KAUNO TECHNOLOGIJOS UNIVERSITETAS
MATEMATIKOS IR GAMTOS MOKSLŲ FAKULTETAS

Ovidijus Samsonovas

DIDŽIŲJŲ LIETUVOS MIESTŲ GYVENTOJŲ IŠLAIDŲ, BŪSTO IŠLAIKYMUI, PROGNOZAVIMO MODELIAI

Baigiamasis magistro projektas

Vadovas
Doc. dr. Audrius Kabašinskas

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS

MATEMATIKOS IR GAMTOS MOKSLŲ FAKULTETAS

DIDŽIŲJŲ LIETUVOS MIESTŲ GYVENTOJŲ IŠLAIDŲ, BŪSTO IŠLAIKYMUI, PROGNOZAVIMO MODELIAI

Baigiamasis magistro projektas

Taikomoji matematika (kodas 621G10003**)**

Vadovas
Doc. dr. Audrius Kabašinskas

Recenzentas
Doc. dr. Mindaugas Kavaliauskas

Projektą atliko
Ovidijus Samsonovas

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS

MATEMATIKOS IR GAMTOS MOKSLŲ FAKULTETAS

Ovidijus Samsonovas

Taikomoji matematika (621G10003)

Baigiamojo projekto „Didžiųjų Lietuvos miestų gyventojų išlaidų, būsto išlaikymui, prognozavimo modeliai“

AKADEMINIO SĄŽININGUMO DEKLARACIJA

2015 m. birželio mėn. 2 d.

Kaunas

Patvirtinu, kad mano, **Ovidijaus Samsonovo,** baigiamasis darbas tema „Didžiųjų Lietuvos miestų gyventojų išlaidų, būsto išlaikymui, prognozavimo modeliai“ yra parašytas visiškai savarankiškai, o visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena darbo dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymu nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

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

__________________________________ ________________

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

Samsonovas, O. The forecasting models for household costs in major cities of Lithuania. *Master's work in applied mathematics* / supervisor dr. assoc. prof. A. Kabašinskas; Mathematical *Modelling department*, Faculty of Mathematics and Natural Sciences, Kaunas University of Technology.

Kaunas, 2015. 99 p.

SUMMARY

A time series is a collection of observations made sequentially in time. Forecasting the future values of an observed time series is an important problem in many areas, including economics, production planning, sales, weather forecasting, stock control and etc. A wide variety of different forecast procedures are available and it's significant to realize that no single method is universally applicable. Rather the analyst must choose the procedure which is most appropriate for a given set of conditions.

In this research paper were used household costs data of three major cities of Lithuania: Vilnius, Kaunas, Klaipėda. Various models were fitted to them: seasonal additive and multiplicative time series models, ARIMA – autoregressive integrated moving average process.

In our case, every time series of household costs data was non-stationary, so we used simple and seasonally differentiation of time series for ARIMA models. To estimate the precision and correctness of each model there was calculated MAPE - the mean absolute percentage error. Off course, all forecasting models were checked if they really provide an adequate description of data. Generalized Autoregressive Conditional Heteroskedasticity models (GARCH) was fitted to volatility of residuals.

In conclusion, the results have shown, that the most precise forecasting method for Vilnius data is *ARIMA(5,1,4)* time series model with *ARCH(1)* process for residuals, for Kaunas data - seasonal multiplicative time series model with *AR(2)* process for residuals and for Klaipėda - seasonal multiplicative time series model with *AR(1)* process for residuals. As it is common for ARIMA models, good predictions are gained for short – time forecasting and for those periods, which had more observations.

TURINYS

LENTELIŲ SĄRAŠAS

PAVEIKSLŲ SĄRAŠAS

SANTRUMPŲ SĄRAŠAS

ACF – autokoreliacijos funkcija

ADF – išplėstinis Dickey ir Fullerio testas

AIC – Akaike informacinis kriterijus

AR – autoregresijos procesas

ARCH – autoregresinis sąlyginio heteroskedastiškumo modelis

ARIMA – autoregresinis integruotasis slenkančio vidurkio procesas

ARMA – autoregresinis slenkančio vidurkio procesas

BIC – Bajes‘o informacinis kriterijus

BL – Box‘o ir Ljung‘o testas

GARCH – apibendrintasis autoregresinis sąlyginio heteroskedastiškumo modelis

JB – Jarque ir Bera testas

MA – slenkančio vidurkių procesas

MAPE – Vidutinė procentinė absoliutinė paklaida

PACF – dalinės autokoreliacijos funkcija

SARIMA – sezoninis integruotasis autoregresinis slenkančio vidurkio procesas

SSE – paklaidų kvadratų suma

ĮVADAS

Šiuolaikinėje visuomenėje, esant sparčiam gyvenimo tempui, nuolat didėja informacijos srautas, vis didesnį vaidmenį įgyja prognozavimas. Jis naudojamas įvairiose srityse, tokiose kaip meteorologija, socialiniai reiškiniai, ekonomika, kurioje itin svarbi yra įvairių rodiklių prognozė, optimalus įvairių išteklių panaudojimas, medicina, ar įvairių faktorių poveikių vertinimas finansinėms sistemoms. Norint suprasti ir prognozuoti įvairius procesus, naudojamas modeliavimas, kuris yra neatskiriama bet kurio mokslo dalis.

Šiame darbe buvo nagrinėjamas vienas iš ekonominių rodiklių, su kuriuo susiduriame daugelis iš mūsų – išlaidos, būsto išlaikymui. Tyrime buvo naudojami UAB „Viena Sąskaita" trijų didžiųjų Lietuvos miestų gyventojų bendrų išlaidų, būsto išlaikymui, duomenys. Šių ekonominių laiko eilučių prognozavimui buvo pasirinkti sezoniniai, multiplikatyvusis ir adityvusis, laiko eilutės modeliai, autoregresinis integruotasis slenkamųjų vidurkių (ARIMA) metodas, kadangi atlikus stacionarumo tyrimą gavome, jog visų trijų miestų duomenų eilutės yra nestacionarios, ir sezoniniai ARIMA modeliai. Siekiant kuo objektyviau įvertinti bei palyginti prognozavimo modelius, atlikta šių modelių adekvatumo analizė ir apskaičiuotos kiekvieno jų vidutinės procentinės absoliutinės paklaidos. Modeliams, kurių paklaidų dispersijos nebuvo pastovios, taikėme sąlyginio heteroskedastiškumo modelius.

Atlikus visų trijų miestų gyventojų išlaidų prognozavimo modelių tyrimą, paaiškėjo, jog Vilniaus gyventojų išlaidoms, būsto išlaikymui, prognozuoti labiausiai tinka *ARIMA(5,1,4)* laiko eilutės modelis su *ARCH(1)* procesu paklaidoms, Kauno gyventojų išlaidoms, būsto išlaikymui, labiausiai tinka multiplikatyvusis sezoninis regresinis laiko eilutės modelis su autoregresiniu antros eilės procesu liekanoms, Klaipėdos gyventojų išlaidoms, būsto išlaikymu, labiausiai tinka multiplikatyvusis sezoninis regresinis laiko eilutės modelis su autoregresiniu antros eilės procesu liekanoms, tačiau ir kiti modeliai mažai skiriasi tiek MAPE reikšme, tiek ir prognozėmis.

Darbo tikslas: išsiaiškinti, kuris iš pasirinktų prognozavimo metodų yra tinkamiausias prognozuoti Vilniaus, Kauno ir Klaipėdos gyventojų išlaidoms, būsto išlaikymui, bei atlikti prognozę.

1. ANALITINĖ DALIS

Analitinėje dalyje aprašyti surinkti duomenys, apžvelgti sezoniniai, adityvusis ir multiplikatyvusis, regresiniai modeliai, autoregresinis integruotasis slenkamųjų vidurkių (ARIMA) ir sąlyginio heteroskedastiškumo modeliai (GARCH), kuriuos naudosime esant nepastoviai liekanų dispersijai, metodai.

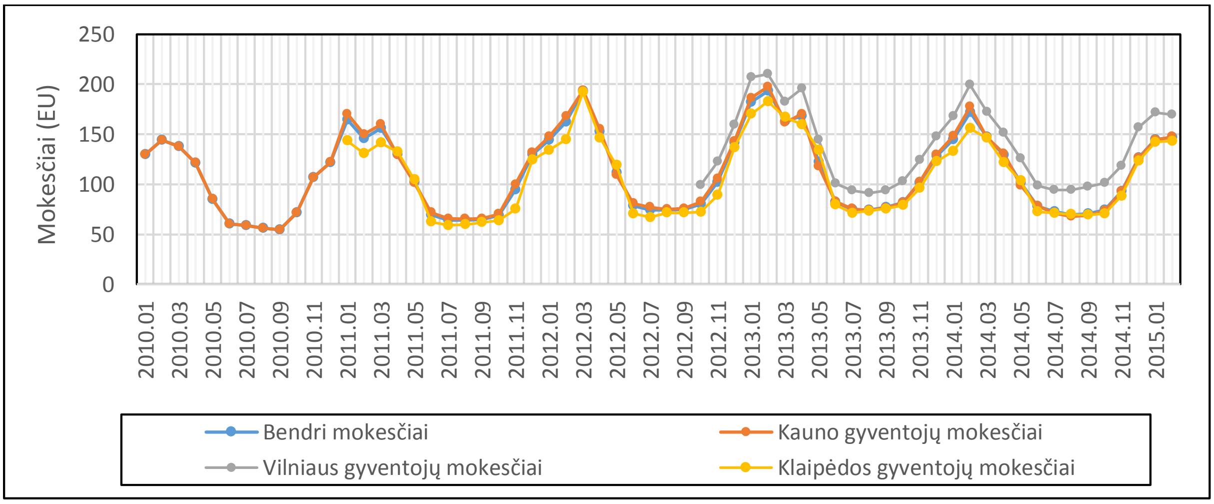
1.1. TIRIAMASIS OBJEKTAS

Magistriniame darbe nagrinėsime didžiųjų Lietuvos miestų: Vilniaus, Kauno ir Klaipėdos gyventojų vidutines mėnesines išlaidas už būsto išlaikymą. Šias išlaidas sudaro tokie mokesčiai kaip:

- Šildymas, vandens pašildymas ir dujos;
- Elektra;
- Vanduo;
- Komunalinės paslaugos;
- Telekomunikacijos ir internetas.

Gyventojų išlaidų duomenys buvo gauti analizuojant bendrovės „Viena sąskaita“ duomenis. Buvo išanalizuoti virš 30 000 namų ūkių trijuose didžiausiuose Lietuvos miestuose.

Kiekvieno miesto surinktų duomenų eilutės turi skirtingą duomenų kiekį. Kauno miesto gyventojų mokesčių duomenys surinkti per 2010.01.01- 2015.02.01 laikotarpį (62 stebėjimai), Klaipėdos - 2011.01.01- 2015.02.01 (50 stebėjimų), Vilniaus - 2012.10.01- 2015.02.01 (29 stebėjimai).



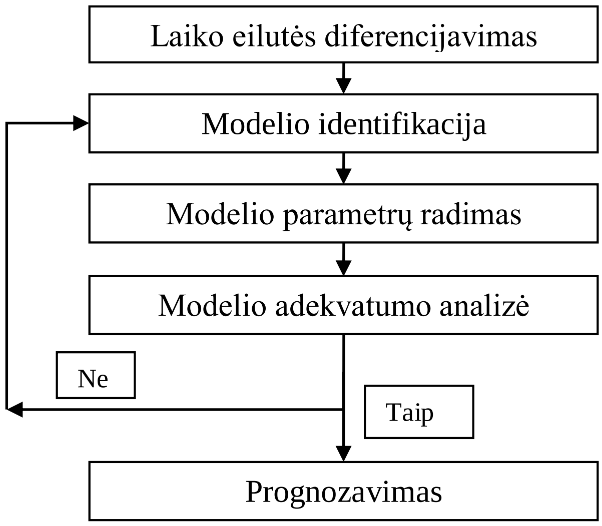
1.1.1 pav. Gyventojų bendri mokesčiai būsto išlaikymui

Iš 1.1.1 pav. matyti, kad didžiausios išlaidos, būsto išlaikymui, yra vilniečių, mažiausios Klaipėdos gyventojų. Taip pat pastebima visų miestų mokesčių sezoniškumo tendencija, į ką atsižvelgsime tolesniame tyrime.

1.2. TYRIME NAUDOJAMŲ METODŲ APŽVALGA

Norint gauti kuo didesnį prognozavimo tikslumą, kuriami ir atrandami nauji metodai, modifikuojami senieji ar pritaikomi tiesioginės paskirties prognozavimui neturėję metodai. Laiko eilučių pritaikymas praktikoje yra labai platus. Iki šių dienų plėtojami ir toliau nagrinėjami bei tobulinami įvairūs modeliai, aprašantys specifinį laiko eilučių elgesį. Vieni žinomiausių yra *autoregresiniai slenkamųjų vidurkių metodai,* finansų rinkose ypač dažnai naudojami naudojami *sąlyginio heteroskedastiškumo* - *ARCH*, $GARCH$. Šie modeliai dažniausiai skirti trumpų duomenų eilučių modeliavimui.

Laiko eilučių modeliai yra vieni iš tinkamiausių prognozavimo uždaviniams spręsti. Vienas seniausių yra autoregresinis slenkančio vidurkio modelis $ARMA$ (AutoRegressive Moving-Average). Iš šio modelio išaugo visa eilė daugiau ar mažiau ištirtų jo modifikacijų. Galima sakyti, jog $ARMA$ modelį išpopuliarino Box ir Jenkins 1970 metais išleidę knygą: „Time series analysis: Forecasting and control“. Nors AR ir MA modeliai buvo žinomi ir naudojami gan seniai. Box ir Jenkins pateikė sisteminį požiūrį kaip apjungti AR ir MA modelius į vieną, taip pat suformulavo modifikaciją $ARIMA$ (AutoRegressive Integrated Moving-Average) - autoregresinis integruotas slenkančio vidurkio modelis.



