



Kauno technologijos universitetas
Matematikos ir gamtos mokslų fakultetas

Europos šalių investicinio patrauklumo vertinimas taikant duomenų tyrybą

Baigiamasis magistro studijų projektas

Gabrielė Japertaitė

Projekto autorė

Prof. dr. Jurgita Bruneckienė

Vadovė

Doc. dr. Tomas Ruzgas

Vadovas

Kaunas, 2022



Kauno technologijos universitetas
Matematikos ir gamtos mokslų fakultetas

Europos šalių investicinio patrauklumo vertinimas taikant duomenų tyrybą

Baigiamasis magistro studijų projektas
Didžiųjų verslo duomenų analitika (6213AX001)

Gabrielė Japertaitė

Projekto autorė

Prof. dr. Jurgita Bruneckienė

Vadovė

Doc. dr. Tomas Ruzgas

Vadovas

Doc. dr. Ineta Zykienė

Recenzentė

Doc. dr. Mantas Landauskas

Recenzentas

Kaunas, 2022



Kauno technologijos universitetas

Matematikos ir gamtos mokslų fakultetas

Gabrielė Japertaitė

Europos šalių investicinio patrauklumo vertinimas taikant duomenų tyrybą

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

1. baigiamąjį projektą parengiau savarankiškai ir sąžiningai, nepažeisdama(s) kitų asmenų autoriaus ar kitų teisių, laikydamasi(s) Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymo nuostatų, Kauno technologijos universiteto (toliau – Universitetas) intelektinės nuosavybės valdymo ir perdavimo nuostatų bei Universiteto akademinės etikos kodekse nustatytų etikos reikalavimų;
2. baigiamajame projekte visi pateikti duomenys ir tyrimų rezultatai yra teisingi ir gauti teisėtai, nei viena šio projekto dalis nėra plagijuota nuo jokių spausdintinių ar elektroninių šaltinių, visos baigiamojo projekto tekste pateiktos citatos ir nuorodos yra nurodytos literatūros sąrašė;
3. įstatymų nenumatytų piniginių sumų už baigiamąjį projektą ar jo dalis niekam nesu mokėjęs (-usi);
4. suprantu, kad išaiškėjus nesąžiningumo ar kitų asmenų teisių pažeidimo faktui, man bus taikomos akademinės nuobaudos pagal Universitete galiojančią tvarką ir būsiu pašalinta(s) iš Universiteto, o baigiamasis projektas gali būti pateiktas Akademinės etikos ir procedūrų kontrolieriaus tarnybai nagrinėjant galimą akademinės etikos pažeidimą.

Gabrielė Japertaitė

Patvirtinta elektroniniu būdu

Japertaitė, Gabrielė. Europos šalių investicinio patrauklumo vertinimas taikant duomenų tyrybą. Magistro studijų baigiamasis projektas / vadovai prof. dr. Jurgita Bruneckienė ir doc. dr. Tomas Ruzgas; Kauno technologijos universitetas, Matematikos ir gamtos mokslų fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Taikomoji matematika.

Reikšminiai žodžiai: investicinis patrauklumas, tiesioginės užsienio investicijos, klasterizavimas, regresinė analizė.

Kaunas, 2022. 57 p.

Santrauka

Greitas prisitaikymas prie kintančių sąlygų, tokių kaip spartėjanti globalizacija, auganti technologijų pažanga, Covid-19 pandemijos įtakoti darbo organizavimo pokyčiai ir kitos aplinkybės, didina šalių konkurencingumą. Pritraukti tiesiogines užsienio investicijas gali būti vis sunkiau, dėl to ypač svarbu didinti šalies investicinį patrauklumą. Šio darbo tikslas - palyginant skirtingus duomenų tyrybos metodus, klasterizuoti Europos šalis pagal investicinį patrauklumą lemiančius veiksnius bei prognozuoti tiesiogines užsienio investicijas.

Pirmiausiai darbe apibendrinami Lietuvos ir užsienio autorių tyrimuose išskirti investicinį patrauklumą ir tiesiogines užsienio investicijas lemiantys veiksniai. Remiantis literatūros analize, išskiriamos šios veiksmių grupės: rinkos patrauklumas, darbo rinkos patrauklumas, inovacinis patrauklumas, infrastruktūros prieinamumas, socialinė ir kultūrinė aplinka, verslo ir mokesstinė aplinka, tvarumo patrauklumas ir gamtos išteklių prieinamumas.

Vėliau aprašomi skirtingi klasterizavimo ir prognozavimo metodai, kurie naudojami empiriniame tyrime. Klasterinei analizei naudoti k-vidurkių, hierarachiniai, Gauso mišinių, Bajeso Gauso mišinių ir tankiu grįsti DBSCAN modeliai. Geriausiai rezultatus parodžiusiais metodais, 2020 metais buvo atskirti 3 Europos šalių klasteriai bei aprašyti jiems būdingi požymiai. Taip pat buvo palyginti klasterinės analizės rezultatai skirtingais laiko momentais. Geriausiam prognozavimo modeliui atrinkti buvo naudojami įvairūs regresijos metodai (pvz. LASSO, LARS ir kt.). 2021 metų tiesioginės užsienio investicijos buvo prognozuojamos dviem būdais: naudojant pilnų duomenų modelius ir klasterių modelius. Remiantis geriausiais modeliais kiekvienam klasteriui, tiesioginių užsienio investicijų mažėjimas prognozuojamas 5 valstybėse: Vokietijoje, Vengrijoje, Airijoje, Ispanijoje ir Švedijoje. Tuo tarpu pilnų duomenų modeliu net 14. Reikšmingiausi veiksniai užsienio investicijų įplaukoms vakarų klasterio šalyse yra gyventojų tankis, darbo jėgos kiekis ir NAWRU (angl. *Non-accelerating wage rate of unemployment*), tuo tarpu rytų klasterio šalims – kapitalo mokesčiai ir darbo jėgos kiekis.

Japertaitė, Gabrielė. Evaluation of investment attractiveness of European countries by data mining. Master's Final Degree Project / supervisors prof. dr. Jurgita Bruneckienė and doc. dr. Tomas Ruzgas; Faculty of Mathematics and Natural Sciences, Kaunas University of Technology.

Study field and area (study field group): Applied Mathematics.

Keywords: investment attractiveness, foreign direct investment, clustering, regression analysis.

Kaunas, 2022. 57 pages.

Summary

Rapid adaptation to changing conditions, such as accelerating globalization, growing technological advances, changes in the organization of work that are affected by the Covid-19 pandemic, and other circumstances are increasing countries' competitiveness. Attracting foreign direct investment may be more difficult, making it particularly important to increase the country's investment attractiveness. The aim of this work is to cluster European countries according to the factors determining investment attractiveness and to forecast foreign direct investment by comparing different data analysis methods.

First of all, the paper summarizes the factors that determine investment attractiveness and foreign direct investment in the research of Lithuanian and foreign authors. Based on literature analysis, the following groups of factors are distinguished: market attractiveness, labor market attractiveness, innovation attractiveness, accessibility of infrastructure, social and cultural environment, business and tax environment, sustainability attractiveness and availability of natural resources.

The different clustering and forecasting methods used in the empirical study are described later. K-means, hierarchical, Gaussian mixtures, Bayesian Gaussian mixtures, and density-based DBSCAN models were used for cluster analysis. In 2020 3 clusters of European countries were distinguished by using methods that showed the best results. Their characteristics were described. The results of the cluster analysis at different time points were also compared. Various regression methods (eg LASSO, LARS, etc.) were used to select the best forecasting model. Foreign direct investment in 2021 was projected in two ways: using full data models and cluster models. Based on the best models for each cluster, a decline in foreign direct investment is projected for 5 countries: Germany, Hungary, Ireland, Spain and Sweden. Meanwhile, the full data model shows decrease for even 14 countries. The most important factors for foreign investment inflows in the western countries' cluster are population density, labor force and NAWRU (Non-accelerating wage rate of unemployment), while in the eastern countries' cluster - capital taxes and labor force.

Turinys

Lentelių sąrašas	7
Paveikslų sąrašas	8
Įvadas	9
1. Literatūros apžvalga	10
1.1. Investicinio patrauklumo samprata.....	10
1.2. Investicinį patrauklumą lemiantys veiksniai	11
1.2.1. Ekonominiai veiksniai	11
1.2.2. Socialiniai ir kultūriniai veiksniai	12
1.2.3. Technologiniai veiksniai	12
1.2.4. Teisiniai ir politiniai veiksniai.....	13
1.2.5. Gamtiniai, geografiniai ir aplinkosauginiai veiksniai	14
1.3. Investicinio patrauklumo vertinimo metodai.....	15
1.4. Investicinio patrauklumo vertinimo modelis.....	20
1.5. Literatūroje naudojamų investicijų vertinimo metodų apžvalga	22
2. Šalių klasterizavimo ir prognozavimo metodų apžvalga	25
2.1. Šalių klasterizavimo metodai	25
2.1.1. Padalinimo metodai	28
2.1.2. Hierarchiniai metodai	28
2.1.3. Tankiu grįsti metodai.....	29
2.1.4. Pasiskirstymu grįsti metodai.....	30
2.1.5. Klasterizavimo metodų vertinimas.....	30
2.2. Šalių tiesioginių užsienio investicijų prognozavimo metodai	30
3. Empirinis tyrimas	35
3.1. Duomenys.....	35
3.2. Duomenų žvalgomoji analizė	36
3.2.1. Tiesioginių užsienio investicijų apžvalga tiriamuoju laikotarpiu.....	36
3.2.2. Išorinių tyrimė naudojamų rodiklių žvalgomoji analizė	37
3.3. Klasterinė analizė	42
3.3.1. Duomenų klasterizavimas, naudojant pradinį duomenų rinkinį.....	43
3.3.2. AMECO duomenų baze papildytas duomenų rinkinys	45
3.4. Duomenų regresinė analizė ir prognozavimas	49
3.4.1. Panelinių duomenų regresinė analizė (PooledOLS).....	49
3.4.2. Kintamųjų įtakos nustatymas remiantis atsitiktinio miško metodu.....	51
3.4.3. 2021 metų tiesioginių užsienio investicijų įplaukų prognozė, naudojant geriausius regresijos metodus.....	52
Išvados	54
Literatūros sąrašas	55
Priedai	58

Lentelių sąrašas

1 lentelė. Investicinio patrauklumo vertinimo veiksniai	21
2 lentelė. Klasterizavimo kategorijos [41]	27
3 lentelė. Geriausi kiekvienų metų klasterizavimo rezultatai, naudojant pradinį duomenų rinkinį..	43
4 lentelė. Geriausi kiekvienų metų klasterizavimo rezultatai, naudojant papildytą duomenų rinkinį.	45
5 lentelė. Pirmojo klasterio sutelktinės MKM regresinės analizės rezultatai.	49
6 lentelė. Antrojo klasterio sutelktinės MKM regresinės analizės rezultatai.....	50
7 lentelė. Tiesioginių užsienio investicijų įplaukų 2021 metų prognozė, remiantis geriausiais regresijos modeliais.	53

Paveikslų sąrašas

1 pav. GFICA indekso sandara [27].....	16
2 pav. Investicinio patrauklumo vertinimas [11]	17
3 pav. Regionų investicinį patrauklumą, sumanaus vystymosi kontekste, lemiančių veiksnių modelis [25]	19
4 pav. Elementarus regiono investicinio patrauklumo modelis [2].....	20
5 pav. Teorinis investicinio patrauklumo modelis	21
6 pav. Klasterių įvairovė [36]	27
7 pav. Hierarchinių klasterių sudarymo metodai [42].....	29
8 pav. Duomenų rinkinių sudarymo schema.....	35
9 pav. Tiesioginių užsienio investicijų įplaukų stačiakampė diagrama	37
10 pav. Bendro vidaus produkto, tenkančio vienam gyventojui, šalių stačiakampės diagramos	38
11 pav. Perkamosios galios pariteto šalių stačiakampės diagramos	38
12 pav. Skirtingų šalių kapitalo investicijų stačiakampės diagramos	39
13 pav. Nedarbo lygio šalių stačiakampės diagramos	40
14 pav. Skirtingų šalių prekybos atvirumo rodiklio stačiakampės diagramos.....	40
15 pav. Skirtingų šalių prekių ir paslaugų eksporto rodiklio stačiakampės diagramos	41
16 pav. Skirtingų šalių korupcijos kontrolės indekso stačiakampės diagramos	42
17 pav. Klasterių vizualizavimas skirtingais tiriamo laikotarpio momentais, naudojant pradinį duomenų rinkinį.....	45
18 pav. Klasterių vizualizavimas skirtingais tiriamo laikotarpio momentais, naudojant papildomą duomenų rinkinį.....	47
19 pav. Skirtingų kintamųjų įtaka atsitiktinio miško regresiniam modeliui, nenaudojant atskirų klasterių	51
20 pav. Skirtingų kintamųjų įtaka atsitiktinio miško regresiniam modeliui pirmojo klasterio atveju	52
22 pav. Skirtingų kintamųjų įtaka atsitiktinio miško regresiniam modeliui antrojo klasterio atveju	52

Įvadas

Tyrimo aktualumas. Tiesioginės užsienio investicijos yra svarbios šalies ekonomikos vystymuisi. Jos padeda kurti naujas darbo vietas, didinti gyventojų pajamas ir perkamąją galią. Greitas prisitaikymas prie kintančių sąlygų, tokių kaip spartėjanti globalizacija, auganti technologijų pažanga, Covid-19 pandemijos įtakoti darbo organizavimo pokyčiai ir kitos aplinkybės, didina šalių konkurencingumą. Pritraukti tiesiogines užsienio investicijas gali būti vis sunkiau, dėl to ypač svarbu didinti šalies investicinį patrauklumą.

Investicinio patrauklumo įvertinimas yra svarbus bet kokio investavimo sprendimo priėmimo aspektas. Sprendimai yra grindžiami daugiakrypčiu, daugiakriteriniu daugelio veiksnių ir tendencijų vertinimu. Pasekmės tiek investuotojui, tiek regiono ir visos šalies ekonomikai priklauso nuo vertinimo teisingumo [1]. Taikant duomenų tyrybą sukuriama modeliai, padedantys įžvelgti tendencijas, priimti ekonominius sprendimus.

Geriausio duomenų tyrybos metodo pasirinkimas gali tapti dideliu iššūkiu tyrėjui. Dėl gausaus galimų metodų pasirinkimo gali kilti abejonių, kuris metodas yra taikytinas sprendžiant konkrečios srities problemą. Šie sprendimai, augant įrankių komercinėse programinėse įrangose skaičiui, reikalauja vis daugiau metodologinės patirties.

Darbo tikslas: Palyginant skirtingus duomenų tyrybos metodus, klasterizuoti Europos šalis pagal investicinį patrauklumą lemiančius veiksnius bei prognozuoti tiesiogines užsienio investicijas vienus metus į priekį.

Uždaviniai:

1. Apibrėžti investicinio patrauklumo ir tiesioginių užsienio investicijų sampratą ir aprašyti šalies investicinį patrauklumą lemiančius veiksnius;
2. Apžvelgti metodus, naudojamus šalių klasterizavimui ir investicinio patrauklumo prognozavimui;
3. Atlikti šalių klasterinę analizę, išskiriant šalis, turinčias panašius tiesiogines užsienio investicijas į šalį lemiančius rodiklius;
4. Sudaryti tiesiogines užsienio investicijas prognozuojančius regresijos modelius bei prognozuoti tiesiogines užsienio investicijas, remiantis geriausiais atrinktais modeliais.

Tyrimo metodai: mokslinės literatūros analizė, sisteminimas ir išvadų generavimas, klasterizavimas, prognozavimas, vizualizavimas.

1. Literatūros apžvalga

Šalies (regiono) investicinį patrauklumą ir tiesiogines užsienio investicijas lemiančių veiksnių tema yra plačiai analizuota tiek Lietuvos, tiek užsienio literatūroje. Šiuolaikiniam mokslui trūksta bendro sutarimo dėl investicinio patrauklumo esmės ir struktūrinio vientisumo [2].

1.1. Investicinio patrauklumo samprata

Vienas pagrindinių kiekvienos šalies (regiono) tikslų yra ekonomikos vystymasis. Stiprėjant ekonomikai gerėja šalies (regiono) gyvenimo lygis, auga įmonių ir asmenų realiosios pajamos, didėja gamybos apimtys bei šalies (regiono) bendras vidaus produktas. Viena iš sąlygų tolygiam ekonomikos vystymuisi yra investicijų pritraukimas [3].

Remiantis Lietuvos Respublikos investicijų įstatymu, investicijos yra piniginės lėšos ir įstatymais bei kitais teisės aktais nustatyta tvarka įvertintas materialusis, nematerialusis ir finansinis turtas, kuris investuojamas siekiant iš investavimo objekto gauti pelno (pajamų), socialinį rezultatą (švietimo, kultūros, mokslo, sveikatos ir socialinės apsaugos bei kitose panašiose srityse) arba užtikrinti valstybės funkcijų įgyvendinimą [4].

Pagal investuotojo nuolatinę buveinę, investicijos yra skirstomos:

1. Vidaus investicijos – šalies (regiono) fizinių ir juridinių asmenų bei juridinio asmens teisių neturinčių įmonių investicijos toje šalyje (regione).
2. Užsienio investicijos - užsienio valstybių, tarptautinių organizacijų, užsienio fizinių ir juridinių asmenų investicijos toje šalyje (regione) [4].

Taip pat investicijos skirstomos:

- pagal įtaką ūkio subjektui - tiesioginės ir netiesioginės (portfelinės) investicijos;
- investuotojo statusą - vidaus, privačios, užsienio valstybių ir tarptautinių organizacijų investicijos;
- investavimo objektą – kapitalo ir finansinės [4].

Svarbia ekonominių augimą skatinančia sąlyga laikomos tiesioginės užsienio investicijos. Jos pritraukia kapitalą į projektus, kurių vietinis kapitalas negalėtų patenkinti [5]. Tiesioginės užsienio investicijos siejamos su kuriama ilgalaikiu interesu ir tikslu užmegzti naujus ryšius, galinčius gauti tam tikrą ekonominį, socialinį, kultūrinį, technologinį ar kitokį rezultatą, kuris padėtų asmeniniams, visuomeniniams tikslams [6].

Ruplienė ir Garšvienė (2008), analizuodamos tiesioginių užsienio investicijų įtaką šalies ekonomiam augimui, pabrėžė, kad investicijos naudingos tiek kapitalą eksportuojančiai, tiek kapitalą importuojančiai pusei. Investuotojai gali eksportuoti kapitalą į šalį (regioną), kuriame palūkanų norma ir pelno norma dėl kapitalo trūkumo aukštesnė, ir tokiu būdu užsitikrinti didesnes kapitalo pajamas. Tuo tarpu šaliai (regionui), kurioje investuojama, naudinga bendrojo produkto augimo, investicijų į ekonomiką ir gamybą bei nedarbo mažėjimo prasme. Taip pat gali būti gaunama galimybė naudotis naujausiomis technologijomis, valdymo žiniomis ir patirtimi, kas gali padidinti darbo jėgos kokybę, vidutinį darbo užmokestį, nulemti vidaus rinkos prisotinimą kokybiškų ir santykinai pigių prekių, struktūrinius ekonomikos pokyčius, padėti vykdyti regioninę plėtros politiką [7].

Žitkus ir Mickevičienė (2013) analizuodami konkurencingumą, kaip regiono plėtros siekinį, pastebėjo, kad literatūroje kai kurie autoriai konkurencingumą sutapatina su investiciniu patrauklumu [8]. Kaip vieną iš pavyzdžių pateikė Camagni (2002), kuris pabrėžia, jog regiono investicinio patrauklumo ir konkurencingumo veiksniai yra gana panašūs: regiono pasiekiamumas, infrastruktūra, natūralios aplinkos kokybė, inovatyvumas ir pan. Tačiau sutapatinti šias sąvokas būtų neteisinga. Investicinis patrauklumas gali būti laikomas požymiu, kurį stiprinant šalis gali tapti konkurencingesnė [9]. Kruk (2010) teigė, kad investicinis patrauklumas yra susijęs su ūkinės veiklos vykdymo vietos pasirinkimu, o konkurencingumas – su to pasirinkimo rezultatais [10].

Investuotojai, rinkdamiesi kur eksportuoti kapitalą, atsižvelgia į šalies (regiono) investicinį patrauklumą. Fisher (1999) teigė, jog investicinis patrauklumas yra finansinių ir ekonominių rodiklių rinkinys, sudarantis bendrą išorinės aplinkos vertinimą, įskaitant politinę, ekonominę, socialinę bei teisinę šalies (regiono) situaciją [11].

Taip pat, jis gali būti suprantamas kaip šalies (regiono) gebėjimas pritraukti investicijas į šį regioną. Norėdami didinti konkurencingumą bei pritraukti išorės kapitalo, šalys (regionai) turi skirti dėmesį palankios investicinės aplinkos formavimui. Takhumova ir kiti autoriai (2018) analizuodami investicinio patrauklumo augimo kryptis, taip pat pabrėžė, kad investiciniu patrauklumu gali būti laikomas konkretaus investuotojo požiūris į investicinį objektą [12].

1.2. Investicinį patrauklumą lemiantys veiksniai

Atliekant literatūros analizę pastebėta, kad tyrimuose išskiriami skirtingi ekonominiai, socialiniai, technologiniai, politiniai ir geografiniai veiksniai.

1.2.1. Ekonominiai veiksniai

Ruplienė ir kiti autoriai (2008), analizuodami tiesiogines užsienio investicijas lemiančius veiksnius, kaip vieną iš svarbiausių makroekonomikos rodiklių išskyrė rinkos dydį, kurį būtų galima apibūdinti gyventojų skaičiumi ir jų pajamomis. Šis veiksnys ypač svarbus ieškant naujų rinkų. Ekonometriniuose tyrimuose rinkos dydis dažniausiai vertinamas, naudojant bendrojo vidaus produkto ir vienam gyventojui tenkančio bendrojo vidaus produkto dydžius [14].

Literatūroje vienu iš investicinio patrauklumo veiksnių taip pat minimas ekonominio atvirumo lygis. Jis skaičiuojamas kaip užsienio prekybos apyvartos (t. y. eksporto ir importo sumos) santykis su bendruoju vidaus produktu [15]. Teoriškai šio rodiklio įtaka šalyje gali būti apibūdinama keliais aspektais:

1. Mažesni užsienio prekybos apribojimai;
2. Didelė importo apimtis, dėl ko atsiranda galimybė perkelti gamybą į tą pačią šalį ir sumažinti transporto kaštus;
3. Šalies aktyvumas pasaulinėje prekyboje. Tai lemia didesnę gamybos veiksnių (kapitalo) migracijos laipsnį [14].

Kitas ekonominis veiksnys, kuris yra minimas moksliniuose tyrimuose – eksportas. Stankevičienė ir Lakštutienė (2012) teigė, kad jis yra vienas iš svarbių veiksnių, susijusių su užsienio prekybos atvirumu bei prekybos liberalizavimu [16]. Radionov (2021) teigė, kad eksporto augimas rodo potencialios investicijos grąžos didėjimą [2].

Investuotojams, besirenkantiems šalį, taip pat gali būti svarbu šalies ekonomikos augimas. Jis siejamas su galimu investicijų pelningumu ir lemia didesnės apimties tiesioginių užsienio investicijų pritraukimą [15].

1.2.2. Socialiniai ir kultūriniai veiksniai

Vienas dažniausiai literatūroje minimų veiksnių, renkantis kur investuoti, yra darbo jėgos pajėgumai. Mustafakulov (2017) pagrindiniais rodikliais išskiria gyventojų tankumą, darbo jėgos išteklius ir kokybę, vidutinį mėnesinį darbo užmokestį, bedarbių skaičių, dirbančiųjų amžių, bedarbių amžių, įgūdžius ir kvalifikaciją. Dėl Covid-19 pandemijos metu pasikeitusio darbo organizavimo, kuomet tam tikros veiklos gali būti vykdomos skirtingose šalyse, darbo jėgos kainos bei įgūdžių ir kvalifikacijos svarba gali didėti [17].

Mustafakulov (2017) taip pat išskyrė ir socialines rizikas, kurios gali turėti neigiamą įtaką investicijų pritraukimui:

1. Gyvenimo būdas;
2. Socialinės infrastruktūros būklė;
3. Demografinė būklė (populiacijos skaičius ir amžius);
4. Šeimų skaičius ir jų struktūra;
5. Migracija ir stratifikacija;
6. Santykis tarp turtingų ir nepasiturinčių;
7. Pajamų lygis ir šaltinis;
8. Perkamoji galia;
9. Išlaidų ir poreikių struktūra bei lygis;
10. Švietimas ir kultūra,
11. Nedarbas ir darbuotojų kaita [17].

Kriminalinė rizika yra minima kaip viena svarbiausių rizikų, į kurias atsižvelgia investuotojai. Mustafakulov (2017) teigia, jog pagrindinės kryptys norint įvertinti šią riziką yra: kriminalinių įvykių lygis bei korupcijos lygis regione [17].

Goraieb ir kiti (2018) analizavo kultūrinių veiksnių įtaką tiesioginėms užsienio investicijoms. Autoriai teigė, jog, nors globalizacijos dėka auga integracija tarp šalių, kultūriniai skirtumai vis dar turi įtakos investuotojams. Šie skirtumai gali sukelti trintį, abejingumą ir nesuderinamus verslo santykius, bet taip pat gali turėti ir teigiamos įtakos, papildyti viena kitą [18].

Hiestand (2005), analizuodamas tiesiogines užsienio investicijas lemiančius veiksnius Taivane, teigė, kad literatūroje apie žmogiškąjį kapitalą, raštingumo lygis dažnai naudojamas kaip išsilavinusios darbo jėgos pakaitinis rodiklis. Numatyta, kad kuo didesnis raštingumas, tuo didesnė ir darbininko įdarbinimo kaina. Taigi šis kintamasis, kuris teigiamai koreliuoja su realiuoju darbo užmokesčiu, netiesiogiai fiksuoja informaciją apie žmogiškojo kapitalo įdarbinimo išlaidas investuojamoje šalyje. Jei auga raštingumo lygis, motyvacija investuoti užsienyje yra didesnė, nes kitoje šalyje patiriamos išlaidos gali būti mažesnės [19].

1.2.3. Technologiniai veiksniai

Infrastruktūra, technologijos ir jų plėtra, inovacijos yra svarbūs investicinio patrauklumo vertinimui ir yra dažnai išskiriami moksliniuose tyrimuose (Dzwigol ir kt. (2019); Kharlamova (2014); Dorożyński

(2016); Mustafakulov (2017); Zykienė (2018); Ruplienė ir kt. (2008); Tocar (2019)). Snieška ir kiti (2019) teigė, jog veiksniai, lemiantys tinklo ir infrastruktūros patrauklumą, sudaro sąlygas, formuoja ir skatina subjektus pasinaudoti vietovėje siūlomomis galimybėmis ir ištekliais. Tuo tarpu inovacijų patrauklumą lemiantys veiksniai susiję su gebėjimu kurti vertę ir veikti pritaikant naujus, geresnius (nestandartinius) sprendimus [20].

Infrastruktūros aprūpinimas yra susijęs su turimo vandens ir elektros energijos būkle regione, oro uostų, automobilių kelių ir geležinkelių prieinamumu bei informacijos–komunikacijos technologijų plėtra [17]. Tocar (2019) sudarydamas investicinio patrauklumo indeksą, infrastruktūros svarbos vertinimui naudojo prekybos ir transporto infrastruktūros indeksą, logistinės veiklos indekso elementą (World bank) [15].

Tocar (2019) taip pat pabrėžė ir technologijų įtaką. Ją vertino pagal aukštų technologijų eksporto kintamąjį [15]. Technologijų svarba taip pat pabrėžiama Stankevičienės ir Lakštutienės (2012) atliktame tiesioginių užsienių investicijų pritraukimą lemiančių veiksnių Baltijos šalyse tyrime. Autorės teigė, kad išlaidų moksliniams tyrimams ir technologijų plėtrai veiksnyms turėjo teigiamą ir reikšmingą įtaką investicijų pritraukimui [16].

Rodionov (2021), analizavęs investicinio patrauklumo vertinimo pasikeitimus po Covid-19 pandemijos, kaip vieną iš pagrindinių veiksnių išskyrė inovacijas, kurias išreiškė išlaidomis technologijoms ir pažangių gamybos technologijų kiekiu [2].

Tuo tarpu Mustafakulov (2017) inovacijų pajėgumus apibrėžė indikatoriais:

1. Regiono mokslinis - techninis pajėgumas;
2. Moksliniai – techniniai pasiekimai;
3. Mokslinį laipsnį turintys asmenys;
4. Mokslinių – techninių projektų kiekis;
5. Asmenys, dalyvaujantys mokslinėje veikloje;
6. Tyrimų ir plėtros centrai bei jų padaliniai regionuose [17].

1.2.4. Teisiniai ir politiniai veiksniai

Globerman ir Shapiro (2003) analizavo valdymo infrastruktūros svarbą tiesioginėms užsienio investicijoms iš Jungtinių Amerikos valstijų. Valdymo infrastruktūra apima įstatymų, reguliavimo ir teisinės sistemas, sąlygojančias sandorių laisvę, nuosavybės teisių saugumą ir vyriausybės bei teisinių procesų skaidrumą [21]. Autoriai tyrime naudoja Kaufmann ir kitų (1999) išskirtus aspektus:

- Balso, politinė ir pilietinė laisvė.
- Politinis nestabilumas, terorizmas ir smurtas.
- Įstatymo taisyklė, nusikalstamumas, sutarties vykdymo ir nuosavybės teisės.
- Korupcijos lygis valstybinėse ir privačiose institucijose.
- Reguliavimo mastas ir rinkos atvirumas, įskaitant tarifus ir importo kontrolę.
- Valdžios efektyvumo priemonės [22].

Globerman ir Shapiro (2003) atliktas tyrimas parodė, kad šalys, kurios nepasiekia minimalaus veiksmingo valdymo slenksčio, greičiausiai negaus jokių JAV investicijų. Šalys, kurios negauna šių investicijų, paprastai neskatina laisvų ir skaidrių rinkų, jų vyriausybės yra neveiksmingos ir dažnai turi kitokias teisinės sistemas [21].

Ruplienė ir kiti (2008) kaip vieną pagrindinių tiesiogines užsienio investicijas lemiančių veiksnių išskyrė institucijas. Dėl jų nestabilumo, instituciniai veiksniai gali daryti neigiamą įtaką. Nuo jų priklauso rinkos efektyvumas, galutiniai gamybos kaštai, informacijos prieinamumas [14]. Vertinant institucijų pajėgumus, Mustafakulov (2017) išskiria vartotojų teises, įmonės valdymo principus, sveiką konkurencinę aplinką, sąlygas vykdyti verslą ir informacijos teikimą, buhalterinės ir statistinės ataskaitos, finansų rinkos ir institucijų plėtrą [17].

Snieskos ir Zykienės (2015) atliktame Alytaus miesto investicinį patrauklumą lemiančių veiksnių tyrime, apklausti investuotojai kaip efektyviausią priemonę, skatinančią verslo kūrimąsi ir plėtrą, išskyrė finansines priemones, verslo paskolas [23].

Taip pat labai svarbus veiksnys – mokesstinė aplinka. Užsienio investuotojams taikoma mažesnė mokesčių našta veikia kaip paskata investuoti toje šalyje [16]. Doroczyński (2016), analizuodamas Višegrado grupės šalių investicinį patrauklumą, pastebėjo, kad Lenkijoje sumažinus fiskalinę naštą, verslo pajamų mokesčiams, investuotojai padidino tiesioginių užsienio investicijų kiekį šios šalies padaliniuose [24].

Vertinant politinę riziką Mustafakulov (2017) teigia, jog reiktų atkreipti dėmesį į žmonių požiūrį į politinių partijų veiklą ir procesus, rinkimų aktyvumą, valdžios reputaciją, šalies įstatymų leidžiamosios ir vykdomosios valdžios veiklą, tarptautinių santykių būklę, tautinių ir etninių grupių veiklą [17].

1.2.5. Gamtiniai, geografiniai ir aplinkosauginiai veiksniai

Hiestand (2005) teigė, kad investuotojai yra linkę investuoti geografiškai artimose šalyse. Viena iš priežasčių yra ta, kad tai gali sumažinti kai kurias transportavimo išlaidas. Taip pat svarbiu veiksniu galima laikyti kultūrinius skirtumus. Užsienio investuotojai gali labiau investuoti į šalį, kurios kultūra yra panaši. Tai gali sumažinti informacines išlaidas verslo sandoriams tarp dviejų šalių. Be to, užsienio firmoms lengviau stebėti ir bendrauti su darbuotojais šalyje, kur bendra kalba ir panaši kultūra [19].

Geografiniai veiksniai yra susiję ne tik su atstumu iki šalies, kurioje investuojama. Tai taip pat aktualu šalies dydis, jos prieiga prie jūros, topografija, vidutinis atstumas iki jos sienų [18].

Mustafakulov (2017) išskiria gamtinius išteklius: mineralinius išteklius, žemę, vandenį, kuro energiją, įvairių rūšių rūdis ir metalus [17].

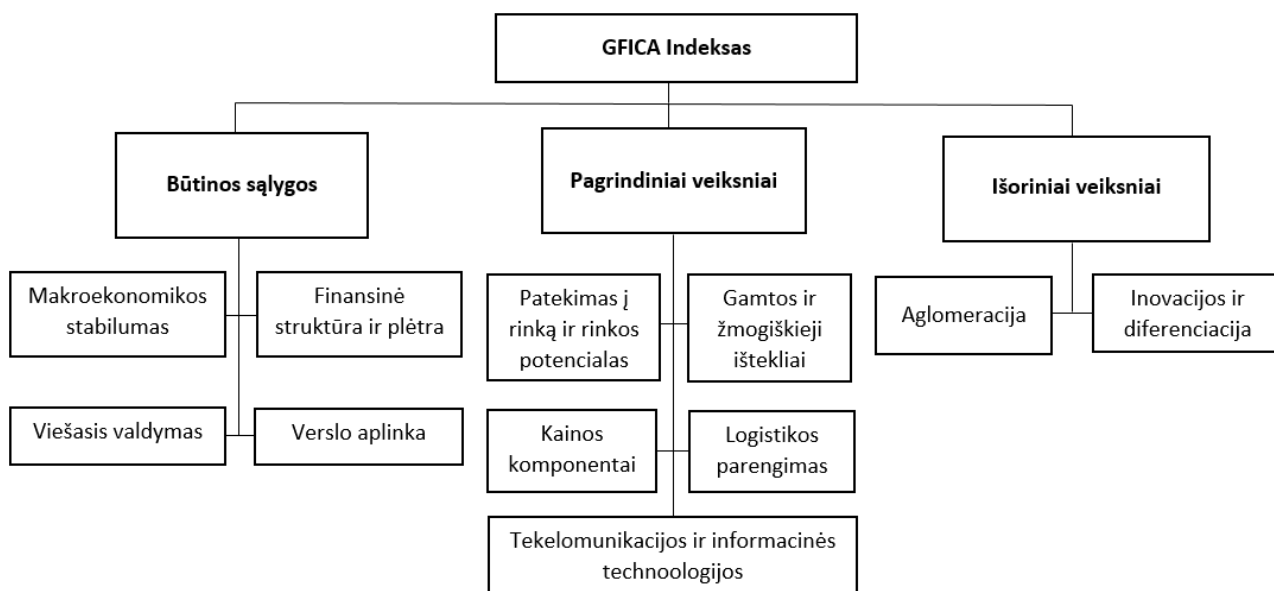
Žmonėms, įstaigoms, organizacijoms ir šalims vis labiau rūpinantis tvarumu ir aplinkosauga, vienu iš svarbiausių veiksnių, renkantis kur investuoti, tapo ekologija. Investuotojai taip turi atsižvelgti į su ekologija susijusius rizikas: aplinkos taršą, aplinką be spinduliuotės, vandens ir biologinius išteklius, dirvožemio ir žemės išteklių pokyčius [17].

Zykienė (2018) vienu iš veiksnių, turinčių įtakos investiciniam patrauklumui, sumanaus vystymosi kontekste, minėjo darnaus vystymosi principų įgyvendinimą [25]. Remiantis Lietuvos Respublikos aplinkos ministerijos internetiniame puslapyje (2021) pateiktu apibrėžimu, darnus vystymasis - modernios ir atsakingos valstybės bei jos visuomenės raidos kelias, kuris remiasi trimis lygiavertėmis politikos sritimis – aplinkos apsauga, ekonominiu vystymusi ir socialine gerove. Darnaus vystymosi tikslai iki 2030 m.:

1. Panaikinti visų formų skurdą;
2. Panaikinti bada, užtikrinti apsirūpinimą maistu ir geresnę mitybą, skatinti darnų žemės ūkį;
3. Užtikrinti sveiką gyvenimą ir skatinti visų amžiaus grupių gerovę;
4. Užtikrinti visaapimančią ir lygiavertį kokybišką švietimą ir skatinti visą gyvenimą trunkančią mokymąsi;
5. Pasiiekti lyčių lygybę ir įgalinti moteris ir mergaites;
6. Užtikrinti visiems vandens prieinamumą, darnų valdymą ir sanitariją;
7. Užtikrinti visiems prieigą prie prieinamos, patikimos, darnios ir modernios energijos;
8. Skatinti tvarų, visaapimančią ir darnų ekonominį augimą, produktyvų įdarbinimą ir tinkamą darbą;
9. Plėtoti atsparią infrastruktūrą, skatinti visa apimančią ir darnią industrializaciją ir skatinti inovacijas;
10. Sumažinti nelygybę valstybėse ir tarp valstybių;
11. Padaryti miestus ir žmonių apgyventas vietas saugias, atsparias ir darnias;
12. Užtikrinti darnaus vartojimo ir gamybos modelius;
13. Imtis skubių kovos su klimato kaita ir jos padariniais veiksmų;
14. Išsaugoti ir darniam vystymui naudoti vandenynų ir jūrų išteklius;
15. Saugoti, atkurti, skatinti darnų žemės ekosistemų naudojimą, darniai valdyti miškus, kovoti su dykumėjimu, sustabdyti žemės degradaciją, sustabdyti bioįvairovės praradimą;
16. Skatinti taikias visuomenės darniam vystymuisi, užtikrinti prieigą prie teisingumo visiems ir sukurti efektyvias, atskaitingas visų lygių institucijas;
17. Stiprinti vystymosi darbotvarkės įgyvendinimo priemones ir pagyvinti globalią partnerystę darniam vystymuisi [26].

