezultatai



18 pav. Bitcoin kriptovaliutos kainų kintamumo prognozavimo rezultatai KNN metodu.



19 pav. Litecoin kriptovaliutos kainų kintamumo prognozavimo rezultatai KNN metodu.

4.3.4. LSTM modelis

Kitas tyrimui panaudotas modelis yra LSTM. Svarbu atkreipti dėmesį, kad prieš pradėdant dirbti būtent su šiuo modeliu, naudojamus laiko eilučių duomenis reikia normalizuoti. Duomenys bus normalizuojami *MinMaxScaler()* funkcijos pagalba. Kaip ir kitur, pasitelkiant LSTM modeliu bus bandoma atsižvelgti į praeitas 7, dienos dažnumo eilučių atveju, arba 12, valandos dažnumo eilučių atveju, reikšmes. LSTM kontekste jos dar vadinamos įvestimi. Vėliau jos bus palygintos su 8 arba 13 reikšmėmis, kurios LSTM kontekste gali būti vadinamos išvestimi.

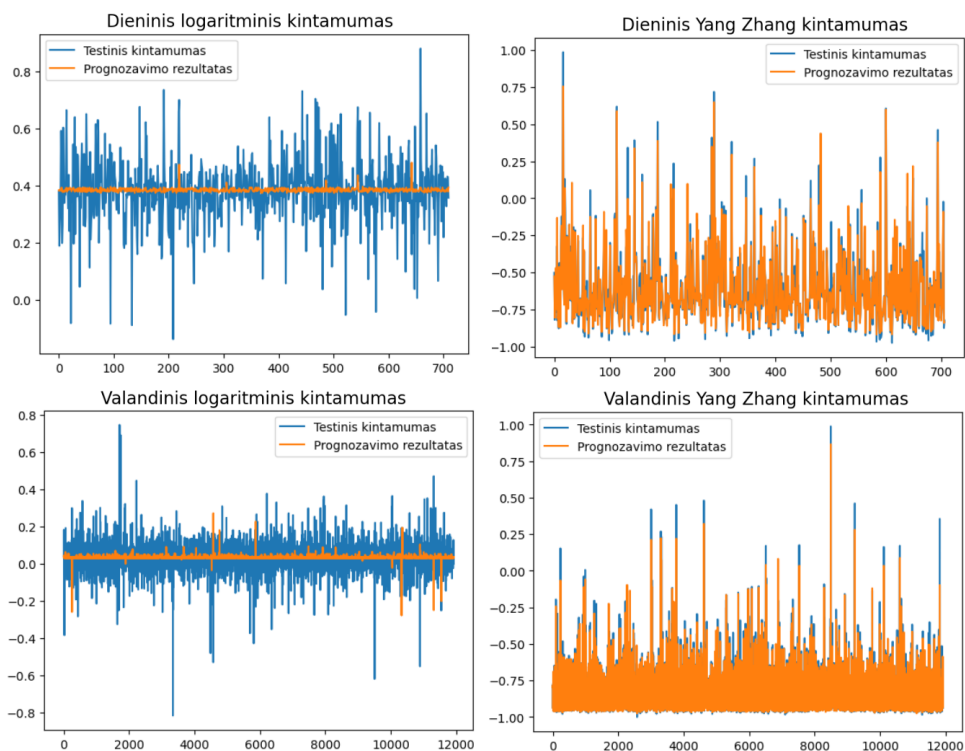
Šiame darbe LSTM modelis turi septynis sluoksnius: LSTM sluoksnį su 256 parametrais, "gelu" aktyvacijos sluoksnį, normalizacijos sluoksnį, dar vieną LSTM sluoksnį tik su 64 parametrais, dar vieną "gelu" aktyvacijos sluoksnį, „Dense“ funkcijos sluoksnį, kur lieka 1 parametras ir paskutinį "tanh" aktyvacijos sluoksnį. Modelis optimizuotas "Adam" funkcijos pagalba, o praradimams (angl. loss) skaičiuoti panaudota MSE metrika. Modelis buvo mokinamas ir testuojamas su skirtingais epochų pasirinkimais: 25, 50 ir 100.

Ištreniravus ir ištestavus LSTM modelį buvo padarytos tokios išvados: geriausi rezultatai pasiekti prognozuojant Yang Zhang kintamumo eilutes. Taip pat, pasirinkus daugiau epochų, prognozavimo rezultatai Yang Zhang atveju yra geresni, nei prognozuojant su mažesnių epochų kiekiu. Taip pat buvo atkreipta dėmesys ir į tai, kad prognozuojant LSTM modelio pagalba, treniravimosi rezultatai parodė kiek didesnes MDA, MAE ir RMSE paklaidas, lyginant su SVM, KNN ir RF modeliais.

Dažnumas	Kintamumas	Epochos	MAE	R2	RMSE	MDA
Dienos	Logaritminis	25	0.086	-0.017	0.127	0.520
		50	0.092	-0.071	0.131	0.516
		100	0.088	-0.059	0.130	0.517
	Yang Zhang	25	0.054	0.093	0.093	0.923
		50	0.051	0.908	0.083	0.912
		100	0.049	0.911	0.082	0.930
Valandos	Logaritminis	25	0.028	-0.040	0.047	0.519
		50	0.028	-0.084	0.048	0.513
		100	0.030	-0.164	0.050	0.489
	Yang Zhang	25	0.011	0.943	0.026	0.948
		50	0.010	0.956	0.023	0.952
		100	0.009	0.958	0.022	0.959

12 lentelė. LSTM modelio skirtingų Bitcoin kriptovaliutų laiko eilučių prognozavimo rezultatai

Padarytos išvados pasitvirtina ir po prognozavimo rezultatų vizualizavimo (20 pav.). Matosi, kad prognozuojant Bitcoin kriptovaliutų kainų kintamumo eilutes, Yang Zhang kintamumo kintamasis prognozuojamas žymiai geriau.