1.2.1 pav. Box-Jenkins schema

ARMA modeliai analizuoja stacionarias laiko eilutes. Modelio stacionarumas – tai tarsi realybės supaprastinimas. Vienas iš žinomesnių nestacionarumo šaltinių yra tiesinė komponentė. Kadangi tiesinės funkcijos išvestinė yra konstanta, tai tiesinė komponentė eliminuojama diferencijuojant. Šis metodas taikomas ARIMA modeliuose. Daugelis autorių rekomenduoja ARMA modeliams ir jų modifikacijoms naudoti bent 50 stebėjimų laiko eilutes (Chatfield, 1996).

Ekonominių laiko eilučių savybė, būdinga daugeliui finansinių laiko eilučių, yra ta, kad jų kintamumas (angl. volatility) keičiasi laiko atžvilgiu [3]. Finansų ekonomikos tyrinėtojams ir finansų rinkų analitikams grąžos kintamumas yra vienas iš svarbiausių klausimų. Akcijų ir kitokio turto kainos priklauso nuo numatomo grąžų kintamumo. Bankai ir kitos finansų įstaigos vertina kintamumą ir tai yra jų patiriamos rizikos stebėsenos dalis [11].

Kintamumo modelių tyrinėjimus inicijavo R. F. Engle, praėjusio amžiaus devintąjį dešimtmetį sukūręs naują koncepciją, kurią pavadino autoregresiniu sąlyginiu heteroskedastiškumu ir davė jam akronimą ARCH. (2.31) lygtis nusako R. F. Engle (1982) sukurtą ARCH modelį, kuriame sąlyginė dispersija yra praėjusių paklaidų kvadratų funkcija. Šiame klasikiniame straipsnyje R. F. Engle pateikė ARCH modelių vertinimo teoriją, nusakė didžiausio tikėtinumo įverčių suderinamumo bei asimptotinio normalumo sąlygas ir pasiūlė Lagrange daugiklių testą hipotezei apie ne-ARCH paklaidas ε_t tikrinti [11].

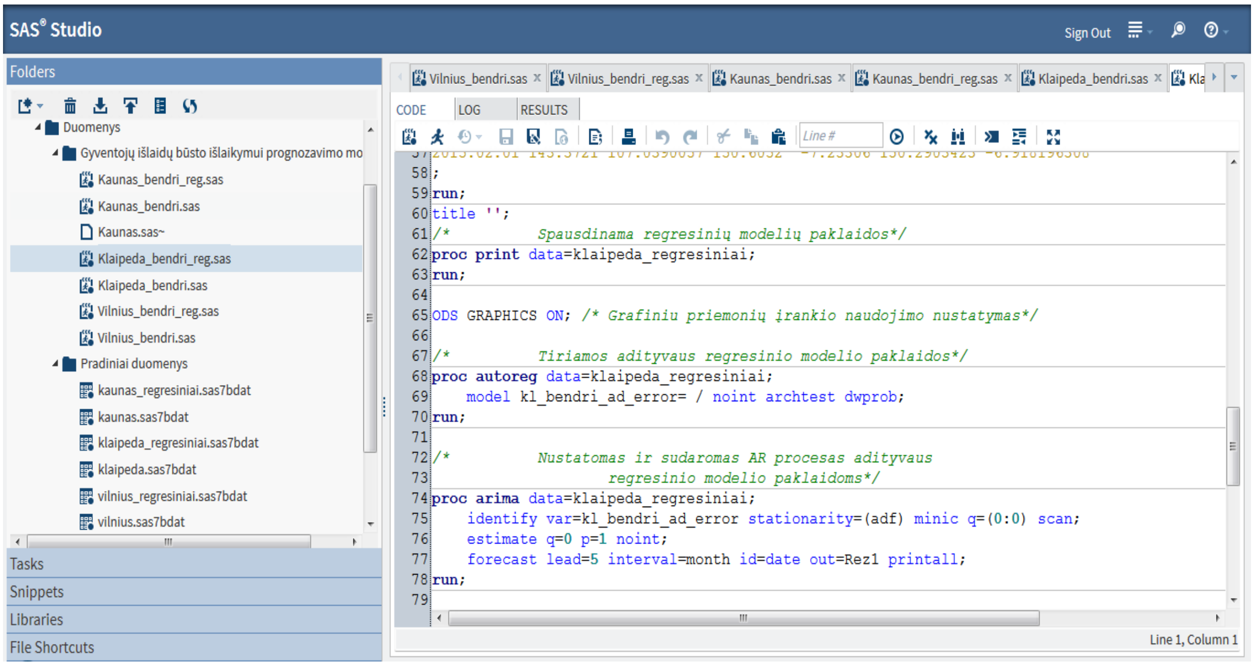
Praktikoje pakankamai aukšto dažnio, pavyzdžiui, dieninių ar savaitinių stebėjimų, grąžų kvadratų ε_t^2 autokoreliacinė funkcija gęsta lėtai. Siekiant tapačiai apibūdinti šį stilizuotą faktą, reikia ARCH modelio su dideliu lagu q. Tačiau jei (2.31) lygties dešinioji pusė yra modifikuota pridedant pavėlintas sąlyginės dispersijos σ_t^2, gautas modelis gali būti sudarytas jau su mažesniu parametrų skaičiumi, kuriame paklaidų ε_t^2 autokoreliacinė funkcija vėlgi gęsta lėtai (Bollerslev, 1986).

Greitai, paskelbus straipsnį apie ARCH, R. F. Engle studentas Tim Bollerslev sukūrė tokį modelį ir pavadino jį apibendrintu ARCH (Generalized Autoregressive Conditional Heteroskedasticity – GARCH) modeliu. Esant tam tikroms papildomoms sąlygoms, *GARCH* modelis gali būti užrašytas kaip *ARCH* (∞) modelis, kitaip tariant, σ_t^2 gali būti išreikštas kaip istorinių kvadratinių gražų r_t^2, (kur $s < t$) slenkantis vidurkis su eksponentiškai gęstančiais koeficientais [6].

Sezoninių laiko eilučių modeliai naudojami tada, kai stebėjimai pasikartoja vienodais laiko tarpais, periodais. Šie modeliai gali būti adityvieji ir multiplikatyvieji. Modelio tipo parinkimas priklauso nuo sezoninių svyravimų pobūdžio. Jei sezoninių svyravimų amplitudė pastovi, naudojamas adityvusis regresijos modelis. Jei svyravimai didėja arba mažėja – multiplikatyvusis.

1.3. PROGRAMINĖS ĮRANGOS APŽVALGA

Duomenų analizei pasirinkta interaktyvi sistema *SAS studio*, kuri yra gana nauja, populiari pasaulyje universalių duomenų analizės sistemų, galinti atlikti įvairias funkcijas. Šis analizės įrankis prieinamas kiekvienam registruotam vartotojui, todėl nereikia siųstis programinės įrangos ar jos papildinių. Tam pakanka turėti internetinės naršyklės programą.



1.3.1 pav. SAS Studio programos langas

SAS – svarbiausias verslo analitikams skirtų sprendimų tiekėjas pasaulyje, siūlantis būdus mokytis iš praeities, stebėti dabartį, suvokti ir numatyti ateitį. Sistema SAS (angl. *Statistical Analysis System*) sparčiai vystoma nuo 1976 metų. Jos pagrindinės funkcijos [10]:

- sąveika su duomenimis (gali sąveikauti su įvairiais duomenų šaltiniais, skaityti ir rašyti į įvairiausių formatų failus, tiesiogiai sąveikauti su populiariausiomis duomenų bazėmis);
- duomenų vadyba (duomenų įvedimas, redagavimas, įvairių duomenų pjūvių formavimas, kelių duomenų failų apjungimas į vieną);
- duomenų analizė (didžiulis rinkinys procedūrų pradedant nuo skaitinių charakteristikų apskaičiavimo ir baigiant specialiais taikomosios statistikos metodais);
- duomenų pristatymas (duomenų analizės rezultatų pateikimas lentelių, diagramų, grafikų, žemėlapių pavidalu).

Darbe buvo naudojamos šios SAS posistemės:

- Laiko eilučių prognozavimo įrankiai (SAS/ETS);
- SAS programavimo kalba (SAS/BASE);

- SAS/ACCESS, kuri užtikrina tiesioginį naudojimąsi iš įvairiausių duomenų tipų šaltinių;
- Taikomosios statistikos metodai (SAS/STAT);
- Grafinis duomenų pateikimas (SAS/GRAPH).

Taip pat darbe buvo naudojamasi MS Exel programa sudaryti sezoninius regresinius modelius bei pateikti, kai kuriuos grafinius rezultatus. Tai Universali skaičiuoklė, kurios pagalba nesudėtingai galima atlikti daug ir įvairių paskaičiavimų.

2. METODOLOGINĖ DALIS

Sudarant ekonometrinius modelius, visuomet reikalingi tiriamo proceso stebėjimo duomenys. Kai duomenys renkami reguliariais laiko intervalais, reikšmę įgyja ne tik patys duomenys, bet ir jų gavimo tvarka. Šie duomenys nusako vieno objekto savybes tam tikrais fiksuotais laiko momentais. Tad, ekonominio proceso reikšmės, nustatytos reguliariais intervalais, sudaro laiko eilutės duomenis [1]. **Laiko eilutė** (angl. *time series*) – statistiniai duomenys surinkti reguliariais laiko intervalais.

Analizuojant laiko eilutes, sprendžiami trys pagrindiniai uždaviniai:

- Identifikacijos. Identifikacijos metu parenkamas preliminarus analizės modelis, nustatomas tiesinės komponentės pobūdis bei sezoniškumo efekto pasireiškimas. Taip pat šiame etape yra nustatomi sudarytąjį modelį apibūdinantys parametrai. Šiuo tikslu naudojamas duomenų glodinimas, regresiniai metodai, Box'o- Jenkins'o ir ARIMA analizės metodai. Jeigu šie parametrai neleidžia tinkamai aprašyti laiko eilutės, reikia sudaryti kitą preliminarųjį laiko eilutės modelį.
- Verifikacijos. Tai galutinis įvertinimas. Šiame etape nustatomas modelio adekvatumas konkrečios laiko eilutės analizei remiantis statistiniais kriterijais.
- Ateities rezultatų prognozavimas. Pagal sudarytąjį laiko eilutės modelį, sudaromos prognozuojamos reikšmės.

Tiriamajame darbe nagrinėsime ARIMA ir regresinius modelius, kai duomenys yra renkami sezoniškai. Taip pat, modeliams, kuriems būdingos heteroskedastiškos liekanos taikysime sąlyginius heteroskedastiškumo modelius. Norėdami padaryti laiko eilutę stacionaria, ją diferencijuojame. Visiems sudarytiems modeliams bus atlikta adekvatumo analizė ir patikrinta, ar sudarytasis prognozavimo modelis yra korektiškas.

2.1. SEZONINIAI LAIKO EILUTĖS REGRESIJOS MODELIAI

2.1.1. ADITYVUSIS LAIKO EILUTĖS REGRESIJOS MODELIS

Sezoninės laiko eilutės dažniausiai modeliuojamos šiais regresijos modeliais: adityviaisiais ir multiplikatyviaisiais. Naudojant regresijos modelį, nesvarbu, ar tai būtų adityvusis, ar multiplikatyvusis, visuomet atskirai reikia įvertinti trendo m_t, sezoniškumo s_t, ir atsitiktinės paklaidos ε_t dėmenų įtaką kiekvienai laiko eilutės reikšmei [1].

Adityvųjį laiko eilutės regresijos modelį užrašysime taip:

$$y_t = m_t + s_t + \varepsilon_t \qquad \textbf{(2.1)}$$

Sezoniškumo $S_t^{(a)}$ įverčio apskaičiavimas t-uoju laiko momentu:

$$S_t^{(a)} = y_t - m_t \qquad \textbf{(2.2)}$$

Atskiro k-oto sezoniškumo indekso vidutinė įverčio reikšmė apskaičiuojama taip:

$$\bar{S}_k^{(a)} = \frac{\sum_{k=1}^{n_k} S_t^{(a)}(k)}{n_k} \qquad \textbf{(2.3)}$$

Apskaičiuojamas sisteminio dėmens įvertis:

$$W_t^{(a)} = m_t^{(a)} + S_t \qquad \textbf{(2.4)}$$

čia $m_t^{(a)}$ - trendo reikšmė apskaičiuota pagal sudarytą trendo regresijos modelį.

Paklaida apskaičiuojama taip: [1]

$$\varepsilon_t^{(a)} = y_t - W_t^{(a)} \qquad \textbf{(2.5)}$$

2.1.2. MULTIPLIKATYVUSIS LAIKO EILUTĖS REGRESIJOS MODELIS

Multiplikatyvųjį laiko eilutės regresijos modelį užrašysime taip:

$$y_t = m_t \cdot s_t + \varepsilon_t \qquad \textbf{(2.6)}$$

čia: y_t- laiko eilutės reikšmė t-uoju laiko momentu, m_t- trendas, s_t - sezoniškumo įvertis, ε_t – atsitiktinės paklaidos.