1.3. Investicinio patrauklumo vertinimo metodai

Tiek literatūroje, tiek žiniasklaidoje naudojamas ne vienas investicinio patrauklumo vertinimo indeksas. Vienas dažniausiai naudojamų yra pasaulinis patrauklumo tiesioginėms užsienio investicijoms indeksas (angl. *global foreign direct investment country attractiveness index (GFICA)*). Remiantis Ben Jelili (2020) GFICA ataskaita, šis indeksas reitinguoja 109 šalių rinkinį, 96% ir 97% pasaulio tiesioginių užsienio investicijų srautų ir atsargų, atsižvelgiant į jų investicinį patrauklumą. Pagrindiniai šio indekso elementai skirstomi į tris grupes: prielaidas, pagrindinius veiksnius ir išorinius veiksnius [27]. Indekso sandara ir šioms grupėms priskiriami veiksniai pateikti 1 paveiksle.



1 pav. GFICA indekso sandara [27]

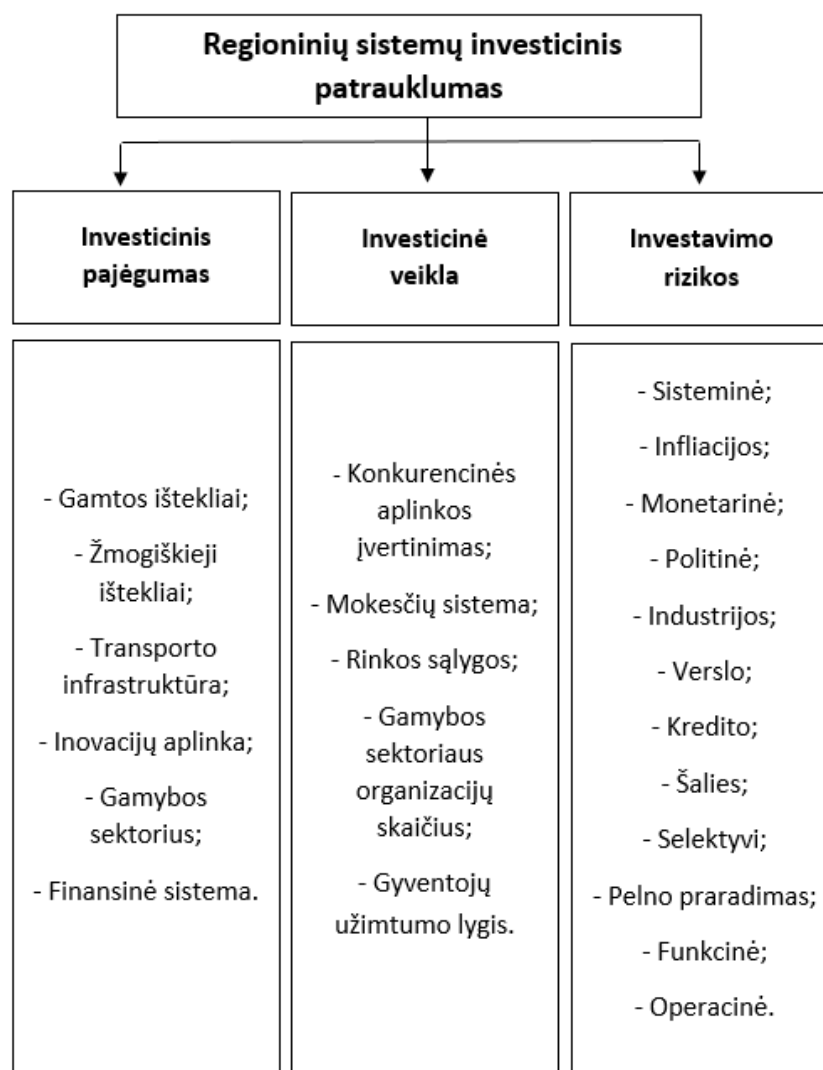
Kitas indeksas, kuris yra naudojamas investicinio patrauklumo vertinimui, yra Kearney Tiesioginių užsienio investicijų pasitikėjimo indeksas (angl. *FDI Confidence Index*). Indeksas sudaromas, naudojant pirminius duomenis, gautus apklausus pirmaujančių pasaulio korporacijų vyresnius vadovus. Reitingai apskaičiuojami remiantis klausimais apie respondentų įmonių tikimybę tiesiogiai investuoti į tam tikrą rinką per ateinančius trejus metus. Indeksas reitinguoja 25 valstybes [28].

Europos šalių investiciniam patrauklumui vertinti taip pat naudojama EY patrauklumo apklausos programa. Vertinimas remiasi dviejų etapų metodika, kurių metu analizuojamas realus ir suvokiamas investicijų patrauklumas šalyje (regione). Išvados grindžiamos tarptautinių ir vietinių lyderių bei sprendimus priimančių atstovų grupių nuomone. Pateikiamos dvylikos Europos šalių investicinio patrauklumo vertinimo ataskaitos: Austrijos, Belgijos, Kipro, Prancūzijos, Vokietijos, Graikijos, Maltos, Portugalijos, Rumunijos, Škotijos, Šveicarijos ir Jungtinės Karalystės [29].

Mustafakulov (2017), Takhumova ir kiti autoriai (2018) regionų investicinio patrauklumo vertinimą siūlo skirstyti į tris etapus:

- investavimo pajėgumo,
- investavimo veiklos,
- investavimo rizikos [16].

Šie veiksniai kartu sudaro šalies (regiono) investicinį klimatą. Takhumovos ir kitų autorių (2018) sudarytas investicinio patrauklumo vertinimo modelis pateiktas 2 paveiksle.



2 pav. Investicinio patrauklumo vertinimas [11]

Mustafakulov (2017) investavimo pajėgumą (potencialą) apibūdino kaip atvirumą investicijų srautui ir ekonominių išteklių prieinamumą regione [17]. Be 1 paveiksle matomų, Takhumovos ir kitų autorių (2018) minėtų veiksnių – gamtos, žmogiškųjų išteklių, infrastruktūros, inovatyvumo, gamybos ir finansinės sistemos, jis taip pat išskyrė institucinį, turistinį ir vartotojų (paklausos) pajėgumus.

Investavimo veikla apibūdina investicijų intergravimo aspektus. Mustafakulov (2017) išskiria veiklos pradžiai reikalingus išteklius, ekonominį potencialą, kapitalo centralizaciją, investicinių projektų įgyvendinimo intensyvumą, produktų energijos vartojimo efektyvumą, projektų įgyvendinimo laiko trumpinimą [17]. Tuo tarpu Takhumova ir kiti (2018) pabrėžia konkurencingumą rinkoje, mokesčių sistemą [12].

Investicinę riziką Mustafakulov (2017) apibrėžia kaip tikimybę gauti pelno ar nuostolių, priklausomai nuo sudarytų sąlygų investuotojams. Autorius išskiria šešis rizikos tipus: finansinę, ekonominę, socialinę, kriminalinę, aplinkosaugos ir administracinę riziką [17]. Takhumova ir kiti (2018) taip pat išskiria politinę, veiklos ir kitas smulkesnes rizikos rūšis [12].

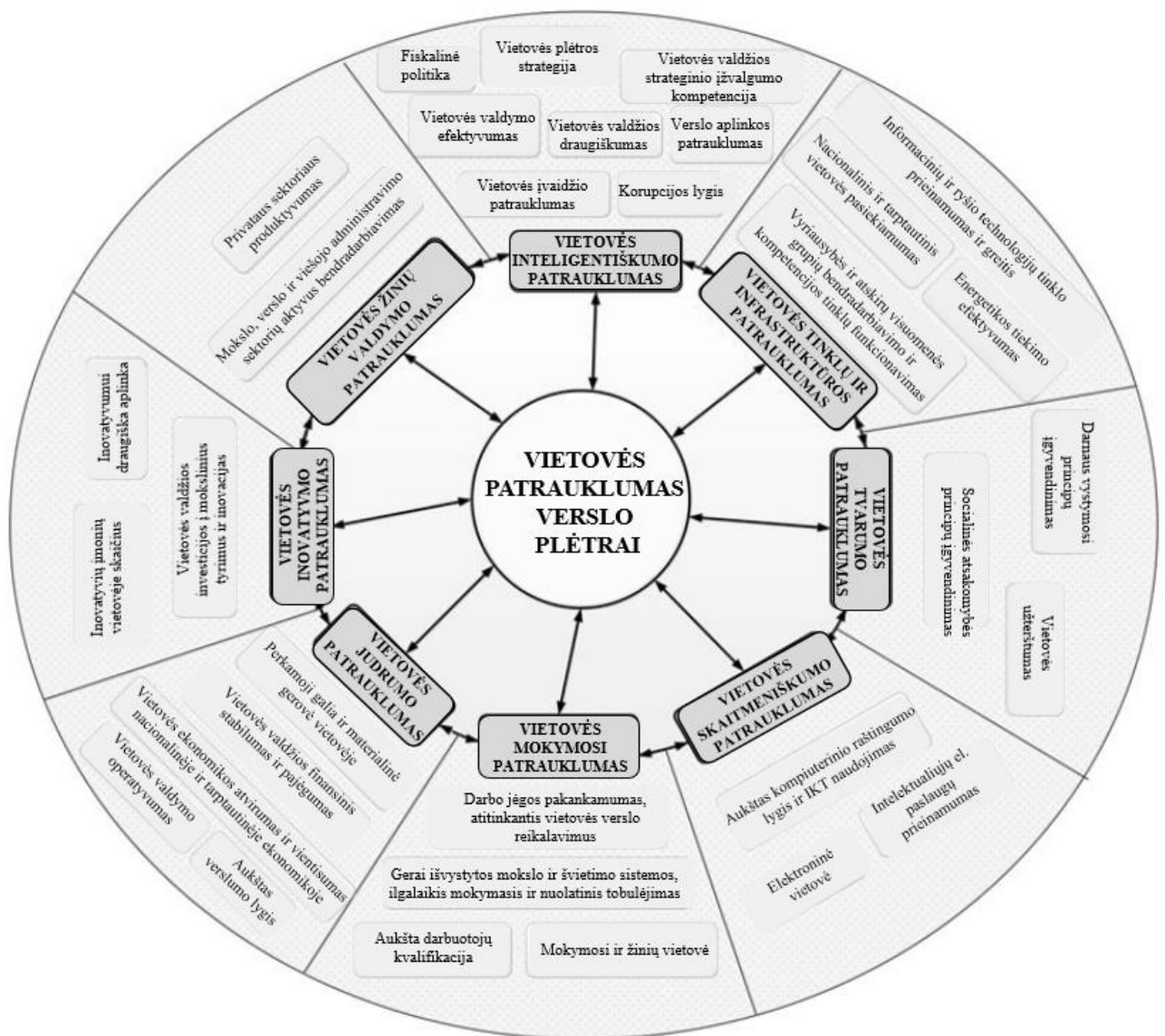
Ilgalaikėje perspektyvoje, remiantis minėtais veiksniais, investicinis klimatas turi būti stabilus, lankstus išteklių, paklausos pokyčiams. Be to, labai svarbus inovacijų vystymas bei novatoriškas verslumas [17].

Šiek tiek kitoks investicijų patrauklumą lemiančių veiksnių skirstymas pateiktas Dzwigol ir kitų autorių (2019) tyrime. Remdamiesi 2012 metais pateikta UNCTAD metodologija Pasaulio Investicijų ataskaitoje (angl. *World Investment Report*) bei kitais šaltiniais, autoriai išskyrė penkias veiksnių grupes:

1. Rinkos patrauklumas (rinkos dydis, perkamoji galia, rinkos augimo potencialas);
2. Darbo kaina ir kokybė (darbo kaštai, pasiekiamumas, sveikata, išsilavinimas);
3. Būtinios infrastruktūros prieinamumas (transportas, galia, komunikacija);
4. Gamtos išteklių prieinamumas (žemės ūkio ištekliai, gamtos išteklių išnaudojimas);
5. Investicijų saugumas (kriminalinis, ekologinis, politinis) [1].

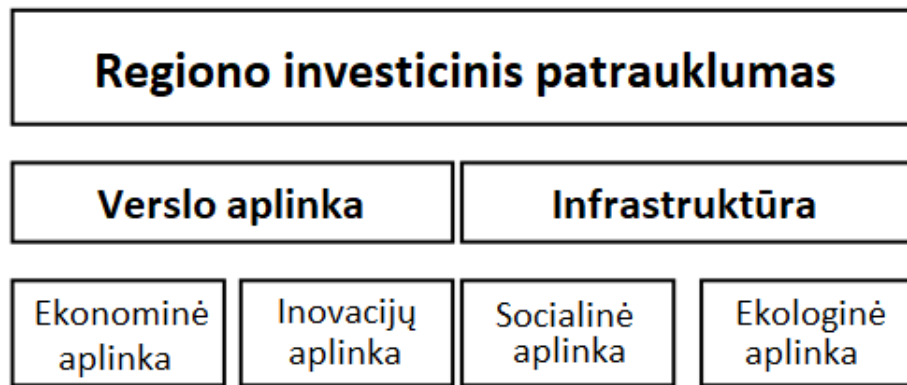
Investicinio patrauklumo tema plėtojama ir lietuvių autorių darbuose. Zykienė (2018) disertacijoje „Vietovės patrauklumo verslo plėtrai sumanaus vystymosi kontekste vertinimas“ (angl. „*Evaluation of location's attractiveness for business growth in the context of smart development*“) sudarė regionų investicinį patrauklumą lemiančių veiksnių modelį sumanaus vystymosi kontekste. (3 pav.)

Investicinio patrauklumo vertinime dėmesys skiriamas vietovės inteligentiškumo patrauklumo, tinkliškumo bei infrastruktūros patrauklumo, darnumo patrauklumo, skaitmeniškumo patrauklumo, mokymosi patrauklumo, judrumo patrauklumo, inovatyvumo patrauklumo ir grindžiamumo žiniomis patrauklumo kokybinėms bei kiekybinėms charakteristikoms. [25]



3 pav. Regionų investicinį patrauklumą, sumanaus vystymosi kontekste, lemiančių veiksnių modelis [25]

Rodionov ir kiti autoriai (2021) analizavo investicinio patrauklumo vertinimo pasikeitimus, kuriuos lėmė Covid-19 pandemija. Dėl karantino įvedimo, nemaža dalis profesinės komunikacijos buvo perkelta į nuotolinio bendravimo priemones. Šis skaitmeninių technologijų integravimas į beveik visus verslo procesus galiausiai gali visiškai pakeisti pasaulio ekonomiką ir sumažinti darbo jėgos teritorinio priklausymo reikšmę. Tradicinių šalies (regiono) investicinio patrauklumo komponentų įtaka yra pastoviai mažėjanti, nes šiuolaikinį verslą galima kurti ir be teritorinės vienybės. Autoriai pabrėžė šių charakteristikų reikšmės augimą: darnios regiono plėtros, socialinės aplinkos savybių ir socialinės infrastruktūros nuoseklumo [2]. Rodionov ir kitų autorių (2021) regiono investicinio patrauklumo modelis pateiktas 5 paveiksle.



4 pav. Elementarus regiono investicinio patrauklumo modelis [2]

Regiono investicinį patrauklumą apibrėžia verslo aplinkos būklė, išreikšta potencialia investicijų graža, ekonominės ir politinės rizikos lygiu, inovaciniu potencialu, ekonominių santykių nuoseklumu ir stabilumu. Taip pat integracijos ir plėtros sąnaudų lygis bei ne verslo infrastruktūros padėtis, kur išskiriama aukštas gyvenimo lygis, gerai išvystyta švietimo ir sveikatos priežiūros aplinka, pakankamas ekologinių reikalavimų lygis bei kultūrinių ir etninių kliūčių verslui nebuvimas [2].

1.4. Investicinio patrauklumo vertinimo modelis

Apibendrinant autorių iškirtus veiksnius ir metodus, investicijų patrauklumą lemiančius veiksnius galėtume sugrupuoti pagal 8 savybes, apibūdinančias šalies investicinį patrauklumą:

- rinkos patrauklumą,
- darbo rinkos patrauklumą,
- inovacinį patrauklumą,
- infrastruktūros prieinamumą,
- socialinę ir kultūrinę aplinką,
- verslo ir mokestinę aplinką,
- tvarumo patrauklumą bei gamtos išteklių prieinamumą (5 pav.).

Veiksnių grupės, veiksniai ir šaltiniai pateikti 1 lentelėje.



5 pav. Teorinis investicinio patrauklumo modelis

1 lentelė. Investicinio patrauklumo vertinimo veiksniai

Veiksnių grupė	Veiksniai	Šaltiniai
Rinkos patrauklumas	Rinkos dydis	Dzwigol ir kt. (2019); Kharlamova (2014); Rodionov ir kt. (2021); Ruplienė ir kt. (2008); Tocar (2019); Bhardwaj ir kt. (2007); Dorożyński (2016); Stankevičienė ir Lakštutienė (2012), Hiestand (2005).
	Perkamoji galia	
	Ekonomikos atvirumo indeksas	
	Eksportas	
	Rinkos augimo potencialas	
Gamtos išteklių prieinamumas	Žemės ūkio ištekliai	Mustafakulov (2017); Dzwigol ir kt. (2019).
	Gamtos išteklių išnaudojimas	

Veiksnių grupė	Veiksniai	Šaltiniai
Infrastruktūros prieinamumas	Energijos ištekliai	Dzwigol ir kt. (2019); Kharlamova (2014); Dorożyński (2016); Mustafakulov (2017); Zykienė (2018); Ruplienė ir kt. (2008); Tocar (2019)), Hiestand (2005).
	Transportas	
	Komunikacija	
	Ekologija	
Darbo rinkos patrauklumas	Darbo vietos kaina	Dzwigol ir kt. (2019); Kharlamova (2014); Dorożyński (2016); Mustafakulov (2017); Zykienė (2018); Ruplienė ir kt. (2008); Tocar (2019); Bhardwaj ir kt. (2007); Stankevičienė ir Lakštutienė (2012)), Hiestand (2005).
	Išsilavinimas	
	Darbo jėgos pasiekiamumas	
Inovacinis patrauklumas	Inovacijoms palanki aplinka	Zykienė (2018); Rodionov ir kt. (2021); Takhumova ir kt. (2018); Stankevičienė ir Lakštutienė (2012).
Verslo ir mokestinė aplinka	Mokesčių sistema	Takhumova ir kt. (2018); Ruplienė ir kt. (2008); Tocar (2019); Bhardwaj ir kt. (2007); Kharlamova (2014); Dorożyński (2016); Stankevičienė ir Lakštutienė (2012).
	Institucijos	
	Politis stabilumas	
Socialinė ir kultūrinė aplinka	Socialinė infrastruktūra	Dzwigol ir kt. (2019); Rodionov ir kt. (2021); Tocar (2019)), Hiestand (2005).
	Kriminalinė rizika	
	Kultūra	
	Sveikata	
Tvarumo patrauklumas	Tvarumo principų įgyvendinimas	Dzwigol ir kt. (2019); Zykienė (2018); Rodionov ir kt. (2021).

1.5. Literatūroje naudojamų investicijų vertinimo metodų apžvalga

Literatūroje dažniausiai naudojami analizės metodai yra statistiniai. Jie apibūdina tiriamųjų ekonominių, socialinių ir kitų sričių ar procesų kiekybines charakteristikas. Statistiniai metodai skirstomi į duomenų sisteminimo ir kiekybinio aprašymo (aprašomosios statistikos) bei statistinių išvadų gavimo (indukcinės statistikos) metodus. Aprašomosios statistikos pagrindiniai metodai – duomenų klasifikavimas ir grupavimas, jų apibendrintų statistinių rodiklių skaičiavimas ir

interpretavimas, dinamikos eilučių tyrimai, koreliacinė analizė, regresinė analizė, dispersinė analizė, daugiamatės statistinės analizės metodai, naudojami tiriant daugelio nepriklausomų ir priklausomų kintamųjų sąveiką (pavyzdžiui, klasterinė analizė). Indukcinės statistikos metodai leidžia nustatyti statistinės visumos dėsningumus pagal jos objektų dalies (imties) tyrimo rezultatus. Imtis sudaroma atrankos metodu, o statistinių sprendimų metodai padeda su nedidele paklaida statistinių rodiklių reikšmes, gautas tiriant imtį, pritaikyti visai statistinei visumai [30].

Kharlamova (2014) vertindama Ukrainos regionų investicinį patrauklumą rėmėsi statistinio modeliavimo technika. Pirmieji analizės etapai buvo statistikos pasirinkimas bei indikatorių standartizavimas. Toliau buvo atlika koreliacijos ir klasterių analizė. Autorė atliko regionų klasterizavimą pagal kiekvieną rodiklį, naudodant k-vidurkių metodą. Regionai paskirstyti įvertinant kiekvieno klasterio vidutines reikšmes. Paskutinis etapas – integralus vertinimas. Remiantis klasterių, kuriems buvo priskirti regionai, reikšmėmis apskaičiuojamas suminis vertinimas [31].

Šečkutės ir Tvaronavičiaus (2007) tiesioginių užsienio investicijų Baltijos šalyse tyrime buvo naudota koreliacinė analizė. Veiksnių reikšmingumai nustatyti įvertinant kiekvieno analizėje naudoto kintamojo reikšmingumą Stjudento kriterijaus metodu [32].

Dzwigol ir kiti (2019), regionų investicinio patrauklumo vertinimui naudojo ekonominius ir statistinius metodus: daugiamačiai vidurkiai (multidimensional means), analitinis grupavimas (analytical grouping), indeksavimas (index), klasterinė analizė (cluster analysis), reitingavimas (ranking), paskirstymo serijų analizė (distribution series analysis), negriežtos logikos aparatas (fuzzy logic aparatus). Reitingavimas buvo atliekamas remiantis apklausomis. Tyrimo duomenys vėliau buvo grupuojami ir integralinio rodiklio pagrindu nustatytas bendras regiono įvertinimas. Būtent šis metodas yra plačiai įdiegtas pirmaujančių pasaulio konsultacinių ir analitinių agentūrų praktikoje. Kaip šio metodo privalumą reikėtų paminėti galimybę, kartu nustatant bendrą regiono įvertį, analizuoti respondentų atsakymus pagal investuotojo informacijos poreikius. Tačiau taikant šį metodą patiriamos didelės analizės išlaidos ir kyla problemų formuojant reprezentatyvią imtį pagal regioną [1].

Negriežtos logikos metodas taip pat minimas Rodionov ir Nasrutdinov (2021) tyrime, kuriame analizuojama požiūrio į regiono investicinio patrauklumo vertinimą transformacija Covid-19 pandemijos kontekste. Autoriai tyrimo metodologiją susikirstė į tris etapus. Pirmojo etapo metu, naudodantis negriežtos logikos daugkartiniu modeliavimo metodu, apskaičiuojamas regiono investicinio patrauklumo lygis. Antrojo etapo metu prognozuojamas regiono investicinis patrauklumas ateityje. Tam naudojami regARIMA klasės statistiniai modeliai. regARIMA sukuria regresijos modelį su ARIMA laiko eilučių paklaidomis, kad išlaikytų regresijos koeficientų jautrumo interpretaciją. Modelių tinkamumo patikrinimas, atsižvelgiant į F kriterijus ir p-reikšmes, esant 95% pasikliautinumo lygiui. Taip apskaičiuojami apibrėžtumo koeficientai. Paskutinio etapo metu, naudodant analitinės geometrijos įrankius, apskaičiuojamas regionų investicinio patrauklumo vystymosi vektorius [2].

Valinurava ir kiti (2015), vertindami investicinį regiono patrauklumą skaičiavo indikatorių, kuris rėmėsi ekonominiais ir rizikos veiksniais. Ekonominiu veiksmiu buvo laikomas investicinių lėšų gražos lygis. Tuo tarpu rizikos veiksmiu apibendrina bendrinę riziką, susijusią su ekonomine ir socialine sistema. Rizikos komponentės skaičiavimui buvo naudojama formulė [3]:

$$k_2 = \frac{\sum_{i=1}^n P_i * j_i}{\sum_{i=1}^n j_i}, \quad (1.1)$$

kur k_2 - regiono investicinio patrauklumo rizikos komponentė;

n - veiksmų skaičius;

P_i - veiksmo charakteristika;

j_i – veiksmo svoris.

Rizikos komponentė padeda apibrėžti, kokia pajamų dalis gali būti prarasta, tad sudėtinį regiono investicinio patrauklumo rodiklį autoriai pateikė šia formule:

$$k = k_1 * (1 - k_2), \quad (1.2)$$

kur k - investicinio patrauklumo indeksas;

k_1 – ekonominė komponentė;

k_2 – rizikos komponentė [3].

Ši formulė taip pat buvo naudota Mustafakulov (2017) regionų investicinio patrauklumo tyrime.

Lukauskas ir Bruneckienė (2018) analizavo mašininio mokymosi panaudojimo galimybes regionų investicinio patrauklumo vertinime. Autoriai teigė, kad norint išanalizuoti didelius duomenų kiekius, paskutiniiais metais vis sparčiau yra naudojami mašininio mokymosi algoritmai, leidžiantys visus skaičiavimus atlikti kur kas greičiau. Vienas iš mašininio mokymosi būdų, kuris autorių naudojamas investicinio patrauklumo vertinimui yra dirbtiniai neuroniniai tinklai [13].

Dagilienė ir kiti (2019) naudodamiesi neuroniniais tinklais analizavo Europos šalių investicinį patrauklumą. Atliekant empirinius tyrimus buvo klasterizuojamos šalys bei identifikuojami pagrindiniai veiksniai, lemiantys šalių patrauklumą. Taip pat atlikta į šalį pritrauktų užsienio investicijų prognozė. Šalys buvo klasterizuojamos hierarchiškai saviorganizuojančiais neuroniniais tinklais (self-organizing mapping), naudojant Euklido atstumą, remiantis visų rodiklių logaritminėmis ir pagrindinių komponentių reikšmėmis. Laiko eilutės prognozavimui buvo naudojami šie metodai: pasikartojantys neuroniniai tinklai (recurrent neural networks, RNN), ilgalaikės trumpalaikės atminties (long short-term memory, LSTM) neuroniniai tinklai, uždaras pasikartojantis vienetas (gated recurrent unit, GRU) ir ekstremalaus mokymo mašinos (extreme learning machines, ELM) (Dagilienė ir kiti, 2019) [33].

Hiestand (2015) prognozavo tiesiogines užsienio investicijas Taivane. Tam jis naudojo sutelktinio modelio (angl. *Pooled Model*), atsitiktinio modelio (angl. *Random Model*) ir fiksuoto modelio (angl. *Fixed Model*) daugialypes regresijas [19].

2. Šalių klasterizavimo ir prognozavimo metodų apžvalga

Duomenų tyryba - tai duomenų analizė, kuria siekiama atrasti nežinomus dėsningumus ir aprašyti (apibendrinti) duomenis, suteikiant naujų žinių. Remiantis šiuo apibrėžimu, galima teigti, kad duomenų tyrybos tikslas – iš duomenų rinkinio (dažnai didelės apimties) gauti naujų žinių ir gilesnį suvokimą, kurie toliau gali būti panaudoti sprendimams priimti [34].

Duomenų tyrybos metodai yra skirstomi į 3 pagrindines grupes:

- mokymas su mokytoju (angl. *supervised learning technique*);
- mokymas be mokytojo (angl. *unsupervised learning technique*);
- kita [34].

Į pirmą kategoriją „mokymas su mokytoju“ patenka klasifikacijos ir prognozavimo uždaviniai. Antrai kategorijai „mokymas be mokytojo“ priskiriamas klasterizavimo ir asociacinių taisyklių paieškos uždavinys. Vizualizavimas, nuokrypių identifikavimas ir ryšių analizė neskirstomi į „mokymo su mokytoju“ arba „mokymo be mokytojo“ klases [34].

Šiame tyrime naudojami duomenų tyrybos metodai yra klasterizavimas, prognozavimas ir vizualizavimas.

2.1. Šalių klasterizavimo metodai

Klasterizavimas (angl. *clustering*) yra duomenų tyrybos metodas, kuris suskirsto objektus į grupes (klasterius), kuriose objektai yra kuo panašesni, o skirtumai nuo kitų klasterių yra kuo didesni. Šis metodas yra dažnai maišomas su klasifikacija, kurioje objektai priskiriami iš anksto nustatytoms klasėms. Klasterizavimo atveju, klasės nėra iš anksto žinomos ir yra identifikuojamos tyrimo metu [35].

Duomenų klasterizavimas yra naudojamas šiems tikslams:

- įgyti įžvalgų apie duomenis, generuoti hipotezes, aptikti anomalijas ir nustatyti svarbiausius bruožus;
- nustatyti panašumus bei jų lygį tarp įvairių objektų;
- naudojant klasterių prototipus, organizuoti bei apibendrinti duomenis [36].

Svarbiausi klasterizavimo etapai yra šie:

1. Pasirenkami klasterizavimo objektai;
2. Pasirenkami požymiai, t.y. nusprendžiama pagal kokius požymius bus klasterizuojama;
3. Pasirenkamas objektų artumo matas;
4. Vienu ar kitu metodu objektai suskirstomi į klasterius;
5. Peržiūrimi gauti rezultatai [37].

Artumo (panašumo) matas gali būti apibūdinamas kaip atstumas tarp įvairių duomenų taškų. Daugelio algoritimų veikimas priklauso nuo gero artumo mato pasirinkimo. Dažniausiai literatūroje naudojami artumo matai [38]:

1. Euklido atstumas (angl. *Euclidean distance*) yra įprastas atstumas tarp taškų ir yra laikomas standartine metrika geometrinėms problemoms. Taip pat yra naudojamas kaip numatytasis matas k-vidurkių metode. Euklido atstumas apibrėžiamas kaip kvadratinė šaknis iš objektų koordinatų skirtumo kvadrato. Skaičiuojamas formule:

$$d_E(i, j) = \sqrt{\sum_{s=1}^p (x_{is} - x_{js})^2} \quad (2.1)$$

2. Manhateno atstumas (angl. *Manhattan distance*) apskaičiuoja absoliutų skirtumą tarp objektų koordinatų. Skaičiuojamas formule:

$$d_{MH}(i, j) = \sum_{s=1}^p |x_{is} - x_{js}| \quad (2.2)$$

3. Čebyševio atstumas (angl. *Chebyshev distance*), taip pat vadinamas maksimalios vertės atstumu, apskaičiuoja didžiausią absoliutų skirtumą tarp objektų koordinatų. Skaičiuojamas formule:

$$d_{CH}(i, j) = \max_{1 \leq s \leq p} |x_{is} - x_{js}| \quad (2.3)$$

4. Minkovskio atstumas (angl. *Minkowski distance*), taip pat vadinamas apibendrintu atstumu. Skaičiuojamas formule:

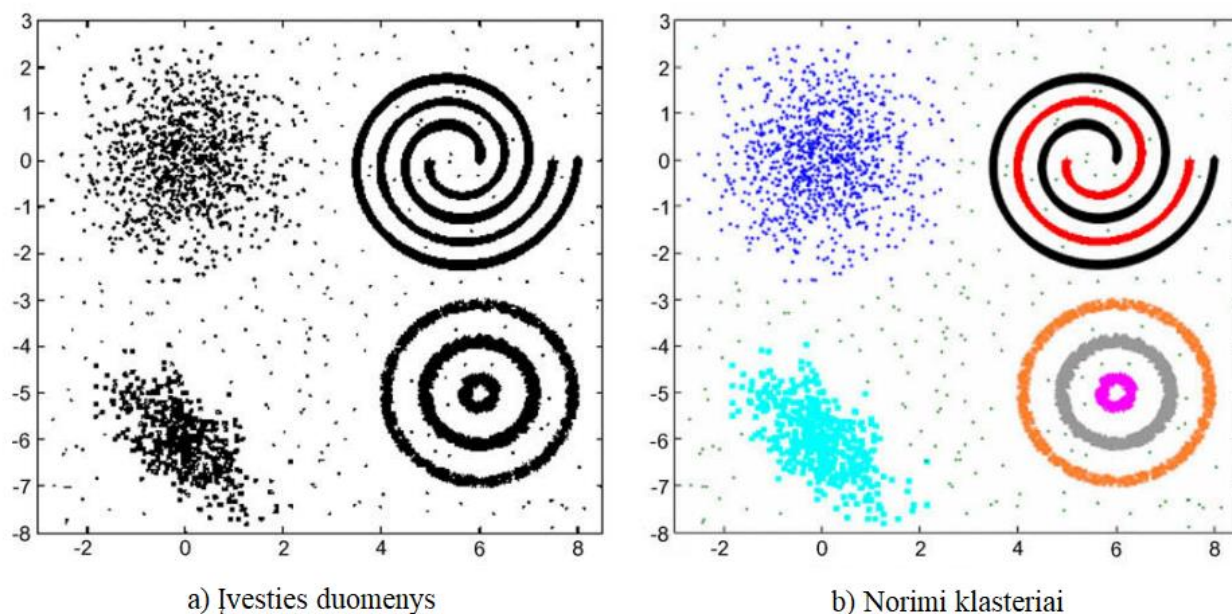
$$d_{MK}(i, j) = \sqrt[h]{\sum_{s=1}^p (x_{is} - x_{js})^h} \quad (2.4)$$

Atstumui vertinti taip pat naudojamas kosinuso panašumas (angl. *cosine similarity*). Šis dydis fiksuoja objektų orientaciją (kampą). Matematiškai, jis matuoja kampo tarp dviejų vektorių, projektuojamų daugiamatėje erdvėje, kosinusą. Kosinuso atstumas skaičiuojamas:

$$1 - \cos \theta = \frac{a^T b}{\|a\| \|b\|} \quad (2.5)$$

Šiame tyrime naudojami atstumo matai: Euklido atstumas, Manhateno atstumas ir Kosinuso atstumas.

Klasteriai gali skirtis pagal savo formą, dydį ir tankį (7 pav.). Literatūroje išskiriami klasterių tipai: gerai atskirti (angl. *well-separated*), sferiniai (angl. *center-based*), artimi arba susiliečiantys (angl. *contiguous*), tankiu grįsti (angl. *density-based*) ir konceptualūs (angl. *conceptual*). Klasterizavimą gali apsunkinti triukšmo buvimas duomenyse. Žmogus, turintis analizuojamos srities žinių, gali išvelgti dviejų – trijų demensijų klasterius, tačiau didelės apimties duomenims suklasterizuoti yra reikalingi automatiniai algoritmai. Dėl šios priežasties bei dėl nežinomo pateiktų duomenų grupių skaičiaus, buvo pasiūlyta daug klasterizavimo algoritmų, iš kurių kelis apžvelgsime tolimesnėje metodų analizėje [36].



6 pav. Klasterių įvairovė [36]

Visi klasterizavimo algoritmai apytiksliai buvo suskirstyti į keturias klases:

- padalijimo (angl. *partitioning*) metodai;
- hierarchiniai (angl. *hierarchical*) metodai;
- tankiu grįsti (angl. *density-based*) metodai;
- tinkleliu grįsti (angl. *grid-based*) metodai [39].

Swarndeeep ir kiti autoriai (2016) be šių metodų, taip pat išskiria modelių grįstą klasterizavimą [40]. Tuo tarpu Xu ir Tian (2015) dar išplečia grupavimą ir tradicinius klasterizavimo algoritmus skirsto į 9 kategorijas (2 lentelė).

2 lentelė. Klasterizavimo kategorijos [41]

Kategorija	Algoritmai
Klasterizavimas, grįstas padalijimu (angl. <i>partition</i>)	K-vidurkių, k-medoidų, PAM, CLARA, CLARANS
Klasterizavimas, grįstas hierarchija (angl. <i>hierarchy</i>)	BIRCH, CURE, ROCK, Chameleon
Klasterizavimas, grįstas negriežta teorija (angl. <i>fuzzy theory</i>)	FCM, FCS, MM
Klasterizavimas, grįstas pasiskirstymu (angl. <i>distribution</i>)	DBCLASD, GMM, BGMM
Klasterizavimas, grįstas tankiu (angl. <i>density</i>)	DBSCAN, OPTICS, Mean-shift
Klasterizavimas, grįstas grafine teorija (angl. <i>graph theory</i>)	CLICK, MST
Klasterizavimas, grįstas tinkleliu (angl. <i>grid</i>)	STING, CLIQUE
Klasterizavimas, grįstas fraktalų teorija (angl. <i>fractal theory</i>)	FC
Klasterizavimas, grįstas modelių (angl. <i>model</i>)	COBWEB, GMM, SOM, ART

Šiame tyrime naudoti padalijimo, hierarchiniai, tankiu ir pasiskirstymu grįsti metodai. Jie plačiau aprašomi tolimesniuose skirsniuose.