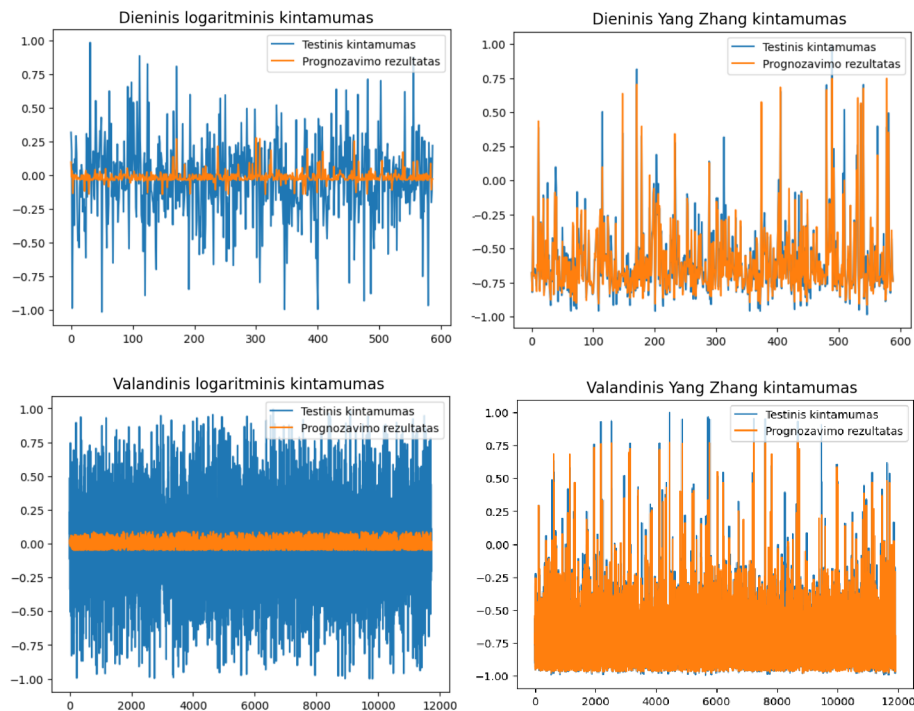


20 pav. LSTM modelio skirtingų Bitcoin kriptovaliutų laiko eilučių prognozavimo rezultatai.

Panašias išvadas buvo padarytas ir testuojant LSTM modelį su Litecoin kriptovaliutos kainų kintamumo eilutėmis. Rezultatus galima matyti žemiau pristatytoje 13 lentelėje ir 21 paveikslėlyje.

Dažnumas	Kintamumas	Epochos	MAE	R2	RMSE	MDA
Dienos	Logaritminis	25	0.095	0.003	0.143	0.498
		50	0.095	0.030	0.141	0.546
		100	0.096	0.008	0.142	0.505
	Yang Zhang	25	0.070	0.834	0.108	0.887
		50	0.052	0.865	0.097	0.904
		100	0.062	0.852	0.102	0.899
Valandos	Logaritminis	25	0.036	-0.052	0.058	0.513
		50	0.037	-0.058	0.058	0.512
		100	0.037	-0.058	0.059	0.503
	Yang Zhang	25	0.020	0.950	0.039	0.955
		50	0.018	0.955	0.037	0.954
		100	0.024	0.947	0.040	0.951

13 lentelė. LSTM modelio skirtingų Litecoinin kriptovaliutų laiko eilučių prognozavimo rezultatai



21 pav. LSTM modelio skirtingų Litecoinin kriptovaliutų laiko eilučių prognozavimo rezultatai.

4.3.5. GRU modelis

Šiame poskyryje aprašomas modelis yra GRU modelis. Kaip ir LSTM atveju, prieš pradėdant dirbti su šiuo modeliu, naudojamus laiko eilučių duomenis reikia normalizuoti. Duomenys bus normalizuojami *MinMaxScaler()* funkcijos pagalba. Kaip ir kitur, pasitelkiant GRU modeliu bus bandoma atsižvelgti į praeitas 7, dienos dažnumo eilučių atveju, arba 12, valandos dažnumo eilučių atveju, reikšmes.

Susitvarkius duomenis prieš modelio taikymą yra aprašomas pats modelis. GRU neuroninis tinklas sukuriama penkiais sluoksniais, kur pirmas yra GRU sluoksnis su 256 parametru, antras

- "tanh" aktyvacijos sluoksnis ir trečias - „Dense“ funkcijos sluoksnis, kur lieka 1 parametras. Modelio optimizavimui panaudota "Adam" funkcija, o praradimams skaičiuoti naudojama MSE metrika. Kiekvienai laiko eilutei sudarytas modelis buvo išbandomas su skirtingais epochų kiekiais: 25, 50 ir 100.

Ištestavus abiejų kriptovaliutų kainų kintamumo laiko eilutes, galima matyti, kad kaip ir kitus Yang Zhang kintamumo eilutes rodo geresnius prognozavimo įvertinimus. Tačiau lyginant su LSTM, GRU rezultatai yra kiek prastesni. Rezultatus galima matyti 14 ir 15 lentėse.

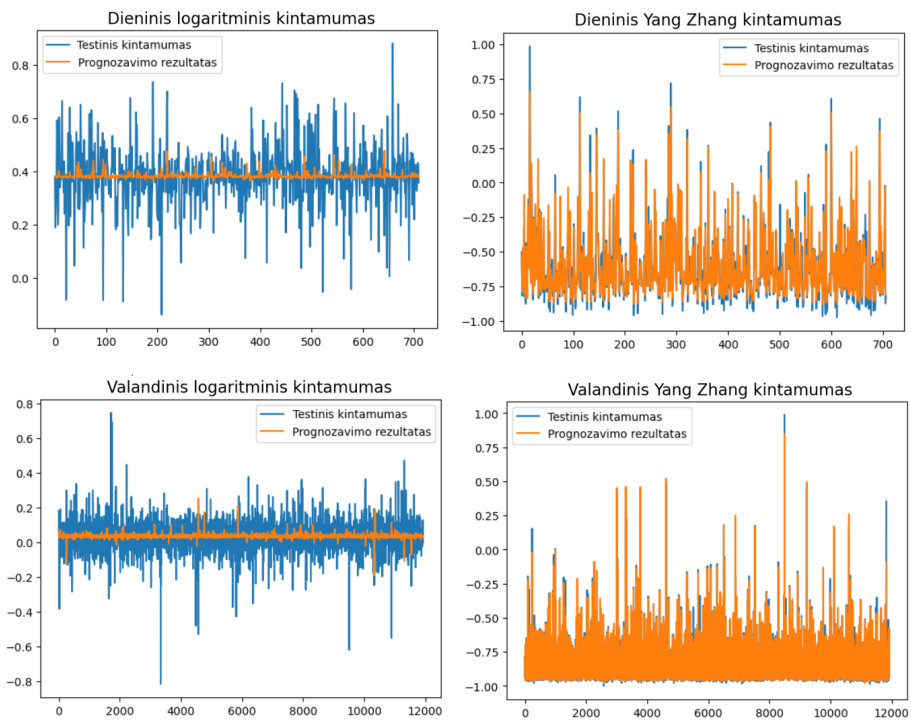
Dažnumas	Kintamumas	Epochos	MAE	R2	RMSE	MDA
Dienos	Logaritminis	25	0.088	-0.071	0.131	0.516
		50	0.086	-0.024	0.128	0.531
		100	0.086	-0.003	0.127	0.534
	Yang Zhang	25	0.069	0.846	0.108	0.895
		50	0.065	0.867	0.100	0.908
		100	0.067	0.854	0.105	0.899
Valandos	Logaritminis	25	0.028	-0.011	0.046	0.515
		50	0.028	-0.021	0.047	0.515
		100	0.028	-0.031	0.047	0.508
	Yang Zhang	25	0.017	0.906	0.033	0.917
		50	0.018	0.911	0.033	0.924
		100	0.021	0.946	0.040	0.940

14 lentelė. GRU modelio skirtingų Bitcoin kriptovaliutų laiko eilučių prognozavimo rezultatai