Sezoniškumo $S_t^{(m)}$ įverčio apskaičiavimas t-uoju laiko momentu:

$$S_t^{(m)} = \frac{y_t}{m_t} \qquad \textbf{(2.7)}$$

Atskiro k-oto sezoniškumo indekso vidutinė įverčio reikšmė apskaičiuojama taip:

$$\bar{S}_k^{(m)} = \frac{\sum_{k=1}^{n_k} S_t^{(m)}(k)}{n_k} \qquad \textbf{(2.8)}$$

Apskaičiuojamas sisteminio dėmens įvertis:

$$W_t^{(m)} = m_t^{(m)} \cdot S_t \qquad \textbf{(2.9)}$$

čia $m_t^{(m)}$ - trendo reikšmė apskaičiuota pagal sudarytą trendo regresijos modelį.

Paklaida apskaičiuojama taip [1]:

$$\varepsilon_t^{(m)} = y_t - W_t^{(m)} \qquad \textbf{(2.10)}$$

2.2. STACIONARIEJI PROCESAI

Procesas vadinamas stacionariuoju siaurąja prasme, jei jo daugiamačiai skirstiniai nepriklauso nuo postūmio laike:

$$\forall t_1, \dots, t_k \in T, k = 1,2, \dots, F_{t_1,\dots t_k}(\cdot) = F_{t_1+\tau,\dots,t_k+\tau}(\cdot), jei\ \ t_i + \tau \in T. \qquad \textbf{(2.11)}$$

Procesas ξ_t vadinamas stacionariuoju plačiąja prasme, jei jo matematinis vidurkis ir kovariacinė funkcija nepriklauso nuo postūmio laike:

$$\forall t, s \in T \quad m(t) = m(0), R(t,s) = R(t-s,0) \qquad \textbf{(2.12)}$$

Jei procesas ξ_t yra Gausso, tai stacionarusis siaurąja prasme procesas yra stacionarusis plačiąja prasme ir atvirkščiai.

Atsitiktinių dydžių procesas $Y_t, t \in T$ yra stacionarusis plačiąja prasme, jei:

1. $m_t = EY_t,\ t \in T$.
2. $DY_t = cov(Y_t, Y_t) = \gamma_0$ yra pastovi.
3. $cov(Y_t, Y_{t+s}) = cov(Y_{t+h}, Y_{t+s+h}) = \gamma_s\ \forall h$. Proceso autokovariacinė funkcija γ priklauso tik nuo atstumo tarp laiko momentų, bet ne nuo pačių momentų.

Stacionarus procesas $Y_t, t \in T$ vadinamas baltuoju triukšmu, jei [2]:

1. $EY_t = 0$.
2. $$cov(Y_t, Y_s) = \begin{cases} \sigma^2, jei\ t = s \\ 0, jei\ t \neq s. \end{cases} \qquad \textbf{(2.13)}$$

Laiko eilutės stacionarumui nustatyti, naudojamas išplėstinis Dickey-Fuller testas (angl. *Augmented Dickey-Fuller*) ir grafinė analizė.

2.3. ARIMA MODELIAI

Jei laiko eilutės stebimos reikšmės stipriai koreliuotos tarpusavyje, tai ateities reikšmę galima prognozuoti naudojantis praeityje stebėtomis reikšmėmis, dažniausiai turinčiomis didžiausią įtaką. Paprasčiausias autoregresinis laiko eilutės modelis (*AR()*) su vienu parametru yra apibrėžiamas:

$$Y_t = \mu + \phi Y_{t-1} + \varepsilon_t, \qquad \textbf{(2.14)}$$

kur ε_t yra nepriklausomų atsitiktinių dydžių stacionari seka vadinama baltu triukšmu, su vidurkiu lygiu nuliui ir dispersija σ_z^2.

Pagal šį metodą kiekviena laiko eilutės reikšmė yra tiesinė prieš tai buvusios reikšmės ar reikšmių funkcija. Pirmos eilės autoregresinėje lygtyje yra naudojama tik viena prieš tai buvusi reikšmė, antros eilės – dvi prieš tai esančios reikšmės ir t.t. Prieš tas reikšmes esantys koeficientai nusako, kaip stipriai kiekviena laiko eilutės reikšmė priklauso nuo prieš tai buvusių reikšmių.

Apibendrintas autoregresinis *p* eilės modelis (*AR(p)*) apibrėžiamas:

$$Y_t = \mu + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t, \quad \textbf{(2.15)}$$

kur $\phi_1, \phi_2, \ldots$ - AR koeficientai.

Pažymėję: $L^j y_t = y_{t-j}$, kad $\phi(L) = 1 - \phi_1 L - \cdots - \phi_p L^p$, galime užrašyti:

$$\phi(L)Y_t = \varepsilon_t, Y_t = Y_t - EY_t, \mu = \phi(1)EY_t. \quad \textbf{(2.16)}$$

Kitas pagrindinis Box-Jenkins modelis vadinamas slenkančiųjų vidurkių modeliu. Kitaip nei autoregresinio modelio atveju, slenkančio vidurkio modelio parametrai priklauso nuo atsitiktinių paklaidų ankstesniame laikotarpyje. Vieno parametro MA modelis užrašomas:

$$Y_t = \varepsilon_t - \theta_1 \varepsilon_{t-1}, \quad \textbf{(2.17)}$$

kas reiškia, kad Y_t laiko eilutėje tiesiogiai priklauso tik nuo atsitiktinės paklaidos ε_{t-1} iš ankstesnio stebėjimo ir tam tikros paklaidos ε_t.

Apibendrintas slenkančių vidurkių *q* eilės modelis (*MA(p)*) apibrėžiamas:

$$Y_t = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}, \quad \textbf{(2.18)}$$

kur $\theta_1, \theta_2, \ldots$ - MA modelio parametrai.

Apungiant autoregresijos modelį su slenkančio vidurkio modeliu gaunamas *ARMA*(*p*,*q*) modelis.

Bendroji ARMA modelio išraiška:

$$Y_t = \mu + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}$$

(2.19)

Išraišką galime supaprastinti įvedę vėlinimo operatorių *L* [3]:

$$\phi(L)Y_t = \theta(L)\varepsilon_t. \quad \textbf{(2.20)}$$

ARIMA modelis laiko eilutę išskaido į autoregresinį procesą AR, aprašantį praeities įvykius, integruotą procesą, padedantį stabilizuoti duomenis, ir slenkančio vidurkio MA procesą, vertinantį modelio paklaidų poveikį duomenims. Matematiškai modelis užrašomas taip:

$$\phi(L)(1 - L)^d Y_t = \theta(L)\varepsilon_t, \quad \textbf{(2.21)}$$

kur $\phi(z)$ ir $\theta(z)$ yra *p* ir *q* eilės polinomai, *L* – vėlavimo operatorius, *d*- integravimo eilė, ε_t yra modelio liekanų procesas.

Sezoniniai *ARIMA(p,d,q)(P,D,Q)s* modeliai užrašomi taip:

$$\phi(L)\tilde{\phi}(L^S)(1 - L)^d (1 - L)^S Y_t = \theta(L)\tilde{\theta}(L^S)\varepsilon_t, \quad \textbf{(2.22)}$$

čia *S* – proceso periodiškumas [3].

2.3.1. ARIMA EILĖS NUSTATYMAS

ARIMA modelio eilės nustatymui padeda autokoreliacinė ir dalinė autokoreliacinė funkcijos.

AR(p) proceso eilė *p* nustatoma tiriant dalinės autokoreliacijos koeficientus. AR procesui būdinga tai, jog dalinės autokoreliacijos koeficientas p vėlavimų yra didelis, o likusiuose vėlavimuose dalinė autokoreliacija yra nebereikšminga [4].

MA procesui būdinga tai, jog autokoreliacijos koeficientas yra didelis q vėlavimų, o likusiuose vėlavimuose autokoreliacija yra nebereikšminga. Taigi, radę ACF ir PACF grafikuose reikšmingumo lygmenį kertančias reikšmes, nustatome, kurios eilės modeliais galėtume aprašyti turimą laiko eilutę [4].

Paprastai vienai ir tai pačiai laiko eilutei yra keli galimi ARIMA modeliai, todėl parinkti eilę naudojami įvairūs kriterijai: AIC, BIC, SSE ir kt.

AIC informacinis kriterijus yra vienas iš dažniausiai naudojamų, kai reikia pasirinkti tinkamiausią modelį. Akaike informacinis kriterijus (AIC) geriausiam modelio parinkimui naudoja tikėtinumo funkciją bei parodo sudaryto modelio kokybę ir yra aprašomas formule:

$$AIC = 2k - 2\ln(L), \qquad \textbf{(2.23)}$$

čia *k* yra vertinamų parametrų skaičius, o *L* - maksimizuota modelio tikėtinumo funkcijos reikšmė [14].

**Bajes`o informacinis kriterijus (BIC)** dar vadinamas Schwarz`o kriterijumi. Jis parodo parametrinio modelio tinkamumą analizuojamiems duomenims. Pagrindinis AIC ir BIC skirtumas yra tai, kad Bajes`o informacinis kriterijus labiau atsižvelgia į vertinamų parametrų skaičių. Taigi, BIC naudojamas modelio parinkimui ir jo matematinė išraiška yra:

$$BIC = kln(n) - 2\ln(L), \qquad \textbf{(2.24)}$$

čia *L* – tikėtinumo funkcija, *k* – vertinamų parametrų skaičius, o *n* – stebėjimų skaičius.

Bayesian informacinis kriterijus – BIC – gali būti užrašytas taip:

$$BIC(p,q) = \ln\left(\widehat{\sigma^2_{(p,q)}}\right) + 2(p+q)\ln(n)\,/n, \qquad \textbf{(2.25)}$$

čia p – autoregresinio proceso eilė, q – slenkančių vidurkių narių skaičius, *n* yra laiko eilutės duomenų skaičius, $\sigma^2_{(p,q)}$- standartinis nuokrypis [14].

SSE - liekanų kvadratų suma, kuri susidaro dėl atsitiktinių klaidų [14]:

$$SSE = \sum_{i=1}^{n} \left(\hat{Y}_i - Y_i\right)^2. \qquad \textbf{(2.26)}$$

2.3.2. BALTASIS TRIUKŠMAS. BOX`o - LJUNG`o TESTAS

Modelis yra adekvatus, jei jo liekanos sudaro baltąjį triukšmą. Procesas W_t vadinamas baltuoju triukšmu, jei jis yra stacionarus, vidurkis $EW_t = 0, \forall t,$ ir kovariacija $cov(W_t, W_s) = 0, t \neq s.$

Viena iš baltojo triukšmo sąlygų yra duomenų nepriklausomumas. Liekanų nepriklausomumui patikrinti galima naudoti įvairius testus. Vienas jų yra Box – Ljung testas. Box – Ljung testas tikrina, ar kuri nors iš laiko eilutės autokoreliacijų grupės skiriasi nuo nulio.

$$\begin{cases} H_0: \rho_1 = \cdots = \rho_m = 0, t.y. liekanos\ nekoreliuotos, \\ H_1: \exists i \leq m: \rho_i \neq 0, \qquad t.y. galima\ liekanų\ autokoreliacija. \end{cases}$$

Naudojama statistika:

$$Q = n(n+2)\sum_{j=1}^{h}\frac{\hat{\rho}_j^{\ 2}}{n-j}, \tag{2.27}$$

kur n yra imties dydis, $\hat{\rho}_j^{\ 2}$ yra vėlavimo j autokoreliacija, h yra skaičius tikrinamų vėlavimų. Ši statistika yra lyginama su χ – kvadrato su h laisvės laipsnių ir α reikšmingumo lygmeniu skirstiniu [14].

2.3.3. LAIKO EILUTĖS STACIONARUMO NUSTATYMAS. ADF TESTAS

Norint taikyti ARIMA modelį, turi būti išpildytas pagrindinis reikalavimas: laiko eilutė turi būti stacionari, t.y. vidurkis ir autokoreliacijos funkcija nekinta laike.

Laiko eilutės stacionarumo nustatymo būdai:

- grafinė analizė;
- autokoreliacijos analizė;
- mažiausios dispersijos testas;
- Dickey-Fuller testai [4].

Tyrime, išlaidų, būto išlaikymui, laiko eilučių stacionarumui nustatyti, pasirinkome išplėstinį Dickey-Fuller testą (ang. *Augmented Dickey-Fuller*) ir grafinę analizę.

Augmented Dickey-Fuller arba ADF yra vienas populiariausių stacionarumo hipotezės tikrinimo testų. Šis testas skirtas patikrinti, ar laiko eilutė turi vienetinių šaknų. Vienetinė šaknis –

tai laiko eilutės autoregresijos parametras, lygus 1. Jei laiko eilutė turi vienetinių šaknų, sakoma, kad ji yra nestacionari.

ADF atliekama modeliui:

$$\Delta Y_t = \alpha + \beta t + \gamma Y_{t-1} + \delta_1 \Delta Y_{t-1} + \ldots + \delta_p \Delta Y_{t-p} + \varepsilon_t, \tag{2.28}$$

čia α ir β konstantos, p – vėlavimų eilė. Vienetinės šaknies testas yra vykdomas nulinei hipotezei $\gamma = 0$ su alternatyvia $\gamma < 0$. Naudojama testo statistika:

$$DF_t = \frac{\hat{\gamma}}{SE(\hat{\gamma})} \tag{2.29}$$

lyginama, su atitinkama Dickey-Fuller testo, taikomo autoregresiniam modeliui, kritine reikšme [4].