2.1.1. Padalinimo metodai

Padalinimo metodai yra laikomi populiariausiais klasterizavimo algoritmais. Jie padalina n objektų duomenų rinkinį į k dalis (klasterius). Kiekvienas objektas priklauso tik vienai grupei ir kiekviena grupė turi turėti bent vieną objektą. Metodai yra naudingi tyrimuose, kuriuose reikalingų grupių skaičius yra statinis. Padalinimo algoritmų trūkumas – kai atstumas tarp klasterių yra mažas, rezultatas tampa prastas arba klaidinantis, dėl duomenų taškų sutapimo. [38,41]

Šiame tyrime naudojamas padalinimo metodas yra k -vidurkių. Lengvo įgyvendinimo, paprastumo, efektyvumo ir empirinės sėkmės dėka, šis klasterizavimo būdas yra vienas populiariausių ir dažniausiai naudojamų įvairiuose tyrimuose [36]. K -vidurkių algoritmo žingsniai:

1. Objektai suskirstomi į k pradinių klasterių;
2. Apskaičiuojamas kiekvieno objekto atstumas iki klasterių centrų (Euklido atstumas). Objektas skiriamas į artimiausią klasterį. Tada vėl perskaičiuojami klasterių centrai;
3. Antras žingsnis kartojamas tol, kol perskirstymų daugiau nėra [37].

Pagrindiniai metodo privalumai:

- Jei yra daug kintamųjų, k -vidurkių metodas dažniausiai apskaičiuoja greičiau negu hierarchiniai klasterizavimo metodai.
- K -vidurkių metodu sukurti klasteriai yra glaudesni palyginus su hierarchiniais metodais [40].

Tuo tarpu pagrindiniai trūkumai: sudėtinga numatyti k reikšmę, sunkiau palyginti klasterių kokybę, prastai pritaikomas globaliems klasteriams [40].

2.1.2. Hierarchiniai metodai

Hierarchiniu klasterizacijos metodu siekiama suskirstyti objektus medžio principu – visi objektai laikomi vienu dideliu klasteriu, kurį sudaro mažesni klasteriai, taip pat sudaryti iš dar mažesnių klasterių ir t.t. Šis metodas yra vaizduojamas dendograma. Hierarchinio klasterizavimo tipai [38]:

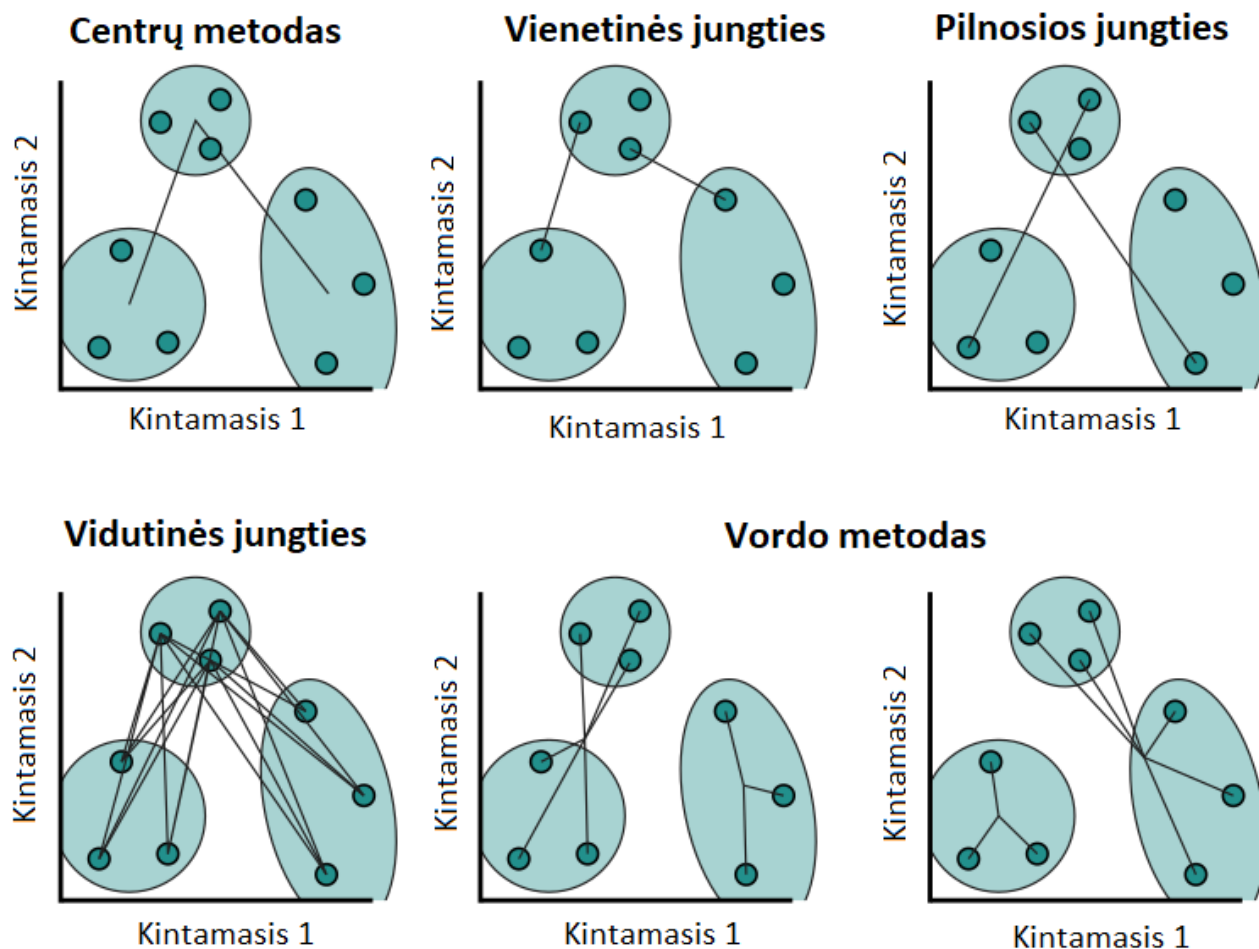
- Jungimo (angl. *agglomerative*) metodas. Klasterizavimas prasideda su vienu duomenų objektu viename klasteryje. Toliau grupuojami artimiausi objektai iki tol kol visi objektai yra sugrupuojami kartu viename klasteryje.
- Skaidymo (angl. *divisive*) metodas. Klasterizavimas vyksta priešingai. Pradžioje visi duomenų objektai yra kartu viename klasteryje ir toliau yra skaidomi iki vienetinių klasterių.

Pagrindinis hierarchinių metodų trūkumas yra ilgas proceso laikas, kuomet yra daug duomenų, dėl to yra sunku atlikti pakeitimus procesui prasidėjus [40].

Dažniausiai naudojami hierarchinių klasterių sudarymo metodai (7 pav.):

1. Vienetinės jungties (angl. *Single link*) arba artimiausio kaimyno (angl. *Nearest neighbor*): apibrėžiamas atstumu tarp skirtinguose klasteriuose esančių dviejų artimiausių objektų;
2. Pilnosios jungties (angl. *Complete link*) arba tolimiausio kaimyno (*Furthest neighbor*): apibrėžiamas atstumu tarp skirtinguose klasteriuose esančių dviejų tolimiausių objektų;
3. Vidutinės jungties (angl. *Between-group linkage* arba *Group average*): apibrėžiamas vidutiniu atstumu tarp visų galimų dviejų klasterių objektų porų;
4. Centrų (angl. *Centroid*): apibrėžiamas atstumu tarp klasterių centrų;

5. Vordo (angl. *Ward's*): apibrėžiama mažiausia klasterių vidinių kvadratinių nuokrypių suma.



7 pav. Hierarchinių klasterių sudarymo metodai [42]

2.1.3. Tankiu grįsti metodai

Tankiu pagrįsti klasterizavimo algoritmai remiasi idėja, kad objektai, sudarantys tankų regioną, turėtų būti sugrupuoti į vieną grupę, t.y. objektai grupuojami ne pagal atstumų matą, bet pagal objektų tankį. Tai padeda išskirti sudėtingesnės formos klasterius. Taikant tokį klasterizacijos metodą, klasteris yra laikomas regionu, kuriame duomenų objektų tankis viršija tam tikrą ribinę vertę [38,43].

Vienas iš tankiu grįstų metodų yra DBSCAN (angl. Density-Based Spatial Clustering of Applications with Noise). Šiame algoritme su tašku susietas tankis gaunamas skaičiuojant taškų kiekį nurodyto spindulio apie tašką srityje. Taškai, kurių tankis viršija nurodytą slenkstį, sugrupuojami į klasterį. Šiuo metodu sudaryti klasteriai gali būti įvairių formų (pvz., linijinė, įgaubta, ovali ir kt.). DBSCAN taip pat pasižymi galimybe apdoroti labai dideles duomenų bazes [43].

Šių metodų trūkumas, jog jie fiksuoja tik tam tikrų tipų triukšmo taškus, kai egzistuoja skirtingų tankių klasteriai. Be to, jie yra tinkami, kai klasteriai yra nutolę vienas nuo kito, tačiau nepakankami, kai klasteriai yra artimi, šalia vienas kito [43].

2.1.4. Pasiskirstymu grįsti metodai

Pasiskirstymu grįsti metodai yra paremti idėja, kad jei yra keli pasiskirstymai duomenų rinkinyje, duomenys sukurti to pačio pasiskirstymo, priklauso tam pačiam klasteriui. Vienas iš šių metodų yra Gauso mišinių (angl. *Gaussian mixture models*, GMM). GMM susideda iš kelių Gauso skirstinių, iš kurių generuojami pirminiai duomenys, o duomenys, paklūstantys tam pačiam nepriklausomam Gauso skirstiniui, laikomi priklausančiais tam pačiam klasteriui [41].

Metodo privalumai: tiksliau pateikiama priklausymo tikimybė, santykinai didelė mastelio keitimo galimybė, keičiant pasiskirstymą, klasterių skaičių ir kt., palaikomas gerai išvystyto statistikos mokslo. Trūkumai: prielaida nėra visiškai teisinga, susiję su daug parametru, turinčių didelę įtaką klasterizacijos rezultatui ir ilgam skaičiavimo laikui [41].

Kitas metodas, naudojamas šiame tyrime yra Bajeso Gauso mišinių modelis (angl. *Bayesian Gaussian mixture models*, BGMM). Šis klasterizavimo būdas paremtas Bajeso metodu, kuris reikalauja aproksimacijos vėlesniam pasiskirstymui įvertinti. Tai gali padėti spręsti persimokymo problemą [44].

2.1.5. Klasterizavimo metodų vertinimas

Suskirsčius objektus į klasterius, svarbu įvertinti klasterizavimo kokybę. Tam gali būti naudojamas Silueto koeficientas (angl. *Silhouette score*), parodantis kaip gerai objektai buvo priskirti klasteriams, remiantis grupių glaudumu ir atskyrimu [45]. Jo reikšmės svyruoja nuo -1 iki 1, kur 1 reiškia, jog klasteriai yra gerai nutolę vienas nuo kitos ir aiškiai išsiskiria, 0 – atstumas tarp klasterių nereikšmingas, -1 – klasteriai yra priskirti neteisingai. Silueto koeficientas apskaičiuojamas šia formule:

$$\text{Silueto koeficientas} = \frac{b-a}{\max(a,b)}, \quad (2.6)$$

kur a – vidutinis atstumas tarp kiekvieno taško klasteryje, b – vidutinis atstumas tarp visų klasterių [46].

2.2. Šalių tiesioginių užsienio investicijų prognozavimo metodai

Bendrai visam duomenų rinkiniui ir atskirai kiekvienam klasteriui sudaromi regresijos ir atsitiktinių miškų modeliai. Vėliau, remiantis geriausiais atrinktais metodais, prognozuojami tiesioginių užsienio investicijų šrantai vieneriems metams į priekį.

Regresija – statistinis modelis, leidžiantis vieno kintamojo reikšmes prognozuoti pagal kito kintamojo reikšmes. Taip pat galima teigti, kad tai yra statistinė vieno atsitiktinio dydžio reikšmių priklausomybė nuo kito – neatsitiktinio – dydžio (arba kelių kitų dydžių), turinti griežtą funkcinį ryšį (t.y., aprašoma lygtimi) [47].

Regresinės analizės eigą galima suskirstyti į 3 pagrindinius etapus:

- Pirmiausia sudaromas modelis (parenkami kintamieji, užrašoma regresijos lygtis, apskaičiuojami koeficientai);
- Tada vykdoma modelio diagnostika (patikrinamas modelio tinkamumas duomenims, duomenų tinkamumas modeliui, modelio prielaidos);
- Galiausiai tinkamai sudarytas modelis aprašomas ir naudojamas prognozavimui [48].

Modelių reikšmingumui įvertinti yra naudojamas F-kriterijus (Fišerio). Gerą tinkamumą duomenims parodo tai, jog kriterijus yra statistiškai reikšmingas (daug skiriasi nuo nulio) [47].

$$F = \frac{\bar{R}^2 \times (n-k)}{(1-\bar{R}^2) \times (k-1)} \quad (2.7)$$

Apibrėžtumo koeficientas (R^2) yra svarbiausia regresijos modelio tinkamumo duomenims metrika. Jis yra modeliujamų ir stebimų priklausomo kintamojo reikšmių skirtumų matas. Apibrėžtumo koeficiento reikšmės gali siekti nuo 0 iki 1. Kuo koeficiento reikšmė didesnė, tuo modelis geriau tinka duomenims. Modelis yra netinkamas, jei $R^2 < 0,20$. Jei modelio R^2 yra 0,8, tai reiškia, kad nepriklausomi kintamieji gali paaiškinti 80% priklausomojo kintamojo kitimo. Svarbu paminėti, kad didelė R^2 reikšmė neužtikrina, kad visi regresoriai jame būtini, o modelis yra prasmingas [47]. Apibrėžtumo koeficientas apskaičiuojamas pagal šią formulę:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y}_i)^2} \quad (2.8)$$

ANOVA $p < 0,05$ parodo, kad yra reikalingas bent vienas regresorius, t.y., bent vieno koeficiento vertė statistiškai reikšmingai skiriasi nuo nulio [48].

Visiems regresoriams t kriterijus $p < 0,05$ parodo, kurių konkrečiai koeficientų vertė statistiškai reikšmingai skiriasi nuo 0 [48].

Vertinant regresijos modelį tikrinama, ar liekamosios paklaidos normalios. Tai galima padaryti sprendžiant pagal q-q grafiką bei Shapiro-Wilk kriterijaus $p \geq 0,05$. Taip pat, ar liekamųjų paklaidų vidurkis lygus 0. Tai galima patikrinti iš liekamųjų paklaidų pagal prognozuojamas reikšmes grafiko bei vienos imties t testo $p \geq 0,05$ [48].

Taip pat yra tikrinama, ar nėra heteroskedatiškumo problemos. Tai galima matyti liekamųjų paklaidų grafike bei Breusch-Pagan testo $p \geq 0,05$ [48].

Daugialypės regresijos atveju yra tikrinama, ar nėra multikolinearumo problemos, dėl kurios neteisingai apskaičiuojami lygties koeficientai. Problemos nėra, jei regresijos koeficientų ženklai atitinka koreliacijos koeficiento ženklus tarp Y ir kiekvieno X arba visi $VIF \leq 4$ (dispersijos mažėjimo daugiklis, angl. *variance inflation factor*) [48].

Laiko eilučių atveju, tikrinama, ar nėra autokoreliacijos. Durbin-Watson autokoreliacijos kriterijaus $p \geq 0,05$, kai alternatyva $\rho \neq 0$ (čia ρ – (auto)koreliacijos koeficientas) arba Durbin-Watson statistikos reikšmė yra tarp 1,5-2,5 [48].

Pirmiausia tyrime buvo naudojama sutelktinė MKM regresinė analizė (angl. *PooledOLS*). Šis metodas buvo pasirinktas, nes jis yra taikomas, esant paneliniams duomenims. Tyrime naudojami

duomenys atitinka šį duomenų tipą, nes tie patys duomenys yra stebimi keliais laiko momentais. Paneliniai duomenys suteikia daugiau informatyvių duomenų, kintamumo ir efektyvumo bei mažiau kolinearumo tarp kintamųjų [19].

Sutelktinės MKM regresijos modeliui yra būdinga, jog visi stebimi objektai yra homogeniški, pastovūs koeficientai - pastovioji regresijos dalis (angl. *intercept*) ir krypties koeficientas (angl. *slope*). Daroma prielaida, kad santykis tarp priklausomojo kintamojo ir išorinių kintamųjų yra toks pats visais periodais. Šiam modeliui tyrėjai gali sujungti visus duomenis ir paleisti įprastą mažiausių kvadratų regresijos modelį [19]. Sutelktinės regresijos modelis:

$$Y_{it} = \alpha + X_{it}\beta + u_{it}, \quad i = 1, \dots, N, \quad t = 1, \dots, T \quad (2.9)$$

Sutelktinės MKM regresijos modeliai sudaryti su visais kintamaisiais, naudojant atgalinę atranką (angl. *backward stepwise*). Iš pradžių į modelį yra įtraukiami visi kintamieji, bet vėliau laipsniškai šalinami mažiausiai įtakos priklausomajam kintamajam turintys kintamieji, kol lieka tik reikšmingi.

Tiesioginių užsienio investicijų prognozavimui geriausias modelis buvo atrinktas palyginus šių modelių rezultatus:

1. LARS regresija;
2. Daugiasluoknis perceptrono regresorius;
3. Bajeso ARD regresija;
4. Elastiniai tinklai;
5. Bagging regresorius;
6. Atsitiktinių miškų modelis;
7. LASSO regresija;
8. Huber regresorius;
9. Gradiento didinimo regresija;
10. Kernel Ridge regresija;
11. Gauso proceso regresija;
12. AdaBoost regresorius;
13. Tiesinė regresija.

LARS (angl. *Least Angle Regression*) metodo procedūra gali būti lyginama su pažingsninio kintamųjų įtraukimo procedūra (angl. *forward stepwise regression*) tiesinėje regresijoje. Be to, LARS glaudžiai susijęs su LASSO regresija [49].

Daugiasluoknis perceptrono regresorius (angl. *Multi-layer Perceptron regressor*, MLP) optimizuoja kvadratinę paklaidą, naudodamas LBFGS arba stochastinį gradiento nusileidimą [50].

Bajeso ARD regresija (angl. *Bayesian Automatic Relevance Determination regression*, ARD) pritaiko regresijos modelio svorius, naudojant pirminį ARD. Manoma, kad regresijos modelio svoriai yra pasiskirstę pagal Gauso skirstinį. Taip pat iteracinėmis procedūromis įvertina svorių ir triukšmo pasiskirstymo tikslumus [50].

Bagging regresorius (angl. *Bagging regressor*) - grupinis metaįvertis, kuris kiekvieną bazinį regresorių pritaiko atsitiktiniams pradinio duomenų rinkinio poaibiams, o tada sujungia jų individualias prognozes (balsuojant arba apskaičiuojant vidurkį), kad sudarytų galutinę prognozę [50].

Atsitiktinių miškų (angl. *random forest*) modelis pritaiko daugybę klasifikuojamųjų sprendimų medžių įvairiuose duomenų rinkinio pavyzdžiuose ir naudoja vidurkį, kad pagerintų numatymo tikslumą ir kontroliuotų perteklinį suderinimą [50].

Elastiniai tinklai (angl. *Elastic Net*) yra LASSO ir RIDGE regresijų kombinacija. Šis metodas siūlomas naudoti, kai kyla problemų dėl multikolinearumo ir atvejų, kai kintamųjų yra daugiau nei stebinių [49].

LASSO (angl. *Least Absolute Shrinkage and Selection Operator*) yra efektyvus metodas, kuris atlieka du pagrindinius uždavinius: reguliarizaciją ir kintamųjų atrinkimą. LASSO regresijos modelio koeficientai sumažinami, nustatant tam tikrą baudą dėl modelio sudėtingumo. Gauti LASSO koeficientai minimizuoja paklaidų kvadratų sumą, atsižvelgiant į nustatytą baudą [49].

Huber regresorius (angl. *Huber regressor*, HB) – tiesinės regresijos modelis, atsparus nuokrypiams. Naudojama kitokia nuostolių funkcija, o ne tradiciniai mažiausi kvadratai [50].

Gradiento didinimas regresijai (angl. *Gradient Boosting for regression*, GB) optimizuoja savavališkas diferencijuojamų nuostolių funkcijas [50].

Kernel Ridge regresija sujungia Ridge regresiją (tiesinius mažiausius kvadratus su L2 normos reguliarumu) su branduolio „gudrybe“. Netiesiniams branduoliams tai atitinka netiesinę funkciją pradinėje erdvėje [50].

Gauso proceso regresija (angl. *Gaussian Process regression*) apskaičiuoja tikimybių pasiskirstymą pagal visas leistinas funkcijas, kurios atitinka duomenis.

AdaBoost regresorius yra metaįvertis, kuris pradėdamas pritaikant regresorių į pradinį duomenų rinkinį, o vėliau pritaiko papildomas regresoriaus kopijas tame pačiame duomenų rinkinyje, bet kur atvejų svoriai koreguojami pagal dabartinės prognozės paklaidą. Todėl vėlesni regresoriai daugiau dėmesio skiria sudėtingiems atvejams [50].

Modelių sudarymui buvo naudojami keturi duomenų mastelio tipai:

- Raw: naudojami originalūs duomenys;
- `MinMaxScaler()`: naudojami normalizuoti duomenys. Kiekvienas požymis yra individualiai padidinamas ar paverčiamas taip, kad jis būtų nurodytame mokymo rinkinio diapazone, pvz. tarp nulio ir vienetų;
- Standard: naudojami standartizuoti duomenys. Pašalinamas vidurkis ir pakoreguojamas mastelis iki vienetų dispersijos;
- `RobustScaler()`: pašalinama mediana ir duomenys pakeičiami pagal kvantilio intervalą [50].

Geriausių metodų įvertinimui naudoti rodikliai:

- Vidutinė kvadratinė paklaida (MSE);
- Vidutinė absoliutinė paklaida (MAE);
- Vidutinės kvadratinės paklaidos šaknis (RMSE).

Vidutinė kvadratinė paklaida (angl. *mean square error*, MSE) apskaičiuojama pagal formulę:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.10)$$

Vidutinė absoliutinė paklaida (angl. *mean absolute error*, MAE) apskaičiuojama pagal formulę:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.11)$$

Vidutinės kvadratinės paklaidos šaknis (angl. *rooted mean square error*, RMSE) apskaičiuojama pagal formulę:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}} \quad (2.12)$$

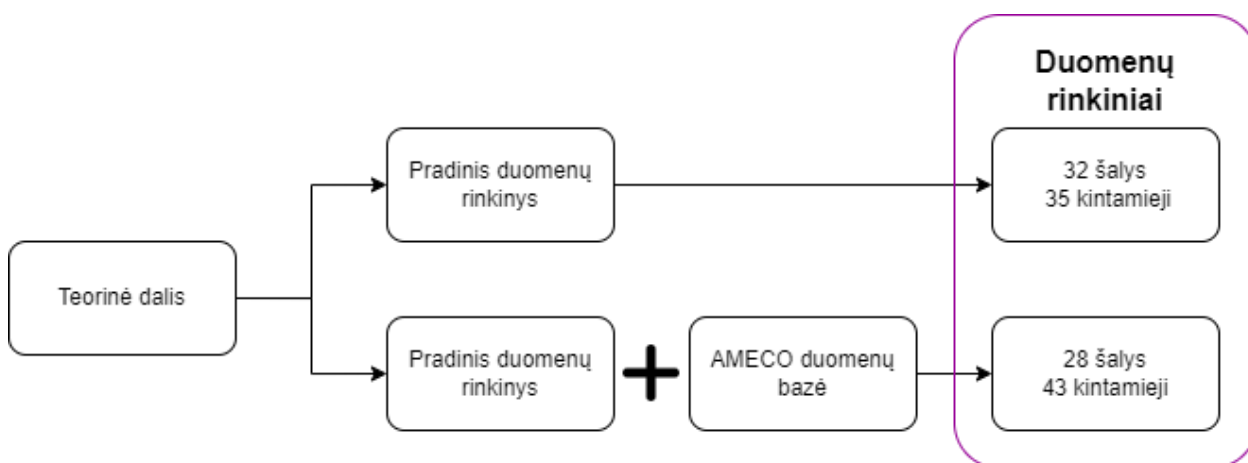
3. Empirinis tyrimas

3.1. Duomenys

Tyrime buvo naudojami skirtingi ekonominiai šalių duomenys, kurie yra susiję su tiesioginėmis užsienio investicijomis (TUI). Duomenų surinkimui buvo naudojami šie šaltiniai:

1. TheGlobalEconomy – tai duomenų bazė, kurioje pateikiami įvairių pasaulio šalių ekonominiai rodikliai. Ši duomenų bazė buvo pasirinkta dėl to, kad joje yra daug skirtingų rodiklių. TheGlobalEconomy yra rekomenduojamas Amerikos Ekonomikos asociacijos.
2. WorldBank – tai pasaulio banko departamento duomenų bazė, kurioje kaip ir anksčiau minėtoje duomenų bazėje pateikiami įvairūs pasaulio šalių ekonominiai rodikliai. Ši duomenų bazė remiasi įvairiais duomenų šaltiniais, tokiais kaip TVF (tarptautinis valiutos fondas) ir kt., todėl galima teigti, kad duomenys yra tikslūs ir juos galima naudoti tyrime. Taip pat svarbus faktorius, kad šie duomenys yra laisvai prieinami.

Kadangi tiriamasis laikotarpis yra nuo 2006 iki 2020 metų, todėl tampa sudėtinga rasti patikimus duomenų šaltinius, kuriuose būtų visi reikiami duomenys. Pirmajame tyrimo etape buvo naudojamas mažesnis kintamųjų rinkinys, taip išlaikant šias šalis: Rusiją, Baltarusiją, Ukrainą, Šiaurės Makedoniją. Tuomet antrajame tyrimo etape buvo panaudota mažesnė šalių imtis, tačiau papildomai buvo pridėti AMECO duomenų bazės duomenys, kurie yra labai artimai susiję su tiesioginėmis užsienio investicijomis. AMECO yra metinė Europos Komisijos Ekonomikos ir finansų reikalų generalinio direktorato makroekonominė duomenų bazė. Duomenų bazė naudojama analizei ir generalinio direktorato parengtoms ataskaitoms. Jame pateikiami ES, euro zonos, ES šalių ir šalių kandidačių bei kitų EBPO šalių duomenys. Tuomet šalys, kurioms trūksta duomenų buvo pašalintos, taip siekiant patikrinti, ar šie rodikliai turi reikšmingos įtakos klasterių struktūrai. (8 pav.)



8 pav. Duomenų rinkinių sudarymo schema

3.2. Duomenų žvalgomoji analizė

Tyrimo pradžioje yra svarbu atlikti žvalgomąją duomenų analizę. Pirmiausiai apžvelgiamas pagrindinis tyrimo rodiklis – TUI (tiesioginės užsienio investicijos), tuomet aptariami keli išoriniai kintamieji, kurie yra aktualiausi arba kuriuose pastebėtos įdomiausios tendencijos.

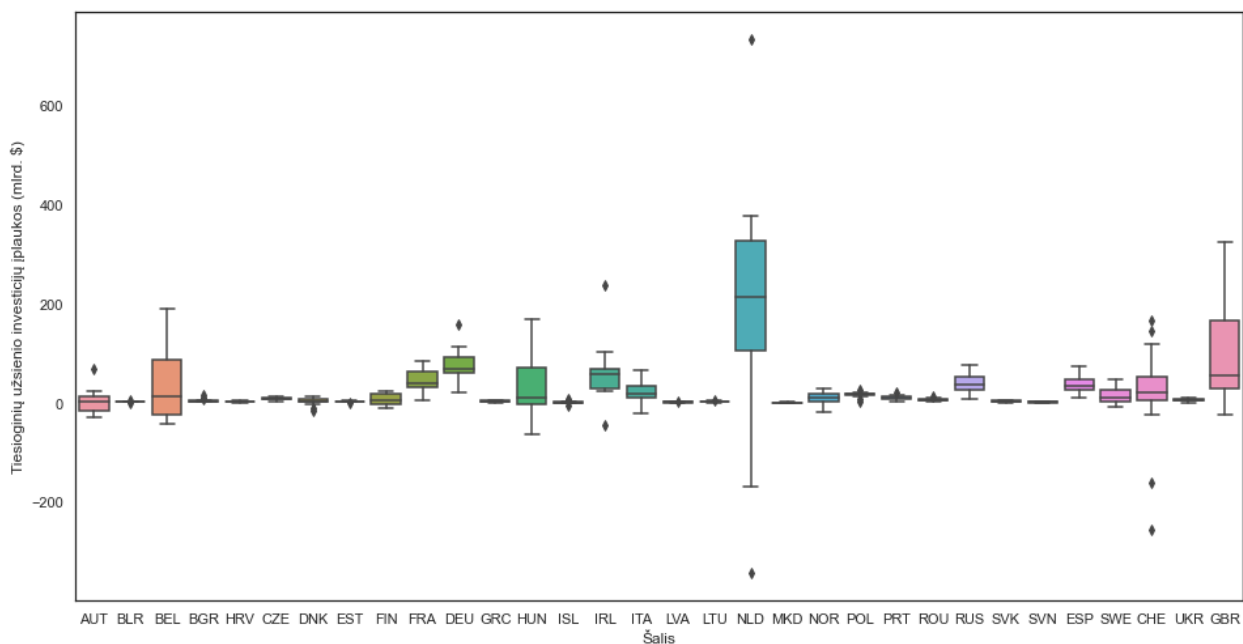
3.2.1. Tiesioginių užsienio investicijų apžvalga tiriamuoju laikotarpiu

Pirmajame šio poskyrio skirsnyje aptariamos tiesioginių užsienio investicijų tendencijos. 9 paveiksle pateikta stačiakampė diagrama padeda įvertinti vidutines tiesioginių užsienio investicijų įplaukų reikšmes, jų pasiskirstymą. Taip pat išvelgti išskirtinai didelės teigiamas arba neigiamas reikšmes. Remiantis pateiktu paveikslu galima pastebėti, kad didžiausias vidutinių tiesioginių užsienio investicijų įplaukų kiekis 2006–2020 metų laikotarpiu buvo Nyderlanduose. Taip pat, galime matyti, kad daugelyje šalių tiesioginių užsienio investicijų įplaukos yra artimos 0 (su tam tikru svyravimu). Didesnis tiesioginio užsienio investicijų įplaukų svyravimas pastebimas tik Belgijoje, Vengrijoje, Jungtinėje Karalystėje, pats didžiausias - Nyderlanduose. Įdomu tai, kad vertinant Nyderlandų stačiakampę diagramą, matomos išskirtinai didelės, tačiau ir išskirtinai mažos tiesioginių užsienio investicijų įplaukų reikšmės. Detaliau nagrinėjant rezultatus pastebima tai, kad 2007 metais Nyderlandai pasiekė absoliutų tiesioginių užsienio investicijų rekordą, kuomet augimas lyginant su 2006 metais siekė net 2285%.¹ Ši šalis pasižymi aukštos kvalifikacijos darbuotojais, taip pat palankiais verslo įstatymais bei geografinė padėtimi, dideliu uostų kiekiu. Toliau vertinant neigiamas reikšmes pastebima tai, kad Nyderlandų neigiama reikšmė nustatyta tuo laikotarpiu, kai visoje Europoje vyravo neigiamos įplaukos. Nagrinėjant Airijos vieną išskirtinai didelę reikšmę pastebima tai, kad ji nustatyta 2015 metais. Remiantis papildoma informacija matoma, kad 2015 metais Airijai tai buvo absoliutus tiesioginių užsienio investicijų įplaukų rekordas. Pagrindinė to priežastis, buvo tai, kad Jungtinių Amerikos Valstijų įmonės nusprendė daugiau investuoti Airijoje dėl kur kas palankesnių mokesčių sąlygų lyginant su kitomis šalimis.² Taip pat kelios išskirtinai mažos tiesioginių užsienio investicijų reikšmės pastebimos Šveicarijoje. Remiantis papildoma informacija, nors šalis sukūrė tvirtą pramoninę bazę, pastarojo meto neigiamos vertės daugiausia atspindi didelės šalies metinių srautų reikšmes.³ Vertinant Lietuvos tiesioginių užsienio investicijų įplaukas pastebima tai, kad jos visais metais yra gana stabilios ir neturi išskirtinai didelių ar išskirtinai neigiamų reikšmių. Lyginant su kitų šalių vidutinėmis reikšmėmis galima teigti, kad Lietuvos įplaukų srautai yra panašūs į kitų šalių.

¹<https://unctad.org/press-material/foreign-direct-investment-reached-new-record-2007>

²<https://www.irishtimes.com/business/economy/ireland-s-gdp-figures-why-26-economic-growth-is-a-problem-1.2722170>

³<https://unctad.org/news/foreign-investment-flows-developed-countries-slump-58-2020>

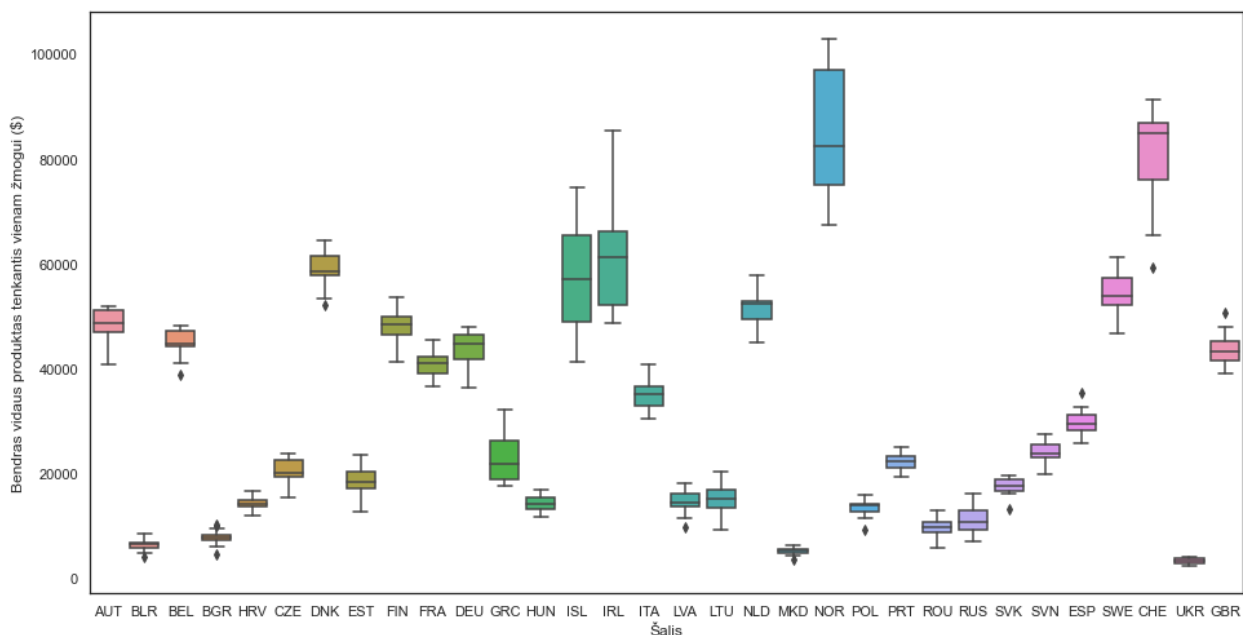


9 pav. Tiesioginių užsienio investicijų įplaukų stačiakampė diagrama

3.2.2. Išorinių tyrime naudojamų rodiklių žvalgomoji analizė

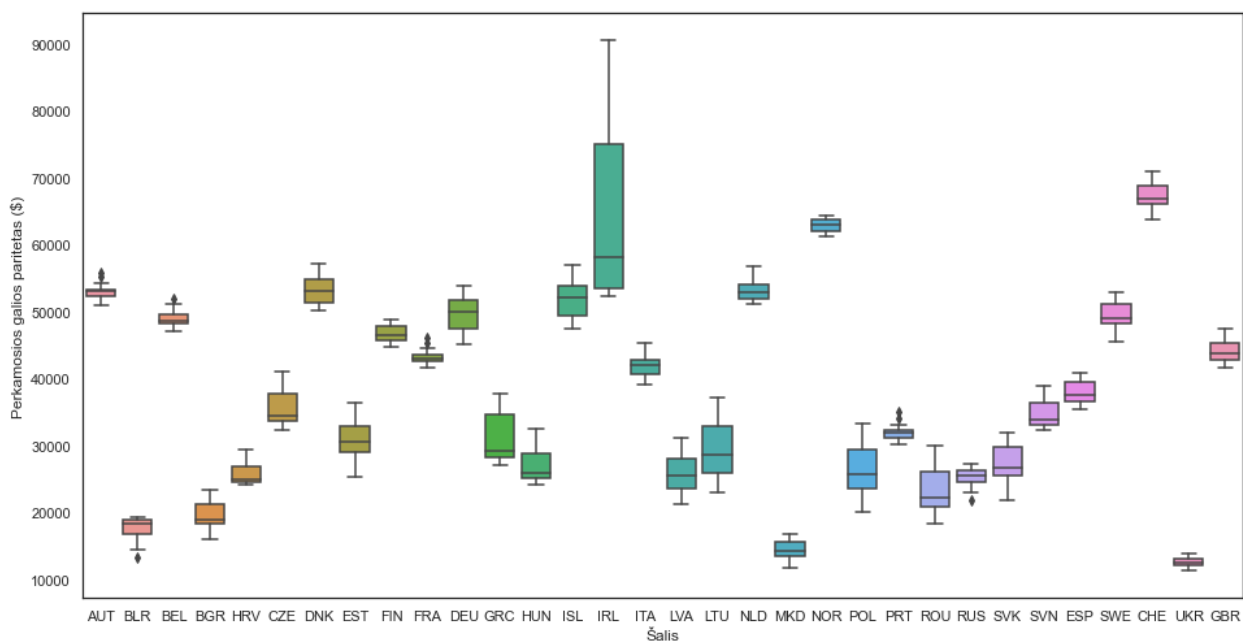
Ypatingai svarbu apžvelgti ne tik pagrindinį tyrime naudojamą tiesioginių užsienio investicijų kintamąją, tačiau ir išorinius kintamuosius, kurie toliau naudojami tyrime. Atsižvelgiant į tai, šiame skirsnyje pateikiama dalies išorinių kintamųjų, kurie yra naudojami tyrime, apžvalga. Kadangi rodiklių yra daug, šiame skirsnyje aptariami tik tie išoriniai kintamieji, kurie yra svarbiausi arba pasižymi tam tikromis išskirtinėmis reikšmėmis, tendencijomis.