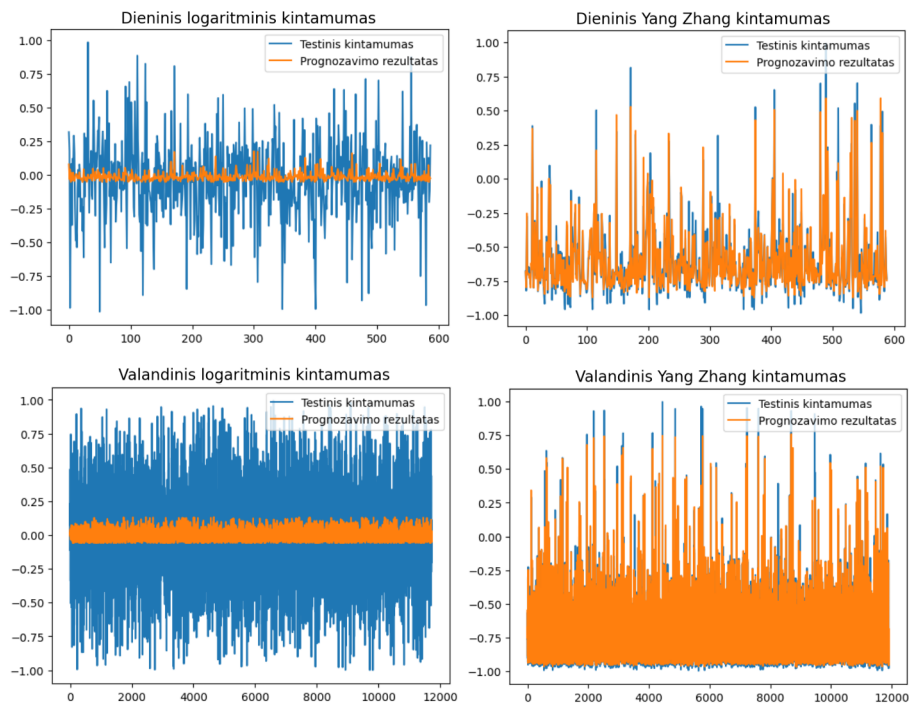
Dažnumas	Kintamumas	Epochos	MAE	R2	RMSE	MDA
Dienos	Logaritminis	25	0.095	-0.013	0.144	0.498
		50	0.095	-0.012	0.144	0.490
		100	0.095	-0.009	0.143	0.498
	Yang Zhang	25	0.161	0.489	0.189	0.836
		50	0.075	0.790	0.121	0.862
		100	0.069	0.810	0.115	0.889
Valandos	Logaritminis	25	0.035	-0.001	0.056	0.526
		50	0.036	-0.003	0.057	0.522
		100	0.036	-0.002	0.057	0.532
	Yang Zhang	25	0.018	0.952	0.038	0.944
		50	0.020	0.949	0.039	0.943
		100	0.020	0.947	0.040	0.939

15 lentelė. GRU modelio skirtingų Litecoin kriptovaliutų laiko eilučių prognozavimo rezultatai

Atvaizdavirus testavimo imties prognozavimo rezultatus, matosi, kad išvados, padarytos aukščiau, pasitvirtina (22 pav., 23 pav.).



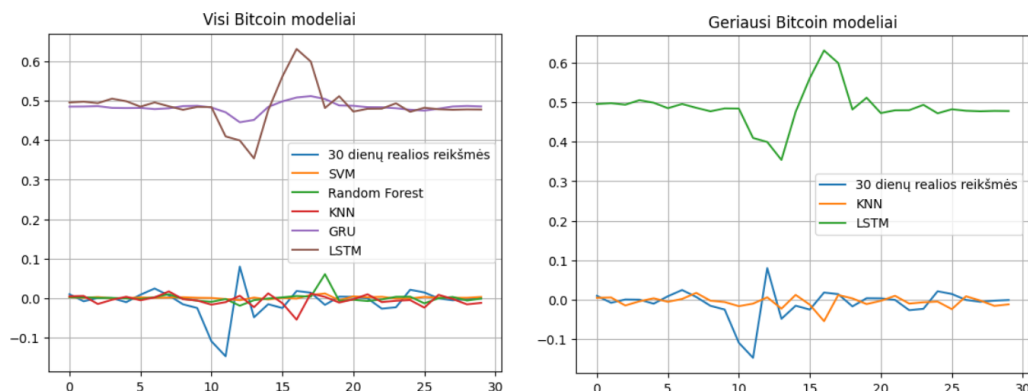
22 pav. GRU modelio skirtingų Bitcoin kriptovaliutų laiko eilučių prognozavimo rezultatai.



23 pav. GRU modelio skirtingų Litecoin kriptovaliutų laiko eilučių prognozavimo rezultatai.

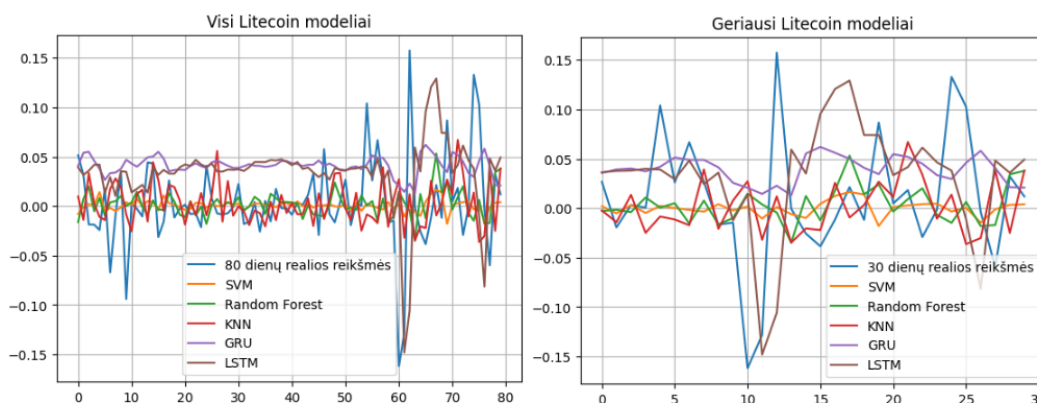
4.3.6. Rezultatai

Prognozuojant dieninį logaritminį Bitcoin kriptovaliutos kainų kintamumą buvo pastebėta, kad nei vienas iš modelių neturėjo idealių rezultatų, ką patvirtina ir rezultatų metrikos, aprašytos aukščiau. Tačiau žemiau esančiame paveikslėlyje matosi, kad LSTM modelio rezultatas, nors ir pilnai neatitinka reikšmių rėžių, geriausiai iš visų tirtų modelių atitinka kintamumo tendencijas. Jo MDA yra vienas geriausių ir lygus 0.517. KNN modelio MDA kriterijus buvo kiek didesnis, lygus 0.518, tačiau iš paveikslėlio žemiau matosi, kad prognozavimo rezultatai vis tiek yra prasti. Tai galėjo parodyti R2 kriterijus, nurodantis, ant kiek geras modelis buvo sudarytas. Visų tirtų modelių R2 metrika buvo neigiama (24 pav.).