2.4. SĄLYGINIO HETEROSKEDASTIŠKUMO MODELIAI

Praktikoje dažna situacija, kai modelio liekanos nors ir yra nekoreliuotos, tačiau jų dispersija nėra pastovi. Su tokia situacija dažnai susiduriama regresiniuose modeliuose. Tiesinėje regresijoje tokiu atveju taikomas apibendrintas mažiausių kvadratų metodas. Laiko eilučių analizėje nepastovios liekanų dispersijos atvejis yra taip pat labai dažnai. Pvz., jei analizuotume akcijų grąžas, tai pastebėtume, kad jos linkusios įgauti didesnes išsibarstymo reikšmes tam tikram laikotarpiui, o po to grąžos nusistovi ir jų kintamumas vėl įgauna ankstesnes nedideles reikšmes. Kyla natūralus noras tokį atsitiktinio proceso svyravimo pobūdį įtraukti į modelį [3].

Heteroskedastiškumo nagrinėjimui dažniausiai naudojami autoregresiniai sąlyginio heteroskedastiškumo (ARCH) ir apibendrinti sąlyginio heteroskedastiškumo modeliai (GARCH).

2.4.1. ARCH MODELIS

Sakysime, kad Y_t tenkina autoregresinį sąlyginio heteroskedastiškumo *ARCH(q)* (angl. autoregressive conditional heteroskedastic) modelį, jei

$$Y_t = \sigma_t \varepsilon_t, \tag{2.30}$$

kur $\varepsilon_t \sim N(0,1)$ ir nepriklausomi, o σ_t tenkina lygtį:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i Y_{t-i}^2 \tag{2.31}$$

Tam, kad šios lygtys tinkamai apibrėžtų atsitiktinį procesą, koeficientai α_i turi tenkinti tam tikras sąlygas, pvz. [3],

$$\alpha_1 + \cdots \alpha_q < 1, \alpha_i \geq 0. \tag{2.32}$$

ARCH(1) atvejis:

$$\begin{cases} Y_t = \sigma_t \varepsilon_t, \\ \sigma_t^2 = \alpha_0 + \alpha_1 Y_{t-1}^2, t \in \mathbb{Z} \end{cases} \tag{2.33}$$

Taikant šias formules rekurentiškai, turėsime

$$\sigma_t^2 = \alpha_0 + \alpha_1 Y_{t-1}^2 = \alpha_0 + \alpha_1 \sigma_{t-1}^2 \varepsilon_{t-1}^2 = \alpha_0 + \alpha_0 \alpha_1 \sigma_{t-2}^2 \varepsilon_{t-1}^2 = \cdots$$

$$= \alpha_0 + \alpha_0 \sum_{j=1}^{n} a_1^j \varepsilon_{t-1}^2 \dots \varepsilon_{t-j}^2 + a_1^{n+1} \varepsilon_{t-1}^2 \dots \varepsilon_{t-n}^2 Y_{t-n-1}^2.$$

Todėl galima spėti, kad, jeigu $\alpha_1 < 1$, tai lygčių (2.4.1.4) sprendinys turi pavidalą

$$Y_t^* = \sqrt{\sigma_t^{*2}} \varepsilon_t \text{ su } \sigma_t^{*2} = \alpha_0 + \alpha_0 \textstyle\sum_{j=1}^{n} a_1^j \varepsilon_{t-1}^2 \dots \varepsilon_{t-j}^2. \tag{2.34}$$

Pastebėsime, kad

$$Y_t^2 = \sigma_t^2 + Y_t^2 - \sigma_t^2 = \alpha_0 + \alpha_1 Y_{t-1}^2 + \sigma_t^2(\varepsilon_t - 1) = \alpha_0 + \alpha_1 Y_{t-1}^2 + \nu_t, \tag{2.35}$$

kur $\nu_t = \sigma_t^2(\varepsilon_t - 1)$ yra baltasis triukšmas. Taigi *ARCH(1)* procesas gali būti išreikštas kaip *AR(1)* modelis procesui Y_t^2, bendru atveju *ARCH(q)* į *AR(q)* [3].

Natūralu bandyti ARCH modelį išplėsti laikant, kad proceso dispersija gali priklausyti ne tik nuo proceso kvadrato reikšmių praeityje, bet ir nuo pačių proceso dispersijų praeityje [2].

2.4.2. GARCH MODELIS

Sakysime, kad Y_t tenkina autoregresinį sąlyginio heteroskedastiškumo *GARCH(p,q)* (angl. general autoregressive conditional heteroskedastic) modelį, jei

$$Y_t = \sigma_t \varepsilon_t, \tag{2.36}$$

kur $\varepsilon_t \sim N(0,1)$ ir nepriklausomi, o σ_t tenkina lygtį [3]:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i Y_{t-i}^2 + \sum_{i=1}^{p} \beta_i \sigma_{t-i}^2. \tag{2.37}$$

GARCH(1,1) atvejis:

$$\begin{cases} Y_t = \sigma_t \varepsilon_t, \\ \sigma_t^2 = \alpha_0 + \alpha_1 Y_{t-1}^2 + \beta_1 \sigma_{t-1}^2, t \in \mathbb{Z}' \end{cases} \tag{2.38}$$

čia ε_t yra nepriklausomų vienodai pasiskirsčiusių dydžių seka su parametrais (0,1).

$$Y_t^2 = \sigma_t^2 + (Y_t^2 - \sigma_t^2) = \alpha_0 + \alpha_1 Y_{t-1}^2 + \beta_1 \sigma_{t-1}^2 + Y_t^2 - \sigma_t^2 =$$

$$= \alpha_0 + (\alpha_1 + \beta_1) Y_{t-1}^2 - \beta_1 (Y_{t-1}^2 - \sigma_{t-1}^2) + Y_t^2 - \sigma_t^2 =$$

$$= \alpha_0 + (\alpha_1 + \beta_1) Y_{t-1}^2 - \beta_1 \nu_{t-1} + \nu_t; \tag{2.39}$$

čia $\nu_t = Y_t^2 - \sigma_t^2$. Taigi, panašiai kaip (2.4.1.6), kintamumas *GARCH(1,1)* modelyje gali būti interpretuojamas kaip *ARMA(1,1)* su triukšmu $\nu_t \sim BT(0, \sigma_\nu^2)$ [2].

2.4.3. ARCH IR GARCH PARAMETRŲ VERTINIMAS

Vienas iš didžiausių ARCH modelių pranašumų yra lengvai užrašomas daugiamatis stebėjimų tankis, todėl, norint vertinti parametrus, nesunku taikyti didžiausio tikėtinumo metodą. Bet kokiam atsitiktinių dydžių vektoriui $(X_1, \dots, X_n)$ jo tankis (jei jis egzistuoja ir yra teigiamas) gali būti užrašytas tokia sąlyginių tankių sandauga [2]:

$$f_{X_1,\dots,X_N}(x_1, \dots, x_n) = f_{X_1}(x_1) \prod_{i=2}^{n} f_{X_i|X_{i-1},\dots,X_1}(x_i|x_{i-1}, \dots, x_1). \tag{2.40}$$

ARCH(1) modelio $(X_t = Y_t)$ su $\varepsilon_t \sim N(0,1)$ atveju turime

$$f_{X_i|X_{i-1}}(x_i|x_{i-1}, \dots, x_1) = f_{X_i|X_{i-1},\dots,X_1}(x_i|x_{i-1}) = \frac{1}{\sqrt{2\pi(\alpha_0 + \alpha_1 x_{i-1}^2)}} exp\left\{-\frac{x_i^2}{2(\alpha_0 + \alpha_1 x_{i-1}^2)}\right\}. \tag{2.41}$$

Taigi duotam Y_1 sąlyginė tikėtinumo funkcija yra

$$L \equiv L(Y_2, \dots, Y_n; \alpha_0, \alpha_1) =$$

$$= \prod_{i=2}^{n} \frac{1}{\sqrt{2\pi\sigma_i^2}} exp\left\{-\frac{Y_i^2}{2\sigma_i^2}\right\} = \prod_{i=2}^{n} \frac{1}{\sqrt{2\pi(\alpha_0 + \alpha_1 Y_{i-1}^2)}} exp\left\{-\frac{Y_i^2}{2(\alpha_0 + \alpha_1 Y_{i-1}^2)}\right\}, \tag{2.42}$$

taigi

$$\ln(L) = -\frac{n-1}{2}\ln(2\pi) - \frac{1}{2}\sum_{i=2}^{n} \ln(\alpha_0 + \alpha_1 Y_{i-1}^2) - \frac{1}{2}\sum_{i=2}^{n} \frac{Y_i^2}{\alpha_0 + \alpha_1 Y_{i-1}^2}. \tag{2.43}$$

GARCH(1,1) atveju pirma pastebėsime, kad panašiai, kaip ARMA modelio atveju, galima gauti tokį σ_t^2 skleidinį begaline eilute:

$$\sigma_t^2 = (1 - \beta_1 B)^{-1}(\alpha_0 + \alpha_1 Y_{t-1}^2) = \frac{\alpha_0}{1 - \beta_1} + \alpha_1 \sum_{j=1}^{\infty} \beta_1^{j-1} Y_{t-j}^2. \tag{2.44}$$

Kadangi teoriškai kintamumas σ_t^2 priklauso nuo be galo daug praėjusių reikšmių Y_{t-1}^2, Y_{t-2}^2, ..., tai praktikoje vietoj Y_t^2 įvedami „nupjauti“ dydžiai $\tilde{Y}_t^2 = 0$ su $t \leq 0, \tilde{Y}_t^2 = Y_t^2$ su $t > 0$, o vietoj σ_t^2- rekurentiškai skaičiuojami dydžiai $\tilde{\sigma}_t^2$, $t = 1,2, \ldots$ [2]

$$\tilde{\sigma}_t^2 = \alpha_0 + \alpha_1 \tilde{Y}_{t-1}^2 + \beta_1 \tilde{\sigma}_{t-1}^2, \; \tilde{\sigma}_t^2 = 0, \; t \leq 0. \qquad \textbf{(2.45)}$$

Gauname tokią rekurentinę procedūrą:

$$\begin{aligned} &\tilde{\sigma}_1^2 = \alpha_0, \\ &\tilde{\sigma}_2^2 = \alpha_0 + \alpha_1 \tilde{Y}_1^2 + \beta_1 \tilde{\sigma}_1^2 = \alpha_0 + \alpha_0 \beta_1 + \alpha_1 \tilde{Y}_1^2, \\ &\tilde{\sigma}_3^2 = \alpha_0 + \alpha_1 \tilde{Y}_2^2 + \beta_1 \tilde{\sigma}_2^2 = \alpha_0 + \alpha_0 \beta_1 + \alpha_0 {\beta_1}^2 + \alpha_1 \beta_1 \tilde{Y}_2^2 + \alpha_1 \beta_1 \tilde{Y}_1^2, \\ &\quad \vdots \end{aligned} \qquad \textbf{(2.46)}$$

Įstatę išraiškas į tikėtinumo funkciją gauname:

$$L(\alpha_0, \alpha_1, \beta_1) = \prod_{i=2}^{n} \frac{1}{\sqrt{2\pi \tilde{\sigma}_i^2}} exp\left\{-\frac{Y_i^2}{2\tilde{\sigma}_i^2}\right\}. \qquad \textbf{(2.47)}$$

2.5. NORMALUMO KRITERIJUS

Liekamųjų paklaidų normalumui įvertinti dažniausiai naudojamas Jarque Bera (str. JB) kriterijus.

Nulinė hipotezė formuluojama taip H_0: liekamųjų paklaidų reikšmės pasiskirsčiusios pagal normalųjį skirstinį. JB kriterijaus statistika modeliui su laisvuoju nariu:

$$JB = (n - K)(\tfrac{S^2}{6} - \tfrac{(K-3)^2}{24}). \qquad \textbf{(2.48)}$$

Statistika pasiskirsčiusi pagal χ^2 su dviem laisvės laipsniais: $JB \sim \chi 2(2)$, kur S asimetrijos koeficientas, K – ekscesas [1].

2.6. ARCH TESTAS

ARCH testas yra naudojamas tikrinant modelio likučių dispersijos pastovumą. Testas yra atliekamas naudojant pagalbinę regresiją:

$$e_t^2 = c + \beta_1 \cdot e_{t-1}^2 + \nu_t, \;\; \nu_t \sim N(0, \sigma^2). \qquad \textbf{(2.49)}$$

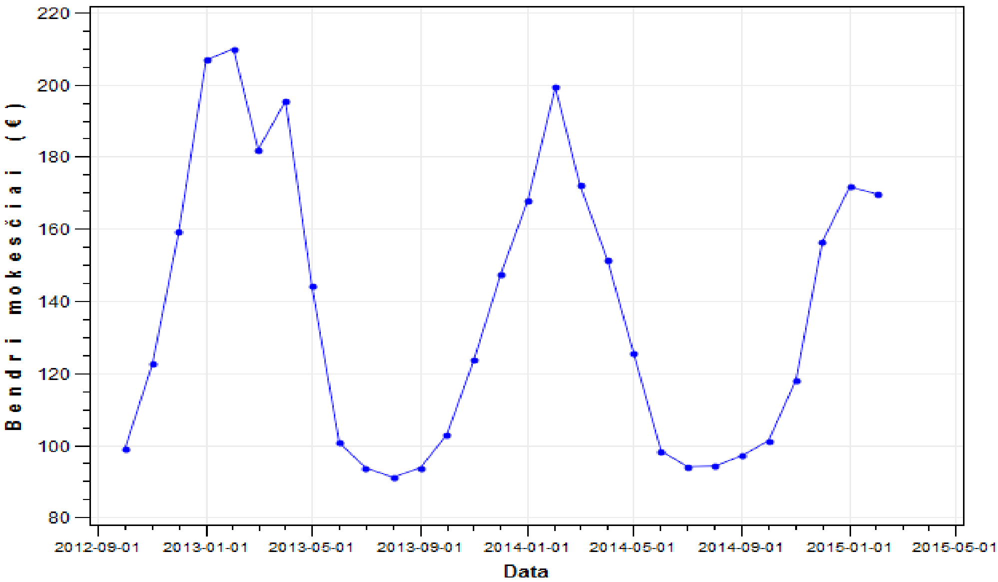
Atliekant testavimą yra tikrinama nulinė hipotezė : $H_o: \beta_1 = 0$ ir alternatyvi $H_a: \beta_1 \neq 0$. Jeigu nulinė hipotezė yra priimama, modelio paklaidų dispersija yra pastovi. SAS programinė įranga nulinei hipotezei tikrinti naudoja F - statistika [13].