Kaip galima pastebėti literatūroje bei ekonomikoje, siekiant apibūdinti šalis, vienas iš dažniausiai naudojamų kintamųjų yra bendras vidaus produktas (BVP), o taip pat ir dar geriau šalį apibūdinantis bendras vidaus produktas vienam gyventojui. Šis rodiklis yra naudojamas siekiant įvertinti šalies ekonomikos būklę. 10 paveiksle galima pamatyti, kad BVP, tenkantis vienam gyventojui, yra didžiausias Šveicarijoje bei Norvegijoje. Tuo tarpu mažiausia reikšmė pastebima Baltarusijoje, Ukrainoje, Šiaurės Makedonijoje. Vertinant Lietuvos BVP, tenkantį vienam gyventojui, įdomu tai, kad ši reikšmė yra labai panaši į tokių šalių kaip Italija, Lenkija, Estija (remiantis visu vertinimo laikotarpiu).



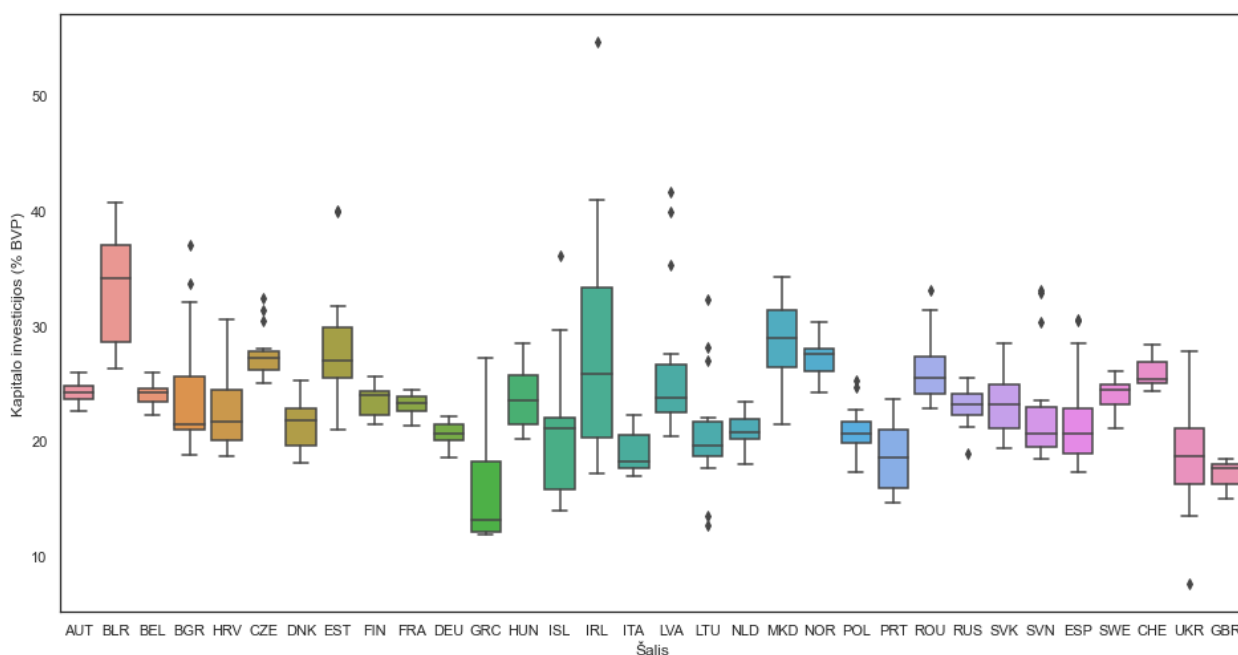
10 pav. Bendro vidaus produkto, tenkančio vienam gyventojui, šalių stačiakampės diagramos

Vertinant skirtingų šalių pragyvenimo lygį yra labai svarbu atsižvelgti į tas pačias sąlygas. Didesnis darbo užmokestis šalyje ne visada reiškia geresnes gyvenimo sąlygas, kadangi didesnės kainos gali lemti tuo pačiu ir didesnes išlaidas. Siekiant įvertinti skirtingų šalių pragyvenimo lygį dažnai yra naudojamas perkamosios galios paritetas. Šis metodas yra taikomas siekiant apskaičiuoti skirtingų valiutų tarpusavio kursus, pašalinant kainų skirtumus, bei palyginant šalių gyvenimo lygį. Remiantis toliau pateikta diagrama (11 pav.) galima pastebėti, kad didžiausia šio rodiklio reikšmė yra matoma Airijoje, taip pat šioje šalyje pastebimas ir gana didelis šio rodiklio nestabilumas. Verta pabrėžti ir tai, kad perkamosios galios paritetas yra mažiausias Ukrainoje, Baltarusijoje, Šiaurės Makedonijoje. Vertinant Lietuvos perkamosios galios paritetą, galima teigti, kad Lietuva turi vidutinišką perkamosios galios paritetą bendroje šalių imtyje.



11 pav. Perkamosios galios pariteto šalių stačiakampės diagramos

Įvairios pasaulio įmonės, norėdamos investuoti šalyje, dažnai turi skirti ir nemažai kapitalo, todėl svarbu apžvelgti ir kapitalo investicijas (12 pav.). Įdomu tai, kad didžiausios vidutinės kapitalo investicijos pastebimos Baltarusijoje, o mažiausios - Graikijoje. Vertinant atskiras reikšmes matoma, kad Airijoje yra išskirtinai didelė kapitalo investicijų reikšmė 2019 metais. Apžvelgiant papildomą informaciją galima teigti, kad tai lemia Airijoje esanti palanki mokesstinė aplinka⁴, taip pat antroji ir turbūt viena iš svarbiausių priežasčių šiai reikšmei yra BREXIT procesas, dėl kurio kompanijos, norinčios dirbti Europos Sąjungos rinkoje, yra perkeliamos į Airiją⁵. Ukrainoje 2020 metais pastebimas išskirtinai mažas kapitalo investicijų kiekis, to priežastis didžiaja dalimi yra COVID-19 pandemija, dėl kurios buvo atmesta daug žaliojo lauko (angl. *Greenfield*) investicijų ir NT sektoriaus vystymosi⁶. Vertinant Lietuvos kapitalo investicijas matoma, kad lyginant su kitomis Baltijos šalimis jos yra pačios mažiausios.



12 pav. Skirtingų šalių kapitalo investicijų stačiakampės diagramos

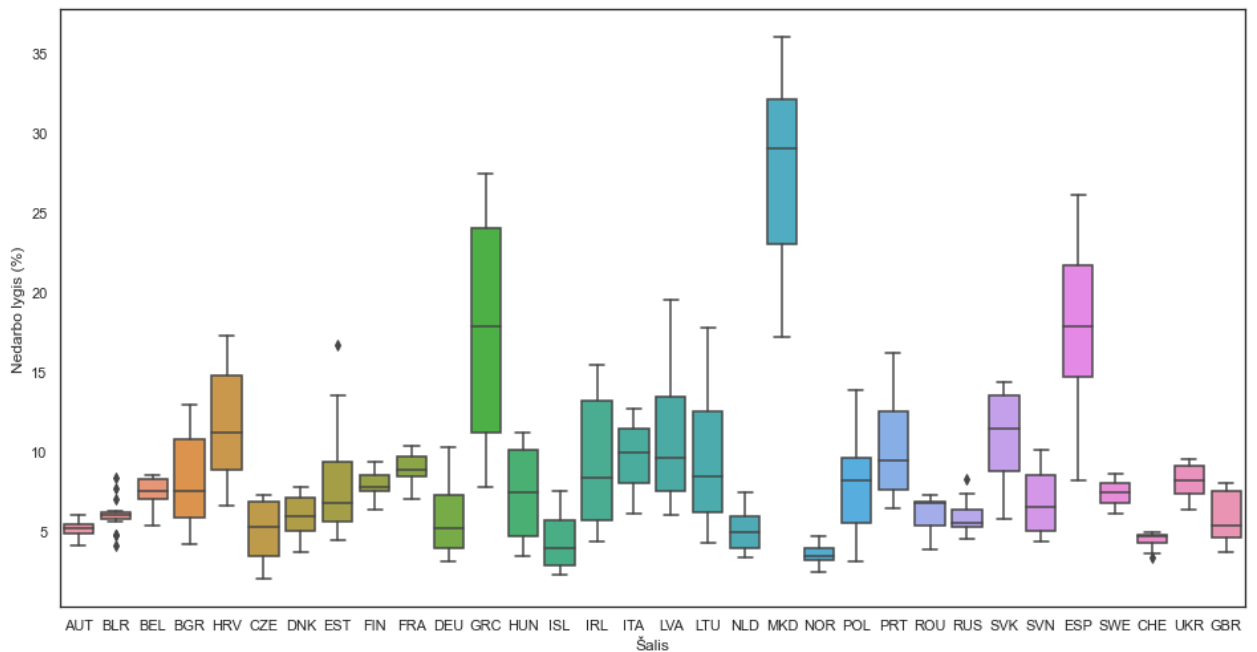
Toliau apžvelgiamas atskirų šalių nedarbo lygis (13 pav.). Galima pastebėti, kad didžiausiu vidutiniu nedarbo lygiu pasižymi Šiaurės Makedonija, Ispanija ir Graikija. Šiose šalyse didelis nedarbo lygis dažniausiai yra nulemtas valstybių darbo rinkos politikos problemų, pasyvių užimtumo programų ir palankios išmokų sistemos.⁷ 2010 metais pastebima išskirtinai didelė Estijos nedarbo reikšmė, kurią galėjo nulemti gana lėtas ekonomikos atsigavimas po krizės. Vėlesniais laikotarpiais pastebima, kad situacija stabilizavosi.

⁴<https://www.irishtimes.com/business/economy/foreign-direct-investment-in-ireland-exceeds-1-trillion-for-first-time-1.4565120>

⁵<https://www.investmentmonitor.ai/analysis/ireland-benefited-brexite-dublin-financial-services>

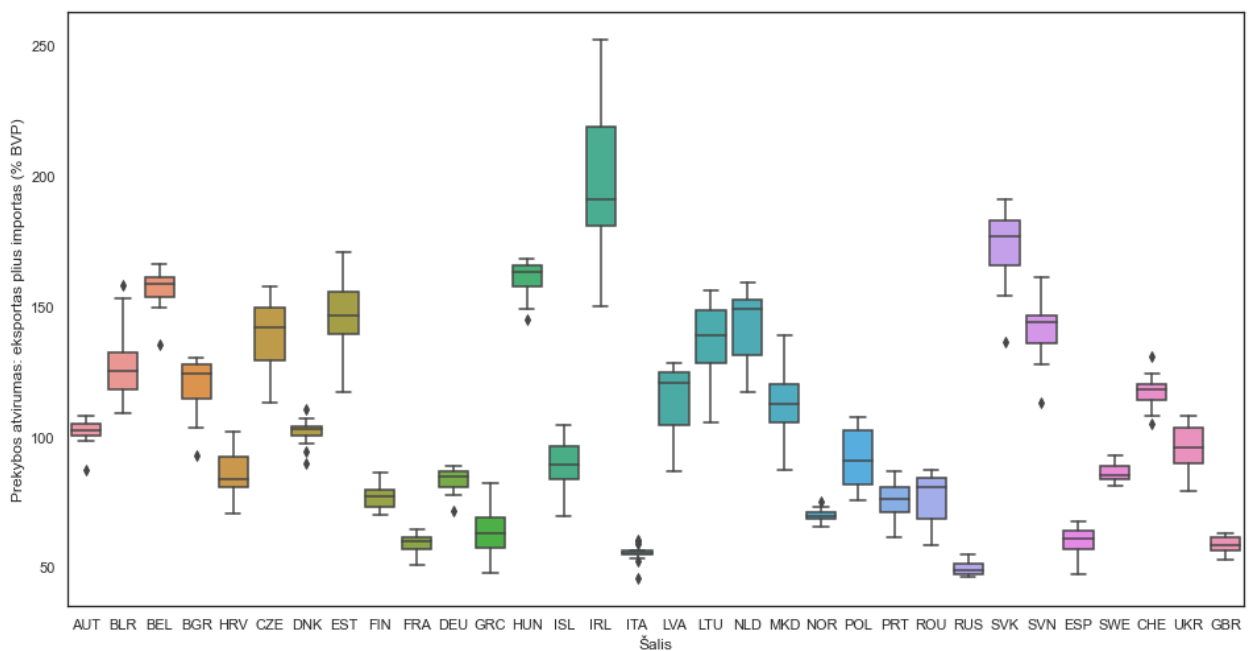
⁶<https://www.investmentmonitor.ai/analysis/ukraine-fdi-snapshot-foreign-investment>

⁷[https://www.europarl.europa.eu/RegData/etudes/IDAN/2017/607331/IPOL_IDA\(2017\)607331_EN.pdf](https://www.europarl.europa.eu/RegData/etudes/IDAN/2017/607331/IPOL_IDA(2017)607331_EN.pdf)



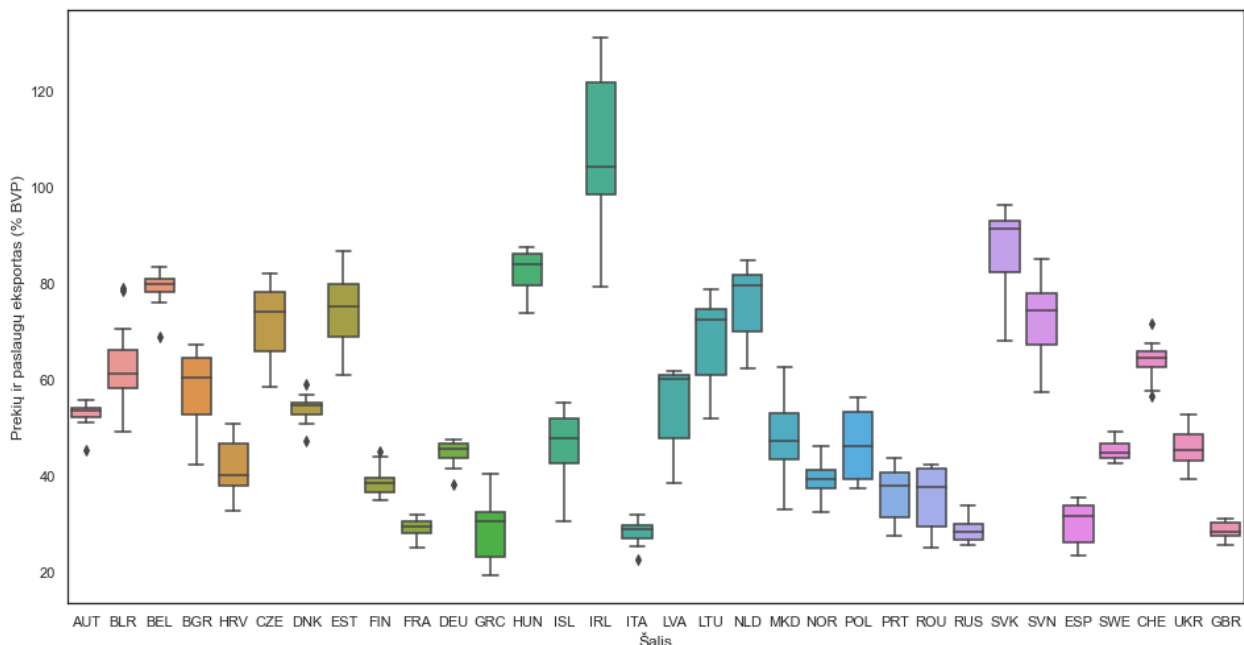
13 pav. Nedarbo lygio šalių stačiakampės diagramos

Vertinant prekybos atvirumą pastebima tai, kad Airija ir Slovakija pasižymi ypatingai didele šio rodiklio reikšmė (14 pav.). Visa tai lemia, kad įprastai mažos atviros ekonomikos turi nedideles rinkas, todėl atvirumas padeda vietos gamintojams plėsti savo produkciją pasaulinėje rinkoje. Įmonėms parduodant prekes tik savo rinkoje būtų sudėtinga jas išlaikyti, todėl tokia strategija ypatingai padeda gamintojams. Tuo tarpu priešingas atvejis pastebimas Rusijoje, Prancūzijoje, Graikijoje ir Italijoje. Šios šalys turi mažiausią vidutinį prekybos atvirumą. Galima teigti, kad didelės šalys, ribojančios įėjimą į savo rinką, siekia apsaugoti vietos gamintojus, kurie įprastai yra mažiau konkurencingi pasaulinėje rinkoje.



14 pav. Skirtingų šalių prekybos atvirumo rodiklio stačiakampės diagramos

Vertinant ne tik prekybos atvirumą, bet ir prekių bei paslaugų eksportą, galima pastebėti, kad didžiausia vidutine prekių ir paslaugų eksporto verte pasižymi Airija (15 pav.). Remiantis Airijos profiliu galima teigti, kad tai yra pirmaujanti šalis pagal kraujo eksportą pasaulyje. Taip pat Airija dėl palankios mokesstinės aplinkos yra pritraukusi tokių įmonių, kaip Microsoft, Apple bei kitas technologijų kompanijas. Šalis pagamina labai daug aukštųjų technologijų produkcijos, kurios vertė iškelia prekių ir paslaugų eksporto reikšmę, taip pat pastebima tai, kad medicinos sektorius šioje šalyje yra labai išvystytas.⁸ Prancūzija, Graikija, Italija ir Jungtinė Karalystė pasižymi labai mažu prekių ir paslaugų eksportu. Tai yra didelės rinkos, kuriose vietos gamintojai tenkina paklausą, o trūkumas šiose rinkose yra importuojamas. Dėl šių priežasčių dalinai yra siekiama riboti įėjimą į rinką kitų šalių įmonėms.

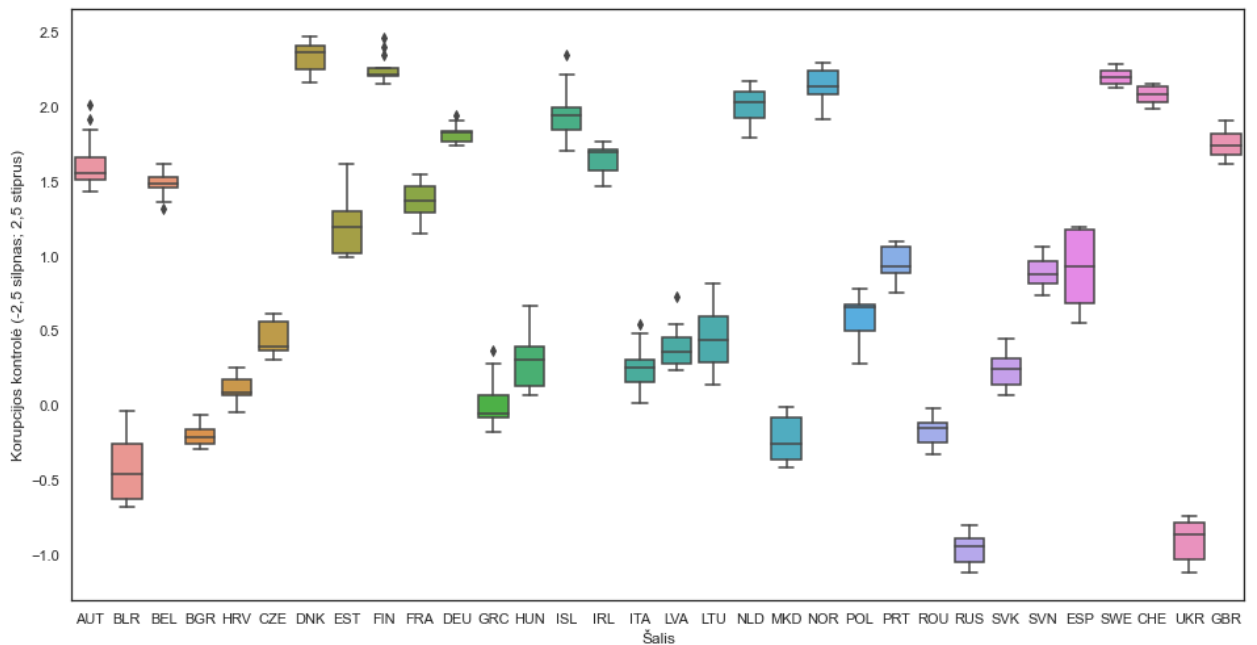


15 pav. Skirtingų šalių prekių ir paslaugų eksporto rodiklio stačiakampės diagramos

Vienas iš įdomių rodiklių, naudojamų šiame tyrime, yra korupcijos kontrolės vertinimas (16 pav.). Galima pastebėti, kad geriausiu korupcijos kontrolės indeksu pasižymi tokios šalys, kaip Danija, Suomija, Nyderlandai, Norvegija, Švedija ir Šveicarija. Tuo tarpu mažiausia šio rodiklio reikšmė pasižymi Rusija ir Ukraina. Lietuvos korupcijos kontrolės indeksas yra vertinamas geriau nei Latvijos, tačiau verta pastebėti tai, kad Estijos rodiklis yra reikšmingai geresnis. Lietuvos korupcijos kontrolės indeksas yra kur kas labiau panašus į Rytų Europos šalių korupcijos kontrolės indeksą: Lenkijos, Slovakijos ir kt.

21 priede galima pamatyti kitų rodiklių stačiakampės diagramas.

⁸ <https://oec.world/en/profile/country/irl>



16 pav. Skirtingų šalių korupcijos kontrolės indekso stačiakampės diagramos

3.3. Klasterinė analizė

Šiame skyriuje pateikiami šalių klasterinės analizės rezultatai. Klasterizavimo tikslas yra atskirti šalis, remiantis skirtingais tiesiogines užsienio investicijas lemiančiais veiksniais. Kadangi tyrime yra naudojamas 2006-2020 metų laikotarpis, šiuose duomenyse taip pat yra laiko kintamasis. Įprastai laiko kintamasis nėra naudojamas klasterinėje analizėje, todėl šalių klasterizavimui galima būtų naudoti vidutines rodiklių reikšmes visame laikotarpyje. Tačiau įtraukus laiko kintamąjį į analizę galima pastebėti ir kitimo laike tendencijas. Tyrime klasteriai buvo apskaičiuoti skirtingais laiko intervalais siekiant įvertinti, koks klasterių stabilumas ir kokie pasikeitimai yra pastebimi per visą laikotarpį. Šiame tyrime naudojami 6 laiko taškai:

1. 2006 metai – tyrimo duomenų pradinis laiko momentas;
2. 2008 metai – prieš krizinis laikotarpis, siekiant įvertinti, kurios šalys buvo panašiausios prieš ekonominę krizę;
3. 2009 metai – krizės laikotarpis, įvertinama šalių reakcija į krizę;
4. 2013 metai – tarpinis laiko taškas tarp krizės laikotarpio ir naujausių duomenų. Galima teigti, kad šiuo laiko momentu turėtų būti pastebimas atsigavimas po krizės. Šis laiko momentas leidžia įvertinti skirtingų šalių atsigavimą po ekonominės krizės;
5. 2019 metai – laikotarpis prieš COVID-19;
6. 2020 metai – COVID-19 laikotarpis, naujausi tyrime naudojami duomenys.

Kadangi klasterinė analizė yra atliekama su daug rodiklių, rezultatų vizualizacijai yra naudojami žemėlapiai, išskiriantys atskiras šalis pagal klasterius. Kaip jau buvo minėta anksčiau, klasterinėje analizėje yra naudojami du atskiri duomenų rinkiniai: pirmajame duomenų rinkinyje yra pradiniai duomenys, o antrajame duomenų rinkinyje naudojamas AMECO duomenų bazės duomenimis papildytas duomenų rinkinys, kur dėl duomenų trūkumo vertinama mažiau šalių. Tolimesniuose skirsniuose pateikiami atskirų duomenų rinkinių klasterizavimo rezultatai. Klasterinė analizė buvo atliekama skirtingais metodais bei keičiant šių metodų parametrus kiekvieniems metams atskirai. Bendrai buvo sudaryta daugiau nei 30 tūkstančių skirtingų klasterizavimo metodų, kurie vėliau buvo

vertinami remiantis Silueto koeficientų metrikomis. Galutinis sprendimas dėl geriausių metodų buvo priimtas remiantis Silueto metrika su Euklido atstumu. Pirmajame šio poskyrio skirsnyje yra aptariamas pradinių duomenų, antrajame - papildytų duomenų klasterizavimas.

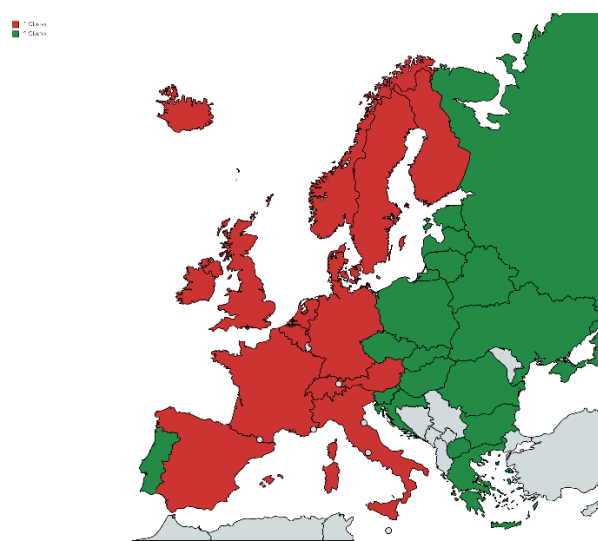
3.3.1. Duomenų klasterizavimas, naudojant pradinį duomenų rinkinį

Šiame skirsnyje pateikiama pagrindinė informacija apie klasterizavimo rezultatus, naudojant pradinį duomenų rinkinį. 3 lentelėje pateikiami geriausi klasterizavimo modeliai kiekvienais metais. Galima pastebėti, kad geriausiais visu tiriamuoju laikotarpiu buvo išskiriami 2 arba 3 klasteriai. Taip pat verta paminėti tai, kad nors čia pateikti tik geriausi modeliai, tačiau daugumoje metų trys arba penki geriausi modeliai parodė vienodus rezultatus. Pavyzdžiui, 2006 metais pateiktas geriausias modelis yra k-vidurkių, tačiau antroje ir trečioje vietoje esantys Gauso mišinių ir Bajeso Gauso mišinių metodai pasižymėjo tokiais pačiais rezultatais. Taip pat pastebima tai, kad išskirti klasteriai buvo lygiai tokie patys su visais trimis metodais, todėl galima teigti, kad šie klasteriai tikrai gerai išskiriami. Daugiau tyrimo rezultatų yra pateikiama 1-9 prieduose, kuriuose galima matyti ne tik geriausius kiekvienų metų duomenis, tačiau ir kitus palyginimui naudotus metodus.

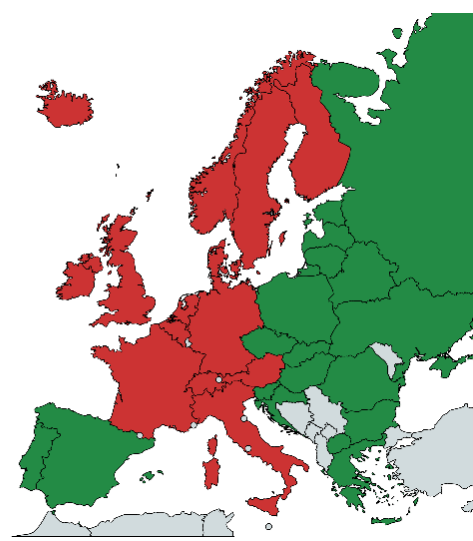
3 lentelė. Geriausi kiekvienų metų klasterizavimo rezultatai, naudojant pradinį duomenų rinkinį

Metai	Metodas	Parametrai	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
2006	kmeans	{'k': 2}	0,7648	0,6637	0,6686
2007	kmeans	{'k': 2}	0,7143	0,6543	0,6568
2008	kmeans	{'k': 2}	0,6892	0,6532	0,6538
2009	GMM	{'k': 3, 'init_params': 'kmeans'}	0,5625	0,6589	0,6566
2010	GMM	{'k': 3, 'init_params': 'kmeans'}	0,6000	0,6784	0,6750
2011	kmeans	{'k': 3}	0,6070	0,7079	0,7022
2012	Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	0,6474	0,7179	0,7158
2013	kmeans	{'k': 3}	0,6366	0,7170	0,7140
2014	BGMM	{'k': 3, 'init_params': 'kmeans'}	0,6377	0,7156	0,7093
2015	kmeans	{'k': 2}	0,7396	0,6663	0,6691
2016	GMM	{'k': 2, 'init_params': 'kmeans'}	0,7300	0,6591	0,6622
2017	kmeans	{'k': 2}	0,7414	0,6455	0,6489
2018	GMM	{'k': 2, 'init_params': 'kmeans'}	0,7345	0,6399	0,6412
2019	kmeans	{'k': 2}	0,7572	0,6371	0,6377
2020	Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	0,5181	0,6379	0,6393

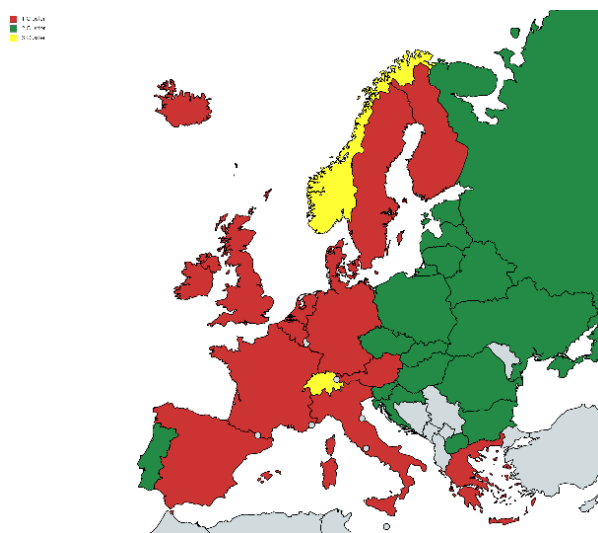
17 paveiksle pateikiamas klasterių vaizdavimas žemėlapyje. Verta paminėti tai, kad 2006 ir 2008 metais išskiriami tik 2 klasteriai, kurie galėtų būti įvardijami, kaip Rytų ir Vakarų Europos klasteriai. Pastebima, kad šiuo atveju, vertinant abu laikotarpius, Portugalija yra panašesnė į Rytų Europą. Tuo tarpu Ispanija tik 2008 metais yra priskiriama tam pačiam klasteriui, kaip Rytų Europos šalys, nors 2006 metais buvo priskiriama tam pačiam klasteriui, kaip kitos Vakarų Europos šalys. 2009 metais Ispanija ir vėl yra priskiriama tam pačiam klasteriui, kaip ir kitos Vakarų Europos šalys. Taip pat svarbus aspektas yra, kad 2009 ir 2013 metais yra išskiriami ne du klasteriai, tačiau trys. Verta pastebėti, kad remiantis geriausių rezultatų lentele, trys klasteriai yra išskiriami nuo 2009 metų iki 2014 metų imtinai. Taip pat, vertinant naujausius laikotarpius, svarbus faktorius yra tai, kad Italija, taip pat kaip ir Ispanija, yra priskiriama prie kitų Rytų Europos šalių. Klasterių vidutinės reikšmės pateikiamos 10-11 prieduose.



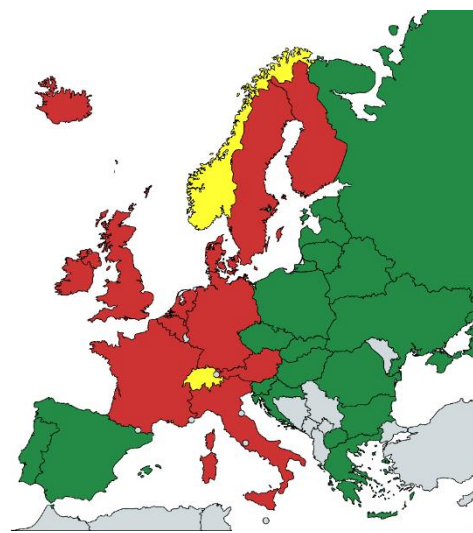
2006



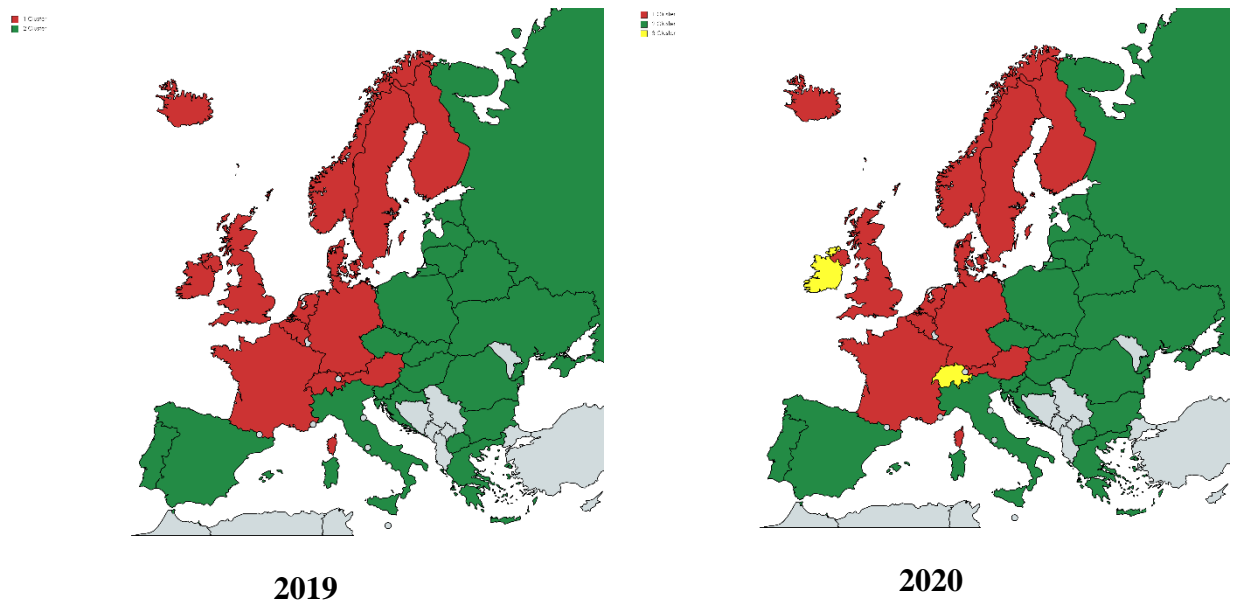
2008



2009



2013



17 pav. Klasterių vizualizavimas skirtingais tiriamo laikotarpio momentais, naudojant pradinį duomenų rinkinį.

3.3.2. AMECO duomenų baze papildytas duomenų rinkinys

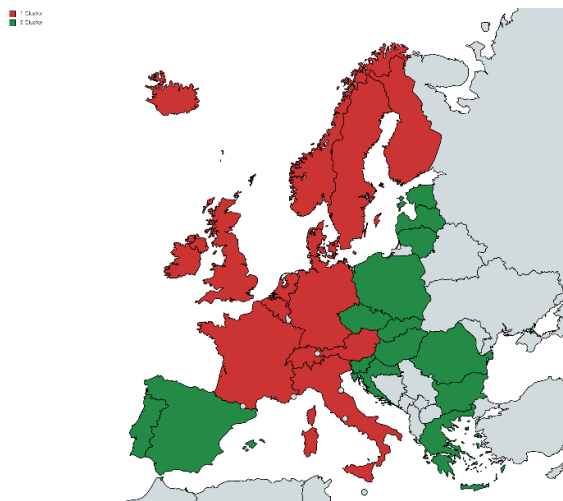
Šiame skirsnyje pateikiama antra klasterinė analizė, kurioje naudojamas duomenų rinkinys papildytas keliais rodikliais. 4 lentelėje pateikiami geriausi klasterizavimo modeliai kiekvienais metais. Galima pastebėti, kad geriausiai buvo išskiriami 2 arba 3 klasteriai visu tiriamuoju laikotarpiu. Taip pat verta paminėti tai, kad nors čia pateikti tik geriausi modeliai, tačiau daugumoje metų 3-7 geriausi modeliai parodė vienodus rezultatus. Galima pastebėti, kad dažniau nei ankstesniame duomenų rinkinyje yra išskiriami ne du, o trys klasteriai. Daugiau tyrimo rezultatų yra pateikiama 12-19 prieduose, kuriuose galima matyti ne tik geriausius kiekvienų metų duomenis, tačiau ir kitus metodus. Lyginant pradinio duomenų rinkinio ir naujo duomenų rinkinio klasterizavimo rezultatus, galima pastebėti, kad Silueto metrikos reikšmės yra labai panašios, tačiau dažnu atveju papildytas duomenų rinkinys pasižymi geresniais klasterizavimo rezultatais.

4 lentelė. Geriausi kiekvienų metų klasterizavimo rezultatai, naudojant papildytą duomenų rinkinį.