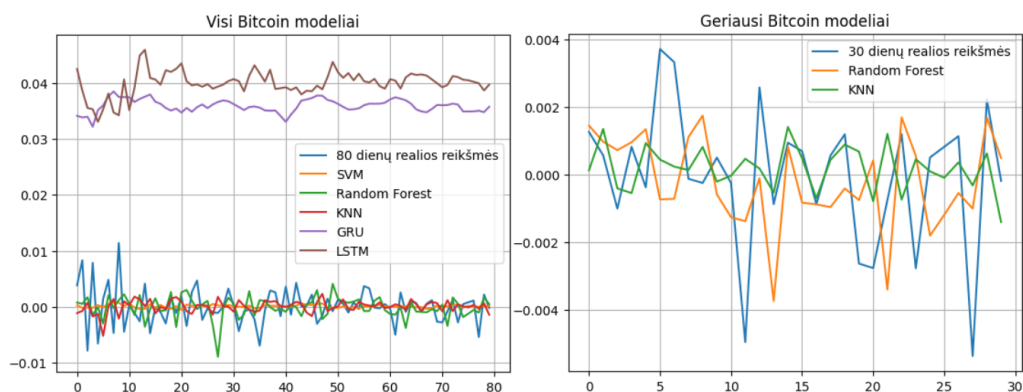
24 pav. Dieninio logaritminio Bitcoin kainų kintamumo laiko eilučių prognozavimo rezultatai.

Dėja, to paties negalima pasakyti apie dieninio logaritminio Litecoin kriptovaliutos kainų kintamumo prognozavimą. Šiuo atveju, nors SVM ir LSTM modeliai ir turėjo geriausius MDA rezultatus (atitinkamai, 0.534 ir 0.505), o R2 buvo teigiamas tik LSTM atveju, visi jie prognozavimo užduotį atliko blogai (25 pav.).



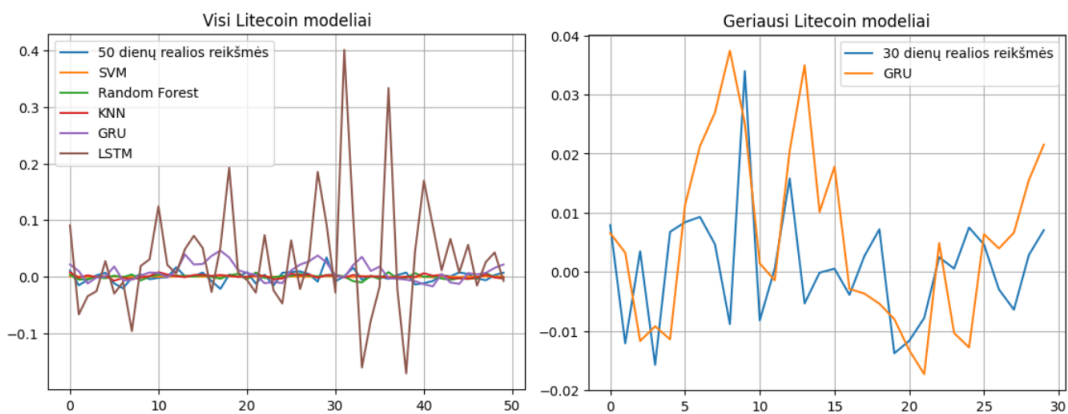
25 pav. Dieninio logaritminio Litecoin kainų kintamumo laiko eilučių prognozavimo rezultatai.

Prognozuojant valandinį logaritminį Bitcoin kriptovaliutos kainų kintamumą taip pat buvo pastebėta, kad nei vienas iš modelių neturėjo gerų rezultatų. Visi modeliai turėjo neigiamą R2 įvertį, panašius MAE ir RMSE įverčius. O MDA buvo arti 50 % ribos. Tačiau yra du modeliai, kuriuos visgi būtų galima išskirti - RF ir KNN. Jie, nors ir neturėjo geresnių už kitus modelių rezultatų, turi didelį potencialą, kadangi labiausiai atitaiso realių reikšmių, nors ir su postumių į dešinę (26 pav.).



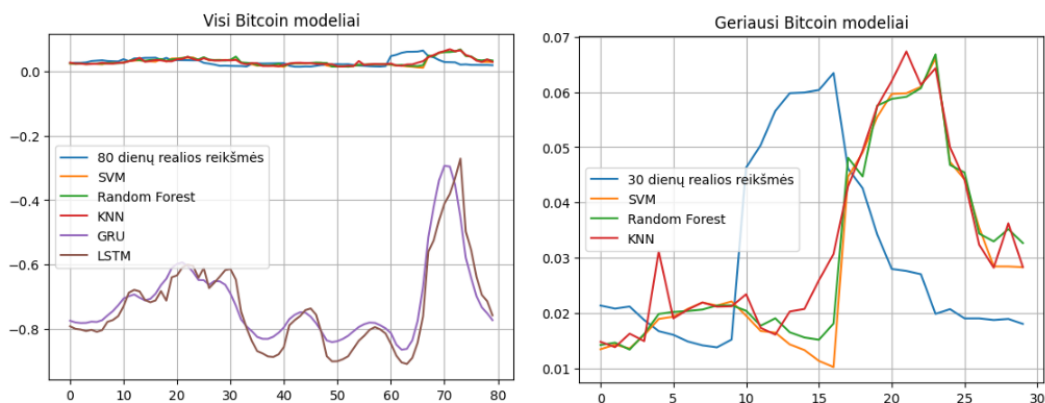
26 pav. Valandinio logaritminio Bitcoin kainų kintamumo laiko eilučių prognozavimo rezultatai.

Prognozuojant valandinį logaritminį Litecoin kriptovaliutos kainų kintamumą geriausiai pasirodė GRU modelis, jo MDA buvo 0.532. Jis, kaip matosi paveikslėlyje žemiau, labiausiai atitiko krypties tendencijas. Taip pat buvo pastebėta, kad SVM modelis, turintis geriausias metrikų įverčius, susitvarkė su prognozavimų prasčiausiai. Galima teigti, kad metrikos prognozavimo modeliams įvertinti buvo parinktos blogai (27 pav.).