3. TIRIAMOJI DALIS

Tiriamojoje dalyje sudaryti ARIMA, sezoniniai, adityvusis ir multiplikatyvusis laiko eilutės modeliai, patikrintas visų metodų adekvatumas bei esant liekanų heteroskedaštikumui papildomai taikomi GARCH modeliai paklaidoms. Atlikta kiekvieno jų prognozė bei lyginamoji analizė. ARIMA modeliams yra būdinga skaičiuoti tik trumpalaikes prognozes, dėl to sudaroma tik 5 mėnesių prognozė.

3.1. ARIMA MODELIAI VILNIAUS GYVENTOJŲ BŪSTO IŠLAIKYMO IŠLAIDOMS

Norint turimiems duomenims parinkti tinkamą modelį, pirmiausiai reikia pažiūrėti, kaip šie duomenys atrodo ir ką galima pasakyti iš pirminės jų grafinės analizės (3.1.1 pav.)



3.1.1 pav. Vilniaus gyventojų išlaidos, būsto išlaikymui

Iš pirmo žvilgsnio pakankamai sunku teigti, kad Vilniaus gyventojų išlaidų eilutė yra nestacionari. Kad būtume užtikrinti, panaudosime vienetinių šaknų ADF testą stacionarumui nustatyti.

3.1.1 lentelė

ADF stacionarumo testas Vilniaus gyventojų išlaidų eilutei

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	0.0539	0.6870	0.06	0.6934		
	1	-1.0721	0.4556	-0.64	0.4286		
	2	-1.1648	0.4399	-0.76	0.3761		
Single Mean	0	-5.5904	0.3477	-1.72	0.4090	1.64	0.6605
	1	-24.4721	0.0004	-3.27	0.0263	5.35	0.0369
	2	-70.7626	<.0001	-3.68	0.0105	6.77	0.0066

Išlaidų eilutės ADF testo p reikšmė – *0,3477* > 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą neatmetama, t. y. eilutė nėra stacionari. Dažnai stacionarumą gauti padeda eilutės diferenciajavimas. Šiuo atveju, sezoninio diferenciajavimo nenaudosime, nors ir žinoma, kad eilutės duomenys rinkti kas mėnesį. Eilutė turi tik 29 stebėjimus, o kaip jau buvo minėta anksčiau (žr. 1.2 sk.) rekomenduojama turėti 50 stebėjimų. Diferencijuojame eilutę vieną kartą ir vėlgi tikriname hipotezę apie eilutės stacionarumą.

3.1.2 lentelė

ADF stacionarumo testas Vilniaus gyventojų išlaidų eilutę diferencijavus vieną kartą

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-13.6096	0.0058	-3.03	0.0038		
	1	-15.5249	0.0028	-2.88	0.0056		
	2	-18.9808	0.0006	-2.90	0.0054		
Single Mean	0	-13.6582	0.0320	-2.96	0.0512	4.41	0.0746
	1	-15.4600	0.0167	-2.80	0.0714	3.99	0.0974
	2	-18.7972	0.0044	-2.83	0.0682	4.11	0.1014

Eilutės ADF testo p reikšmė < 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą atmetama, t.y. vieną kartą integruota eilutė yra stacionari.

Vieną kartą integruotos eilutės ACF išsiskiriantis stulpelis grafike (3.1.2 pav.) yra penktas, todėl patikrinami galimi modeliai iki *ARIMA(5,1,5).* Atsižvelgus į AIC įvertį, pasirinktas modelis yra *ARIMA(1,1,0)*, tačiau pagal mažiausią standartinį nuokrypį – *ARIMA(5,1,4).* Sudarysime abu modelius.

ARIMA(1,1,0):

$$Y_t = Y_{t-1} + \phi_1 Y_{t-1} + \phi_1 Y_{t-2} + \varepsilon_t$$

$$Y_t = Y_{t-1} - 0{,}496 Y_{t-1} - 0{,}496 Y_{t-2} + \varepsilon_t, \tag{3.1}$$

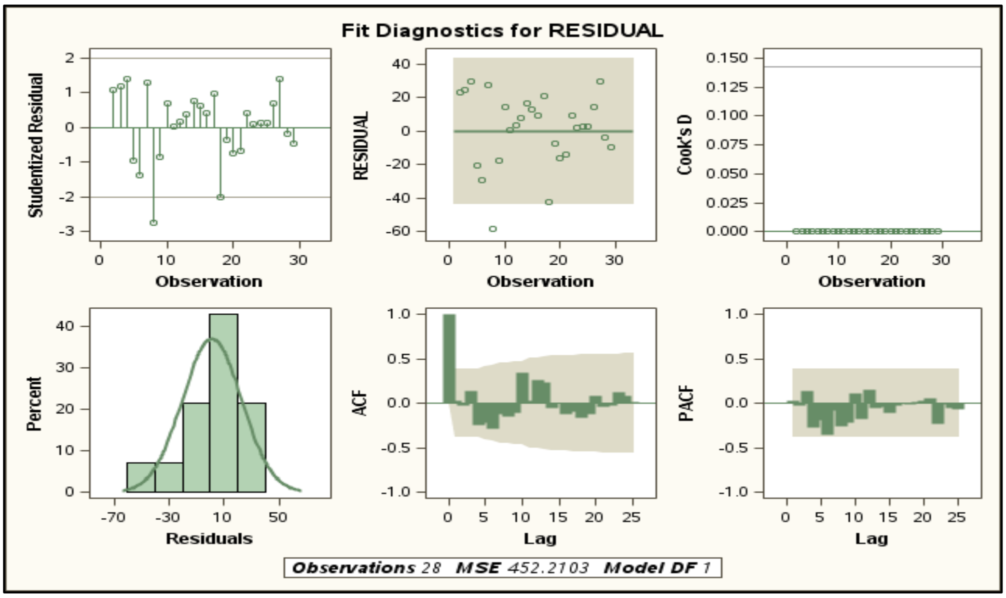
kur ε_t yra modelio liekanų procesas.

3.1.3 lentelė

***ARIMA(1,1,0)* modelio liekanų Box – Ljung testas**

Autocorrelation Check of Residuals									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	7.96	5	0.1587	0.037	-0.028	0.132	-0.254	-0.218	-0.294
12	19.04	11	0.0604	-0.115	-0.150	-0.108	0.340	0.030	0.264
18	26.17	17	0.0714	0.235	-0.056	-0.000	-0.128	-0.094	-0.156
24	33.40	23	0.0743	-0.125	0.085	-0.042	-0.015	0.133	0.093

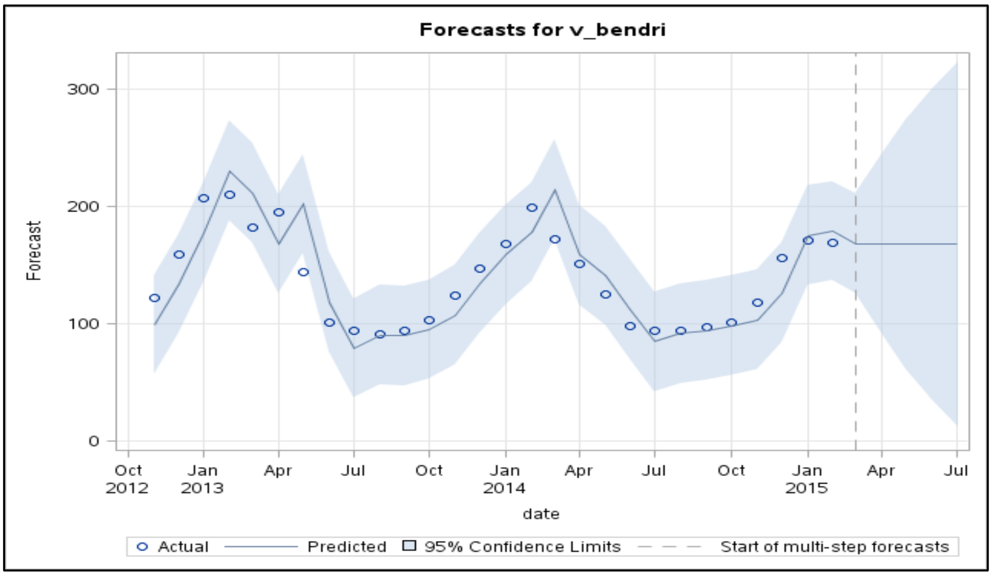
Apskaičiuotos *p* reikšmės visais atvejais yra didesnės už reikšmingumo lygmenį *α=0,05*, todėl nulinę hipotezę H_0: reikšmės yra neautokoreliuotos, priimame.



3.1.2 pav. Modelio *ARIMA (1,1,0)* liekanų grafinė analizė

Sudarytas modelis yra adekvatus, kadangi liekanos yra baltojo triukšmo procesas. Modelio liekanų grafinė analizė rodo, kad liekanos atitinka keliamus reikalavimus, t. y. liekanų reikšmės išsidėsčiusios apie nulinę tiesę, ACF grafike stulpeliai nekerta kritinės reikšmės ribų, Box – Ljung statistikos *p* reikšmės > 0,05. Taip pat, atlikus ARCH testą liekanoms (žr. 1 priedas), nulinė hipotezė apie ARCH poveikį priimama, todėl galima teigti, kad liekanos nėra heteroskedastiškos.

Prognozuojame Vilniaus gyventojų išlaidas, būsto išlaikymui, 5 mėnesiams.



3.1.3 pav. *ARIMA (1,1,0)* modelio prognozė

Iš 3.1.3 paveikslo matyti, kad prognozė neatspindi laiko eilutės. Išlaidų dydžiai yra pakankamai arti konstantos, dėl to teigti, kad prognozė yra tiksli negalima. Modelio vidutinė procentinė absoliutinė paklaida (MAPE) – 11,67%.

Sudarome *ARIMA (5,1,4)* modelį:

$$Y_t = Y_{t-1} + \phi_1(Y_{t-1} - Y_{t-2}) + \phi_2(Y_{t-2} - Y_{t-3}) + \phi_3(Y_{t-3} - Y_{t-4}) + \phi_4(Y_{t-4} - Y_{t-5}) + \\ + \phi_5(Y_{t-5} - Y_{t-6}) + \theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \theta_3\varepsilon_{t-3} + \theta_4\varepsilon_{t-4} + \varepsilon_t,$$

$$Y_t = Y_{t-1} + 0{,}07(Y_{t-1} - Y_{t-2}) - 0{,}951(Y_{t-2} - Y_{t-3}) + 0{,}008(Y_{t-3} - Y_{t-4}) + 0{,}745(Y_{t-4} - Y_{t-5}) + \\ + 0{,}185(Y_{t-5} - Y_{t-6}) + 0{,}407\varepsilon_{t-1} - 1{,}081\varepsilon_{t-2} - 0{,}268\varepsilon_{t-3} + 0{,}552\varepsilon_{t-4} + \varepsilon_t, \quad \textbf{(3.2)}$$

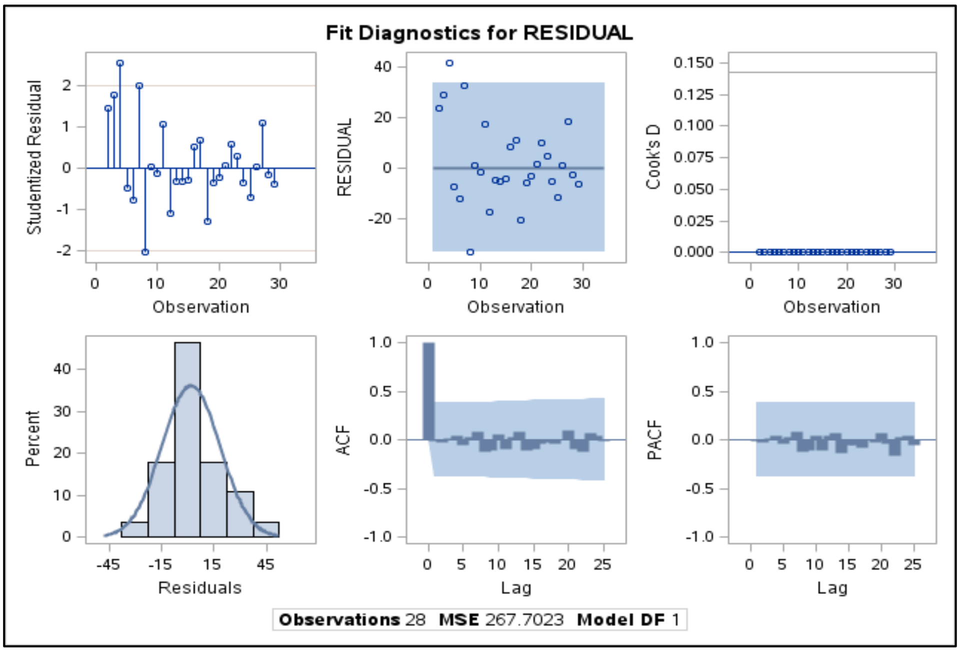
kur ε_t yra modelio liekanų procesas.