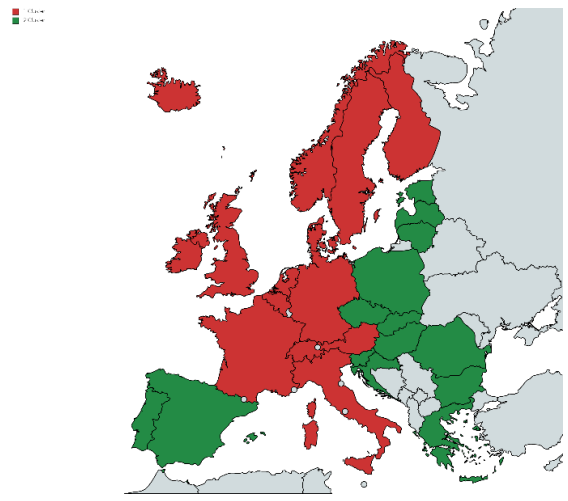
Metai	Metodas	Parametrai	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
2006	kmeans	{'k': 2}	0,7488	0,6474	0,6517
2007	GMM	{'k': 2, 'init_params': 'kmeans'}	0,7553	0,6442	0,6485
2008	kmeans	{'k': 2}	0,7446	0,6447	0,6470
2009	kmeans	{'k': 3}	0,5685	0,6547	0,6542

Metai	Metodas	Parametrai	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
2010	Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	0,6024	0,6822	0,6797
2011	Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	0,6640	0,7166	0,7126
2012	kmeans	{'k': 3}	0,6838	0,7273	0,7280
2013	GMM	{'k': 3, 'init_params': 'kmeans'}	0,6714	0,7249	0,7247
2014	kmeans	{'k': 3}	0,6911	0,7255	0,7206
2015	GMM	{'k': 3, 'init_params': 'kmeans'}	0,5556	0,6781	0,6821
2016	kmeans	{'k': 2}	0,7909	0,6566	0,6609
2017	kmeans	{'k': 3}	0,6295	0,6527	0,6488
2018	GMM	{'k': 3, 'init_params': 'kmeans'}	0,6063	0,6555	0,6511
2019	Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	0,5341	0,6439	0,6504
2020	GMM	{'k': 3, 'init_params': 'kmeans'}	0,5727	0,6566	0,6597

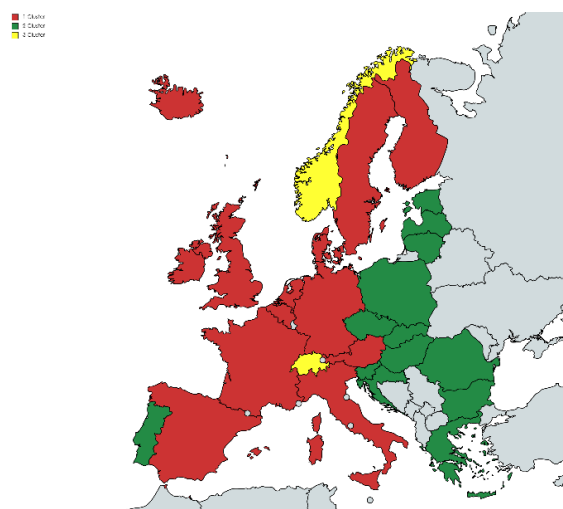
Galima pastebėti, kad klasterizavimo rezultatuose yra matomi nedideli pasikeitimai (18 pav.). Nenaudojant papildomų duomenų 2006 metais Ispanija buvo priskiriama klasteriui, kuriame yra vakarų Europos šalys, tačiau vertinant papildomus duomenis Ispanija jau ir 2006 metais yra priskiriama labiau prie rytų Europos šalių. Taip pat vienas žymesnių pasikeitimų klasterizavimo rezultatuose yra tai, kad 2019 metais nenaudojant papildomų duomenų nebuvo išskirtas trečias klasteris: Airija, Šveicarija, Norvegija. Kai tuo tarpu su papildomais duomenimis šios šalys jau yra išskiriamos. Taip pat 2020 metais Italija jau yra pastebimai priskiriama prie rytų, o ne vakarų Europos šalių.



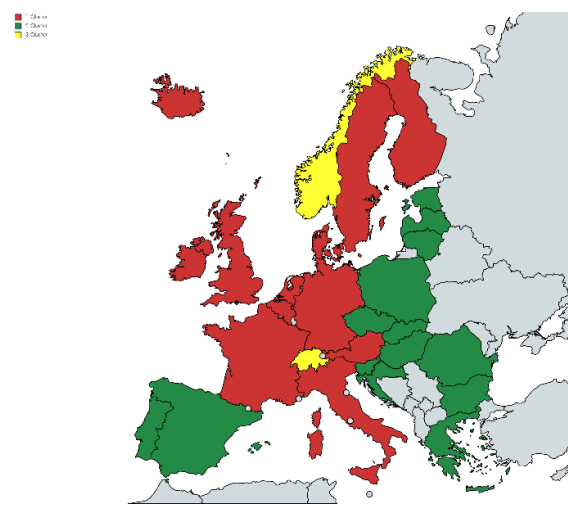
2006



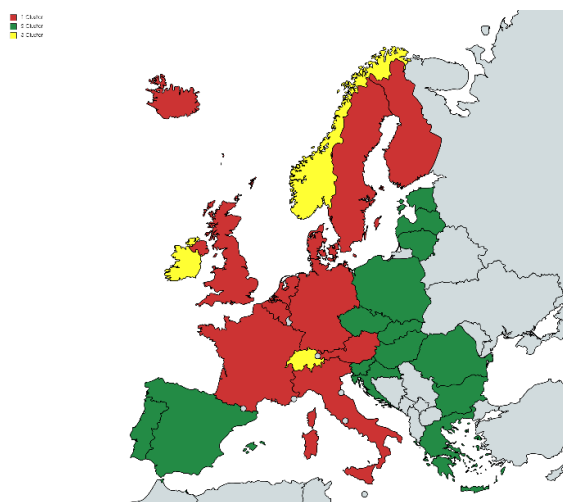
2008



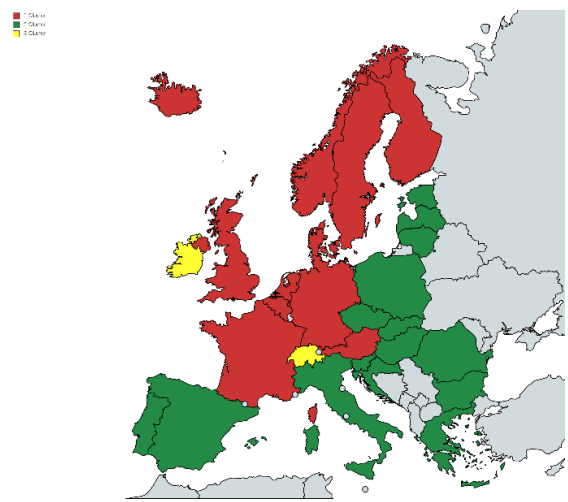
2009



2013



2019



2020

18 pav. Klasterių vizualizavimas skirtingais tiriamo laikotarpio momentais, naudojant papildomą duomenų rinkinį.

Toliau pateikiama informacija apie išskirtų klasterių profilius. Remiantis gautais geriausiais klasterizavimo rezultatais, aptariami naujausi (2020 metais) sudaryti klasteriai:

- **1 Klasteris:** Austrija, Belgija, Danija, Suomija, Prancūzija, Vokietija, Islandija, Nyderlandai, Norvegija, Švedija, Jungtinė Karalystė.
- **2 Klasteris:** Bulgarija, Kroatija, Čekija, Estija, Graikija, Vengrija, Italija, Latvija, Lietuva, Lenkija, Portugalija, Rumunija, Slovakija, Slovėnija, Ispanija.
- **3 Klasteris:** Airija, Šveicarija.

Vertinant klasterių profilius galima pastebėti, kad trečiasis klasteris 2020 metais pasižymi ekonomikos augimu (+1,74%), kai tuo tarpu kiti du klasteriai patiria didesnę ekonomikos nuosmukį (1 klasteris -4,82%, 2 klasteris -5,46%). Taip pat pastebima tai, kad bendras vidutinis vidaus produktas, tenkantis vienam gyventojui, trečiajame klasteryje yra didžiausias (86182 \$), pirmajame klasteryje šis rodiklis yra kur kas mažesnis (51025\$), o antrajame klasteryje vidutinė šio rodiklio reikšmė palyginus su kitais klasteriais yra labai maža (19739\$). Vertinant kapitalo investicijas pirmasis ir antrasis klasteris pasižymi panašiomis vidutinėmis reikšmėmis (23,41 ir 21,02), o trečiasis klasteris yra išskirtinis, kadangi kapitalo investicijos šiame klasteryje siekia 34,62. Taip pat įdomu tai, kad didžiausias vartojimas yra pastebimas rytų Europos šalyse (2 klasteryje), o mažiausias - trečiajame klasteryje. Kalbant apie infliaciją trečiajame klasteryje matoma neigiama infliacija, o kiti du klasteriai pasižymi panašia klasterio vidutine infliacija. Didžiausias darbo jėgos kiekis matomas pirmajame klasteryje (vakarų Europos šalyse), tačiau didžiausias nedarbo lygis matomas antrajame klasteryje (rytų Europos šalyse). Didžiausias prekybos atvirumas matomas trečiajame klasteryje, kai tuo tarpu vakarų Europos šalyse prekybos atvirumas yra santykinai mažas palyginus su kitais klasteriais bei šio rodiklio vidutinė reikšmė nesiekia 100. Eksportas taip pat yra stipriai susietas su prekybos atvirumu ir galima pastebėti tai, kad klasteris, kuriame matomas didžiausias prekybos atvirumas, pasižymi ir didžiausiu vidutiniu eksporto rodikliu. Pirmajame ir trečiajame klasteryje pastebimas panašus vyriausybės efektyvumas, o antrasis klasteris turi santykinai mažą vyriausybės efektyvumą. Ta pati tendencija pastebima ir korupcijos kontrolės atveju, kadangi antrasis klasteris turi kur kas prastesnę šio rodiklio reikšmę. Politinio stabilumo reikšmės turi panašias tendencijas, kadangi vakarų Europos šalyse šis rodiklis vertinamas geriau nei rytų Europos šalyse. Rodiklis, kuris pastebimai turi kitą tendenciją, lyginant su anksčiau aptartais rodikliais, yra fiskalinė kontrolė. Mažiausia šio rodiklio reikšmė pastebima pirmajame klasteryje, o antrajame ir trečiajame klasteryje vidutinės šio rodiklio reikšmės yra panašios. Geriausias verslo laisvė pasižymi pirmasis klasteris, tuo tarpu prasčiausia verslo laisvė matoma antrajame klasteryje. Taip pat šiame klasteryje pastebima ir reikšmingai mažesnė verslo laisvės rodiklio reikšmė, todėl galima teigti, kad investuoti šiame klasteryje yra kur kas sudėtingiau lyginant su kitais klasteriais. Antrajame klasteryje yra daug mažesnė finansinė ir ekonominė laisvė (vidutinė), todėl tai tik dar labiau patvirtina teiginį apie sudėtingesnes investicijas šiam klasteriui priskirtose šalyse. Mieste gyvenančių asmenų procentinė dalis turi gana panašias vidutines reikšmes trečiojo ir antrojo klasterio šalyse, o pirmajame klasteryje šis santykis yra pats didžiausias. Tokią pačią tendenciją turi ir gyventojų tankis. Įdomu tai, kad didžiausi kapitalo mokesčiai yra pastebimi pirmojo klasterio šalyse, o antrojo ir trečiojo klasterio šalyse vidutiniai kapitalo mokesčiai yra panašūs. Tačiau nors šie mokesčiai yra didžiausi, pastebima tai, kad pirmasis klasteris pasižymi didžiausiu produktyvumu, todėl tai ypatingai svarbu įvertinti investuojant šiose šalyse. Išsamūs rezultatai yra pateikiami 18 ir 19 priede.

3.4. Duomenų regresinė analizė ir prognozavimas

Šiame poskyryje yra pateikiami duomenų regresinės analizės rezultatai bei yra siekiama atlikti tiesioginių užsienio investicijų įplaukų prognozavimą ateinantiems metams. Šiuo atveju prognozuojami metai yra 2021, kadangi turimi duomenys buvo tik iki 2020 metų. Atliekant regresinę analizę ir prognozavimą yra remiamasi ir anksčiau atlikta duomenų klasterine analize. Sudarant skirtingus regresijos modelius naudojami tiek pilni, tiek atskirų klasterių duomenys, taip siekiant nustatyti, ar remiantis klasterių informacija gali būti gaunami geresni prognozavimo rezultatai. Šioje darbo dalyje yra priimama prielaida apie dviejų klasterių išskyrimą, kadangi trečiajame klasteryje pastebimas labai mažas šalių skaičius bei tam tikrais laikotarpiais buvo išskiriami tik du klasteriai. Klasteriams yra priskiriamos šios šalys:

- **1 Klasteris:** Austrija, Belgija, Danija, Suomija, Prancūzija, Vokietija, Islandija, Airija, Nyderlandai, Norvegija, Švedija, Jungtinė Karalystė, Šveicarija;
- **2 Klasteris:** Bulgarija, Kroatija, Čekija, Estija, Graikija, Vengrija, Italija, Latvija, Lietuva, Lenkija, Portugalija, Rumunija, Slovakija, Slovėnija, Ispanija.

3.4.1. Panelinių duomenų regresinė analizė (PooledOLS)

Šiame skirsnyje atliekama sutelktinė MKM regresinė analizė (angl. *PooledOLS*). Naudojant šią analizę galima nagrinėti panelinius duomenis, būtent tokius, kokie yra naudojami šiame tyrime. Toliau yra pateikiami atskirų klasterių sudaryti modeliai. Šie modeliai buvo kuriami su visais kintamaisiais. Vėliau, panaudojant žingsninę strategiją (angl. *stepwise*), buvo atmetami statistiškai mažiausias reikšmes turintys kintamieji taip sudarant tik reikšmingų kintamųjų modelius. Toliau pateiktame pirmojo klasterio modelyje (5 lentelė) matoma, kad apibrėžtumo koeficientas yra 0,3639, todėl galima teigti, kad tyrime naudojami išoriniai kintamieji paaiškina tik 36,39% duomenų variacijos. Dėl šios priežasties galime manyti, kad egzistuoja daug papildomų kintamųjų, lemiančių tiesioginių užsienio investicijų įplaukas. Paklaidų vidurkis yra artimas nuliui, todėl galima teigti, kad ši modelio prielaida yra tenkinama. Taip pat verta pastebėti, kad F modelio reikšmė $p < 0,05$, todėl modelis yra prasmingas. Paklaidos nėra paskirstytos pagal normalųjį skirstinį, o tai reiškia, kad standartinės paklaidos ir pasikliautiniai intervalai, susiję su modelio prognozėmis, gali būti nepatikimi. Verta pastebėti ir tai, kad šiame modelyje reikšmingi kintamieji yra tokie kaip perkamosios galios paritetas (PPP), namų ūkių suvartojimas, infliacija, darbo jėgos kiekis, eksportas, fiskalinė ir investavimo laisvė bei kiti toliau pateikiami rodikliai. Įdomu tai, kad didėjant infliacijai, tiesioginės užsienio investicijos yra mažėjančios. Tuo tarpu didėjant fiskalinei ir investavimo laisvei, tiesioginės užsienio investicijos didėja (1,39 ir 3,27 koeficientai atitinkamai).

5 lentelė. Pirmojo klasterio sutelktinės MKM regresinės analizės rezultatai.

Sutelktinės MKM regresinės analizės įvertinimo santrauka			
=====			
Priklausomas kintamasis:	TUI	Apibrėžtumo koeficientas:	0.3639
Įvertintojas: Sutelktinė MKM regresija		Apibrėžtumo koeficientas (tarp):	0.9841
Stebinių skaičius:	210	Apibrėžtumo koeficientas (viduje):	0.1603
Data:	Sun, May 2 2022	Apibrėžtumo koeficientas (bendras):	0.3639
Laikas:	19:18:30	Logaritminis tikėtinumumas	-1207.4
Kovariacijų įvertis:	Nekoreguotas		
		F-statistika:	8.6258
Subjektai:	14	P-reikšmė	0.0000
Vidutiniškai stebinių:	15.000	Pasiskirstymas:	F(13,196)
Mažiausiai stebinių:	15.000		
Daugiausiai stebinių:	15.000	F-statistika (stabili, angl.robust):	8.6258
		P-reikšmė	0.0000
Laiko periodai:	15	Pasiskirstymas:	F(13,196)

Vidutiniškai stebinių: 14.000
 Mažiausiai stebinių: 14.000
 Daugiausiai stebinių: 14.000

Parametru įverčiai

	Parametras	Stand.kl.	T-statistika	P-reikšmė	Žemutinis PI	Viršutinis PI
Pastovus įvertis	16.536	260.07	0.0636	0.9494	-496.37	529.44
Perkamosios galios paritetas	-0.0035	0.0012	-2.8654	0.0046	-0.0059	-0.0011
Namų ūkių vartojimas	-5.7359	1.7028	-3.3685	0.0009	-9.0941	-2.3778
Infliacija	-11.511	4.0616	-2.8342	0.0051	-19.521	-3.5012
Darbo jėgos kiekis	1.7413	0.6345	2.7444	0.0066	0.4900	2.9926
Eksportas	-1.5965	0.5488	-2.9091	0.0040	-2.6789	-0.5142
Fiskalinė laisvė	1.3959	0.7387	1.8896	0.0603	-0.0610	2.8527
Investavimo laisvė	3.2781	0.7606	4.3099	0.0000	1.7781	4.7781
Gyventojų tankumas	0.3596	0.0619	5.8120	0.0000	0.2376	0.4816
65 metų ir vyresni gyventojai	-25.145	4.9188	-5.1120	0.0000	-34.845	-15.444
Vartojimo kainų indeksas	-2.1438	0.7862	-2.7270	0.0070	-3.6942	-0.5934
Grynojo kapitalo sandoriai	45.830	19.552	2.3440	0.0201	7.2711	84.388
Darbo jėgos dalis						
gamybos produktyvumui	7.2241	2.6480	2.7281	0.0069	2.0018	12.446
Atotrūkis tarp faktinio BVP ir tendencijų	6.0230	2.1874	2.7534	0.0065	1.7091	10.337

Toliau pateiktuose regresinės analizės rezultatuose matyti, kad F modelio reikšmė $p < 0,05$, todėl modelis yra prasmingas. Pateiktame antrojo klasterio modelyje (6 lentelė) matoma, kad apibrėžtumo koeficientas yra 0,3805, todėl galima teigti, kad tyrime naudojami išoriniai kintamieji paaiškina tik 38,05% duomenų variacijos, kas reiškia, jog egzistuoja daug papildomų kintamųjų, lemiančių tiesioginių užsienio investicijų įplaukas. Paklaidų vidurkis yra artimas nuliui, todėl ši modelio prielaida yra tenkinama. Paklaidos nėra paskirstytos pagal normalųjį skirstinį, o tai kaip ir ankstesniame modelyje reiškia, kad negalima pilnai pasitikėti pasikliautinaisiais intervalais. Verta pastebėti, kad šiame modelyje reikšmingi kintamieji yra tokie kaip BVP, tenkantis vienam žmogui, produktyvumas ir kt. Įdomu tai, kad didėjant BVP, tenkančiam vienam gyventojui, pastebimas nežymus tiesioginių užsienio investicijų įplaukų didėjimas. Taip pat pastebima ir tai, kad didėjant produktyvumui, tiesioginės užsienio investicijos mažėja. Remiantis tuo galima manyti, kad investuotojai yra labiau linkę investuoti į šalis, kuriose šiuo metu produktyvumas dar nėra toks didelis ir tikimasi, kad šiose šalyse jis padidės.

6 lentelė. Antrojo klasterio sutelktinės MKM regresinės analizės rezultatai.

Sutelktinės MKM regresinės analizės įvertinimo santrauka

Priklausomas kintamasis:	TUI	Apibrėžtumo koeficientas:	0.3805
Įvertintojas:	Sutelktinė MKM regresija	Apibrėžtumo koeficientas (tarp):	0.9791
Stebinių skaičius:	210	Apibrėžtumo koeficientas (viduje):	0.1024
Data:	Sun, May 2 2022	Apibrėžtumo koeficientas (bendras):	0.3805
Laika:	19:38:06	Logaritminis tikėtinumumas	-868.55
Kovariacijų įvertis:	Nekoreguotas		
Subjektai:	14	F-statistika:	17.721
Vidutiniškai stebinių:	15.000	P-reikšmė	0.0000
Mažiausiai stebinių:	15.000	Pasiskirstymas:	F(7,202)
Daugiausiai stebinių:	15.000		
		F-statistika (stabili, angl.robust):	17.721
		P-reikšmė	0.0000
		Pasiskirstymas:	F(7,202)
Laiko periodai:	15		
Vidutiniškai stebinių:	14.000		
Mažiausiai stebinių:	14.000		
Daugiausiai stebinių:	14.000		

Parametru įverčiai

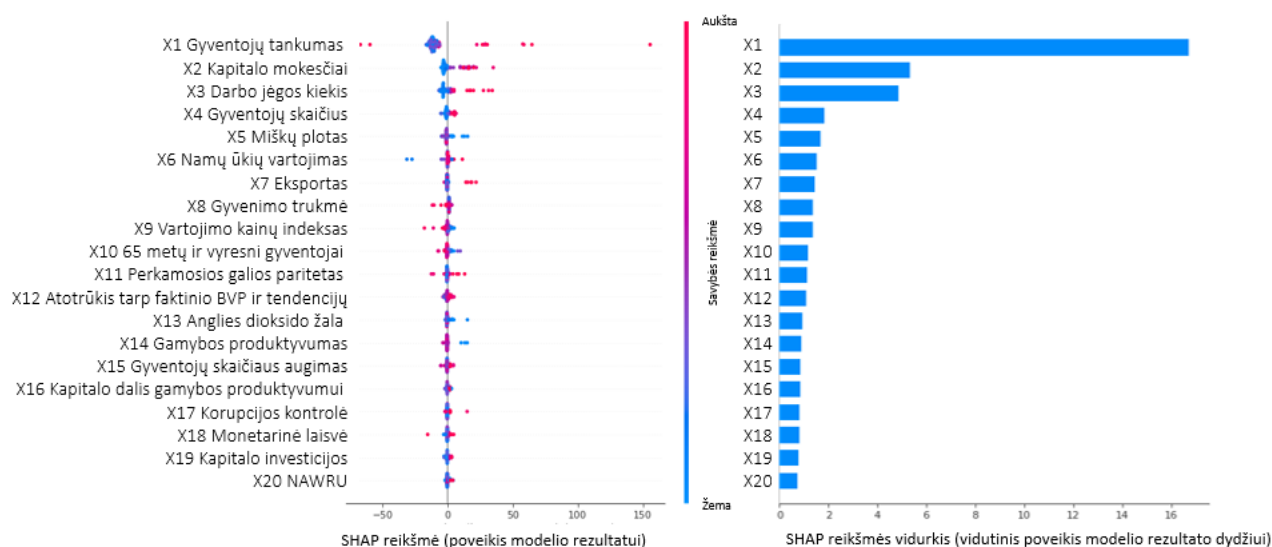
	Parametras	Stand.kl.	T-statistika	P-reikšmė	Žemutinis PI	Viršutinis PI
Pastovus įvertis	58.894	21.432	2.7479	0.0065	16.634	101.15
BVP vienam gyventojui	-0.0014	0.0004	3.9207	0.0001	0.0007	0.0021
Prekybos atvirumas	0.1160	0.0358	-3.2424	0.0014	-0.1866	-0.0455
Anglies dioksido žala	8.9630	2.5795	3.4748	0.0006	3.8769	14.049
Miškų plotas	-0.3828	0.1472	-2.6004	0.0100	-0.6730	-0.0925

Grynojo kapitalo sandoriai	-7.9044	2.4972	-3.1654	0.0018	-12.828	-2.9806
Kapitalo mokesčiai	-2.2059	0.2844	7.7560	0.0000	1.6451	2.7666
Gamybos produktyvumas	0.4419	0.1993	-2.2179	0.0277	-0.8348	-0.0490

Verta pastebėti tai, kad šiuose modeliuose nėra idealiai tenkinamos visos sąlygos, todėl jie yra naudojami tik nustatyti, kurie iš išorinių kintamųjų turi tiesioginį statistiškai reikšmingą ryšį su tiesioginių užsienio investicijų įplaukomis.

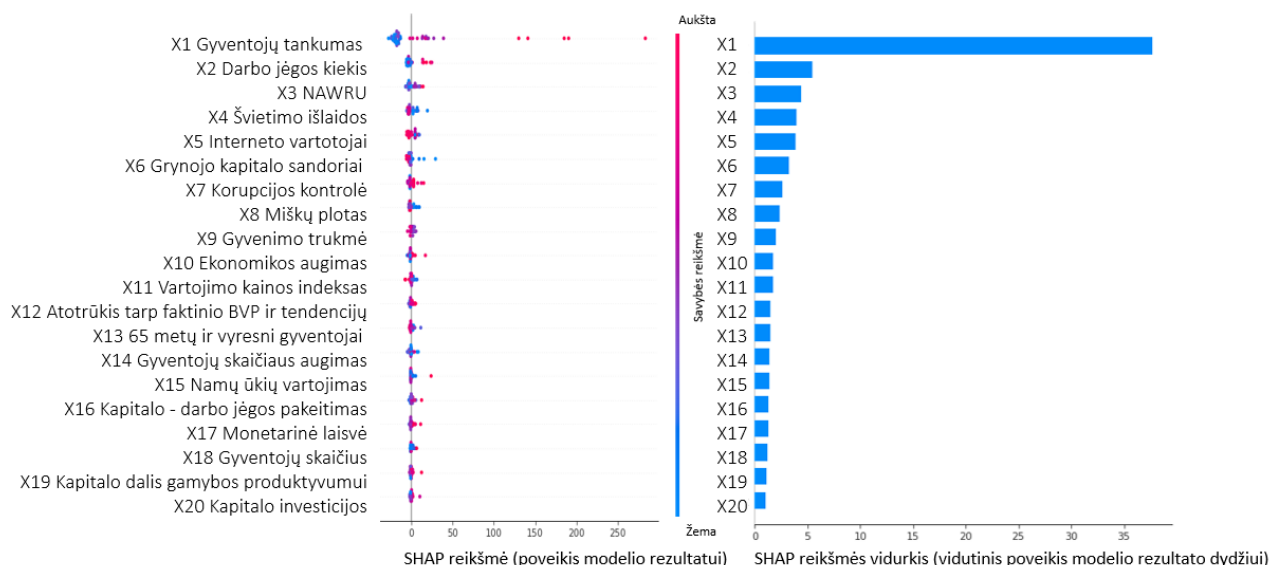
3.4.2. Kintamųjų įtakos nustatymas remiantis atsitiktinio miško metodu

Šiame poskyryje yra apžvelgiami geriausi atsitiktinio miško sukurti regresijos modeliai, tiksliau skirtingų kintamųjų įtaka tiesioginėms užsienio investicijoms. Siekiama įvertinti, kokie kintamieji yra reikšmingiausi modelyje, kuris nėra „juoda dėžė“. 19 paveiksle pateiktas geriausio atsitiktinio miško modelio parametrų reikšmingumas, naudojant visą duomenų rinkinį, o ne atskirus jo klasterius. Iš rezultatų matoma, kad didžiausią įtaką tiesioginių užsienio investicijų įplaukoms turi gyventojų tankis, taip pat didelę įtaką turi tokie kintamieji, kaip kapitalo mokesčiai, darbo jėgos ir gyventojų skaičius.



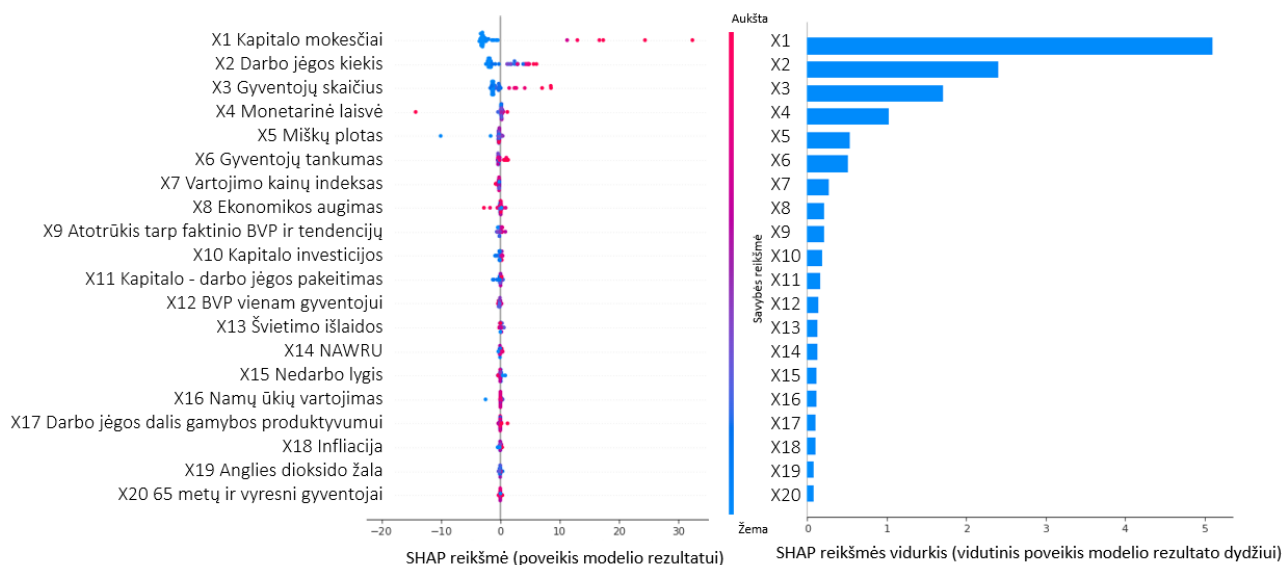
19 pav. Skirtingų kintamųjų įtaka atsitiktinio miško regresiniam modeliui, nenaudojant atskirų klasterių

Toliau pateikiama informacija apie pirmojo klasterio atsitiktinio miško regresijos modelio kintamųjų įtaką tiesioginių užsienio investicijų įplaukoms (20 pav.). Galima pastebėti, kad, kaip ir pirmuoju atveju, tiesiogines užsienio investicijas labiausia lemia gyventojų tankis ir darbo jėga. Tačiau, skirtingai nuo ankstesnio modelio, šiuo atveju didelę reikšmę turi NAWRU (angl. *Non-accelerating wage rate of unemployment*) rodiklis.



20 pav. Skirtingų kintamųjų įtaka atsitiktinio miško regresiniam modeliui pirmojo klasterio atveju

Toliau pateikiama informacija apie antrojo klasterio atsitiktinio miško regresijos modelio kintamųjų įtaką tiesioginėms užsienio investicijų įplaukoms (21 pav.). Galima pastebėti, kad šiuo atveju gyventojų tankis neturi tokios didelės reikšmės. Didžiausią įtaką tiesioginėms užsienio investicijoms turi kapitalo mokesčiai ir darbo jėgos kiekis. Apibendrinant galima teigti, kad investuojant rytų Europoje, t.y. antrajame klasteryje, didžiausias dėmesys skiriamas kapitalo mokesčiams.



21 pav. Skirtingų kintamųjų įtaka atsitiktinio miško regresiniam modeliui antrojo klasterio atveju

3.4.3. 2021 metų tiesioginių užsienio investicijų įplaukų prognozė, naudojant geriausius regresijos metodus

Šiame poskyryje pateikiami geriausi regresijos modelių rezultatai atskirai kiekvienai šaliai. Išsamūs rezultatai yra pateikiami 22-24 prieduose. Sudarant regresijos modelius, buvo naudojami skirtingi metodai tiek bendram duomenų rinkiniui, tiek atskiriems klasteriams, išskirtiems ankstesniame skyriuje. 7 lentelėje pateikiamos tiesioginių užsienio investicijų prognozės 2021-iems metams. Galima pastebėti, kad remiantis geriausiais modeliais atskirai kiekvienam klasteriui, tiesioginių užsienio investicijų mažėjimas prognozuojamas 5 valstybėse: Vokietijoje, Vengrijoje, Airijoje,

Ispanijoje ir Švedijoje. Matoma, kad 4 iš šių 5 valstybių yra priskiriamos pirmajam klasteriui t.y. Vakarų Europos šalims. Tuo tarpu remiantis bendrais duomenimis, tiesioginių užsienio investicijų mažėjimas yra prognozuojamas net 14 valstybių. Svarbu paminėti tai, kad šių modelių prognozės remiasi ir 2020 metų duomenimis. Atsižvelgiant į tai, kad 2020 metais įvyko pandemija, mokslinėje literatūroje dažnai galima rasti, kad prognozuojant ekonominius procesus įtraukus 2020 metus, prognozės gali būti ne visai tikslios. Remiantis Lietuvos banko duomenimis tiesioginės užsienio investicijos Lietuvoje 2021 metais padidėjo, todėl Lietuvos atveju modelis teisingai prognozuoja tiesioginių užsienio investicijų didėjimą.

7 lentelė. Tiesioginių užsienio investicijų įplaukų 2021 metų prognozė, remiantis geriausiais regresijos modeliais.

Šalis	Kodas	TUI 2020	Pilnų duomenų modeliai			Klasterių modeliai		
			TUI 2021	Pokytis	Tendencija	TUI 2021	Pokytis	Tendencija
Austrija	AUT	-18,21	-10,82	40,6%	Didėja	4,84	126,6%	Didėja
Belgija	BEL	-17,57	1,46	108,3%	Didėja	20,43	216,3%	Didėja
Bulgarija	BGR	2,70	-1,04	-138,6%	Mažėja	21,35	690,0%	Didėja
Kroatija	HRV	1,18	-1,03	-187,0%	Mažėja	18,74	1490,1%	Didėja
Čekija	CZE	6,43	13,99	117,6%	Didėja	26,48	311,9%	Didėja
Danija	DNK	1,58	10,14	540,7%	Didėja	20,20	1176,0%	Didėja
Estija	EST	3,54	2,33	-34,0%	Mažėja	27,42	675,5%	Didėja
Suomija	FIN	-2,37	5,84	346,5%	Didėja	5,71	340,9%	Didėja
Prancūzija	FRA	13,24	28,21	113,0%	Didėja	76,59	478,3%	Didėja
Vokietija	DEU	112,62	91,58	-18,7%	Mažėja	66,04	-41,4%	Mažėja
Graikija	GRC	3,30	-1,05	-131,8%	Mažėja	20,54	521,7%	Didėja
Vengrija	HUN	168,93	61,28	-63,7%	Mažėja	23,30	-86,2%	Mažėja
Islandija	ISL	-1,03	-6,66	-548,8%	Mažėja	24,94	2529,8%	Didėja
Airija	IRL	32,45	43,31	33,5%	Didėja	7,82	-75,9%	Mažėja
Italija	ITA	-22,09	-0,81	96,3%	Didėja	24,21	209,6%	Didėja
Latvija	LVA	0,94	-0,27	-129,1%	Mažėja	24,91	2538,0%	Didėja
Lietuva	LTU	4,48	16,14	260,5%	Didėja	26,28	486,9%	Didėja
Nyderlandai	NLD	-149,32	-12,12	91,9%	Didėja	15,31	110,3%	Didėja
Norvegija	NOR	-3,81	4,08	207,1%	Didėja	4,53	219,1%	Didėja
Lenkija	POL	17,39	9,54	-45,1%	Mažėja	24,24	39,4%	Didėja
Portugalija	PRT	4,71	2,61	-44,5%	Mažėja	24,60	422,4%	Didėja
Rumunija	ROU	3,60	-12,78	-454,8%	Mažėja	20,28	462,9%	Didėja
Slovakija	SVK	-0,33	-4,10	-1140,0%	Mažėja	21,46	6591,5%	Didėja
Slovėnija	SVN	0,49	8,60	1668,3%	Didėja	25,05	5051,1%	Didėja
Ispanija	ESP	33,74	27,45	-18,6%	Mažėja	26,66	-21,0%	Mažėja
Švedija	SWE	28,44	17,38	-38,9%	Mažėja	6,53	-77,0%	Mažėja
Šveicarija	CHE	-257,34	-75,27	70,8%	Didėja	2,42	100,9%	Didėja
Jungtinė Karalystė	GBR	31,06	36,34	17,0%	Didėja	44,39	42,9%	Didėja

Išvados

1. Investuotojai, rinkdamiesi kur eksportuoti kapitalą, vertina šalies investicinį patrauklumą, gebėjimą pritraukti tiesiogines užsienio investicijas. Apibendrinant Lietuvos ir užsienio autorių moksliniuose darbuose pateiktus tyrimus, investicinį patrauklumą ir tiesiogines užsienio investicijas lemiančius veiksnius būtų galima suskirstyti į šias grupes: rinkos, darbo rinkos ir inovacinis patrauklumas, infrastruktūros prieinamumas, socialinė ir kultūrinė aplinka, verslo ir mokestinė aplinka, tvarumo patrauklumas ir gamtos išteklių prieinamumas.
2. Duomenų tyrybos metodai padeda nustatyti nežinomus dėsningumus, suteikia daugiau žinių apie analizuojamus objektus, kas vėliau padeda priimant sprendimus. Šiame darbe naudoti duomenų tyrybos metodai – klasterizavimas, prognozavimas, vizualizavimas. Klasterinė analizė padeda išvelgti, kurių šalių tiesiogines užsienio investicijas lemiantys veiksniai yra panašūs, kaip šios šalys gali būti grupuojamos. Tyrime naudoti padalinimo, hierarchiniai, tankiu ir pasiskirstymu grįsti klasterizavimo metodai. Prognozavimo metodai padeda išvelgti ateities tendencijas. Iš skirtingų mašininio mokymosi regresijos metodų, remiantis vidutinių paklaidų metrikomis, išrenkami geriausiai modeliai, kurie vėliau naudojami tiesioginių užsienio investicijų prognozavimui.
3. Remiantis geriausiais klasterizavimo rezultatais, 2020 metais šalys buvo sugrupuotos į 3 klasterius. Pirmajam klasteriui priskiriamos daugiausia vakarų Europos šalys: Austrija, Belgija, Danija, Suomija, Prancūzija, Vokietija, Islandija, Nyderlandai, Norvegija, Švedija, Jungtinė Karalystė. Šios šalys pasižymi didžiausiu darbo jėgos kiekiu, santykinai žemu prekybos atvirumu ir fiskaline kontrole. Taip pat didžiausiais kapitalo mokesčiais ir produktyvumu. Antrajam klasteriui priskiriamos daugiausia rytų Europos šalys: Bulgarija, Kroatija, Čekija, Estija, Graikija, Vengrija, Latvija, Lietuva, Lenkija, Rumunija, Slovakija, Slovėnija, taip pat Italija, Ispanija ir Portugalija. Joms būdingas žemiausias bendras vidutinis vidaus produktas, tenkantis vienam gyventojui, mažesnė verslo, ekonominė laisvė, vyriausybės efektyvumas, politinis stabilumas ir korupcijos kontrolė, taip pat aukščiausias vartojimas, nedarbo lygis. Trečiajam klasteriui priskiriamos šalys: Airija, Šveicarija. Šios šalys pasižymi ekonomikos augimu, kai tuo tarpu kituose klasteriuose esančios šalys patiria ekonomikos nuosmukį. Taip pat aukštesniu bendru vidutiniu vidaus produktu, tenkančiu vienam gyventojui, prekybos atvirumu ir kapitalo investicijų skaičiumi, žemiausiu vartojimu ir neigiama infliacija. Klasterinė analizė skirtingais laiko momentais parodė nežymius klasterių pasikeitimus laikotarpiu po krizės bei covid-19 pandemijos metu. Kiekvienais metais 3-7 geriausi klasterizavimo modeliai, rodė tokius pat rezultatus.
4. Remiantis geriausiais modeliais kiekvienam klasteriui, tiesioginių užsienio investicijų mažėjimas prognozuojamas 5 valstybėse: Vokietijoje, Vengrijoje, Airijoje, Ispanijoje ir Švedijoje. 4 iš šių 5 valstybių yra priskiriamos pirmajam klasteriui t.y. Vakarų Europos šalims. Tuo tarpu remiantis modeliais, sudarytais su visu duomenų rinkiniu, mažėjimas matomas net 14 šalių. Reikšmingiausi veiksniai užsienio investicijų įplaukoms vakarų klasterio šalyse yra gyventojų tankis, darbo jėgos kiekis ir NAWRU (angl. *Non-accelerating wage rate of unemployment*), tuo tarpu rytų klasterio šalims – kapitalo mokesčiai ir darbo jėgos kiekis. Geriausius rezultatus pilnam duomenų rinkiniui parodė Huber regresorius, pirmojo klasterio duomenų rinkiniui - LARS regresija, antrojo - LASSO regresija.