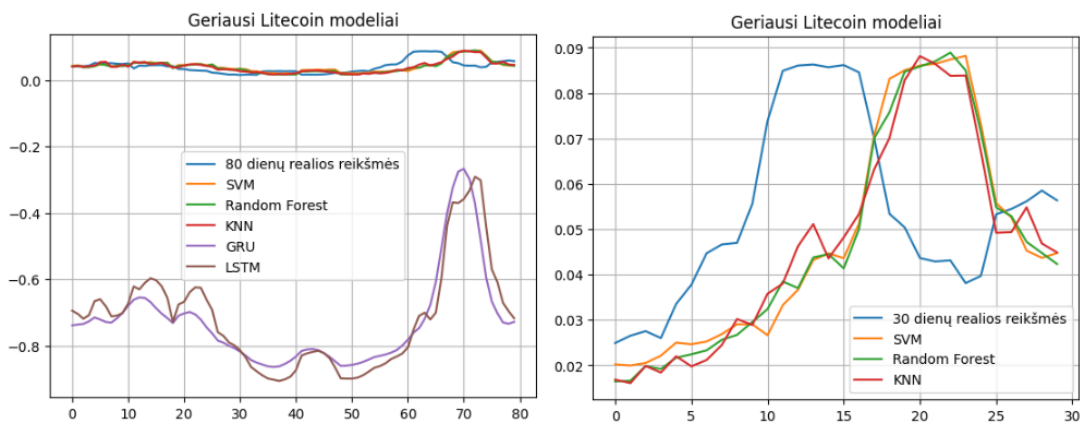
27 pav. Valandinio logaritminio Litecoin kainų kintamumo laiko eilučių prognozavimo rezultatai.

Prognozuojant dieninį Yang Zhang Bitcoin kriptovaliutos kainų kintamumą buvo pastebėta, kad visų modelių rezultatai yra žymiai geresni, nei prognozuojant logaritminį kintamumą. Pagal gautus metrikų įverčius galima išskirti SVM ir KNN modelius. SVM turėjo MAE - 0.003, R2 - 0.931, RMSE - 0.006 ir MDA - 0.930, o KNN parodė MAE - 0.000, R2 - 0.963, RMSE - 0.001 ir MDA - 0.954. Tačiau vizualiai KNN susitvarkė gana prastai, ko negalima pasakyti apie RF, kuris beveik atkartojė SVM modelio trajektoriją. Abu modeliai, tiek SVM, tiek RF gerai atitakė realių kintamumo reikšmių, tačiau su vėlavimu (28 pav.).



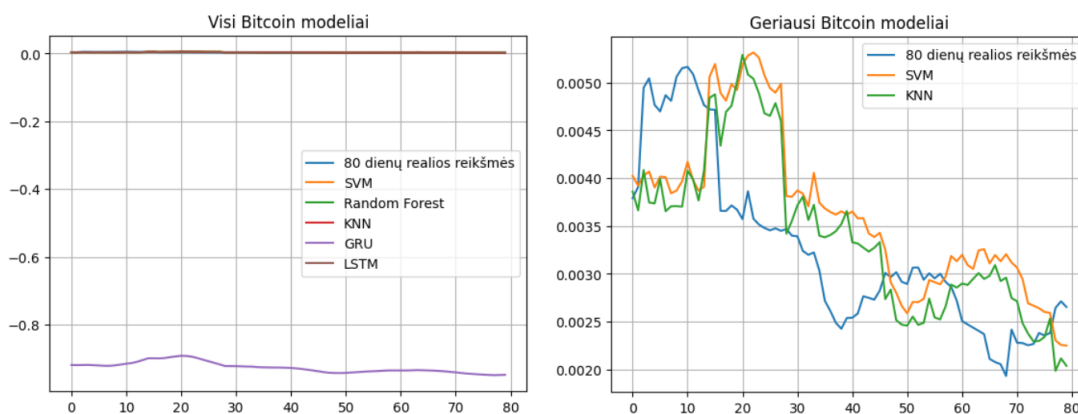
28 pav. Dieninio Yang Zhang Bitcoin kainų kintamumo laiko eilučių prognozavimo rezultatai.

Prognozuojant dieninį Yang Zhang Litecoin kriptovaliutos kainų kintamumą pagal gautus metrikų įverčius galima išskirti SVM ir RF modelius. SVM turėjo MAE - 0.005, R2 - 0.885, RMSE - 0.010 ir MDA - 0.916, o RF parodė MAE - 0.005, R2 - 0.886, RMSE - 0.010 ir MDA - 0.922. Taip pat visai neblogai susitvarkė ir KNN modelis. Visi trys neblogai atitakė realias reikšmes ir tendencijas, bet, kaip ir kitur, su vėlavimu. LSTM ir GRU algoritmai turėjo geresnius R2 ir MDA įverčius, tačiau, kaip matosi paveikslėlyje žemiau, buvo labiausiai atitolę nuo realiųjų reikšmių (29 pav.).



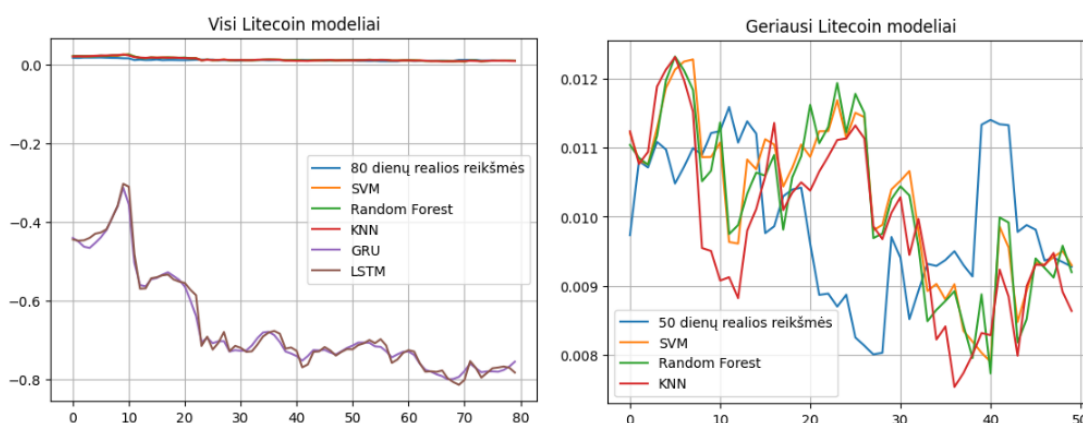
29 pav. Dieninio Yang Zhang Litecoin kainų kintamumo laiko eilučių prognozavimo rezultatai.

Vizualizavus valandinio Yang Zhang Bitcoin kriptovaliutos kainų kintamumo prognozavimo rezultatus galima išskirti SVM ir KNN modelius. SVM turėjo geriausius rezultatus. Jis pasirodė su MAE - 0.000, R2 - 0.966, RMSE - 0.001 ir MDA - 0.961. Nedaug atsiliko ir KNN. Jo R2 - 0.939, o MDA - 0.944. LSTM ir GRU algoritmai turėjo neblogus R2 ir MDA įverčius, tačiau, taip pat kaip matosi paveikslėlyje žemiau, buvo labiausiai atitolę nuo realiųjų reikšmių (30 pav.).



30 pav. Valandinio Yang Zhang Bitcoin kainų kintamumo laiko eilučių prognozavimo rezultatai.