3.1.4 lentelė

ARIMA(5,1,4) **modelio liekanų Box – Ljung testas**

Autocorrelation Check of Residuals									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	.	0	.	-0.005	-0.023	0.002	0.033	-0.050	0.017
12	2.35	3	0.5029	0.084	-0.127	-0.116	0.053	-0.096	0.030
18	3.98	9	0.9130	0.083	-0.106	-0.087	-0.027	-0.011	-0.014
24	9.64	15	0.8415	0.016	0.117	-0.084	-0.101	0.080	0.064

Apskaičiuotos *p* reikšmės visais atvejais yra didesnės už reikšmingumo lygmenį *α=0,05*, todėl nulinę hipotezę H_0: reikšmės yra neautokoreliuotos, priimame.



3.1.4 pav. Modelio *ARIMA (5,1,4)* liekanų grafinė analizė

Sudarytas modelis yra adekvatus, kadangi liekanos yra baltojo triukšmo procesas. Modelio liekanų grafinė analizė rodo, kad liekanų reikšmės išsidėsčiusios apie nulinę tiesę, ACF grafike stulpeliai nekerta kritinės reikšmės ribų, Box – Ljung statistikos *p* reikšmės > 0,05 (modelio liekanos neautokoreliuotos). Atlikus ARCH testą liekanoms (žr. 1 priedas), nulinė hipotezė apie ARCH poveikį atmetama, todėl galima teigti, kad liekanų dispersija nėra pastovi.

Modelio *ARIMA (5,1,4)* liekanoms taikome GARCH modelius:

$$\varepsilon_t = \sigma e_t,$$

kai

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{i=1}^{p} \beta_i \sigma_{t-i}^2.$$

Remiantis AIC ir BIC įverčiu pasirinktas modelis yra *ARCH(1)*.

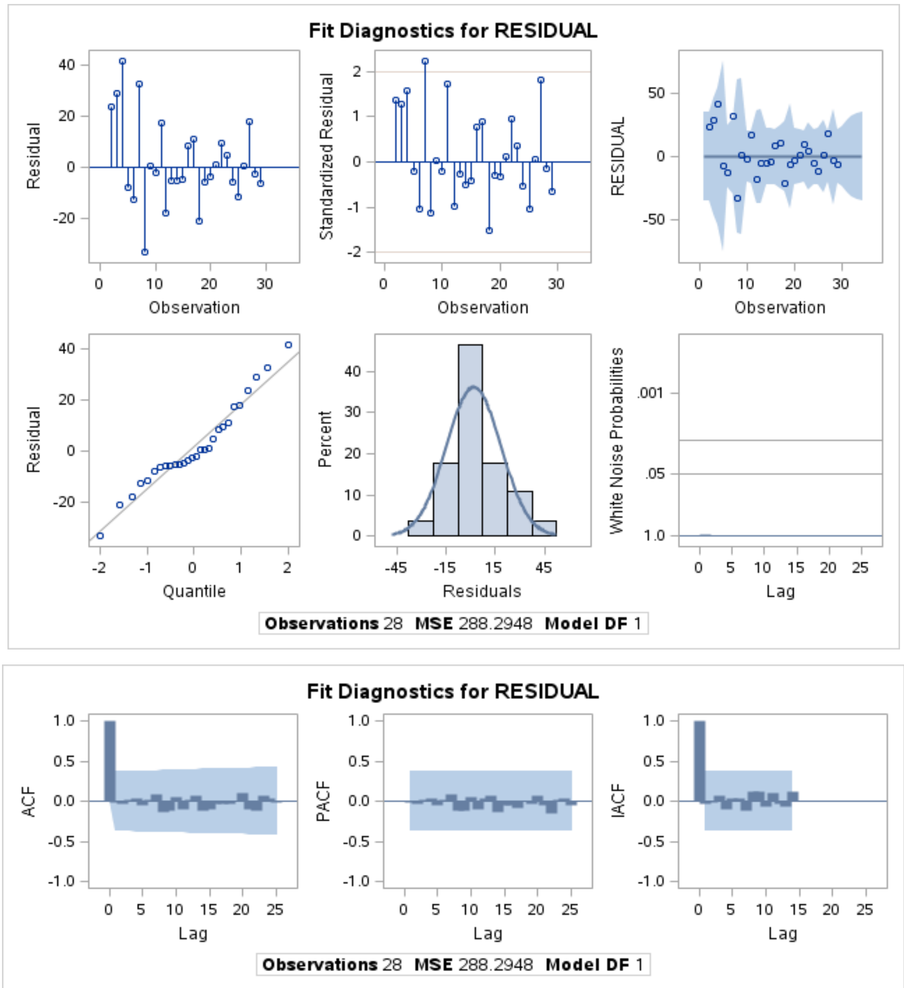
ARCH(1) modelis:

$$\varepsilon_t = \sigma e_t,$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-i}^2$$

$$\sigma_t^2 = 98{,}408 + 0{,}731 \varepsilon_{t-i}^2. \quad \textbf{(3.3)}$$

e_t – balto triukšmo komponentė.



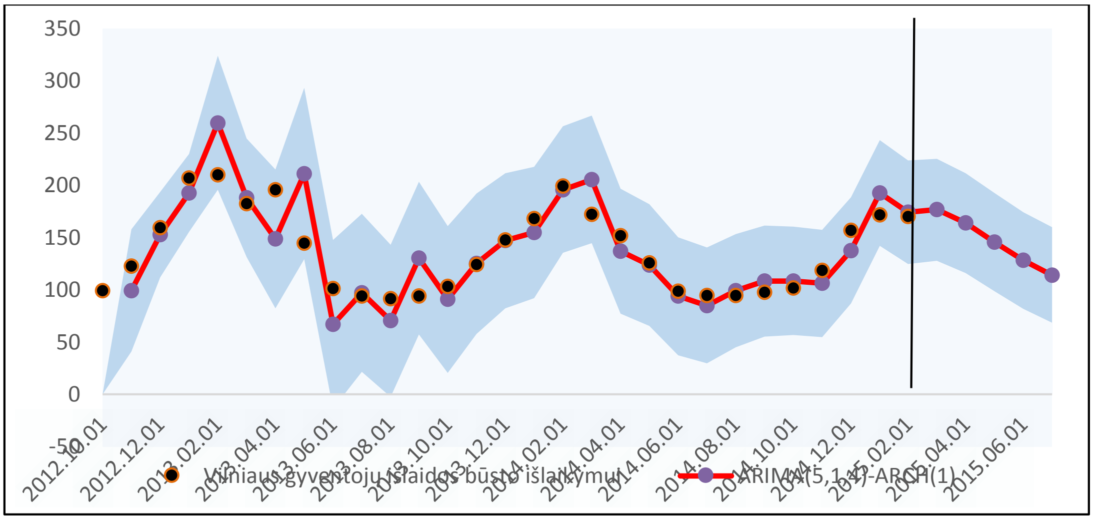
3.1.5 pav. Modelio *ARIMA(5,1,4)* liekanų ARCH modelio grafinė analizė

Sudarytas modelis ARCH liekanoms yra geras, kadangi liekanos yra baltojo triukšmo procesas. Modelio liekanų grafinė analizė rodo, kad liekanų reikšmės išsidėsčiusios apie nulinę tiesę, ACF grafike stulpeliai nekerta kritinės reikšmės ribų, baltojo triukšmo tikimybė arti vieneto.

Sudarytas modelis papildomas *ARCH(1)* dalimi*:*

$$Y_t = Y_{t-1} + 0{,}07(Y_{t-1} - Y_{t-2}) - 0{,}951(Y_{t-2} - Y_{t-3}) + 0{,}008(Y_{t-3} - Y_{t-4}) + 0{,}745(Y_{t-4} - Y_{t-5}) + \\ +0{,}185(Y_{t-5} - Y_{t-6}) + 0{,}40692\varepsilon_{t-1} - 1{,}08083\varepsilon_{t-2} - 0{,}26790\varepsilon_{t-3} + 0{,}55167\varepsilon_{t-4} + \\ +\sqrt{98{,}408 + 0{,}731\varepsilon_{t-i}^2}\, e_t, \qquad \textbf{(3.4)}$$

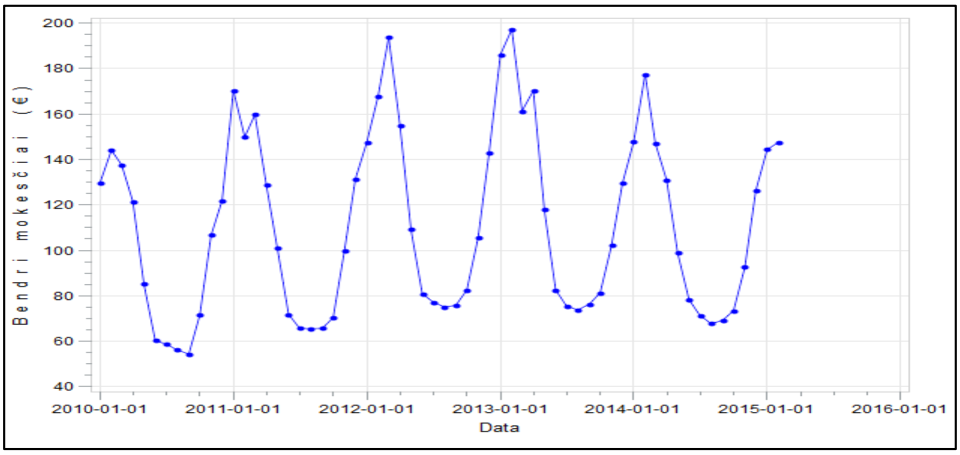
kur e_t baltasis triukšmas.



3.1.6 pav. *ARIMA (5,1,4)* modelio prognozė su ARCH(1) liekanomis

Modelio vidutinė procentinė absoliutinė paklaida (MAPE) - 12,228%.

3.2. ARIMA MODELIAI KAUNO GYVENTOJŲ BŪSTO IŠLAIKYMO IŠLAIDOMS



3.2.1 pav. Kauno gyventojų išlaidos, būsto išlaikymui

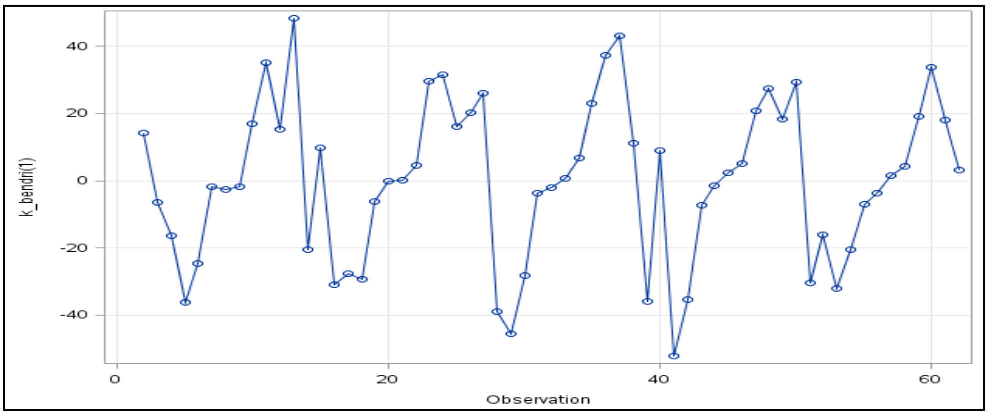
Iš pirmo žvilgsnio vėlgi negalima pasakyti, kad turima eilutė nėra stacionari. Kad būtume užtikrinti, panaudosime vienetinių šaknų ADF testą stacionarumui nustatyti.

3.2.1 lentelė

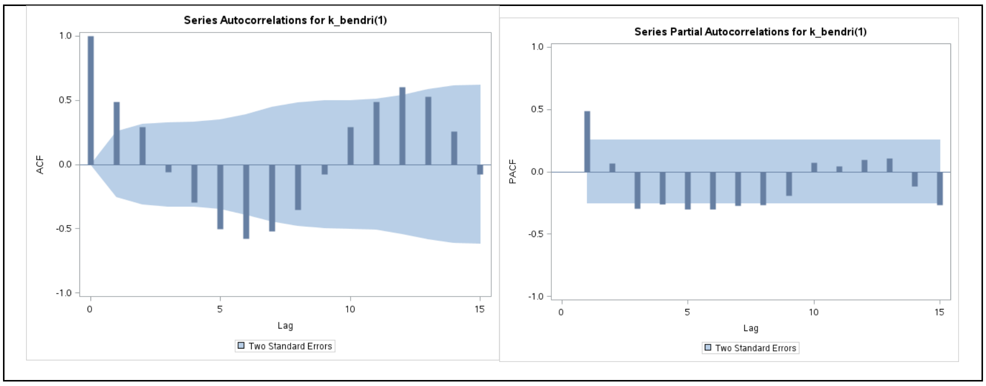
ADF stacionarumo testas Kauno gyventojų išlaidų eilutei

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-1.0489	0.4613	-0.67	0.4246		
	1	-3.8276	0.1748	-1.37	0.1558		
	2	-4.4061	0.1448	-1.38	0.1538		
Single Mean	0	-10.4066	0.1068	-2.31	0.1728	2.67	0.3994
	1	-47.8857	0.0006	-4.77	0.0003	11.37	0.0010
	2	-560.366	0.0001	-6.66	0.0001	22.20	0.0010

Išlaidų eilutės ADF testo p=0,1728 > $\alpha=0,05$, todėl hipotezė apie vienetinės šaknies egzistavimą neatmetama, t. y. eilutė nėra stacionari. Taip, kaip ir Vilniaus atveju, eilutę diferencijuosime. Kadangi iš grafinės analizės matyti ryškus sezoniniai svyravimai, be paprasto diferencijavimo naudosime sezoninį – kas 12 mėnesių.