Literatūros sąrašas

1. Dzwigol, H., Aleinikova, O., Umanska, Y., Shmygol, N., Pushak, Y. (2019). An entrepreneurship model for assessing the investment attractiveness of regions. *Journal of Entrepreneurship Education*, 22, 1-7.
2. Rodionov, D., Nasrutdinov, M. (2021). A Transformation of the Approach to Evaluating a Region's Investment Attractiveness as a Consequence of the COVID-19 Pandemic. *Economies*, 9(2), 59. <https://doi.org/10.3390/economies9020059>
3. Valinurova, L. S., Kazakova, O. B., Sulimova, E. A. (2015). Evaluation of investment attractiveness and prediction of investment volumes to the region. *Mediterranean Journal of Social Sciences*, 6(5 S3), 371.
4. Lietuvos Respublikos investicijų įstatymas. <https://e-seimas.lrs.lt/portal/legalAct/lt/TAD/TAIS.84573/asr>
5. Šečkutė, L., Tvaronavičius, V. (2007). Tiesioginių užsienio investicijų Baltijos šalyse tyrimas. Verslas: teorija ir praktika, 8(3), 153-160.
6. Danilevičienė, I., Lukšytė, V. (2017). The assessment of foreign direct investment influence on the country's economic competitiveness. *Mokslas–Lietuvos ateitis/Science–Future of Lithuania*, 9(2), 183-196.
7. Ruplienė, D., Garšvienė, L. (2008). Tiesioginių užsienio investicijų įtaka šalies ekonominiam augimui. *Ekonomika ir vadyba: aktualijos ir perspektyvos*, (3), 262-270.
8. Žitkus, Mickevičienė, M. (2013). Competitiveness as objective of regional development. *Viešojo Politika Ir Administravimas*, 12(3). <https://doi.org/10.5755/j01.ppa.12.3.4008>
9. Camagni, R. On the Concept of Territorial Competitiveness. Sound or Missleading? *Urban Studies*, 2002, Vol. 39, No. 13, p. 2395–2396. <http://dx.doi.org/10.1080/0042098022000027022>
10. Kruk, H. Ekoinowacje jako czynnik rozwoju regionalnego. Kn.: Wozniak, L. Strojny, J., Wojnicka, E. (sud.). Ekoinowacyjnosć dzis i jutro. Warszawa, PARP, 2010.
11. Fisher, P. (1999). Foreign direct investment for Russia: a strategy for the revival of industry. *Finance and statistics*, 2, 512-513.
12. Takhumova, O. V., Kasatkina, E. V., Maslikhova, E. A., Yumashev, A. V., Yumasheva, M. A. (2018). The main directions of increasing the investment attractiveness of the Russian regions in the conditions of institutional transformations. *Espacios*, 39(37), 6-6.
13. Lukauskas, M., Bruneckienė, J. (2018). Mašininio mokymosi panaudojimo galimybės regionų investicinio patrauklumo vertinime.
14. Ruplienė, D., Montvilaitė, K., Grigaliūnienė, Ž. (2008). Tiesiogines užsienio investicijas lemiantys veiksniai. *Ekonomika ir vadyba: aktualijos ir perspektyvos*, (3), 271-280.
15. Tocar. (2019). Culture-specific investment attractiveness index: an original composite indicator. *International Journal of Economics and Financial Issues*, 9(4), 203–211. <https://doi.org/10.32479/ijefi.8293>
16. Stankevičienė, J., Lakštutienė, A. (2012). Tiesioginių užsienio investicijų pritraukimą lemiančių veiksnių ir jų kitimo tendencijų tyrimas Baltijos šalyse. *Vadybos mokslas ir studijos-kaimo verslų ir jų infrastruktūros plėtrai*, 33, 69-79.
17. Mustafakulov, S. (2017). Investment Attractiveness of Regions: Methodic Aspects of the Definition and Classification of Impacting Factors. *European Scientific Journal*, 13(10), 433-449.

18. Goraieb, M. R., Reinert, M., Verdu, F. C. (2019). Cultural influences on foreign direct investment. *Revista Eletrônica de Negócios Internacionais: Internext*, 14(2), 128-144.
19. Hiestand, T. (2005). Using pooled model, random model and fixed model multiple regression to measure foreign direct investment in Taiwan. *International Business Economics Research Journal (IBER)*, 4(12).
20. Snieska, V., Zykiene, I., Burksaitiene, D. (2019). Evaluation of location's attractiveness for business growth in smart development. *Economic research-Ekonomska istraživanja*, 32(1), 925-946.
21. Globerman, S., Shapiro, D. (2002). Global foreign direct investment flows: The role of governance infrastructure. *World development*, 30(11), 1899-1919.
22. Kaufmann, D., Kraay, A. and Zoido-Lobaton, P. (1999a) Aggregating Governance Indicators, World Bank, Working paper no. 2195.
23. Snieska, V., Zykiene, I. (2015). City attractiveness for investment: characteristics and underlying factors. *Procedia-Social and Behavioral Sciences*, 213, 48-54.
24. Dorożyński, Kuna-Marszałek, A. (2016). Investments Attractiveness. The Case Of The Visegrad Group Countries. *Comparative Economic Research. Central and Eastern Europe*, 19(1), 119–140. <https://doi.org/10.1515/cer-2016-0007>
25. Zykienė, Ineta. Evaluation of location's attractiveness for business growth in the context of smart development. Doktoro disertacija. Kaunas: Kauno technologijos universitetas, 2018 m.
26. Lietuvos Respublikos aplinkos ministerija. <https://am.lrv.lt/lt/veiklos-sritys-1/es-ir-tarptautinis-bendradarbiavimas/darnus-vystymasis/darnus-vystymasis-ir-lietuva>
27. Ben Jelili, R. (2020) A Global Foreign Direct Investment Country Attractiveness Index.
28. The 2021 FDI Confidence Index® Executive summary. <https://www.kearney.com/foreign-direct-investment-confidence-index>
29. Attractiveness surveys. https://www.ey.com/en_gl/attractiveness
30. Valkauskas R. Statistiniai metodai. *Visuotinė Lietuvių enciklopedija*.
31. Kharlamova, G. (2014). Investment attractiveness of Ukrainian regions: rating assessment and marketing promotion. *Journal of International Studies*, 7(1).
32. Šečkutė, L., Tvaronavičius, V. (2007). Tiesioginių užsienio investicijų Baltijos šalyse tyrimas. Verslas: teorija ir praktika, 8(3), 153-160.
33. Dagilienė, L., Bruneckienė, J., Jucevičius, R., Lukauskas, M. (2020). Exploring smart economic development and competitiveness in Central and Eastern European countries. *Competitiveness Review: An International Business Journal*.
34. Niakšu O. (2014) Duomenų tyryba medicinoje: taikymas, problemos ir galimybės. *Literatūros apžvalgos. "Visuomenės sveikata"* 2014/4(67), 9-19.
35. Gan, G., Ma, C., Wu, J. (2020). *Data clustering: theory, algorithms, and applications*. Society for Industrial and Applied Mathematics.
36. Jain. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8), 651–666. <https://doi.org/10.1016/j.patrec.2009.09.011>
37. Stabingienė L. (2014) *Klasterizavimas* http://www.ilab.lt/stabingiene/sk2_1.html
38. Irani, J., Pise, N., Phatak, M. (2016). Clustering techniques and the similarity measures used in clustering: A survey. *International journal of computer applications*, 134(7), 9-14.

39. Deng, Hu, Y., Zhu, M., Huang, X., Du, B. (2014). A scalable and fast OPTICS for clustering trajectory big data. *Cluster Computing*, 18(2), 549–562. <https://doi.org/10.1007/s10586-014-0413-9>
40. Swarndeep Saket, J., Pandya, S. (2016). An overview of partitioning algorithms in clustering techniques. *International Journal of Advanced Research in Computer Engineering Technology (IJARCET)*, 5(6), 1943-1946.
41. Xu, D., Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2(2), 165-193.
42. Rhys, H. (2020). *Machine Learning with R, the tidyverse, and mlr*. Simon and Schuster.
43. Birant, D., Kut, A. (2007). ST-DBSCAN: An algorithm for clustering spatial–temporal data. *Data knowledge engineering*, 60(1), 208-221.
44. Zhang, M., Fu, Y., Bennett, K. M. and Wu, T. "Computational efficient Variational Bayesian Gaussian Mixture Models via Coreset," *2016 International Conference on Computer, Information and Telecommunication Systems (CITS)*, 2016, pp. 1-5, doi: 10.1109/CITS.2016.7546405.
45. Lovmar, L., Ahlford, A., Jonsson, M., Syvänen, A. C. (2005). Silhouette scores for assessment of SNP genotype clusters. *BMC genomics*, 6(1), 1-6.
46. Bhardwaj A. (2020). Silhouette Coefficient. Towards Data Science. <https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c>
47. Čekanavičius, V., Murauskas G. (2014). Taikomoji regresinė analizė socialiniuose tyrimuose. <http://www.statistika.mif.vu.lt/wp-content/uploads/2014/04/regresine-analize.pdf>
48. Gėgžna V. (2022) Biostatistinės analizės pagrindai. <https://mokymai.github.io/biostatistika/index.html>
49. Baltušnikaitė, I., Bratčikovienė, N. (2018). Regresinės analizės taikymas didiesiems duomenims. *Lithuanian Journal of Statistics*, 57(1), 56-69.
50. Scikit-learn: Machine Learning in Python, Pedregosa *et al.*, JMLR 12, pp. 2825-2830, 2011.

Priedai

1 priedas. Geriausi klasterizavimo rezultatai 2006-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 2}	Raw	0,7648	0,6637	0,6686
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7648	0,6637	0,6686
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7648	0,6637	0,6686
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7212	0,6579	0,6615
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7212	0,6579	0,6615
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7212	0,6579	0,6615
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7212	0,6579	0,6615
kmeans	{'k': 3}	Raw	0,5118	0,5954	0,5973
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5118	0,5954	0,5973
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5118	0,5954	0,5973
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4107	0,5734	0,5746
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4107	0,5734	0,5746
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4107	0,5734	0,5746
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4107	0,5734	0,5746
kmeans	{'k': 5}	Raw	0,3831	0,5393	0,5473
kmeans	{'k': 4}	Raw	0,3095	0,5332	0,5369
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,3095	0,5332	0,5369
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,3095	0,5332	0,5369
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,3095	0,5332	0,5369
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,3095	0,5332	0,5369
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,3088	0,5212	0,5262
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,3088	0,5212	0,5262
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,3088	0,5212	0,5262
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,3088	0,5212	0,5262
kmeans	{'k': 6}	Raw	0,3130	0,5202	0,5191
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2383	0,5014	0,4977
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2383	0,5014	0,4977
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2383	0,5014	0,4977
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2383	0,5014	0,4977
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,4843	0,4990	0,5023

2 priedas. Geriausi klasterizavimo rezultatai 2008-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 2}	Raw	0,6892	0,6532	0,6538
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,6892	0,6532	0,6538
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,6892	0,6532	0,6538
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,6892	0,6532	0,6538
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,6892	0,6532	0,6538
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,6892	0,6532	0,6538
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,6892	0,6532	0,6538
kmeans	{'k': 3}	Raw	0,5473	0,6293	0,6285
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5113	0,6255	0,6255
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5113	0,6255	0,6255
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5113	0,6255	0,6255
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5113	0,6255	0,6255
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,4701	0,5777	0,5861
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,4701	0,5777	0,5861
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5014	0,5527	0,5469
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5014	0,5527	0,5469
kmeans	{'k': 4}	Raw	0,4025	0,5520	0,5574
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4025	0,5520	0,5574
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4025	0,5520	0,5574
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4025	0,5520	0,5574
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4025	0,5520	0,5574
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2889	0,5512	0,5659
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2889	0,5512	0,5659
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2889	0,5512	0,5659
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2889	0,5512	0,5659
kmeans	{'k': 6}	Raw	0,2889	0,5512	0,5659
kmeans	{'k': 7}	Raw	0,0599	0,5334	0,5419
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,0599	0,5334	0,5419
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,0599	0,5334	0,5419
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,0599	0,5334	0,5419

3 priedas. Geriausi klasterizavimo rezultatai 2009-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 3}	Raw	0,5625	0,6589	0,6566
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5625	0,6589	0,6566
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5625	0,6589	0,6566
kmeans	{'k': 2}	Raw	0,6991	0,6517	0,6536
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,6991	0,6517	0,6536
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,6991	0,6517	0,6536
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,6452	0,6455	0,6471
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,6452	0,6455	0,6471
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,6452	0,6455	0,6471
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,6452	0,6455	0,6471
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4696	0,6422	0,6414
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4696	0,6422	0,6414
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4696	0,6422	0,6414
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4696	0,6422	0,6414
kmeans	{'k': 4}	Raw	0,4228	0,6036	0,6087
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4228	0,6036	0,6087
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4228	0,6036	0,6087
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4228	0,6036	0,6087
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4228	0,6036	0,6087
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,4012	0,5955	0,6032
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,4012	0,5955	0,6032
kmeans	{'k': 5}	Raw	0,1722	0,5743	0,5805
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,1722	0,5743	0,5805
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,1722	0,5743	0,5805
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,1722	0,5743	0,5805
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,1722	0,5743	0,5805
kmeans	{'k': 6}	Raw	0,1506	0,5662	0,5750
GMM	{'k': 6, 'init_params': 'kmeans'}	Raw	0,1506	0,5662	0,5750
BGMM	{'k': 6, 'init_params': 'kmeans'}	Raw	0,1506	0,5662	0,5750
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,1506	0,5662	0,5750

4 priedas. Geriausi klasterizavimo rezultatai 2012-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,6474	0,7179	0,7158
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,6474	0,7179	0,7158
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,6474	0,7179	0,7158
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,6474	0,7179	0,7158
kmeans	{'k': 3}	Raw	0,6474	0,7179	0,7158
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6474	0,7179	0,7158
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6474	0,7179	0,7158
kmeans	{'k': 2}	Raw	0,7397	0,6744	0,6816
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7397	0,6744	0,6816
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7397	0,6744	0,6816
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7397	0,6744	0,6816
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7397	0,6744	0,6816
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7397	0,6744	0,6816
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7397	0,6744	0,6816
kmeans	{'k': 4}	Raw	0,7132	0,6236	0,6284
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7132	0,6236	0,6284
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7132	0,6236	0,6284
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7132	0,6236	0,6284
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7132	0,6236	0,6284
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,6450	0,5979	0,6013
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,6450	0,5979	0,6013
DBSCAN	{'eps': 0.01, 'metric': 'cosine', 'samples': 7}	Raw	0,4695	0,5672	0,5281
DBSCAN	{'eps': 0.01, 'metric': 'cosine', 'samples': 8}	Raw	0,4695	0,5672	0,5281
DBSCAN	{'eps': 0.01, 'metric': 'cosine', 'samples': 9}	Raw	0,4695	0,5672	0,5281
kmeans	{'k': 7}	Raw	0,4022	0,5513	0,5494
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4022	0,5513	0,5494
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4022	0,5513	0,5494
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4022	0,5513	0,5494
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4022	0,5513	0,5494
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,5600	0,5510	0,5522

5 priedas. Geriausi klasterizavimo rezultatai 2013-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 3}	Raw	0,6366	0,7170	0,7140
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6366	0,7170	0,7140
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6366	0,7170	0,7140
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,6640	0,6934	0,6899
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,6640	0,6934	0,6899
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,6640	0,6934	0,6899
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,6640	0,6934	0,6899
kmeans	{'k': 2}	Raw	0,7507	0,6767	0,6831
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7095	0,6694	0,6736
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7095	0,6694	0,6736
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7747	0,6499	0,6565
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7747	0,6499	0,6565
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7747	0,6499	0,6565
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7747	0,6499	0,6565
kmeans	{'k': 4}	Raw	0,6927	0,6168	0,6216
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,2669	0,5767	0,5829
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,2669	0,5767	0,5829
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2669	0,5767	0,5829
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2669	0,5767	0,5829
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2669	0,5767	0,5829
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2669	0,5767	0,5829
kmeans	{'k': 5}	Raw	0,3735	0,5527	0,5633
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,3735	0,5527	0,5633
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,3735	0,5527	0,5633
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,3735	0,5527	0,5633
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,3735	0,5527	0,5633
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4619	0,5525	0,5484
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4619	0,5525	0,5484
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4619	0,5525	0,5484
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4619	0,5525	0,5484

6 priedas. Geriausi klasterizavimo rezultatai 2015-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 2}	Raw	0,7396	0,6663	0,6691
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7396	0,6663	0,6691
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7396	0,6663	0,6691
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7396	0,6663	0,6691
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7396	0,6663	0,6691
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7396	0,6663	0,6691
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7396	0,6663	0,6691
kmeans	{'k': 3}	Raw	0,5274	0,6642	0,6669
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5274	0,6642	0,6669
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5274	0,6642	0,6669
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5115	0,6597	0,6620
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5115	0,6597	0,6620
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5115	0,6597	0,6620
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5115	0,6597	0,6620
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4482	0,5532	0,5579
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4482	0,5532	0,5579
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4482	0,5532	0,5579
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4482	0,5532	0,5579
kmeans	{'k': 4}	Raw	0,4910	0,5323	0,5413
kmeans	{'k': 5}	Raw	0,2771	0,5303	0,5392
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,2771	0,5303	0,5392
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,2771	0,5303	0,5392
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,2804	0,5297	0,5389
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,2804	0,5297	0,5389
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2959	0,5144	0,5227
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2959	0,5144	0,5227
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2959	0,5144	0,5227
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2959	0,5144	0,5227
GMM	{'k': 6, 'init_params': 'kmeans'}	Raw	0,1777	0,5034	0,5079
BGMM	{'k': 6, 'init_params': 'kmeans'}	Raw	0,1777	0,5034	0,5079

7 priedas. Geriausi klasterizavimo rezultatai 2017-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 2}	Raw	0,7414	0,6455	0,6489
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7414	0,6455	0,6489
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7414	0,6455	0,6489
kmeans	{'k': 3}	Raw	0,5650	0,6336	0,6285
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5650	0,6336	0,6285
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5650	0,6336	0,6285
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5689	0,6187	0,6143
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5689	0,6187	0,6143
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5689	0,6187	0,6143
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5689	0,6187	0,6143
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7902	0,6110	0,6162
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7902	0,6110	0,6162
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7902	0,6110	0,6162
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7902	0,6110	0,6162
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,5375	0,5680	0,5662
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,5375	0,5680	0,5662
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5384	0,5582	0,5564
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5384	0,5582	0,5564
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5384	0,5582	0,5564
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5384	0,5582	0,5564
kmeans	{'k': 5}	Raw	0,4134	0,5393	0,5404
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,3420	0,5286	0,5303
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,3420	0,5286	0,5303
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,3350	0,5218	0,5243
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,3350	0,5218	0,5243
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,3350	0,5218	0,5243
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,3350	0,5218	0,5243
kmeans	{'k': 4}	Raw	0,5288	0,5113	0,5134
kmeans	{'k': 6}	Raw	0,1424	0,4875	0,4955
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,1424	0,4875	0,4955

8 priedas. Geriausi klasterizavimo rezultatai 2019-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 2}	Raw	0,7572	0,6371	0,6377
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7572	0,6371	0,6377
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7572	0,6371	0,6377
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7572	0,6371	0,6377
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7572	0,6371	0,6377
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7572	0,6371	0,6377
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7572	0,6371	0,6377
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,4944	0,6177	0,6214
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,4944	0,6177	0,6214
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4944	0,6177	0,6214
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4944	0,6177	0,6214
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4944	0,6177	0,6214
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4944	0,6177	0,6214
kmeans	{'k': 3}	Raw	0,5917	0,6070	0,5999
kmeans	{'k': 4}	Raw	0,5457	0,5416	0,5413
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,4596	0,5398	0,5514
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,4596	0,5398	0,5514
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2865	0,5204	0,5305
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2865	0,5204	0,5305
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2865	0,5204	0,5305
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2865	0,5204	0,5305
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,3531	0,5157	0,5267
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,3531	0,5157	0,5267
kmeans	{'k': 5}	Raw	0,4817	0,5039	0,5006
kmeans	{'k': 6}	Raw	0,3933	0,4976	0,4907
GMM	{'k': 6, 'init_params': 'kmeans'}	Raw	0,3933	0,4976	0,4907
BGMM	{'k': 6, 'init_params': 'kmeans'}	Raw	0,3933	0,4976	0,4907
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2564	0,4863	0,4814
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2564	0,4863	0,4814
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2564	0,4863	0,4814

9 priedas. Geriausi klasterizavimo rezultatai 2020-iems metams (pradinis duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5181	0,6379	0,6393
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5181	0,6379	0,6393
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5181	0,6379	0,6393
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5181	0,6379	0,6393
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5181	0,6379	0,6393
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5181	0,6379	0,6393
kmeans	{'k': 2}	Raw	0,7437	0,6355	0,6330
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7437	0,6355	0,6330
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7437	0,6355	0,6330
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7437	0,6355	0,6330
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7437	0,6355	0,6330
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7437	0,6355	0,6330
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7437	0,6355	0,6330
kmeans	{'k': 3}	Raw	0,4502	0,6185	0,6206
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,4945	0,5387	0,5491
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,4945	0,5387	0,5491
kmeans	{'k': 4}	Raw	0,4293	0,5334	0,5448
kmeans	{'k': 5}	Raw	0,4551	0,4957	0,4998
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,2487	0,4874	0,4975
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,2487	0,4874	0,4975
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,3014	0,4871	0,4894
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,3014	0,4871	0,4894
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,3014	0,4871	0,4894
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,3014	0,4871	0,4894
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,3890	0,4731	0,4770
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,3890	0,4731	0,4770
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,3890	0,4731	0,4770
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,3890	0,4731	0,4770
Agglomerative	{'k': 8, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,0327	0,4697	0,4694
Agglomerative	{'k': 8, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,0327	0,4697	0,4694

10 priedas. Geriausio 2006, 2008 ir 2009 metų klasterizavimo rezultatų klasterių centrai (v1 duomenų rinkinys)

	2006		2008		2009		
	0	1	0	1	0	1	2
Ekonomikos augimas: realaus BVP kitimo tempas, %	6,95	3,63	2,92	0,35	-6,72	-1,90	-4,69
BVP vienam gyventojui, \$	10785	45900	16746	57153	12428	76030	43681
Perkamosios galios paritetas (PPP)	23139	49750	26074	51651	22860	63047	45319
Kapitalo investicijos, % BVP	28,27	24,92	29,61	23,92	23,04	26,15	19,92
Namų ūkių vartojimas, % BVP	59,70	51,92	59,39	50,95	60,19	47,95	54,23
Infliacija, %	4,81	2,31	9,06	4,02	4,31	0,85	1,00
Darbo jėgos kiekis, mln. žmonių	9,52	12,30	10,35	11,82	9,89	3,54	13,38
Nedarbo lygis, %	9,93	6,07	8,45	5,26	10,96	3,61	8,55
Darbo jėgos aktyvumo lygis, %	56,47	61,76	57,29	62,47	57,34	67,34	60,44
Prekybos atvirumas: eksporto ir importo suma, % BVP	102,83	89,80	103,50	97,16	94,11	87,39	82,84
Eksportas, % BVP	48,14	46,37	47,97	50,81	45,65	48,43	42,78
Vyriausybės efektyvumo indeksas (-2,5 silpnas; 2,5 stiprus)	0,38	1,67	0,40	1,67	0,35	1,89	1,50
Korupcijos kontrolė (-2,5 silpna; 2,5 stipri)	0,17	1,82	0,18	1,82	0,11	2,03	1,59
Politinio stabilumo indeksas (-2,5 silpnas; 2,5 stiprus)	0,46	0,97	0,44	1,00	0,43	1,29	0,71
Interneto vartotojai, % gyventojų	37,53	69,33	47,53	77,34	51,57	86,69	74,40
Fiskalinės laisvės indeksas (0–100)	77,47	54,60	75,72	54,93	77,94	59,00	55,36
Verslo laisvės indeksas (0–100)	66,47	89,20	68,50	90,07	68,31	85,50	88,50
Darbo laisvės indeksas (0–100)	59,18	66,40	59,33	68,43	58,81	64,00	64,29
Monetarinės laisvės indeksas (0–100)	78,11	86,68	76,53	82,13	75,67	81,00	81,14
Prekybos laisvės indeksas (0–100)	76,94	81,87	81,22	85,29	83,00	87,00	85,07
Investavimo laisvės indeksas (0–100)	58,24	74,00	58,89	76,43	58,75	65,00	77,14
Finansinės laisvės indeksas (0–100)	64,12	71,33	60,00	75,71	58,12	70,00	75,71
Ekonominė laisvės indeksas (0–100)	61,65	72,27	62,44	73,64	62,19	74,50	72,07
Gyventojų skaičius, mln. žmonių	19,34	25,40	20,73	24,27	19,73	6,28	27,58
Miesto gyventojų, %	64,45	78,51	65,47	78,96	64,19	76,20	79,26
Gyventojų tankumas, žmonių km^2	79,88	150,47	80,28	156,71	79,06	104,50	155,57
65 metų ir vyresni gyventojai, %	15,46	15,91	15,74	16,14	15,59	15,69	16,64
Gyventojų skaičiaus augimas, %	-0,30	0,83	-0,14	0,83	-0,21	1,25	0,53
Gyvenimo trukmė gimus	73,14	79,71	74,00	80,23	73,67	81,29	80,31
Moterų verslo ir teisės indeksas (1–100)	81,10	90,50	84,34	92,37	85,78	91,25	94,06
Anglies dioksido padaryta žala, % BNP	2,30	37,94	1,58	34,91	1,93	0,24	35,03
Švietimo išlaidos, % BNP	4,32	5,17	4,30	5,48	4,54	5,48	5,62
Vartotojų kainų indeksas (2010m.=100)	81,40	91,78	93,59	97,43	96,96	98,48	98,09
Miško plotas (% žemės ploto)	37,36	30,00	37,55	29,70	38,18	32,14	29,94
Tiesioginės užsienio investicijos (grynosios įplaukos), mln.\$	8,50	68,36	17,47	60,44	4,98	28,15	26,00

11 priedas. Geriausio 20013, 2019 ir 2020 metų klasterizavimo rezultatų klasterių centrai (v1 duomenų rinkinys)

	2013			2019		2020		
	0	1	2	0	1	0	1	2
Ekonomikos augimas: realaus BVP kitimo tempas, %	0,74	1,42	0,71	3,08	1,91	-5	-4,82	1,74
BVP vienam gyventojui, \$	15324	95511	49276	17584	57770	16962	51025	86182
Perkamosios galios paritetas (PPP)	25395	64378	48169	30993	57645	29390	51007	79689
Kapitalo investicijos, % BVP	21,88	26,09	19,93	22,39	26,08	21,19	23,41	34,62
Namų ūkių vartojimas, % BVP	59,92	46,34	52,74	58,16	48,88	57,55	49,32	38,01
Infliacija, %	2,59	0,95	1,52	2,55	1,52	1,49	0,97	-0,5
Darbo jėgos kiekis, mln. žmonių	10,33	3,68	13,48	10,93	11,52	10,75	12,87	3,69
Nedarbo lygis, %	13,26	4,08	8,21	6,98	4,89	7,74	5,69	5,22
Darbo jėgos aktyvumo lygis, %	57,26	66,27	60,76	57,82	62,79	57,2	61,78	64,5
Prekybos atvirumas: eksporto ir importo suma, % BVP	113,05	99,22	101,71	114,86	111,51	106,85	90,28	177,92
Eksportas, % BVP	56,49	55,29	53,19	57,95	57,70	53,85	46,44	96,72
Vyriausybės efektyvumo indeksas (-2,5 silpnas; 2,5 stiprus)	0,46	1,85	1,59	0,54	1,62	0,47	1,6	1,75
Korupcijos kontrolė (-2,5 silpna; 2,5 stipri)	0,15	2,21	1,71	0,25	1,82	0,29	1,83	1,82
Politinio stabilumo indeksas (-2,5 silpnas; 2,5 stiprus)	0,38	1,37	0,96	0,41	0,90	0,36	0,84	1,08
Interneto vartotojai, % gyventojų	65,30	90,69	85,57	80,06	91,89	82,32	92,64	93,1
Fiskalinės laisvės indeksas (0–100)	79,44	59,50	55,83	76,42	57,92	76,42	55,36	73
Verslo laisvės indeksas (0–100)	72,89	84,50	88,08	70,79	84,15	69	83,55	78,5
Darbo laisvės indeksas (0–100)	59,72	66,50	62,83	60,26	62,92	60,37	61,09	74
Monetarinės laisvės indeksas (0–100)	75,58	81,55	79,57	79,22	81,54	78,26	80,2	84,85
Prekybos laisvės indeksas (0–100)	85,61	89,50	86,67	84,00	85,54	84,84	85,45	86,5
Investavimo laisvės indeksas (0–100)	64,17	75,00	82,92	68,95	85,00	68,68	83,64	87,5
Finansinės laisvės indeksas (0–100)	57,22	70,00	73,33	56,32	74,62	56,84	73,64	80
Ekonominė laisvės indeksas (0–100)	63,28	76,00	71,58	65,74	74,85	67,37	74,27	81,5
Gyventojų skaičius, mln. žmonių	20,68	6,58	27,75	22,66	22,75	22,62	25,75	6,81
Miesto gyventojų, %	66,23	76,98	80,49	67,56	81,93	67,8	84,53	68,78
Gyventojų tankumas, žmonių km^2	79,56	109,50	170,42	85,11	164,23	85,16	168,64	146
65 metų ir vyresni gyventojai, %	16,75	16,65	17,66	19,13	18,92	19,52	19,61	16,84
Gyventojų skaičiaus augimas, %	-0,29	1,18	0,60	-0,13	0,70	-0,12	0,53	0,97
Gyvenimo trukmė gimus	75,95	82,07	81,19	77,55	82,06	77,82	82,18	83,3
Moterų verslo ir teisės indeksas (1–100)	89,72	91,25	96,20	91,68	97,26	91,48	98,52	92,81
Anglies dioksido padaryta žala, % BNP	1,65	0,21	42,98	34,57	0,45	34,35	0,46	0,33
Švietimo išlaidos, % BNP	4,32	5,47	5,81	4,05	5,63	4,05	5,87	4,28
Vartotojimo kainų indeksas (2010m.=100)	119,29	101,74	107,18	147,49	114,02	150,74	117,17	102,53
Miško plotas (% žemės ploto)	37,97	32,36	29,72	37,94	30,26	37,96	31,85	21,73
Tiesioginės užsienio investicijos (grynosios įplaukos), mln.\$	8,77	-13,15	45,05	13,80	-6,03	12,64	-0,49	-112,44

12 priedas. Geriausi klasterizavimo rezultatai 2006-iems metams (papildytas duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 2}	Raw	0,7488	0,6474	0,6517
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7488	0,6474	0,6517
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7488	0,6474	0,6517
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7488	0,6474	0,6517
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7488	0,6474	0,6517
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7488	0,6474	0,6517
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7488	0,6474	0,6517
kmeans	{'k': 3}	Raw	0,5189	0,5868	0,5886
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5189	0,5868	0,5886
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5189	0,5868	0,5886
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4173	0,5745	0,5763
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4173	0,5745	0,5763
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4173	0,5745	0,5763
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4173	0,5745	0,5763
kmeans	{'k': 4}	Raw	0,3639	0,5494	0,5562
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,3639	0,5494	0,5562
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,3639	0,5494	0,5562
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2760	0,5295	0,5322
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2760	0,5295	0,5322
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2760	0,5295	0,5322
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2760	0,5295	0,5322
kmeans	{'k': 5}	Raw	0,2837	0,5275	0,5239
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,2837	0,5275	0,5239
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,2837	0,5275	0,5239
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,1954	0,5070	0,4996
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,1954	0,5070	0,4996
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,1954	0,5070	0,4996
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,1954	0,5070	0,4996
kmeans	{'k': 6}	Raw	0,0031	0,4769	0,4724
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,0031	0,4769	0,4724

13 priedas. Geriausi klasterizavimo rezultatai 2008-iems metams (papildytas duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 2}	Raw	0,7446	0,6447	0,6470
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7446	0,6447	0,6470
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7446	0,6447	0,6470
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7446	0,6447	0,6470
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7446	0,6447	0,6470
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7446	0,6447	0,6470
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7446	0,6447	0,6470
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5693	0,6307	0,6320
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5693	0,6307	0,6320
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5693	0,6307	0,6320
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5693	0,6307	0,6320
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5693	0,6307	0,6320
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5693	0,6307	0,6320
kmeans	{'k': 3}	Raw	0,6109	0,6271	0,6276
kmeans	{'k': 4}	Raw	0,4553	0,5742	0,5815
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4553	0,5742	0,5815
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4553	0,5742	0,5815
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4553	0,5742	0,5815
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4553	0,5742	0,5815
kmeans	{'k': 5}	Raw	0,2467	0,5434	0,5577
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2467	0,5434	0,5577
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2467	0,5434	0,5577
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2467	0,5434	0,5577
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2467	0,5434	0,5577
kmeans	{'k': 6}	Raw	-0,0149	0,5231	0,5304
GMM	{'k': 6, 'init_params': 'kmeans'}	Raw	-0,0149	0,5231	0,5304
BGMM	{'k': 6, 'init_params': 'kmeans'}	Raw	-0,0149	0,5231	0,5304
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	-0,0149	0,5231	0,5304
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	-0,0149	0,5231	0,5304
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	-0,0149	0,5231	0,5304

14 priedas. Geriausi klasterizavimo rezultatai 2009-iems metams (papildytas duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 3}	Raw	0,5685	0,6547	0,6542
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6156	0,6537	0,6518
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6156	0,6537	0,6518
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5165	0,6492	0,6494
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5165	0,6492	0,6494
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5165	0,6492	0,6494
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5165	0,6492	0,6494
kmeans	{'k': 2}	Raw	0,7000	0,6360	0,6385
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7000	0,6360	0,6385
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7000	0,6360	0,6385
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7000	0,6360	0,6385
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7000	0,6360	0,6385
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7000	0,6360	0,6385
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7000	0,6360	0,6385
kmeans	{'k': 4}	Raw	0,4627	0,6270	0,6326
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,4627	0,6270	0,6326
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,4627	0,6270	0,6326
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,4627	0,6270	0,6326
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,4627	0,6270	0,6326
kmeans	{'k': 5}	Raw	0,1763	0,5935	0,6002
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,1763	0,5935	0,6002
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,1763	0,5935	0,6002
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,1763	0,5935	0,6002
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,1763	0,5935	0,6002
kmeans	{'k': 6}	Raw	0,0431	0,5726	0,5774
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,0431	0,5726	0,5774
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,0431	0,5726	0,5774
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,0431	0,5726	0,5774
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,0431	0,5726	0,5774
kmeans	{'k': 7}	Raw	0,0112	0,5538	0,5491