Tokias pat išvados buvo padarytos vizualizavus Litecoin valandinio kintamumo prognozavimus rezultatus. SVM turėjo geriausias rezultatus. Jis pasirodė su MAE - 0.000, R2 - 0.959, RMSE - 0.001 ir MDA - 0.956. Nedaug atsiliko ir KNN. Jo R2 - 0.940, o MDA - 0.939. LSTM ir GRU algoritmai turėjo vėl neblogus R2 ir MDA įverčius, bet buvo labiausiai atitolę nuo realiųjų reikšmių, kas labiausiai matosi paveikslėlyje žemiau (31 pav.).



31 pav. Valandinio Yang Zhang Litecoin kainų kintamumo laiko eilučių prognozavimo rezultatai.

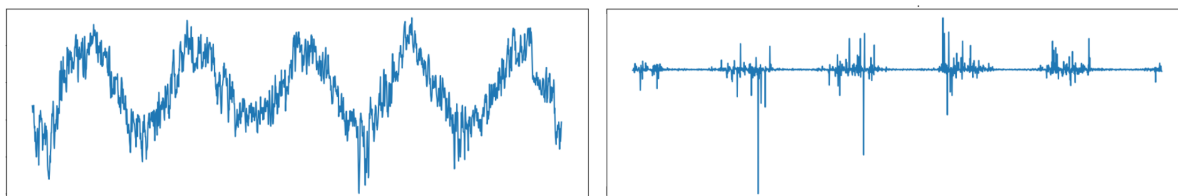
Atlikus skirtingų modelių palyginimą pastebėta, kad nors LSTM ir GRU gavo ganėtinai mažas paklaidas mokymosi ir testavimo metu, jų rezultatai buvo vieni prasčiausių. Taip pat žvelgiant bendrai į visus tirtus modelius ir jų rezultatus, galima teigti, kad absoliučiai visi rezultatai buvo ganėtinai prasti ir tiriami modeliai nesusitvarkė su prognozavimo užduotimi. Taip pat, vertinant treniravimosi ir testavimo rezultatus, Yang Zhang kintamumo eilutės buvo lengviau prognozuojamos, o metrikų įverčiai buvo geresni, nei prognozuojant logaritminio kintamumo eilutes.

4.3.7. Kitos laiko eilutės

Šiame darbe buvo tiriamos tik kriptovaliutų kainų kintamumo laiko eilutės, tad buvo nuspręsta išbandyti ir kitas laiko eilutes: temperatūros Vilniuje.

Vilniaus vidutinės temperatūros 2018.01.01 - 2022.12.08 laikotarpio duomenys buvo paimti iš „Meteostat“ svetainės. „Meteostat“ yra vienas didžiausių atvirų oro ir klimato duomenų tiekėjų [2].

Išanalizuoti oro temperatūros laiko eilutei buvo sudaryti kvantilių grafikai ir padarytas patikrinimas Lillefors testu. Abu patikrinimai parodė, kad duomenys nėra pasiskirstę pagal normalųjį skirstinį, ir nulinė hipotezė, kad imtis yra normalaus skirstinio, gali būti atmesta. Tačiau autokoreliacijos grafikai parodė, kad duomenis turi gan didelę autokoreliacinę priklausomybę, tad Vilniaus oro temperatūros prognozavimas gali turėti didelę prognozavimo galią. O kadangi šio darbo tiriamasis objektas yra kintamumas, bus bandoma iširti ir Vilniaus temperatūros kintamumą, išreikštą taip pat per dviejų reikšmių natūrinį logaritmą (32 pav.).



32 pav. Vilniaus oro temperatūra (kairėje) ir jos kintamumas (dešinėje).

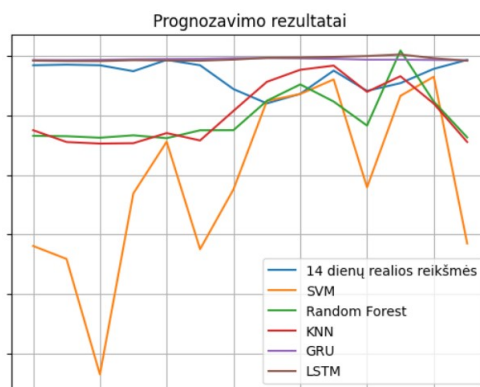
Atlikus oro temperatūros prognozavimus buvo gauti tokie rezultatai:

Modelis	MAE	R2	RMSE	MDA
SVM	1.734	0.937	2.264	0.913
KNN	2.057	0.914	2.645	0.888
RF	1.805	0.931	2.364	0.905
LSTM	0.08592	0.91814	0.11127	0.89405
GRU	0.11550	0.85864	0.14622	0.84944

16 lentelė. Vilniaus oro temperatūros laiko eilučių prognozavimo rezultatai

Pagal gautas metrikų paklaidas matosi, kad geriausiai su prognozavimo užduotimi susitvarkė SVM, GRU ir LSTM modeliai, kadangi jie turi mažiausias MAE ir RMSE paklaidas, o taip pat geriausias R2 ir MDA rodiklius.

Palyginimui, prognozavimo rezultatai buvo atvaizduoti 33 paveikslėlyje:



33 pav. Vilniaus oro temperatūros prognozavimo rezultatai.

Iš atvaizduotų rezultatų matosi, kad nė vienas iš modelių nesugebėjo suprognozuoti laiko eilutę tinkamai. Iš visų modelių tik SVM šiek tiek atitakė realius tendencijų pokyčius.

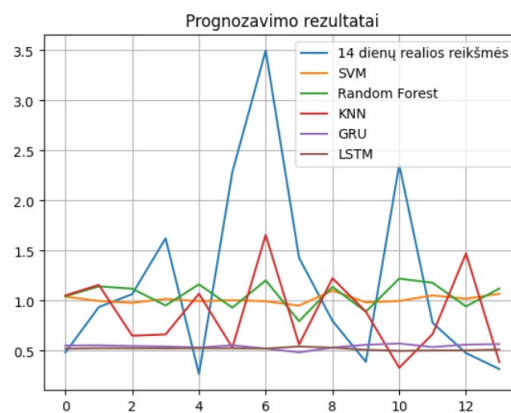
Atlikus oro temperatūros kintamumo prognozavimus buvo gauti tokie rezultatai:

Modelis	MAE	R2	RMSE	MDA
SVM	0.607	-0.016	1.783	0.507
KNN	0.786	-0.535	2.192	0.498
RF	0.721	-0.720	2.320	0.524
LSTM	0.03063	-1.14507	0.08358	0.51859
GRU	0.02292	-0.15611	0.06136	0.50743

17 lentelė. Vilniaus oro temperatūros kintamumo laiko eilučių prognozavimo rezultatai

Pagal gautas metrikų paklaidas galima teigti, kad geriausiai su prognozavimo užduotimi susitvarkė GRU ir SVM modeliai, kadangi abu jie turi mažiausias MAE ir RMSE paklaidas, o taip pat geriausias R2 ir MDA rodiklius.

Palyginimui, prognozavimo rezultatai buvo atvaizduoti 34 paveikslėlyje:



34 pav. Vilniaus oro temperatūros kintamumo prognozavimo rezultatai.