3.2.2 pav. Kauno gyventojų išlaidų eilutė diferencijuota vieną kartą



3.2.3 pav. Vieną kartą diferencijuotos Kauno gyventojų išlaidų eilutės autokoreliacijos ir dalinės autokoreliacijos grafikai

Vieną kartą diferencijuota 3.2.2 pav. eilutė labai panaši į stacionarią. Jos ACF ir PACF grafikuose reikšmingumo lygmenį kerta keli stulpeliai, tačiau tai nėra priežastis atmesti stacionarumo hipotezę. Atliekame ADF testą integruotai eilutei.

3.2.2 lentelė

ADF stacionarumo testas vieną kartą diferencijuotai Kauno gyventojų išlaidų eilutei

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-30.6475	<.0001	-4.52	<.0001		
	1	-26.1982	<.0001	-3.55	0.0006		
	2	-60.6010	<.0001	-4.33	<.0001		
Single Mean	0	-30.6459	0.0006	-4.48	0.0006	10.02	0.0010
	1	-26.1971	0.0010	-3.52	0.0106	6.20	0.0117
	2	-60.5066	0.0005	-4.29	0.0011	9.23	0.0010

3.2.2 lentelėje pateikta išplėstinio Dickey-Fuller stacionarumo hipotezė, kai eilutė buvo diferencijuota vieną kartą, neatmetama ($p<\alpha$), todėl hipotezė apie vienetinės šaknies egzistavimą atmetama, t. y. vieną kartą integruota eilutė yra stacionari.

Palyginus modelių AIC, BIC, SSE, taip pat panaudojus minimalios informacijos kriterijų, buvo pasirinktas modelis *ARIMA(3,1,1):*

$$Y_t = Y_{t-1} - 1{,}077(Y_{t-1} - Y_{t-2}) + 0{,}0198(Y_{t-2} - Y_{t-3}) + 0{,}451(Y_{t-3} - Y_{t-4}) - 0{,}953\varepsilon_{t-1} + \varepsilon_t, \quad \textbf{(3.5)}$$

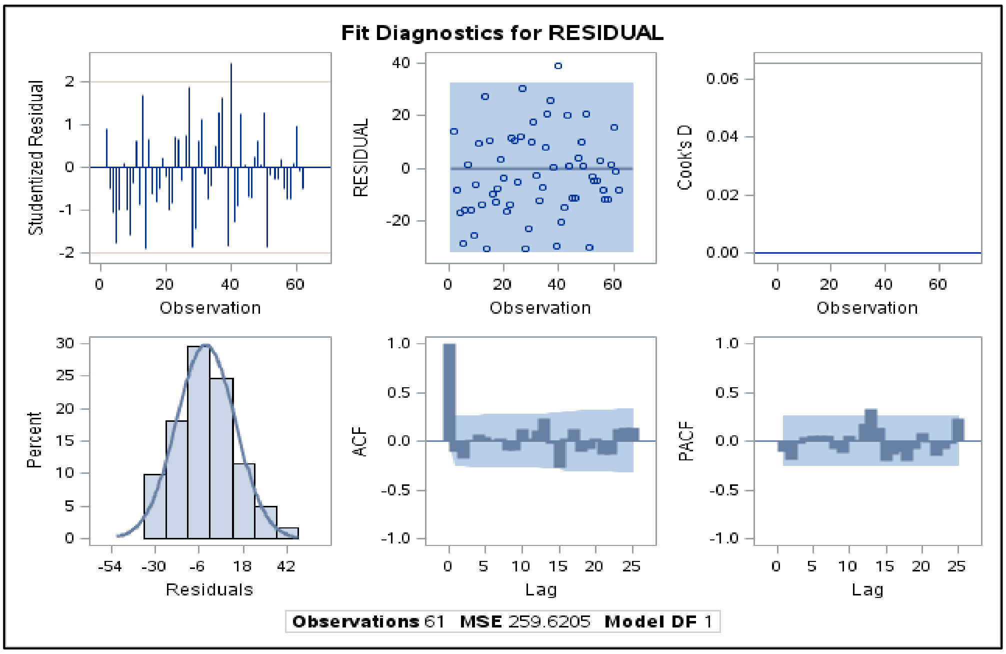
kur ε_t yra modelio liekanų procesas.

3.2.3 lentelė

***ARIMA(3,1,1)* modelio liekanų Box – Ljung testas**

Autocorrelation Check of Residuals									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	2.84	2	0.2417	-0.082	-0.150	0.042	0.087	0.058	0.034
12	6.33	8	0.6103	0.036	-0.083	-0.088	0.124	0.034	0.118
18	18.29	14	0.1940	0.228	-0.025	-0.254	0.038	0.122	-0.103
24	26.20	20	0.1592	-0.083	0.027	-0.133	-0.136	0.131	0.137

Apskaičiuotos p reikšmės visais atvejais yra didesnės už reikšmingumo lygmenį $\alpha=0{,}05$, todėl nulinę hipotezę H_0: reikšmės yra neautokoreliuotos, priimame.



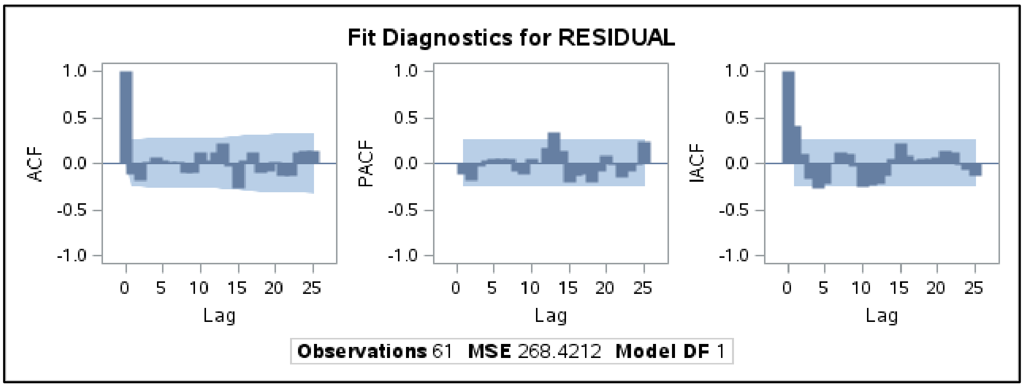
3.2.4. pav. Modelio *ARIMA (3,1,1)* liekanų grafinė analizė

Sudarytas modelis yra adekvatus, kadangi liekanos yra baltojo triukšmo procesas. Modelio liekanų grafinė analizė rodo, kad liekanų reikšmės išsidėsčiusios apie nulinę tiesę, ACF grafike stulpeliai nekerta kritinės reikšmės ribų, Box – Ljung statistikos *p* reikšmės > 0,05. Atlikus ARCH testą liekanoms (žr. 1 priedas), nulinė hipotezė apie ARCH poveikį atmetama, todėl negalima teigti, kad liekanos nėra heteroskedastiškos.

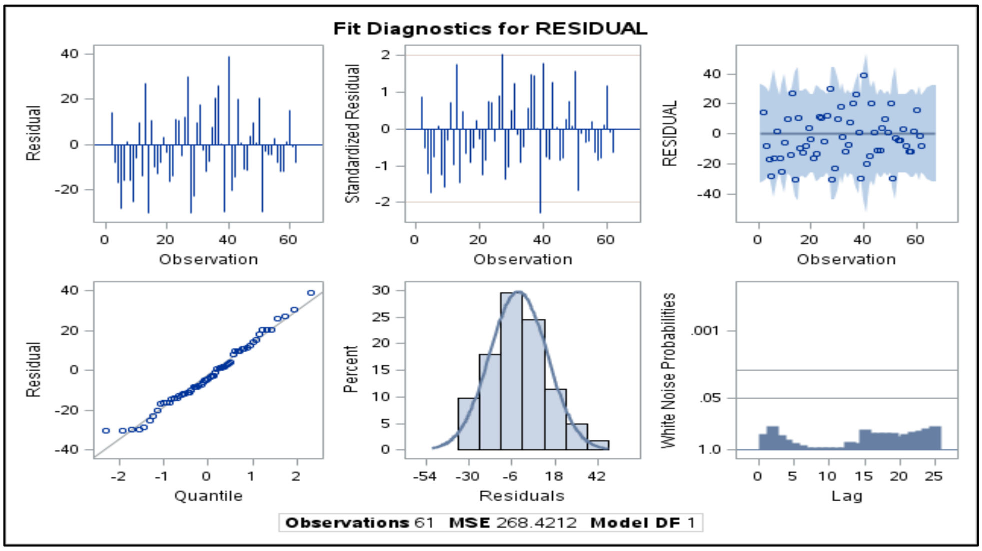
Modelio *ARIMA (3,1,1)* paklaidoms taikome GARCH modelius. Remiantis AIC ir BIC įverčiu pasirinktas modelis yra *ARCH(1)*.

ARCH(1) modelis:

$$\varepsilon_t = \sigma e_t, \qquad \sigma_t^2 = 165{,}927 + 0{,}356\varepsilon_{t-i}^2. \tag{3.6}$$



3.2.5 pav. Modelio *ARIMA(3,1,1)* liekanų ACF ir PACF



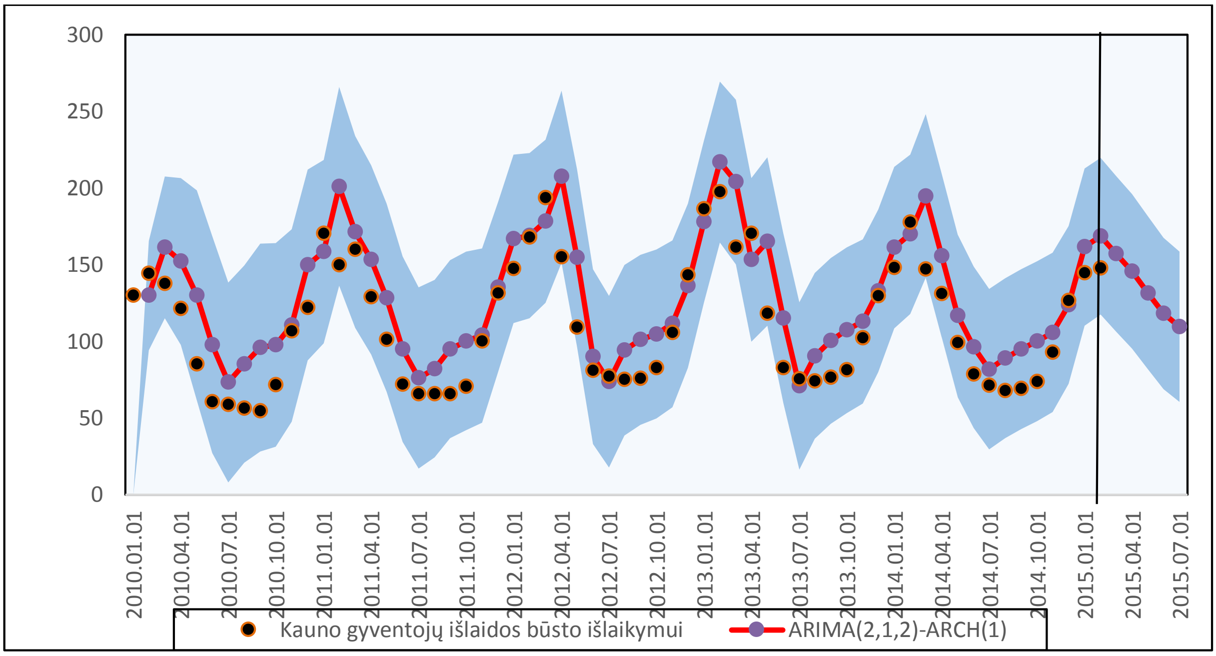
3.2.6 pav. Modelio *ARIMA(3,1,1)* liekanų ARCH modelio grafinė analizė

Sudarytas modelis ARCH liekanoms yra geras, kadangi liekanos yra baltojo triukšmo procesas. Modelio liekanų grafinė analizė rodo, kad liekanų reikšmės išsidėsčiusios apie nulinę tiesę, ACF grafike stulpeliai nekerta kritinės reikšmės ribų, baltojo triukšmo tikimybė arti vieneto.

Sudarytas modelis papildomas *ARCH(1)* dalimi*:*

$$Y_t = Y_{t-1} - 1{,}077\,(Y_{t-1} - Y_{t-2}) + 0{,}0198(Y_{t-2} - Y_{t-3}) + 0{,}451(Y_{t-3} - Y_{t-4}) - 0{,}953\varepsilon_{t-1} + \\ + \sqrt{165{,}927 + 0{,}356\varepsilon_{t-i}^2} \cdot e_t, \qquad \textbf{(3.7)}$$

kur $\mathrm{e_t}$ - baltasis triukšmas.



3.2.7 pav. *ARIMA (3,1,1)* modelio prognozė su *ARCH(1)* liekanomis

Modelio vidutinė procentinė absoliutinė paklaida (MAPE) - 21,824%.