15 priedas. Geriausi klasterizavimo rezultatai 2013-iems metams (papildytas duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
kmeans	{'k': 3}	Raw	0,6714	0,7249	0,7247
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6714	0,7249	0,7247
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6714	0,7249	0,7247
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,6957	0,6929	0,6918
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,6957	0,6929	0,6918
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,6957	0,6929	0,6918
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,6957	0,6929	0,6918
kmeans	{'k': 2}	Raw	0,7338	0,6596	0,6668
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7338	0,6596	0,6668
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7338	0,6596	0,6668
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,7883	0,6291	0,6386
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,7883	0,6291	0,6386
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,7883	0,6291	0,6386
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,7883	0,6291	0,6386
kmeans	{'k': 4}	Raw	0,2659	0,5907	0,6002
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,2659	0,5907	0,6002
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,2659	0,5907	0,6002
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2659	0,5907	0,6002
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2659	0,5907	0,6002
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2659	0,5907	0,6002
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2659	0,5907	0,6002
GMM	{'k': 6, 'init_params': 'kmeans'}	Raw	0,3248	0,5495	0,5414
BGMM	{'k': 6, 'init_params': 'kmeans'}	Raw	0,3248	0,5495	0,5414
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,1637	0,5440	0,5540
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,1637	0,5440	0,5540
kmeans	{'k': 6}	Raw	0,1942	0,5238	0,5179
kmeans	{'k': 5}	Raw	0,1226	0,4944	0,5078
kmeans	{'k': 7}	Raw	0,1520	0,4827	0,4763
GMM	{'k': 7, 'init_params': 'kmeans'}	Raw	0,1358	0,4748	0,4699
BGMM	{'k': 7, 'init_params': 'kmeans'}	Raw	0,1358	0,4748	0,4699

16 priedas. Geriausi klasterizavimo rezultatai 2019-iems metams (papildytas duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5341	0,6439	0,6504
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5341	0,6439	0,6504
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5341	0,6439	0,6504
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5341	0,6439	0,6504
kmeans	{'k': 3}	Raw	0,6260	0,6383	0,6327
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6260	0,6383	0,6327
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,6260	0,6383	0,6327
kmeans	{'k': 2}	Raw	0,8108	0,6345	0,6365
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,8108	0,6345	0,6365
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,8108	0,6345	0,6365
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,8108	0,6345	0,6365
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,8108	0,6345	0,6365
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7404	0,6184	0,6219
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7404	0,6184	0,6219
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,5499	0,5894	0,5810
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,5499	0,5894	0,5810
kmeans	{'k': 4}	Raw	0,4073	0,5368	0,5358
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,3036	0,5261	0,5386
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,3036	0,5261	0,5386
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,3036	0,5261	0,5386
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,3036	0,5261	0,5386
kmeans	{'k': 5}	Raw	0,3385	0,5046	0,5004
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,2692	0,4871	0,4825
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,2692	0,4871	0,4825
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,2692	0,4871	0,4825
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,2692	0,4871	0,4825
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	-0,0108	0,4739	0,4631
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	-0,0108	0,4739	0,4631
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	-0,0108	0,4739	0,4631
Agglomerative	{'k': 6, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	-0,0108	0,4739	0,4631

17 priedas. Geriausi klasterizavimo rezultatai 2020-iems metams (papildytas duomenų rinkinys)

Metodas	Parametrai	Skalė	Silueto koeficientas (Kosinuso)	Silueto koeficientas (Euklido)	Silueto koeficientas (Manhateno)
GMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5727	0,6566	0,6597
BGMM	{'k': 3, 'init_params': 'kmeans'}	Raw	0,5727	0,6566	0,6597
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5727	0,6566	0,6597
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5727	0,6566	0,6597
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5727	0,6566	0,6597
Agglomerative	{'k': 3, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5727	0,6566	0,6597
kmeans	{'k': 3}	Raw	0,4970	0,6422	0,6466
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,8158	0,6293	0,6269
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,8158	0,6293	0,6269
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,8158	0,6293	0,6269
Agglomerative	{'k': 2, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,8158	0,6293	0,6269
GMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7389	0,6214	0,6221
BGMM	{'k': 2, 'init_params': 'kmeans'}	Raw	0,7389	0,6214	0,6221
kmeans	{'k': 2}	Raw	0,6561	0,6193	0,6239
kmeans	{'k': 4}	Raw	0,5209	0,5877	0,5847
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,5209	0,5877	0,5847
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,5209	0,5877	0,5847
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,5209	0,5877	0,5847
Agglomerative	{'k': 4, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,5209	0,5877	0,5847
GMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,4561	0,5451	0,5464
BGMM	{'k': 4, 'init_params': 'kmeans'}	Raw	0,4561	0,5451	0,5464
GMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,1906	0,5100	0,5010
BGMM	{'k': 5, 'init_params': 'kmeans'}	Raw	0,1906	0,5100	0,5010
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,3169	0,4921	0,4925
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,3169	0,4921	0,4925
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,3169	0,4921	0,4925
Agglomerative	{'k': 5, 'affinity': 'euclidean', 'linkage': 'single'}	Raw	0,3169	0,4921	0,4925
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'ward'}	Raw	0,0104	0,4724	0,4703
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'complete'}	Raw	0,0104	0,4724	0,4703
Agglomerative	{'k': 7, 'affinity': 'euclidean', 'linkage': 'average'}	Raw	0,0104	0,4724	0,4703

18 priedas. Geriausio 2006, 2008 ir 2009 metų klasterizavimo rezultatų klasterių centrai (v2 duomenų rinkinys)

	2006		2008		2009		
	0	1	0	1	0	1	2
Ekonomikos augimas: realaus BVP kitimo tempas, %	3,60	6,53	0,35	2,10	-4,69	-7,03	-1,90
BVP vienam gyventojui, \$	47152	13949	57153	19620	43681	14817	76030
Perkamosios galios paritetas (PPP)	50528	26659	51651	28687	45319	25168	63047
Kapitalo investicijos, % BVP	24,52	29,41	23,92	29,61	19,92	22,50	26,15
Namų ūkių vartojimas, % BVP	51,50	59,48	50,95	59,02	54,23	59,19	47,95
Infliacija, %	2,22	4,01	4,02	7,19	1,00	2,44	0,85
Darbo jėgos kiekis, mln. žmonių	11,61	5,81	11,82	5,90	13,38	4,54	3,54
Nedarbo lygis, %	5,90	8,50	5,26	7,05	8,55	9,99	3,61
Darbo jėgos aktyvumo lygis, %	62,03	56,14	62,47	56,84	60,44	56,76	67,34
Prekybos atvirumas: eksporto ir importo suma, % BVP	92,20	103,10	97,16	105,30	82,84	97,83	87,39
Eksportas, % BVP	47,88	47,77	50,81	48,91	42,78	48,15	48,43
Vyriausybės efektyvumo indeksas (-2,5 silpnas; 2,5 stiprus)	1,73	0,67	1,67	0,67	1,50	0,67	1,89
Korupcijos kontrolė (-2,5 silpna; 2,5 stipri)	1,86	0,48	1,82	0,43	1,59	0,39	2,03
Politinio stabilumo indeksas (-2,5 silpnas; 2,5 stiprus)	1,05	0,66	1,00	0,60	0,71	0,66	1,29
Interneto vartotojai, % gyventojų	70,68	44,36	77,34	53,47	74,40	58,25	86,69
Fiskalinės laisvės indeksas (0–100)	54,57	73,71	54,93	74,00	55,36	76,92	59,00
Verslo laisvės indeksas (0–100)	89,93	71,29	90,07	72,21	88,50	73,00	85,50
Darbo laisvės indeksas (0-100)	67,57	58,36	68,43	58,86	64,29	58,17	64,00
Monetarinės laisvės indeksas (0-100)	86,88	80,76	82,13	77,96	81,14	77,07	81,00
Prekybos laisvės indeksas (0–100)	81,86	79,57	85,29	85,79	85,07	86,17	87,00
Investavimo laisvės indeksas (0-100)	74,29	65,71	76,43	66,43	77,14	67,50	65,00
Finansinės laisvės indeksas (0–100)	71,43	70,00	75,71	65,71	75,71	65,00	70,00
Ekonominė laisvės indeksas (0–100)	72,57	64,50	73,64	65,50	72,07	65,75	74,50
Gyventojų skaičius, mln. žmonių	24,05	12,26	24,27	12,33	27,58	9,61	6,28
Miesto gyventojų, %	78,58	64,39	78,96	64,69	79,26	62,80	76,20
Gyventojų tankumas, žmonių km^2	154,86	87,86	156,71	87,71	155,57	87,42	104,50
65 metų ir vyresni gyventojai, %	15,86	15,99	16,14	16,30	16,64	16,23	15,69
Gyventojų skaičiaus augimas, %	0,76	-0,14	0,83	-0,13	0,53	-0,24	1,25
Gyvenimo trukmė gimus	79,68	74,82	80,23	75,16	80,31	74,81	81,29
Moterų verslo ir teisės indeksas (1–100)	90,22	84,06	92,37	87,28	94,06	89,27	91,25
Anglies dioksido padaryta žala, % BNP	40,60	1,53	34,91	1,14	35,03	1,29	0,24
Švietimo išlaidos, % BNP	5,26	4,21	5,48	4,28	5,62	4,61	5,48
Vartotojų kainų indeksas (2010m.=100)	91,76	85,51	97,43	95,75	98,09	97,92	98,48
Miško plotas (% žemės ploto)	29,57	37,54	29,70	37,82	29,94	38,69	32,14
Tiesioginės užsienio investicijos (grynosios įplaukos), mln.\$	70,89	9,53	60,44	16,16	26,00	3,02	28,15
Grynojo kapitalo sandoriai, mlrd.Eur	2,78	2,04	2,81	2,13	3,01	2,16	3,07
Kapitalo mokesčiai, mlrd.Eur	3,96	2,52	7,75	2,23	6,21	1,30	2,31
Kapitalo dalis gamybos produktyvumui, indeksas	101,10	107,86	100,50	105,48	98,99	100,18	99,87
Darbo jėgos dalis gamybos produktyvumui, indeksas	97,43	94,05	97,63	96,53	96,75	94,01	98,14
Gamybos produktyvumo indeksas	98,57	101,52	98,23	101,91	95,95	94,19	98,01
Kapitalo – darbo jėgos pakeitimo indeksas	104,36	116,53	103,42	110,48	102,19	106,83	101,03
NAWRU	7,14	10,06	7,18	9,69	7,98	9,72	10,50
Atotrūkis tarp faktinio BVP ir tendencijų, %	1,74	4,00	1,73	4,25	-3,59	-3,95	0,48

19 priedas. Geriausio 20013, 2019 ir 2020 metų klasterizavimo rezultatų klasterių centrai (v2 duomenų rinkinys)

	2013			2019			2020		
	0	1	2	0	1	2	0	1	2
Ekonomikos augimas: realaus BVP kitimo tempas, %	0,71	0,54	1,42	3,20	1,79	2,29	-4,82	-5,46	1,74
BVP vienam gyventojui, \$	49276	17328	95511	20402	50896	80682	51025	19739	86182
Perkamosios galios paritetas (PPP)	48169	27489	64378	34187	52741	73993	51007	32297	79689
Kapitalo investicijos, % BVP	19,93	20,47	26,09	21,63	22,92	36,62	23,41	21,02	34,62
Namų ūkių vartojimas, % BVP	52,74	59,26	46,34	57,34	51,01	41,80	49,32	56,79	38,01
Infliacija, %	1,52	1,36	0,95	1,97	1,63	1,17	0,97	1,04	-0,5
Darbo jėgos kiekis, mln. žmonių	13,48	5,92	3,68	7,25	13,95	3,41	12,87	7,13	3,69
Nedarbo lygis, %	8,21	13,65	4,08	6,57	5,06	4,34	5,69	7,36	5,22
Darbo jėgos aktyvumo lygis, %	60,76	56,62	66,27	57,35	62,15	64,92	61,78	56,89	64,5
Prekybos atvirumas: eksporto ir importo suma, % BVP	101,71	119,28	99,22	118,20	100,69	147,57	90,28	110,34	177,92
Eksportas, % BVP	53,19	60,45	55,29	60,25	52,04	76,57	46,44	55,91	96,72
Vyriausybės efektyvumo indeksas (-2,5 silpnas; 2,5 stiprus)	1,59	0,74	1,85	0,70	1,60	1,70	1,6	0,65	1,75
Korupcijos kontrolė (-2,5 silpna; 2,5 stipri)	1,71	0,38	2,21	0,45	1,81	1,85	1,83	0,51	1,82
Politinio stabilumo indeksas (-2,5 silpnas; 2,5 stiprus)	0,96	0,62	1,37	0,63	0,82	1,15	0,84	0,63	1,08
Interneto vartotojai, % gyventojų	85,57	67,64	90,69	80,27	91,65	92,72	92,64	82,5	93,1
Fiskalinės laisvės indeksas (0–100)	55,83	77,50	59,50	73,33	54,90	68,00	55,36	73,4	73
Verslo laisvės indeksas (0–100)	88,08	74,29	84,50	69,73	84,70	82,33	83,55	67,53	78,5
Darbo laisvės indeksas (0-100)	62,83	58,36	66,50	59,87	61,60	67,33	61,09	60,33	74
Monetarinės laisvės indeksas (0-100)	79,57	78,43	81,55	82,38	81,24	82,53	80,2	80,55	84,85
Prekybos laisvės indeksas (0–100)	86,67	86,71	89,50	85,67	85,60	85,33	85,45	85,67	86,5
Investavimo laisvės indeksas (0-100)	82,92	73,57	75,00	76,67	85,50	83,33	83,64	76,33	87,5
Finansinės laisvės indeksas (0–100)	73,33	64,29	70,00	62,67	75,00	73,33	73,64	62,67	80
Ekonominė laisvės indeksas (0–100)	71,58	66,14	76,00	67,27	73,70	78,67	74,27	68,8	81,5
Gyventojų skaičius, mln. žmonių	27,75	12,27	6,58	15,35	27,69	6,29	25,75	15,35	6,81
Miesto gyventojų, %	80,49	65,43	76,98	66,83	84,52	73,29	84,53	67,05	68,78
Gyventojų tankumas, žmonių km^2	170,42	86,79	109,50	93,53	183,10	101,33	168,64	93,67	146
65 metų ir vyresni gyventojai, %	17,66	17,61	16,65	20,15	19,57	16,78	19,61	20,56	16,84
Gyventojų skaičiaus augimas, %	0,60	-0,38	1,18	-0,12	0,63	0,92	0,53	-0,06	0,97
Gyvenimo trukmė gimus	81,19	77,08	82,07	78,56	81,80	82,91	82,18	78,85	83,3
Moterų verslo ir teisės indeksas (1–100)	96,20	93,39	91,25	95,25	98,44	93,33	98,52	95	92,81
Anglies dioksido padaryta žala, % BNP	42,98	1,26	0,21	42,78	0,48	0,35	0,46	42,48	0,33
Švietimo išlaidos, % BNP	5,81	4,24	5,47	3,97	5,80	5,07	5,87	3,97	4,28
Vartojimo kainų indeksas (2010m.=100)	107,18	108,45	101,74	114,54	115,58	108,80	117,17	115,77	102,53
Miško plotas (% žemės ploto)	29,72	38,26	32,36	38,10	31,67	25,55	31,85	38,12	21,73
Tiesioginės užsienio investicijos (grynosios įplaukos), mln.\$	45,05	5,82	-13,15	14,84	-5,69	-7,18	-0,49	15,27	-112,44
Grynojo kapitalo sandoriai, mlrd.Eur	2,92	2,47	3,05	2,35	2,78	2,74	3	2,54	2,6
Kapitalo mokesčiai, mlrd.Eur	6,09	1,43	2,25	2,46	8,53	1,26	8,31	1,99	1,89
Kapitalo dalis gamybos produktyvumui, indeksas	98,50	99,08	99,79	102,66	100,76	99,61	97,85	99,22	99,71
Darbo jėgos dalis gamybos produktyvumui, indeksas	98,18	98,66	99,16	103,90	101,56	103,02	99,28	102,19	105,85
Gamybos produktyvumo indeksas	96,80	97,76	98,95	106,62	102,34	102,62	97,15	101,35	105,54
Kapitalo – darbo jėgos pakeitimo indeksas	99,99	100,20	100,40	99,75	99,51	96,89	97,8	97,39	94,31
NAWRU	7,12	10,57	12,18	7,22	5,46	7,51	5,74	7,07	9,85
Atotrūkis tarp faktinio BVP ir tendencijų, %	-2,20	-4,17	-5,56	2,29	1,22	0,49	-5,58	-5,19	-6,84

20 priedas. Sutelktinės MKM regresijos rezultatai, naudojant duomenų rinkinį su papildomais kintamaisiais (neišskiriant klasterių)

Sutelktinės MKM regresinės analizės įvertinimo santrauka

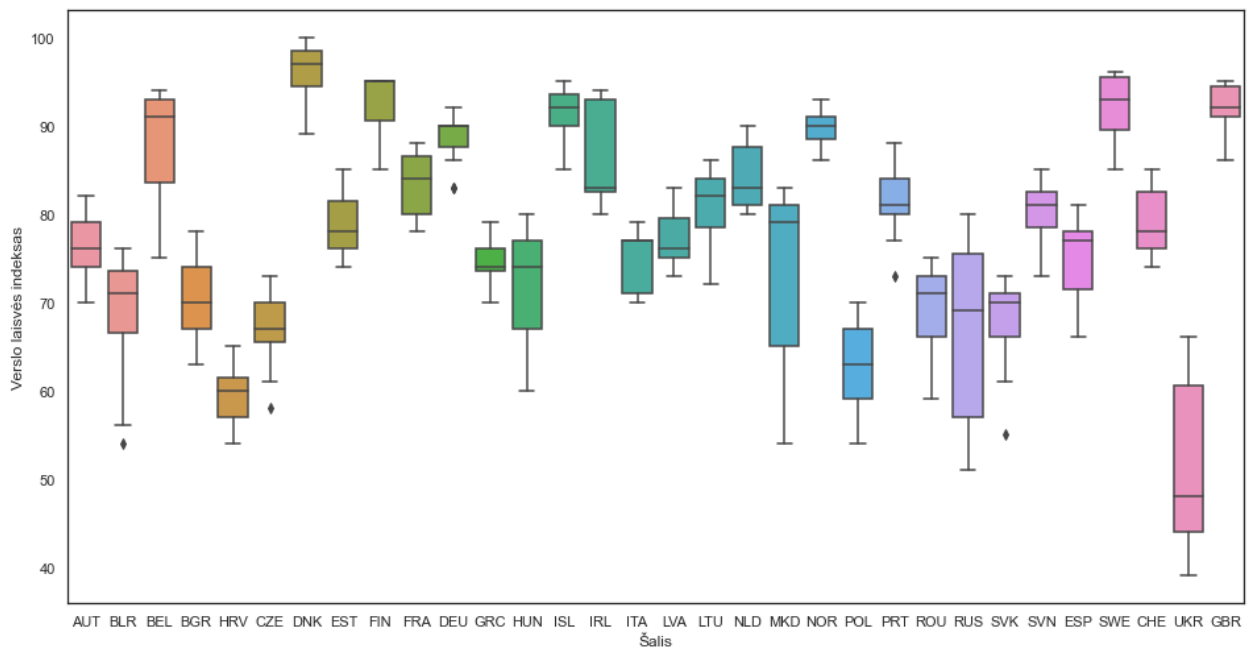
Priklausomas kintamasis:	TUI	Apibrėžtumo koeficientas:	0.3079
Įvertintojas: Sutelktinė MKM regresija		Apibrėžtumo koeficientas (tarp):	0.8848
Stebinių skaičius:	420	Apibrėžtumo koeficientas (viduje):	0.0754
Data:	Sun, May 2 2022	Apibrėžtumo koeficientas (bendras):	0.3079
Laikas:	18:44:38	Logaritminis tikėtinumumas	-2306.1
Kovariacijų įvertis:	Nekoreguotas		
		F-statistika:	3.9929
Subjektai:	28	P-reikšmė	0.0000
Vidutiniškai stebinių:	15.000	Pasiskirstymas:	F(42, 377)
Mažiausiai stebinių:	15.000		
Daugiausiai stebinių:	15.000	F-statistika (stabili, angl.robust):	3.9929
		P-reikšmė	0.0000
		Pasiskirstymas:	F(42, 377)
Laiko periodai:	15		
Vidutiniškai stebinių:	28.000		
Mažiausiai stebinių:	28.000		
Daugiausiai stebinių:	28.000		

Parametru įverčiai

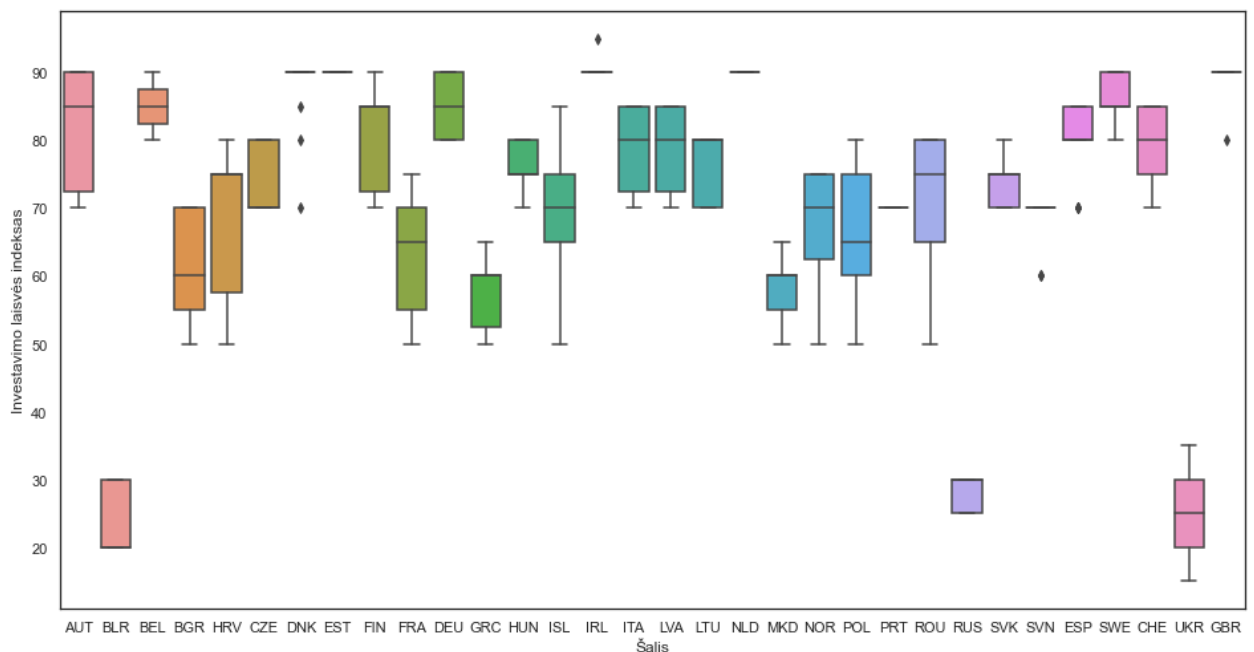
	Parametras	Stand.kl.	T-statistika	P-reikšmė	Žemutinis PI	Viršutinis PI
Pastovus įvertis	14.246	1574.7	0.0090	0.9928	-3082.0	3110.5
Ekonomikos augimas	0.4460	1.3704	0.3255	0.7450	-2.2486	3.1407
BVP vienam gyventojui	-0.0003	0.0008	-0.3518	0.7252	-0.0018	0.0012
Perkamosios galios paritetetas	0.0002	0.0015	0.1391	0.8895	-0.0027	0.0031
Kapitalo investicijos	-3.9531	2.8251	-1.3993	0.1626	-9.5080	1.6019
Namų ūkių vartojimas	-3.1187	2.5819	-1.2079	0.2278	-8.1956	1.9581
Infliacija	-0.8673	2.0520	-0.4226	0.6728	-4.9022	3.1676
Darbo jėgos kiekis	-0.9805	6.2732	-0.1563	0.8759	-13.315	11.354
Nedarbo lygis	-0.7321	2.3197	-0.3156	0.7525	-5.2932	3.8291
Darbo jėgos aktyvumas	0.2492	1.5098	0.1650	0.8690	-2.7194	3.2178
Prekybos atvirumas	1.8364	2.6416	0.6952	0.4874	-3.3577	7.0305
Eksportas	-4.0376	5.1506	-0.7839	0.4336	-14.165	6.0900
Valdžios efektyvumas	-8.6711	23.871	-0.3632	0.7166	-55.608	38.266
Korupcijos kontrolė	31.039	21.131	1.4689	0.1427	-10.509	72.588
Politinis stabilumas	-12.247	17.988	-0.6808	0.4964	-47.616	23.123
Interneto vartotojai	0.5750	0.6554	0.8773	0.3809	-0.7137	1.8637
Fiskalinė laisvė	0.9610	0.7216	1.3317	0.1838	-0.4579	2.3798
Verslo laisvė	0.7283	0.8515	0.8553	0.3929	-0.9461	2.4027
Darbo laisvė	0.0925	0.4148	0.2230	0.8237	-0.7231	0.9081
Monetarinė laisvė	0.9801	1.0250	0.9562	0.3396	-1.0353	2.9955
Prekybos laisvė	1.2926	1.7904	0.7220	0.4708	-2.2278	4.8131
Investavimo laisvė	1.6507	0.6529	2.5282	0.0119	0.3669	2.9344
Finansinė laisvė	-0.3472	0.5594	-0.6206	0.5352	-1.4471	0.7528
Ekonomikos laisvė	-2.7723	2.3329	-1.1884	0.2354	-7.3595	1.8148
Gyventojų skaičius	0.6702	3.1419	0.2133	0.8312	-5.5077	6.8481
Miesto gyventojai	0.0805	0.6322	0.1274	0.8987	-1.1626	1.3237
Gyventojų tankumas	0.2030	0.0728	2.7883	0.0056	0.0598	0.3462
65 metų ir vyr. gyventojai	-7.2968	3.2771	-2.2266	0.0266	-13.741	-0.8530
Gyventojų sk. augimas	-4.3604	9.4045	-0.4636	0.6432	-22.852	14.132
Gyvenimo trukmė	-3.2338	4.1252	-0.7839	0.4336	-11.345	4.8775
Moterų verslas ir teisė	0.5462	1.0828	0.5045	0.6142	-1.5828	2.6752
Anglies dioksido žala	-0.0389	0.0673	-0.5774	0.5640	-0.1712	0.0935
Švietimo išlaidos	-18.558	7.9970	-2.3206	0.0208	-34.282	-2.8334
Vartojimo kainų indeksas	-1.9890	0.7389	-2.6917	0.0074	-3.4420	-0.5361
Miško plotas	-0.2067	0.3615	-0.5720	0.5677	-0.9175	0.5040
Grynojo kapitalo sandoriai	15.310	11.604	1.3193	0.1879	-7.5078	38.127
Kapitalo mokesčiai	0.0954	0.1577	0.6048	0.5457	-0.2148	0.4056
Kapitalo dalis						
gamybos produktyvumui	-12.244	16.872	-0.7257	0.4685	-45.419	20.931
Darbo jėgos dalis						
gamybos produktyvumui	8.4596	15.916	0.5315	0.5954	-22.836	39.755
Productivity	1.1849	15.509	0.0764	0.9391	-29.310	31.680
Kapitalo - darbo jėgos						
pakeitimas	8.3130	4.9022	1.6958	0.0908	-1.3261	17.952
NAWRU	0.0870	2.4800	0.0351	0.9720	-4.7894	4.9634
Atotrūkis tarp faktinio BVP						
ir tendencijų	2.0481	1.8320	1.1180	0.2643	-1.5541	5.6503

21 priedas. Išorinių tyrime naudotų kintamųjų vizualinė apžvalga.

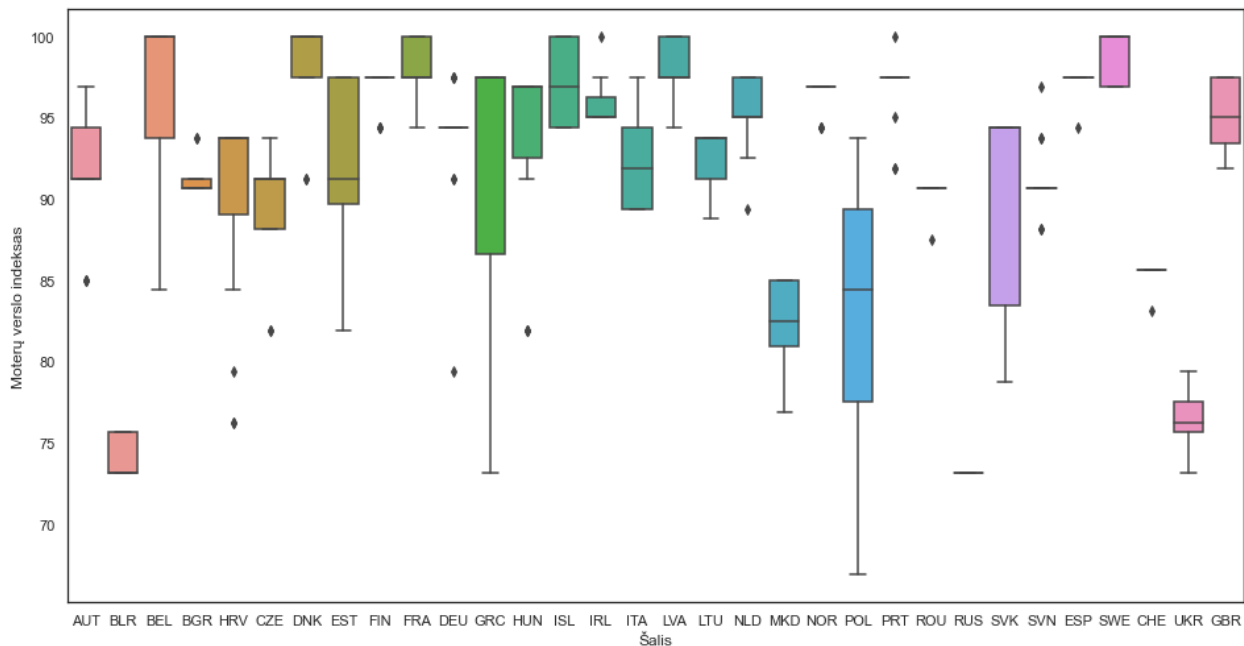
Verslo laisvės indekso apžvalga tiriamuoju laikotarpiu



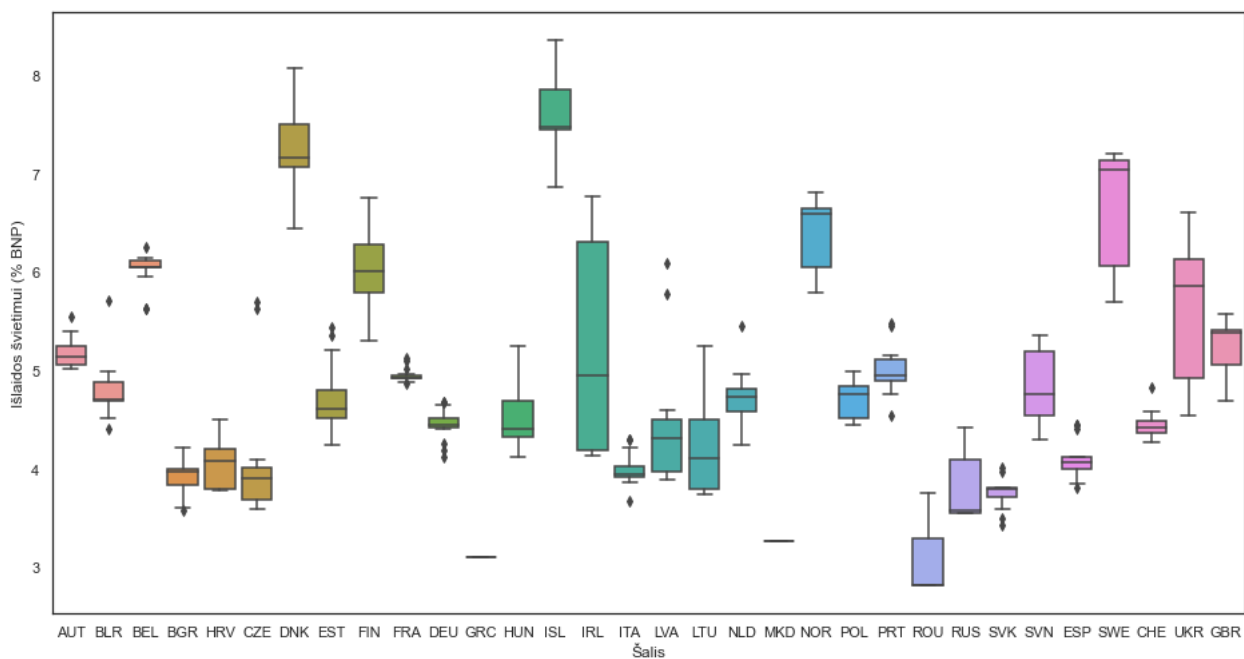
Investavimo laisvės indekso apžvalga tiriamuoju laikotarpiu



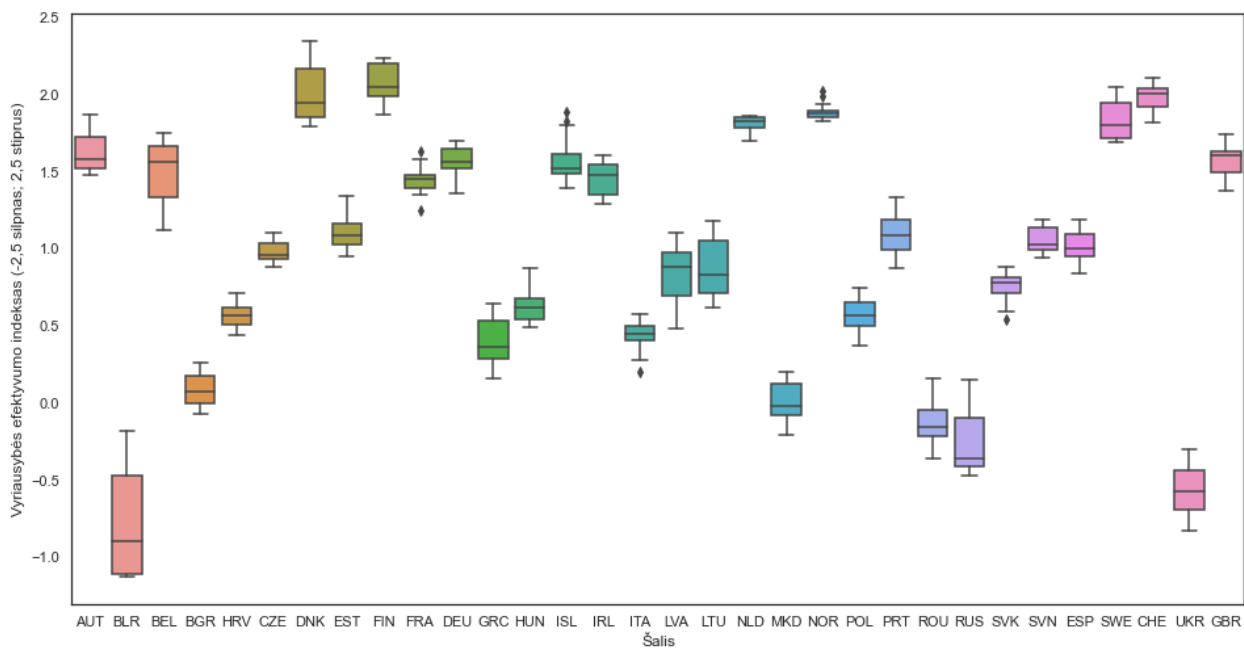
Moterų verslo indekso apžvalga tiriamuoju laikotarpiu



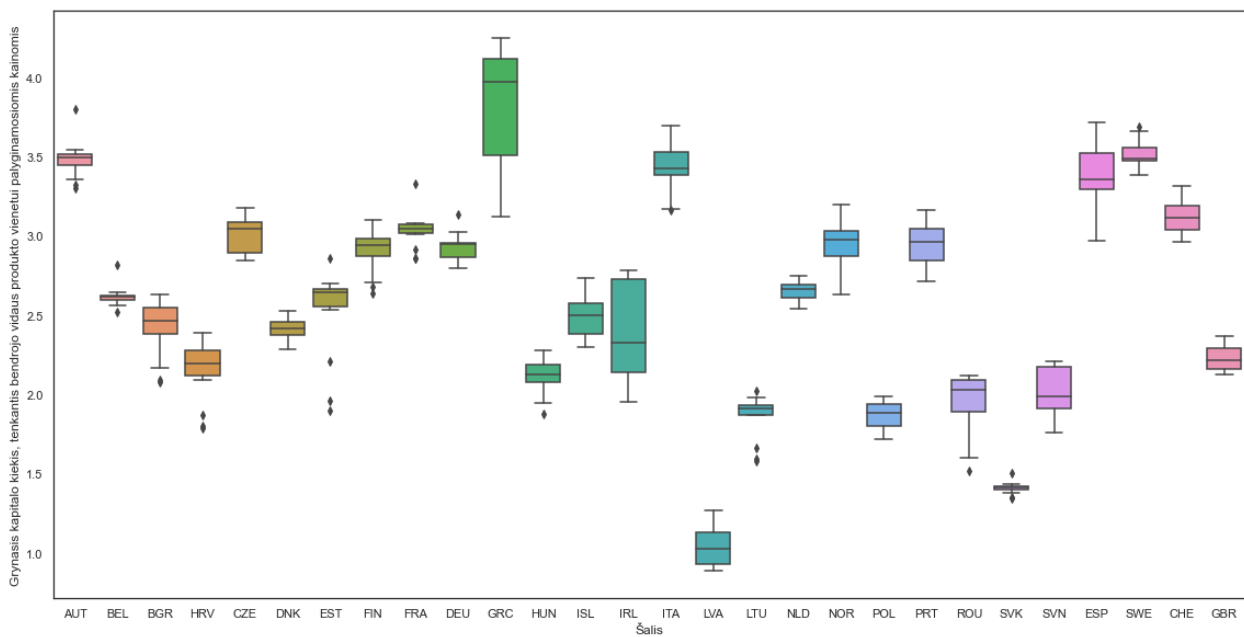
Išlaidų švietimui apžvalga tiriamuoju laikotarpiu