Iš atvaizduotų rezultatų matosi, kad KNN, nors ir turėjo mažiausią MDA, labiausiai atitiko realias tendencijas, kas dar kartą leidžia daryti prielaidą, jog pasirinktos prognozavimo rezultatams vertinti metrikos neatvaizduoja tikrųjų prognozavimo rezultatų.

Išvados ir rekomendacijos

Šiame darbe buvo pristatyta ir iširta kriptovaliutų kainų kintamumo prognozavimo problema. Problemos aktualumui įrodyti buvo apžvelgti su šia problema susiję darbai. Susijusių darbų apžvalgos rezultate buvo išrinkti keli modeliai tolimesniems darbams: SVM, RF, KNN, LSTM ir GRU. Kiekvienas iš modelių buvo pristatytas ir iširtas. Darbo tyrimui buvo panaudotos Bitcoin ir Litecoin kriptovaliutų kainų eilutės. Kiekvienai iš jų buvo apskaičiuoti du kintamumo parametrai: natūrinio logaritmo kintamumas ir Yang Zhang kintamumas. Taip pat buvo nuspręsta tyrimui naudoti ne tik dienos intervalo duomenis, bet ir valandos.

Atlikus tyrimą nebuvo nustatyta tik vieno geriausiai veikiančio modelio, kadangi priklausomai nuo laiko eilutės kiekvienas iš modelių pasirodydavo skirtingai. Tyrimo pabaigoje buvo nustatyta, kad Yang Zhang kintamumo eilučių prognozavimas turėjo geresnius pasirinktų metrikų įverčius. Taip pat, buvo pastebėta, kad net ir geriausius įverčius turintys modeliai nesugebėjo gerai suprognozuoti duotų laiko eilučių kintamumus ir jų tendencijas.

Atlikus tyrimą gautos tokios **išvados**:

- Yang Zhang kintamumas prognozuojamas lengviau ir tiksliau. Prognozuojant Yang Zhang kintamumo eilutes, R2 ir MDA metrikos rodydavo geriausius rezultatus, abiejų metrikų įverčiai visada buvo arti 1.
- KNN modelis neblogai prognozuoja kintamumo laiko eilutes. Pavyzdžiui, prognozuojant Yang Zhang Bitcoin kriptovaliutos kainų valandinį kintamumą, metrikų įvertinimai buvo tokie: MAE - 0.001, R2 - 0.939, RMSE - 0.001, o MDA - 0.944.
- Visi modeliai pasižymėjo didesniu efektyvumu prognozuojant didesnės apimties laiko eilutes Litcoin kriptovaliutos atveju. Tačiau prognozuojant Bitcoin kriptovaliutos eilutes, prognozavimo rezultatai buvo geresni prognozuojant dienos, t.y. mažesnės apimties, laiko eilutes. Pavyzdžiui, SVM modelio Bitcoin kriptovaliutos dienos intervalo logaritminio kintamumo eilutes prognozavimo MDA rezultatas yra 0.480, o valandos intervalo - 0.525. Litecoin atveju, dienos intervalo MDA yra 0.534, o valandos - 0.527.
- Vizualizavus skirtingų modelių prognozavimo rezultatus, SVM, KNN ir RF geriausiai atitaidavo kintamumo pokyčius ir tendencijas, tačiau su vėlavimu.
- Pastebėta, kad pasirinktos metrikos, t.y. MDA, RMSE, MAE ir R2, negali gerai atvaizduoti sudarytų modelių prognozavimo rezultatų tikslumo.
- Pastebėta, kad kitų autorių darbuose GRU ir LSTM algoritmai su prognozavimo užduotimi susitvarko žymiai geriau, nei SVM ar RF. Todėl abu modeliai galėtų būtų sudaryti geriau.
- Visi modeliai gana prastai susitvarkė su prognozavimo užduotimi, nors pasirinktų metrikų įverčiai buvo neblogi.

Pateikiamos tokios darbo **rekomendacijos**:

- Rekomenduojama sudaryti geresnę struktūrą tiek GRU, tiek LSTM modeliams kriptovaliutų kainų kintamumui prognozuoti.
- Rekomenduojama naudotis kryžminio patikrinimo metodu skirtingų modelių geriausioms kombinacijoms atrasti.
- Rekomenduojama pasirinkti kitas modelių vertinimo metrikas, kurios geriau galėtų aprašyti sudaryto modelio efektyvumą.

Ateities tyrimų planas

Vertinant gautus šio magistro baigiamojo darbo rezultatus, atitinkamai susidarė ir ateities darbų sąrašas:

- Apžvelgti visas įmanomas metrikas prognozavimo modelio našumui ir efektyvumui įvertinti. Palyginti jas su AIC, RMSE, R2 ir MDA metrikomis.
- Sudaryti geresnius LSTM ir GRU modelius: pridėti daugiau sluoksnių.
- Naudoti daugiau įvairesnių kriptovaliutų eksperimentiniams tyrimams.
- Atliekant prognozavimą su LSTM arba GRU modeliais, naudoti papildomą veiksnį, pavyzdžiui, valiutos kursą.

Literatūros šaltiniai

- [1] Duomenys paimti iš cryptodatadownload svetainės.
<https://www.cryptodatadownload.com/data/bitstamp>.
- [2] Meteostat.
<https://meteostat.net/en/place/lt/vilnius?s=26730&t=2022-11-24/2022-12-01>.
- [3] Nashirah Abu Bakar and Sofian Rosbi. Autoregressive integrated moving average (arima) model for forecasting cryptocurrency exchange rate in high volatility environment: A new insight of bitcoin transaction. *International Journal of Advanced Engineering Research and Science*, 4(11):130--137, 2017.
- [4] Rahmat Albariqi and Edi Winarko. Prediction of bitcoin price change using neural networks. In *2020 International Conference on Smart Technology and Applications (ICoSTA)*, pages 1--4. IEEE, 2020.
- [5] Khalid Alkhatib, Hassan Najadat, Ismail Hmeidi, and Mohammed K Ali Shatnawi. Stock price prediction using k-nearest neighbor (knn) algorithm. *International Journal of Business, Humanities and Technology*, 3(3):32--44, 2013.
- [6] Aidas Balčiūnas. *Aprašomoji statistika*, 2019.
<http://web.vu.lt/mif/a.balciunas/files/2019/12/Taikomoji-statistika2-1.pdf>.
- [7] Elie Bouri, Konstantinos Gkillas, Rangan Gupta, and Christian Pierdzioch. Forecasting realized volatility of bitcoin: The role of the trade war. *Computational Economics*, 57(1):29--53, 2021.
- [8] Ryan Browne. Bitcoin falls after elon musk tweets breakup meme. 2021.
<https://www.cnn.com/2021/06/04/bitcoin-falls-after-elon-musk-tweets-breakup-meme.html>.
- [9] Sina E Charandabi and Kamyar Kamyar. Survey of cryptocurrency volatility prediction literature using artificial neural networks. *Business and Economic Research*, 12(1), 2022.
- [10] Matthew Chen, Neha Narwal, and Mila Schultz. Predicting price changes in ethereum. *International Journal on Computer Science and Engineering (IJCSE) ISSN*, pages 0975--3397, 2019.
- [11] Sina Esmaeilpour Charandabi and Kamyar Kamyar. Survey of cryptocurrency volatility prediction literature using artificial neural networks. 01 2022.
- [12] Lorella Fatone, Francesca Mariani, and Francesco Zirilli. A hybrid model based on stochastic volatility and machine learning to forecast log returns of a risky asset. In *Methods and Applications in Fluorescence*, pages 235--240. Springer, 2022.
- [13] Andy Field. *Discovering statistics using IBM SPSS statistics*. sage, 2013.
- [14] Ramanathan Gnanadesikan and Martin B Wilk. Probability plotting methods for the analysis of data. *Biometrika*, 55(1):1--17, 1968.