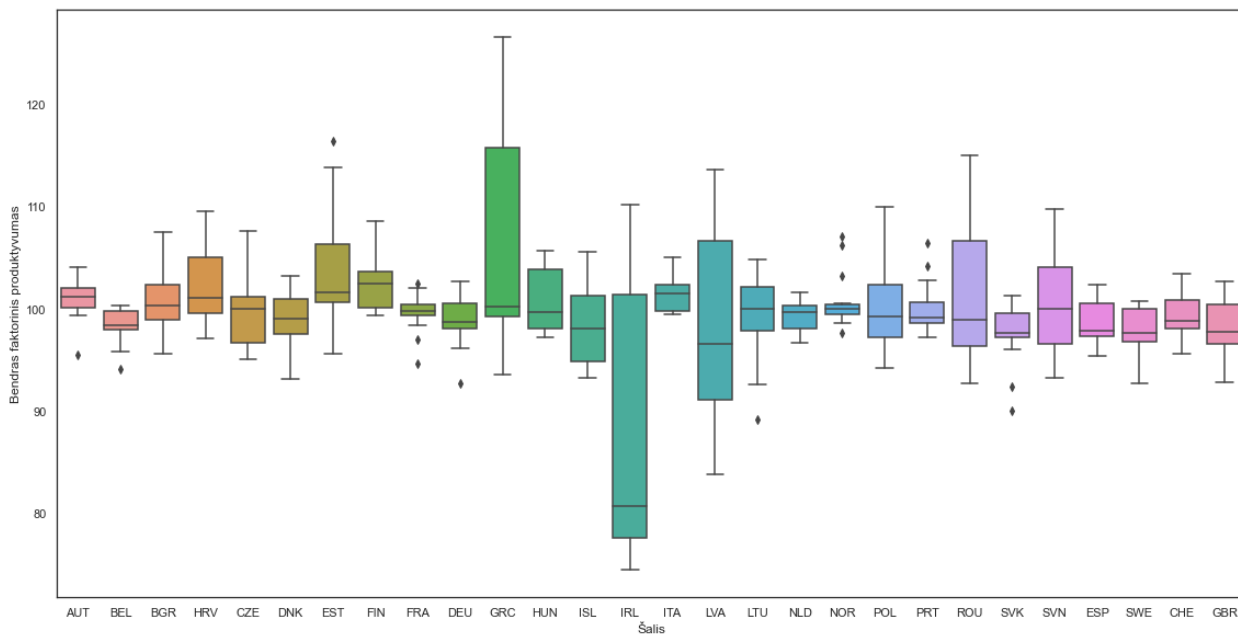
Vyriausybės efektyvumo apžvalga tiriamuoju laikotarpiu



Grynojo kapitalo kiekio apžvalga tiriamuoju laikotarpiu



Bendro faktorinio produktyvumo apžvalga tiriamuoju laikotarpiu



22 priedas. Bendro duomenų rinkinio geriausi regresijos modeliai

Modelis	Parametrai	Skalė	Savybės	MSE mokymo	MSE testavimo	MAE mokymo	MAE testavimo	RMSE mokymo	RMSE testavimo
Huber	{'epsilon': 2}	MinMaxScaler()	30	3190,71	3458,85	22,10	23,86	56,49	58,81
Huber	{'epsilon': 2}	RobustScaler()	20	3190,71	3458,85	22,10	23,86	56,49	58,81
Huber	{'epsilon': 2}	RobustScaler()	30	3189,52	3495,26	22,05	23,98	56,48	59,12
Huber	{'epsilon': 2}	Standard	30	3189,52	3495,26	22,05	23,98	56,48	59,12
Huber	{'epsilon': 2}	Standard	40	3187,86	3493,89	22,02	24,26	56,46	59,11
ARD	{None}	RobustScaler()	all	3139,90	3417,12	23,18	24,61	56,03	58,46
ARD	{None}	MinMaxScaler()	all	3140,04	3417,29	23,19	24,61	56,04	58,46
ARD	{None}	MinMaxScaler()	40	3139,21	3417,78	23,17	24,61	56,03	58,46
ARD	{None}	Standard	all	3139,21	3417,78	23,17	24,61	56,03	58,46
ARD	{None}	Standard	20	3139,58	3418,20	23,18	24,61	56,03	58,47
ARD	{None}	MinMaxScaler()	30	3138,62	3421,71	23,17	24,69	56,02	58,50
ARD	{None}	RobustScaler()	20	3138,62	3421,71	23,17	24,69	56,02	58,50
ARD	{None}	Raw	all	3143,04	3430,59	23,22	25,05	56,06	58,57
Huber	{'epsilon': 2}	MinMaxScaler()	all	3260,61	3646,23	21,59	25,07	57,10	60,38
MLP	{'hidden_size': 5}	RobustScaler()	all	3083,02	3603,75	22,32	25,51	55,52	60,03
LASSO	{'alpha': 0.2}	MinMaxScaler()	all	3084,61	3705,18	23,42	26,14	55,54	60,87
RandomForest	{'n_estimators': 10, 'depth': 2}	RobustScaler()	30	1976,01	3846,70	23,16	27,49	44,45	62,02
RandomForest	{'n_estimators': 10, 'depth': 2}	Raw	30	2083,34	4017,01	22,67	27,59	45,64	63,38
RandomForest	{'n_estimators': 10, 'depth': 2}	RobustScaler()	all	1976,08	3988,27	23,05	28,27	44,45	63,15
Bagging	{'n_estimators': 10}	MinMaxScaler()	all	4733,69	4892,46	26,78	28,67	68,80	69,95
RandomForest	{'n_estimators': 10, 'depth': 2}	Raw	all	1976,48	4050,89	23,53	29,03	44,46	63,65
RandomForest	{'n_estimators': 10, 'depth': 2}	Raw	20	1925,02	4148,94	22,70	29,51	43,87	64,41
RandomForest	{'n_estimators': 10, 'depth': 2}	MinMaxScaler()	40	1965,90	4049,93	23,01	29,62	44,34	63,64
RandomForest	{'n_estimators': 10, 'depth': 2}	MinMaxScaler()	30	1998,04	4188,11	23,11	29,95	44,70	64,72
RandomForest	{'n_estimators': 10, 'depth': 2}	Standard	40	2015,38	4087,55	23,08	29,96	44,89	63,93
Bagging	{'n_estimators': 10}	MinMaxScaler()	40	4987,42	5028,11	29,88	30,37	70,62	70,91
Bagging	{'n_estimators': 10}	Standard	all	4987,42	5028,11	29,88	30,37	70,62	70,91
Bagging	{'n_estimators': 10}	Standard	40	4987,42	5028,11	29,88	30,37	70,62	70,91
Bagging	{'n_estimators': 10}	Raw	all	4987,43	5028,11	29,88	30,37	70,62	70,91
Bagging	{'n_estimators': 10}	MinMaxScaler()	30	4987,43	5028,11	29,88	30,37	70,62	70,91

23 priedas. Pirmojo klasterio geriausi regresijos modeliai

Modelis	Parametrai	Skalė	Savybės	MSE mokymo	MSE testavimo	MAE mokymo	MAE testavimo	RMSE mokymo	RMSE testavimo
LARS	{'Default'}	RobustScaler()	all	6220,11	6566,34	40,87	41,75	78,87	81,03
MLP	{'hidden_size': 5}	RobustScaler()	40	7306,11	7298,82	42,16	45,50	85,48	85,43
MLP	{'hidden_size': 5}	RobustScaler()	all	6694,93	8182,27	40,05	47,00	81,82	90,46
ARD	{None}	RobustScaler()	40	5744,00	7136,51	38,81	47,22	75,79	84,48
ARD	{None}	Raw	all	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	Raw	40	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	Raw	30	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	Raw	20	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	MinMaxScaler()	40	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	MinMaxScaler()	30	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	MinMaxScaler()	20	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	RobustScaler()	30	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	RobustScaler()	20	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	Standard	all	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	Standard	40	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	Standard	30	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	Standard	20	5740,18	7174,06	38,89	47,64	75,76	84,70
ARD	{None}	RobustScaler()	all	5741,93	7177,92	38,90	47,68	75,78	84,72
ARD	{None}	MinMaxScaler()	all	5742,06	7178,03	38,90	47,68	75,78	84,72
Elastic	{'alpha': 0.2, 'l1': 0}	RobustScaler()	all	5507,89	8447,50	38,18	48,35	74,22	91,91
Bagging	{'n_estimators': 10}	MinMaxScaler()	all	8940,11	9253,59	47,14	50,07	94,55	96,20
Bagging	{'n_estimators': 10}	RobustScaler()	40	8685,62	9098,63	47,65	50,18	93,20	95,39
Bagging	{'n_estimators': 10}	Raw	all	8685,15	9100,10	47,64	50,18	93,19	95,39
Bagging	{'n_estimators': 10}	Raw	40	8685,15	9100,10	47,64	50,18	93,19	95,39
Bagging	{'n_estimators': 10}	Raw	30	8685,15	9100,10	47,64	50,18	93,19	95,39
Bagging	{'n_estimators': 10}	Raw	20	8685,15	9100,10	47,64	50,18	93,19	95,39
Bagging	{'n_estimators': 10}	MinMaxScaler()	40	8685,15	9100,10	47,64	50,18	93,19	95,39
Bagging	{'n_estimators': 10}	MinMaxScaler()	30	8685,15	9100,10	47,64	50,18	93,19	95,39
Bagging	{'n_estimators': 10}	MinMaxScaler()	20	8685,15	9100,10	47,64	50,18	93,19	95,39
Bagging	{'n_estimators': 10}	RobustScaler()	30	8685,15	9100,10	47,64	50,18	93,19	95,39

24 priedas. Antrojo klasterio geriausi regresijos modeliai

Modelis	Parametrai	Skalė	Savybės	MSE mokymo	MSE testavimo	MAE mokymo	MAE testavimo	RMSE mokymo	RMSE testavimo
LASSO	{'alpha': 0.2}	MinMaxScaler()	all	238,91	275,03	5,99	6,62	15,46	16,58
ARD	{None}	MinMaxScaler()	all	234,74	271,19	6,10	6,83	15,32	16,47
ARD	{None}	RobustScaler()	all	234,73	271,21	6,10	6,83	15,32	16,47
ARD	{None}	Standard	30	234,41	271,99	6,09	6,95	15,31	16,49
ARD	{None}	Raw	20	234,40	272,00	6,09	6,95	15,31	16,49
ARD	{None}	MinMaxScaler()	30	234,40	272,00	6,09	6,95	15,31	16,49
ARD	{None}	RobustScaler()	20	234,40	272,00	6,09	6,95	15,31	16,49
ARD	{None}	Standard	all	234,40	272,00	6,09	6,95	15,31	16,49
ARD	{None}	Standard	40	234,40	272,00	6,09	6,95	15,31	16,49
ARD	{None}	Standard	20	234,40	272,00	6,09	6,95	15,31	16,49
Elastic	{'alpha': 0.2, 'l1': 0}	RobustScaler()	all	227,67	275,12	6,00	7,17	15,09	16,59
LASSO	{'alpha': 0.2}	RobustScaler()	all	226,38	287,00	6,09	7,87	15,05	16,94
Bagging	{'n_estimators': 10}	RobustScaler()	all	319,39	373,30	6,20	7,89	17,87	19,32
Huber	{'epsilon': 2}	Raw	20	294,80	341,17	5,39	7,94	17,17	18,47
Huber	{'epsilon': 2}	MinMaxScaler()	30	294,80	341,17	5,39	7,94	17,17	18,47
Huber	{'epsilon': 2}	RobustScaler()	20	294,80	341,17	5,39	7,94	17,17	18,47
Huber	{'epsilon': 2}	Standard	all	294,80	341,17	5,39	7,94	17,17	18,47
Huber	{'epsilon': 2}	Standard	40	294,80	341,17	5,39	7,94	17,17	18,47
Huber	{'epsilon': 2}	Standard	20	294,80	341,17	5,39	7,94	17,17	18,47
MLP	{'hidden_size': 5}	MinMaxScaler()	all	355,89	362,71	7,70	7,95	18,87	19,04
MLP	{'hidden_size': 5}	RobustScaler()	all	281,42	307,51	6,70	7,97	16,78	17,54
Bagging	{'n_estimators': 10}	MinMaxScaler()	all	335,73	384,12	6,18	8,16	18,32	19,60
LARS	{'Default'}	RobustScaler()	all	262,81	291,46	7,48	8,19	16,21	17,07
Ada	{'n_estimators': 10}	RobustScaler()	all	31,65	342,27	3,98	8,55	5,63	18,50
Ada	{'n_estimators': 10}	Raw	40	31,81	343,96	4,04	8,62	5,64	18,55
Ada	{'n_estimators': 10}	Raw	30	31,81	343,96	4,04	8,62	5,64	18,55
Ada	{'n_estimators': 10}	MinMaxScaler()	20	31,81	343,96	4,04	8,62	5,64	18,55
Ada	{'n_estimators': 10}	RobustScaler()	40	31,81	343,96	4,04	8,62	5,64	18,55
Ada	{'n_estimators': 10}	RobustScaler()	30	31,81	343,96	4,04	8,62	5,64	18,55
Ada	{'n_estimators': 10}	Standard	30	31,81	343,96	4,04	8,62	5,64	18,55

25 priedas. Duomenų klasterizavimo programos kodas

```
results = pd.DataFrame(columns = ['Period', 'Method', 'Params', 'Scaler', 'Labels', 'Centers',
                                'SilhouetteCosine', 'SilhouetteEuclidean', 'SilhouetteManhattan',
                                'Calinski', 'Davies'])

scalers = ["Raw", "MinMax", "Robust", "Standard"]

years = list(df["Year"].unique())

for year in years:
    df_year = df[df["Year"]==year]
    period = year

    results.to_csv("results.csv")
    for scaler in scalers:

        X = df_year[df_year.columns[3:]]
        if scaler == "MinMax":
            scaler = MinMaxScaler()
            X = scaler.fit_transform(X)
        elif scaler == "Robust":
            scaler = RobustScaler()
            X = scaler.fit_transform(X)
        elif scaler == "Standard":
            scaler = StandardScaler()
            X = scaler.fit_transform(X)
        elif scaler == "Raw":
            X = X

        for k in range(2, df_year.shape[0]):
            method = "kmeans"
            params = {"k":k}
            kmeans = KMeans(n_clusters=params["k"], random_state=0).fit(X)
            labels = kmeans.labels_
            centers = kmeans.cluster_centers_

            results = save_results(X, labels, results)

        for k in range(2, df_year.shape[0]):
            for init in ["kmeans"]:
                method = "GMM"
                params = {"k":k, "init_params": init}
                clust = GaussianMixture(n_components=params["k"], init_params= params["init_params"],
random_state=0).fit(X)
                labels = clust.predict(X)
                if scaler != "Raw":
                    centers = scaler.inverse_transform(clust.means_)
                else:
                    centers = clust.means_

                results = save_results(X, labels, results)

        for k in range(2, df_year.shape[0]):
            for init in ["kmeans"]:
                method = "BGMM"
                params = {"k":k, "init_params": init}
                clust = BayesianGaussianMixture(n_components=params["k"], init_params=
params["init_params"], random_state=0).fit(X)
                labels = clust.predict(X)
                if scaler != "Raw":
                    centers = scaler.inverse_transform(clust.means_)
                else:
                    centers = clust.means_

                results = save_results(X, labels, results)
```

```

for k in range(2, df_year.shape[0]):
    for affinity in ["euclidean", "l1", "l2", "manhattan", "cosine"]:
        for linkage in ["ward", "complete", "average", "single"]:
            start = time.time()
            try:
                method = "AgglomerativeClustering"
                params = {"k":k, "affinity": affinity, "linkage":linkage}
                clust = AgglomerativeClustering(n_clusters = k, affinity=affinity).fit(X)
                labels = clust.labels_

                clf = NearestCentroid()
                clf.fit(X, labels)

                if scaler != "Raw":
                    centers = scaler.inverse_transform(clf.centroids_)
                else:
                    centers = clf.centroids_

                results = save_results(X, labels, results)
            except Exception as e:
                print(e)
            print(f"Total time to compute clusters: {time.time()-start}")

for eps in [0.01, 0.02, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]:
    for metric in ["euclidean", "l1", "l2", "manhattan", "cosine"]:
        for samples in [2, 3, 4, 5, 6, 7,8, 9, 10]:
            start = time.time()
            try:
                method = "DBSCAN"
                params = {"eps":eps, "metric": metric, "samples":samples}
                clust = DBSCAN(eps=eps, min_samples=samples, metric=metric, n_jobs=-1).fit(X)
                labels = clust.labels_

                if scaler != "Raw":
                    centers = scaler.inverse_transform(clust.components_)
                else:
                    centers = clf.centroids_

                results = save_results(X, labels, results)
            except Exception as e:
                print(e)
            print(f"Total time to compute clusters: {time.time()-start}")

```

26 priedas. Duomenų regresinės analizės programos kodas

```
from sklearn.metrics import r2_score
from sklearn.metrics import mean_absolute_percentage_error
from sklearn.metrics import mean_squared_error
from sklearn.linear_model import ElasticNet
import numpy as np
from sklearn.model_selection import KFold
from sklearn.kernel_ridge import KernelRidge
import numpy as np
from sklearn.gaussian_process import GaussianProcessRegressor
from sklearn.svm import SVR
from sklearn.ensemble import BaggingRegressor
from sklearn.datasets import make_regression
from sklearn.ensemble import AdaBoostRegressor
from sklearn.datasets import make_regression
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.preprocessing import MinMaxScaler
from sklearn.preprocessing import RobustScaler
from sklearn.preprocessing import StandardScaler
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import explained_variance_score

results = pd.DataFrame(columns = ["Data", 'Model', 'Params', 'Scaler', "Features", "FS",
                                "R2_train", "R2_test",
                                "MAPE_train", "MAPE_test",
                                "MSE_train", "MSE_test",
                                "MAE_train", "MAE_test",
                                "EVS_train", "EVS_test"])

n_folds = 5

kf = KFold(n_splits=n_folds, random_state=None)
models = ["GB", "RandomForest", "Elastic", "LARS", "LASSO", "Huber", "ARD", "KernelRidge", "Gaussian", "Bagging", "Ada", "LR", "MLP"]

for data in ["full", "first", "second"]:
    for scaler in ["Raw", "MinMax", "Robust", "Standard"]:
        for features in ["all", 40, 30, 20]:

            if data=="full":
                X = np.array(X_full)
                Y = np.array(Y_full)
            if data=="first":
                X = np.array(X_1clust)
                Y = np.array(Y_1clust)
            if data=="second":
                X = np.array(X_2clust)
                Y = np.array(Y_2clust)

            if scaler == "Raw":
                X = X
                Y = Y
            elif scaler == "MinMax":
                scaler = MinMaxScaler()
                X = scaler.fit_transform(X)
                Y = Y
            elif scaler == "Robust":
                scaler = RobustScaler()
                X = scaler.fit_transform(X)
                Y = Y
            elif scaler == "Standard":
                scaler = StandardScaler()
                X = scaler.fit_transform(X)
                Y = Y

            X, fs = select_features(X, Y, k)
```

```

for model in models:
    print(model)
    r2_train = 0
    r2_test = 0
    MAPE_train = 0
    MAPE_test = 0
    MSE_train = 0
    MSE_test = 0
    MAE_train = 0
    MAE_test = 0
    EVS_train = 0
    EVS_test = 0

if model == "RandomForest":
    for n_estimator in [10,20,50,100,200,500]:
        for depth in [2,3,5,10,None]:
            try:
                params = {"n_estimators": n_estimator, "depth":depth}
                for train_index, test_index in kf.split(X):
                    X_train, X_test = X[train_index], X[test_index]
                    y_train, y_test = Y[train_index], Y[test_index]

                    regr = RandomForestRegressor(n_estimators=n_estimator, max_depth=depth).fit(X_train, y_train)

                    prediction_train = regr.predict(X_train)
                    prediction_test = regr.predict(X_test)

                    r2_train += r2_score(prediction_train, y_train)
                    r2_test += r2_score(prediction_test, y_test)
                    MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
                    MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
                    MSE_train += mean_squared_error(prediction_train, y_train)
                    MSE_test += mean_squared_error(prediction_test, y_test)
                    MAE_train += mean_absolute_error(prediction_train, y_train)
                    MAE_test += mean_absolute_error(prediction_test, y_test)
                    EVS_train += explained_variance_score(prediction_train, y_train)
                    EVS_test += explained_variance_score(prediction_test, y_test)

                results = results.append({'Data': data,
                                        'Model': model,
                                        'Params': params,
                                        'Scaler': scaler,
                                        "Features": features,
                                        "FS": fs,
                                        "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                                        "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                                        "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                                        "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                                        "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                                       ignore_index = True)
            except Exception as e:
                print(e)

if model == "Elastic":
    for alpha in [0.2, 0.5, 1, 1.5, 2]:
        for l1 in [0, 0.1, 0.25, 0.5, 1]:
            try:
                params = {"alpha": alpha, "l1":l1}
                for train_index, test_index in kf.split(X):
                    X_train, X_test = X[train_index], X[test_index]
                    y_train, y_test = Y[train_index], Y[test_index]

                    regr = ElasticNet(random_state=0, alpha = alpha, l1_ratio = l1, max_iter=5000).fit(X_train, y_train)
                    prediction_train = regr.predict(X_train)
                    prediction_test = regr.predict(X_test)

```



```

r2_train += r2_score(prediction_train, y_train)
r2_test += r2_score(prediction_test, y_test)
MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
MSE_train += mean_squared_error(prediction_train, y_train)
MSE_test += mean_squared_error(prediction_test, y_test)
MAE_train += mean_absolute_error(prediction_train, y_train)
MAE_test += mean_absolute_error(prediction_test, y_test)
EVS_train += explained_variance_score(prediction_train, y_train)
EVS_test += explained_variance_score(prediction_test, y_test)

results = results.append({'Data': data,
                        'Model': model,
                        'Params': params,
                        'Scaler': scaler,
                        "Features": features,
                        "FS": fs,
                        "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                        "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                        "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                        "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                        "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                        ignore_index = True)
except Exception as e:
    print(e)

if model == "LARS":
    try:
        params = {"Default"}
        for train_index, test_index in kf.split(X):
            X_train, X_test = X[train_index], X[test_index]
            y_train, y_test = Y[train_index], Y[test_index]

            regr = linear_model.LARS(n_nonzero_coefs=1, normalize=False).fit(X_train, y_train)
            prediction_train = regr.predict(X_train)
            prediction_test = regr.predict(X_test)

            r2_train += r2_score(prediction_train, y_train)
            r2_test += r2_score(prediction_test, y_test)
            MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
            MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
            MSE_train += mean_squared_error(prediction_train, y_train)
            MSE_test += mean_squared_error(prediction_test, y_test)
            MAE_train += mean_absolute_error(prediction_train, y_train)
            MAE_test += mean_absolute_error(prediction_test, y_test)
            EVS_train += explained_variance_score(prediction_train, y_train)
            EVS_test += explained_variance_score(prediction_test, y_test)

            results = results.append({'Data': data,
                                    'Model': model,
                                    'Params': params,
                                    'Scaler': scaler,
                                    "Features": features,
                                    "FS": fs,
                                    "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                                    "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                                    "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                                    "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                                    "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                                    ignore_index = True)
    except Exception as e:
        print(e)

if model == "LASSO":
    for alpha in [0.2, 0.5, 1, 1.5, 2]:
        try:
            params = {"alpha": alpha}

```

```

for train_index, test_index in kf.split(X):
    X_train, X_test = X[train_index], X[test_index]
    y_train, y_test = Y[train_index], Y[test_index]

    regr = linear_model.LASSO(alpha=alpha, max_iter=5000).fit(X_train, y_train)
    prediction_train = regr.predict(X_train)
    prediction_test = regr.predict(X_test)

    r2_train += r2_score(prediction_train, y_train)
    r2_test += r2_score(prediction_test, y_test)
    MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
    MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
    MSE_train += mean_squared_error(prediction_train, y_train)
    MSE_test += mean_squared_error(prediction_test, y_test)
    MAE_train += mean_absolute_error(prediction_train, y_train)
    MAE_test += mean_absolute_error(prediction_test, y_test)
    EVS_train += explained_variance_score(prediction_train, y_train)
    EVS_test += explained_variance_score(prediction_test, y_test)

results = results.append({'Data': data,
                        'Model': model,
                        'Params': params,
                        'Scaler': scaler,
                        "Features": features,
                        "FS": fs,
                        "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                        "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                        "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                        "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                        "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                        ignore_index = True)
except Exception as e:
    print(e)

if model == "Huber":
    for epsilon in [1.05, 1.1, 1.15, 1.20, 1.25, 1.30, 1.35, 1.40, 1.45]:
        try:
            params = {"epsilon": alpha}
            for train_index, test_index in kf.split(X):
                X_train, X_test = X[train_index], X[test_index]
                y_train, y_test = Y[train_index], Y[test_index]

                regr = HuberRegressor(epsilon = epsilon, max_iter=5000).fit(X_train, y_train)
                prediction_train = regr.predict(X_train)
                prediction_test = regr.predict(X_test)

                r2_train += r2_score(prediction_train, y_train)
                r2_test += r2_score(prediction_test, y_test)
                MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
                MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
                MSE_train += mean_squared_error(prediction_train, y_train)
                MSE_test += mean_squared_error(prediction_test, y_test)
                MAE_train += mean_absolute_error(prediction_train, y_train)
                MAE_test += mean_absolute_error(prediction_test, y_test)
                EVS_train += explained_variance_score(prediction_train, y_train)
                EVS_test += explained_variance_score(prediction_test, y_test)

            results = results.append({'Data': data,
                                    'Model': model,
                                    'Params': params,
                                    'Scaler': scaler,
                                    "Features": features,
                                    "FS": fs,
                                    "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                                    "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                                    "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                                    "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                                    "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                                    ignore_index = True)

```

```

        ignore_index = True)
    except Exception as e:
        print(e)

if model == "ARD":
    try:
        params = {None}
        for train_index, test_index in kf.split(X):
            X_train, X_test = X[train_index], X[test_index]
            y_train, y_test = Y[train_index], Y[test_index]

            regr = linear_model.ARDRegression().fit(X_train, y_train)
            prediction_train = regr.predict(X_train)
            prediction_test = regr.predict(X_test)

            r2_train += r2_score(prediction_train, y_train)
            r2_test += r2_score(prediction_test, y_test)
            MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
            MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
            MSE_train += mean_squared_error(prediction_train, y_train)
            MSE_test += mean_squared_error(prediction_test, y_test)
            MAE_train += mean_absolute_error(prediction_train, y_train)
            MAE_test += mean_absolute_error(prediction_test, y_test)
            EVS_train += explained_variance_score(prediction_train, y_train)
            EVS_test += explained_variance_score(prediction_test, y_test)
        results = results.append({'Data': data,
                                'Model': model,
                                'Params': params,
                                'Scaler': scaler,
                                "Features": features,
                                "FS": fs,
                                "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                                "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                                "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                                "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                                "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                                ignore_index = True)
    except Exception as e:
        print(e)

if model == "KernelRidge":
    for kernel in ["linear", "polynomial", "rbf", "sigmoid", "laplacian", "sigmoid", "chi2"]:
        for alpha in [0.2, 0.5, 1, 1.5, 2]:
            try:
                params = {"alpha": alpha, "kernel": kernel}
                for train_index, test_index in kf.split(X):
                    X_train, X_test = X[train_index], X[test_index]
                    y_train, y_test = Y[train_index], Y[test_index]

                    regr = KernelRidge(alpha=alpha, kernel=kernel).fit(X_train, y_train)
                    prediction_train = regr.predict(X_train)
                    prediction_test = regr.predict(X_test)

                    r2_train += r2_score(prediction_train, y_train)
                    r2_test += r2_score(prediction_test, y_test)
                    MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
                    MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
                    MSE_train += mean_squared_error(prediction_train, y_train)
                    MSE_test += mean_squared_error(prediction_test, y_test)
                    MAE_train += mean_absolute_error(prediction_train, y_train)
                    MAE_test += mean_absolute_error(prediction_test, y_test)
                    EVS_train += explained_variance_score(prediction_train, y_train)
                    EVS_test += explained_variance_score(prediction_test, y_test)

                results = results.append({'Data': data,
                                        'Model': model,
                                        'Params': params,
                                        'Scaler': scaler,

```

```

        "Features": features,
        "FS": fs,
        "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
        "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
        "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
        "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
        "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
        ignore_index = True)
except Exception as e:
    print(e)

if model == "Gaussian":
    try:
        params = {None}
        for train_index, test_index in kf.split(X):
            X_train, X_test = X[train_index], X[test_index]
            y_train, y_test = Y[train_index], Y[test_index]

            regr = GaussianProcessRegressor(random_state=0).fit(X_train, y_train)
            prediction_train = regr.predict(X_train)
            prediction_test = regr.predict(X_test)

            r2_train += r2_score(prediction_train, y_train)
            r2_test += r2_score(prediction_test, y_test)
            MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
            MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
            MSE_train += mean_squared_error(prediction_train, y_train)
            MSE_test += mean_squared_error(prediction_test, y_test)
            MAE_train += mean_absolute_error(prediction_train, y_train)
            MAE_test += mean_absolute_error(prediction_test, y_test)
            EVS_train += explained_variance_score(prediction_train, y_train)
            EVS_test += explained_variance_score(prediction_test, y_test)

        results = results.append({'Data': data,
                                'Model': model,
                                'Params': params,
                                'Scaler': scaler,
                                "Features": features,
                                "FS": fs,
                                "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                                "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                                "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                                "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                                "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                                ignore_index = True)

    except Exception as e:
        print(e)

if model == "Bagging":
    for n_estimator in [10,20,50,100,200,500]:
        try:
            params = {"n_estimators": n_estimator}
            for train_index, test_index in kf.split(X):
                X_train, X_test = X[train_index], X[test_index]
                y_train, y_test = Y[train_index], Y[test_index]

                regr = BaggingRegressor(base_estimator=SVR(),n_estimators=n_estimator, random_state=0).fit(X_train, y_train)
                prediction_train = regr.predict(X_train)
                prediction_test = regr.predict(X_test)

                r2_train += r2_score(prediction_train, y_train)
                r2_test += r2_score(prediction_test, y_test)
                MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
                MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
                MSE_train += mean_squared_error(prediction_train, y_train)

```

```

MSE_test += mean_squared_error(prediction_test, y_test)
MAE_train += mean_absolute_error(prediction_train, y_train)
MAE_test += mean_absolute_error(prediction_test, y_test)
EVS_train += explained_variance_score(prediction_train, y_train)
EVS_test += explained_variance_score(prediction_test, y_test)

results = results.append({'Data': data,
                        'Model': model,
                        'Params': params,
                        'Scaler': scaler,
                        "Features": features,
                        "FS": fs,
                        "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                        "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                        "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                        "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                        "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                        ignore_index = True)
except Exception as e:
    print(e)

if model == "Ada":
    for n_estimator in [10,20,50,100,200,500]:
        try:
            params = {"n_estimators": n_estimator}
            for train_index, test_index in kf.split(X):
                X_train, X_test = X[train_index], X[test_index]
                y_train, y_test = Y[train_index], Y[test_index]

                regr = AdaBoostRegressor(random_state=0, n_estimators=n_estimator).fit(X_train, y_train)
                prediction_train = regr.predict(X_train)
                prediction_test = regr.predict(X_test)

                r2_train += r2_score(prediction_train, y_train)
                r2_test += r2_score(prediction_test, y_test)
                MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
                MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
                MSE_train += mean_squared_error(prediction_train, y_train)
                MSE_test += mean_squared_error(prediction_test, y_test)
                MAE_train += mean_absolute_error(prediction_train, y_train)
                MAE_test += mean_absolute_error(prediction_test, y_test)
                EVS_train += explained_variance_score(prediction_train, y_train)
                EVS_test += explained_variance_score(prediction_test, y_test)

            results = results.append({'Data': data,
                                    'Model': model,
                                    'Params': params,
                                    'Scaler': scaler,
                                    "Features": features,
                                    "FS": fs,
                                    "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                                    "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                                    "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                                    "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                                    "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                                    ignore_index = True)
        except Exception as e:
            print(e)

if model == "GB":
    for n_estimator in [2, 5, 10,20,50,100,200,500]:
        try:
            params = {"n_estimators": n_estimator}
            for train_index, test_index in kf.split(X):
                X_train, X_test = X[train_index], X[test_index]
                y_train, y_test = Y[train_index], Y[test_index]

```

```

regr = GradientBoostingRegressor(random_state=0, n_estimators=n_estimator).fit(X_train, y_train)
prediction_train = regr.predict(X_train)
prediction_test = regr.predict(X_test)

r2_train += r2_score(prediction_train, y_train)
r2_test += r2_score(prediction_test, y_test)
MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
MSE_train += mean_squared_error(prediction_train, y_train)
MSE_test += mean_squared_error(prediction_test, y_test)
MAE_train += mean_absolute_error(prediction_train, y_train)
MAE_test += mean_absolute_error(prediction_test, y_test)
EVS_train += explained_variance_score(prediction_train, y_train)
EVS_test += explained_variance_score(prediction_test, y_test)

results = results.append({'Data': data,
                        'Model': model,
                        'Params': params,
                        'Scaler': scaler,
                        "Features": features,
                        "FS": fs,
                        "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                        "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                        "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                        "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                        "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                        ignore_index = True)
except Exception as e:
    print(e)

if model == "LR":
    try:
        params = {None}
        for train_index, test_index in kf.split(X):
            X_train, X_test = X[train_index], X[test_index]
            y_train, y_test = Y[train_index], Y[test_index]

            regr = LinearRegression().fit(X_train, y_train)
            prediction_train = regr.predict(X_train)
            prediction_test = regr.predict(X_test)

            r2_train += r2_score(prediction_train, y_train)
            r2_test += r2_score(prediction_test, y_test)
            MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
            MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
            MSE_train += mean_squared_error(prediction_train, y_train)
            MSE_test += mean_squared_error(prediction_test, y_test)
            MAE_train += mean_absolute_error(prediction_train, y_train)
            MAE_test += mean_absolute_error(prediction_test, y_test)
            EVS_train += explained_variance_score(prediction_train, y_train)
            EVS_test += explained_variance_score(prediction_test, y_test)

        results = results.append({'Data': data,
                                'Model': model,
                                'Params': params,
                                'Scaler': scaler,
                                "Features": features,
                                "FS": fs,
                                "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                                "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                                "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                                "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                                "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                                ignore_index = True)
    except Exception as e:
        print(e)

```

```

if model == "MLP":
    for size in [(5),(10),(20),(50),(5,5),(10,10),(20,20), (50,50), (5,5,5),(20,20,20),(50,50,50)]:
        try:
            params = {"hidden_size": size}
            for train_index, test_index in kf.split(X):
                X_train, X_test = X[train_index], X[test_index]
                y_train, y_test = Y[train_index], Y[test_index]

                regr = MLPRegressor(random_state=1, max_iter=2000, early_stopping=True, hidden_layer_sizes=size).fit(X_train, y_train)
                prediction_train = regr.predict(X_train)
                prediction_test = regr.predict(X_test)

                r2_train += r2_score(prediction_train, y_train)
                r2_test += r2_score(prediction_test, y_test)
                MAPE_train += mean_absolute_percentage_error(prediction_train, y_train)
                MAPE_test += mean_absolute_percentage_error(prediction_test, y_test)
                MSE_train += mean_squared_error(prediction_train, y_train)
                MSE_test += mean_squared_error(prediction_test, y_test)
                MAE_train += mean_absolute_error(prediction_train, y_train)
                MAE_test += mean_absolute_error(prediction_test, y_test)
                EVS_train += explained_variance_score(prediction_train, y_train)
                EVS_test += explained_variance_score(prediction_test, y_test)

            results = results.append({'Data': data,
                                    'Model': model,
                                    'Params': params,
                                    'Scaler': scaler,
                                    "Features": features,
                                    "FS": fs,
                                    "R2_train":r2_train/n_folds, "R2_test":r2_test/n_folds,
                                    "MAPE_train":MAPE_train/n_folds,"MAPE_test":MAPE_test/n_folds ,
                                    "MSE_train":MSE_train/n_folds , "MSE_test":MSE_test/n_folds,
                                    "MAE_train": MAE_train/n_folds, "MAE_test": MAE_test/n_folds,
                                    "EVS_train": EVS_train/n_folds, "EVS_test": EVS_test/n_folds},
                                    ignore_index = True)
        except Exception as e:
            print(e)

results.to_csv("Regression_results_v2.csv")

```

27 priedas. Duomenų vizualizavimo programos kodas

```
import seaborn as sns

df = pd.read_excel("rawdata.xlsx")

sns.set(rc={'figure.figsize':(16,8.27)})
sns.set_theme(style="white", palette=None)
ax = sns.boxplot(x="Code", y="TUI", data=df)
ax.set_xlabel("Šalis", fontsize = 12)
ax.set_ylabel("Tiesioginių užsienio investicijų įplaukos (mlrd. $)", fontsize = 12)

df = pd.read_excel("rawdata.xlsx")
sns.set(rc={'figure.figsize':(16,8.27)})
sns.set_theme(style="white", palette=None)
ax = sns.boxplot(x="Code", y="GDP_per_capita", data=df)
ax.set_xlabel("Šalis", fontsize = 12)
ax.set_ylabel("Bendras vidaus produktas tenkantis vienam žmogui ($)", fontsize = 12)

sns.set(rc={'figure.figsize':(16,8.27)})
sns.set_theme(style="white", palette=None)
ax = sns.boxplot(x="Code", y="PPP", data=df)
ax.set_xlabel("Šalis", fontsize = 12)
ax.set_ylabel("Perkamosios galios paritetas ($)", fontsize = 12)

sns.set(rc={'figure.figsize':(16,8.27)})
sns.set_theme(style="white", palette=None)
ax = sns.boxplot(x="Code", y="Capital_investment", data=df)
ax.set_xlabel("Šalis", fontsize = 12)
ax.set_ylabel("Kapitalo investicijos (% BVP)", fontsize = 12)
```