- [15] Vilmantas Gėgžna. Biostatistinės analizės pagrindai, 2022.
<https://mokymai.github.io/biostatistika/ht-normalumas.html>.
- [16] Yiqing Hua. Bitcoin price prediction using arima and lstm. In *E3S Web of Conferences*, volume 218, page 01050. EDP Sciences, 2020.
- [17] Cyrus Ip. Bitcoin’s seasonality: Doing the right thing at the right season is a great art, 2020.
<https://www.newsbtc.com/all/bitcoins-seasonality-doing-the-right-thing-at-the-right-season-is-a-gre>
- [18] Neil Gandal ir Hanna Hałaburda. Competition in the cryptocurrency market. 2014.
<https://www.bankofcanada.ca/wp-content/uploads/2014/08/wp2014-33.pdf>.
- [19] Ahmed M Khedr, Ifra Arif, Magdi El-Bannany, Saadat M Alhashmi, and Meenu Sreedharan. Cryptocurrency price prediction using traditional statistical and machine-learning techniques: A survey. volume 28, pages 3--34. Wiley Online Library, 2021.
- [20] Kyoung-jae Kim. Financial time series forecasting using support vector machines. *Neuro-computing*, 55(1-2):307--319, 2003.
- [21] Gerrit Köchling, Philipp Schmidtke, and Peter N Posch. Volatility forecasting accuracy for bitcoin. *Economics Letters*, 191:108836, 2020.
- [22] RS Latha, GR Sreekanth, RC Suganthe, M Geetha, R Esakki Selvaraj, S Balaji, KR Harini, and P Priya Ponnusamy. Stock movement prediction using knn machine learning algorithm. In *2022 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1--5. IEEE, 2022.
- [23] W. Zhao M. Shen and N. De. Coindesk - leader in blockchain news. *CoinDesk.com*, 2018.
- [24] Sean McNally, Jason Roche, and Simon Caton. Predicting the price of bitcoin using machine learning. In *2018 26th euromicro international conference on parallel, distributed and network-based processing (PDP)*, pages 339--343. IEEE, 2018.
- [25] Mimoza Mijoska, Blagoj Ristevski, Snezana Savoska, and Vladimir Trajkovik. Predicting bitcoin volatility using machine learning algorithms and blockchain technology. 2022.
- [26] Ryotaro Miura, Lukáš Pichl, and Taisei Kaizoji. Artificial neural networks for realized volatility prediction in cryptocurrency time series. In *International Symposium on Neural Networks*, pages 165--172. Springer, 2019.
- [27] Hakan Pabuçcu, Serdar Ongan, and Ayse Ongan. Forecasting the movements of bitcoin prices: an application of machine learning algorithms. *Quantitative Finance and Economics*, 4(4):679--692, 2020.
- [28] Emmanuel Pintelas, Ioannis E Livieris, Stavros Stavroyiannis, Theodore Kotsilieris, and Panagiotis Pintelas. Investigating the problem of cryptocurrency price prediction: a deep learning approach. In *IFIP International conference on artificial intelligence applications and innovations*, pages 99--110. Springer, 2020.
- [29] M CHAITANYA KISHORE REDDY, MSKP SRI PRADA, K SUDHAKAR, and A SUKERTHI. Stock market trend prediction using k-nearest neighbour (knn) algorithm.

- [30] Samiha Nanjiba ir Amitabha Chakrabarty Shaily Roy. Bitcoin price forecasting using time series analysis. In *2018 21st International Conference of Computer and Information Technology (ICCIT)*, pages 1--5. IEEE, 2018.
- [31] Samuel Sanford Shapiro and Martin B Wilk. An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591--611, 1965.
- [32] Ze Shen, Qing Wan, and David J Leatham. Bitcoin return volatility forecasting: A comparative study between garch and rnn. *Journal of Risk and Financial Management*, 14(7):337, 2021.
- [33] Jinghong Shu and Jin E Zhang. Testing range estimators of historical volatility. *Journal of Futures Markets: Futures, Options, and Other Derivative Products*, 26(3):297--313, 2006.
- [34] Dharminder Singh Virk. *Prediction of bitcoin price using data mining*. PhD thesis, Dublin, National College of Ireland, 2017.

Priedai

1 išėities kodas. Gražos skaičiavimas

```
1 vol = np.log(day_close.close.values/day_close.close.shift().values)
```

2 išėities kodas. Yang Zhang kintamumo skaičiavimas

```
1 def get_yang_zhang_volatility(source_data, T, alpha = 1.0):
2     log_oc = (source_data['open']/source_data['close'].shift(1)).apply(
3         np.log)
4     log_co = (source_data['close']/source_data['open']).apply(np.log)
5     avg_log_oc = log_oc.rolling(window=T).mean()
6     avg_log_co = log_co.rolling(window=T).mean()
7     sigma_o2 = (log_oc - avg_log_oc)**2
8     sigma_o2 = sigma_o2.rolling(window=T).mean()
9     sigma_c2 = (log_co - avg_log_co)**2
10    sigma_c2 = sigma_c2.rolling(window=T).mean()
11    sigma_rs2 = get_rogers_satchell_volatility(source_data, T)**2
12    k = (alpha - 1.0)/(alpha + (T+1)/(T-1))
13    # pagrindine formula
14    vol = (sigma_o2 + k * sigma_c2 + (1.0 - k) * sigma_rs2).apply(np.sqrt)
15    return vol
```

3 išėities kodas. Rogers Satchell kintamumo skaičiavimas

```
1 def get_rogers_satchell_volatility(source_data, T):
2     log_hc = (source_data['high']/source_data['close']).apply(np.log)
3     log_ho = (source_data['high']/source_data['open']).apply(np.log)
4     log_lc = (source_data['low']/source_data['close']).apply(np.log)
5     log_lo = (source_data['low']/source_data['open']).apply(np.log)
6     # pagrindine formula
7     vol = log_hc * log_ho + log_lc * log_lo
8     vol = vol.rolling(window=T).mean().apply(np.sqrt)
9     return vol
```