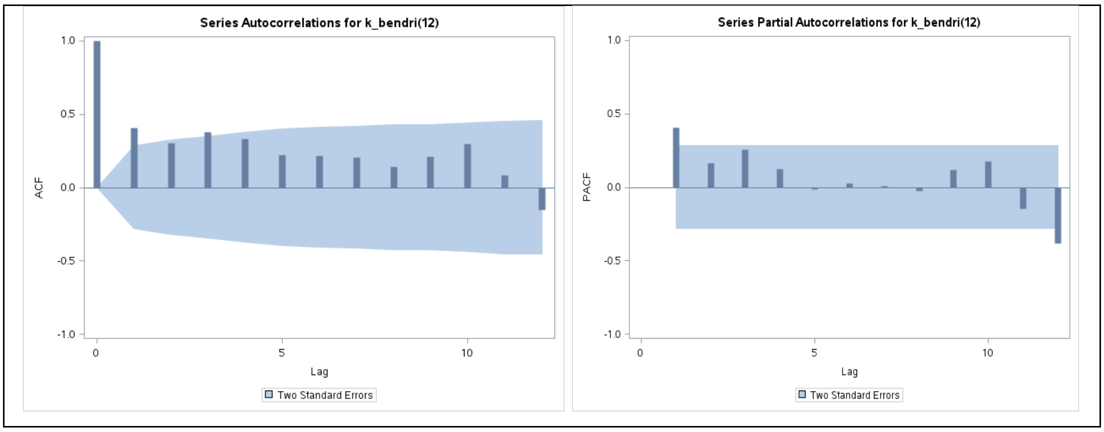
Toliau sudarysime sezoninį ARIMA modelį *SARIMA(p,d,q)(P,D,Q)*$_s$. Integruojame eilutę sezoniškai su sezoniškumo parametru *s=12* ir vėlgi tikriname hipotezę apie eilutės stacionarumą.

3.2.4 lentelė

ADF stacionarumo testas sezoniškai diferencijuotai Kauno gyventojų išlaidų eilutei

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-27.6297	<.0001	-4.48	<.0001		
	1	-18.5242	0.0015	-2.93	0.0042		
	2	-10.6165	0.0196	-2.20	0.0283		
Single Mean	0	-27.6559	0.0005	-4.39	0.0009	9.85	0.0010
	1	-18.4909	0.0095	-2.86	0.0577	4.20	0.0858
	2	-10.2162	0.1077	-2.08	0.2525	2.43	0.4673

Integruotos eilutės ADF testo *p* reikšmė $< 0{,}05$, todėl hipotezė apie vienetinės šaknies egzistavimą atmetama, t. y. eilutė yra stacionari.

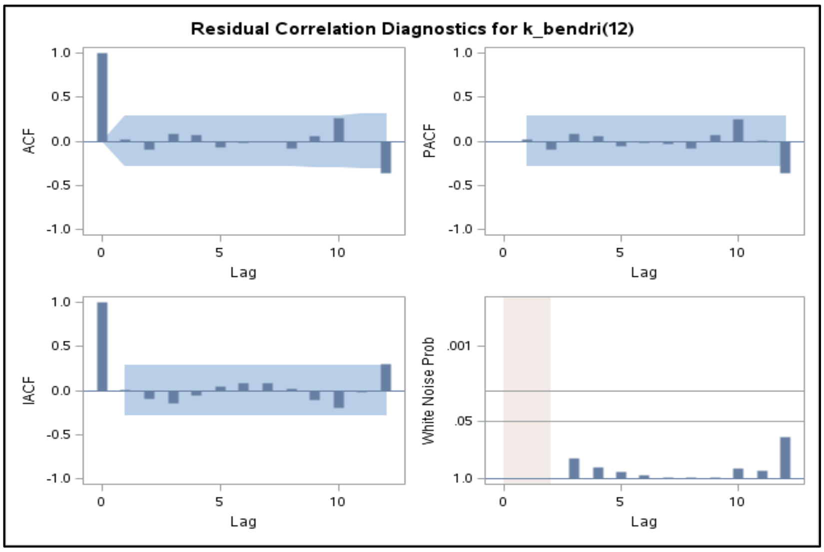


3.2.8 pav. Sezoniškai integruotos Kauno gyventojų išlaidų eilutės autokoreliacijos ir dalinės autokoreliacijos grafikai

Kadangi ACF ir PACF grafikuose išsiskiria 3 stulpelis, tikrinami galimi modeliai iki *SARIMA (3,0,3)(0,1,0)*$_{12}$. Palyginus modelių AIC, BIC, MSE, mažiausias reikšmes turėjo modelis *SARIMA* (1,0,1)(0,1,0)$_{12}$:

$$Y_t = -\,0{,}917Y_{t-1} - Y_{t-12} + 0{,}917Y_{t-13} + 0{,}666\varepsilon_{t-1} + \varepsilon_t, \qquad \textbf{(3.8)}$$

kur ε_t yra modelio liekanų procesas.



3.2.9. pav. Modelio *SARIMA (1,0,1)(0,1,0)₁₂* liekanų grafinė analizė

Modelis yra adekvatus. Modelio liekanų grafinė analizė rodo, kad liekanos atitinka keliamus reikalavimus (3.2.9 pav.):

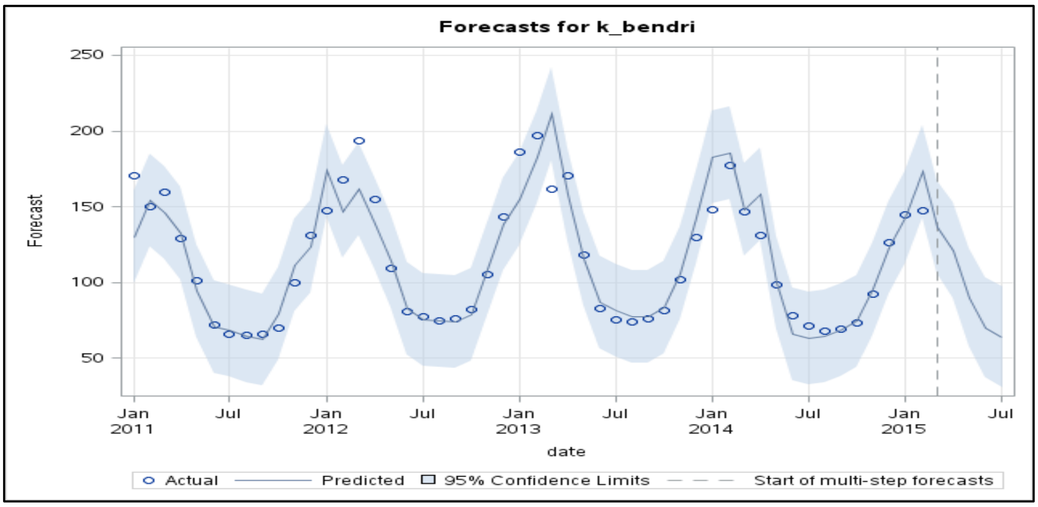
- liekanų reikšmės išsidėsčiusios apie nulinę tiesę t. y. liekanų vidurkis artimas 0;
- ACF grafike stulpeliai nekerta kritinės reikšmės ribų;

3.2.5 lentelė

***SARIMA (1,0,1)(0,1,0)₁₂* modelio liekanų Box – Ljung testas**

Autocorrelation Check of Residuals									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	1.50	4	0.8260	0.023	-0.098	0.082	0.071	-0.069	-0.020
12	15.62	10	0.1111	-0.008	-0.087	0.063	0.262	-0.002	-0.361
18	23.82	16	0.0935	0.168	0.221	-0.181	-0.024	0.052	-0.007
24	28.68	22	0.1543	0.031	0.055	-0.088	-0.156	0.121	0.036

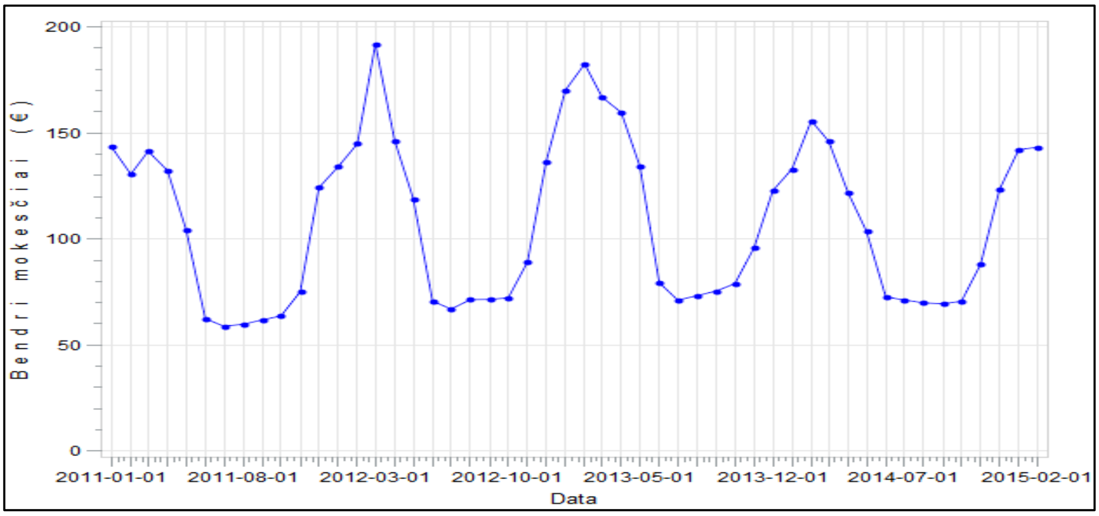
- Box – Ljung statistikos *p* reikšmės > 0,05 – liekanos nekoreliuotos (žr. 3.2.5 lentelė). ARCH testo nulinė hipotezė priimama.



3.2.10. pav. Modelio *SARIMA (1,0,1)(0,1,0)₁₂* prognozė

Modelio vidutinė procentinė absoliutinė paklaida (MAPE) - 7,582%.

3.3. ARIMA MODELIAI KLAIPĖDOS GYVENTOJŲ BŪSTO IŠLAIKYMO IŠLAIDOMS



3.3.1 pav. Klaipėdos gyventojų išlaidos, būsto išlaikymui

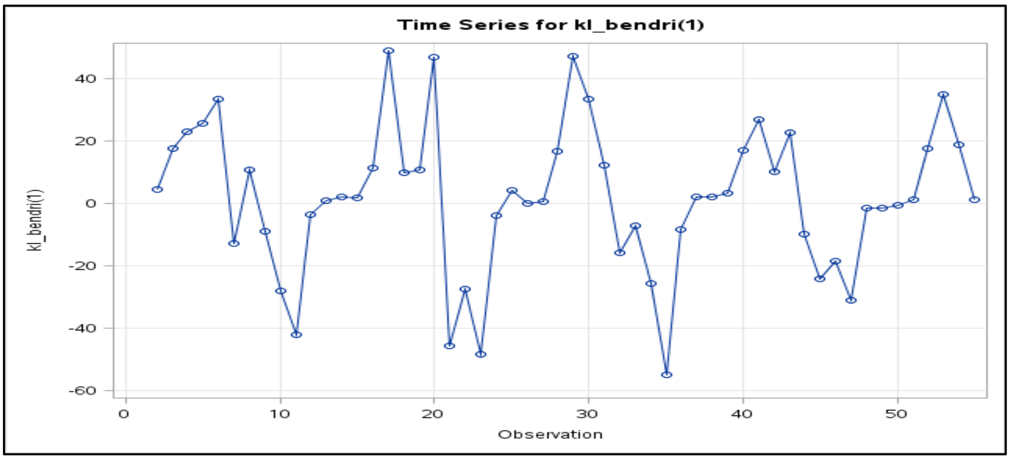
Sunku pažvelgus teigti, kad eilutė yra nestacionari. Nustatyti eilutės stacionarumui panaudosime vienetinių šaknų ADF testą (žr. 3.3.1 lentelė).

3.3.1 lentelė

ADF stacionarumo testas Klaipėdos gyventojų išlaidų eilutei

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-0.4107	0.5862	-0.26	0.5866		
	1	-2.5757	0.2669	-0.96	0.2955		
	2	-2.8775	0.2401	-1.02	0.2737		
Single Mean	0	-10.6357	0.0985	-2.51	0.1184	3.36	0.2263
	1	-40.1102	0.0005	-4.52	0.0006	10.30	0.0010
	2	-145.145	0.0001	-5.51	0.0001	15.20	0.0010

Pagal ADF testo *p* reikšmė - *0,1184* > 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą neatmetama, t. y. eilutė nėra stacionari. Kaip ir Kauno gyventojų išlaidų, būsto išlaikymui, eilutės atveju, eilutę diferencijuosime paprastai ir sezoniškai.



3.3.2 pav. Klaipėdos gyventojų išlaidų laiko eilutė integruotą vieną kartą

3.3.2 lentelė

ADF stacionarumo testas diferencijuotai Klaipėdos gyventojų išlaidų laiko eilutei

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-28.7049	<.0001	-4.40	<.0001		
	1	-27.8120	<.0001	-3.68	0.0004		
	2	-41.9870	<.0001	-3.77	0.0003		
Single Mean	0	-28.8936	0.0005	-4.37	0.0009	9.56	0.0010
	1	-28.0076	0.0005	-3.65	0.0077	6.66	0.0018
	2	-42.1997	0.0004	-3.74	0.0061	6.98	0.0010

3.3.2 lentelėje pateikta išplėstinio Dickey-Fuller stacionarumo hipotezė, kai Klaipėdos gyventojų išlaidų laiko eilutė buvo diferencijuota vieną kartą, neatmetama ($p<\alpha$). Laiko eilutė yra stacionari.

Vienas mažiausių AIC, BIC, SSE reikšmių gavome su *ARIMA (2,1,2):*

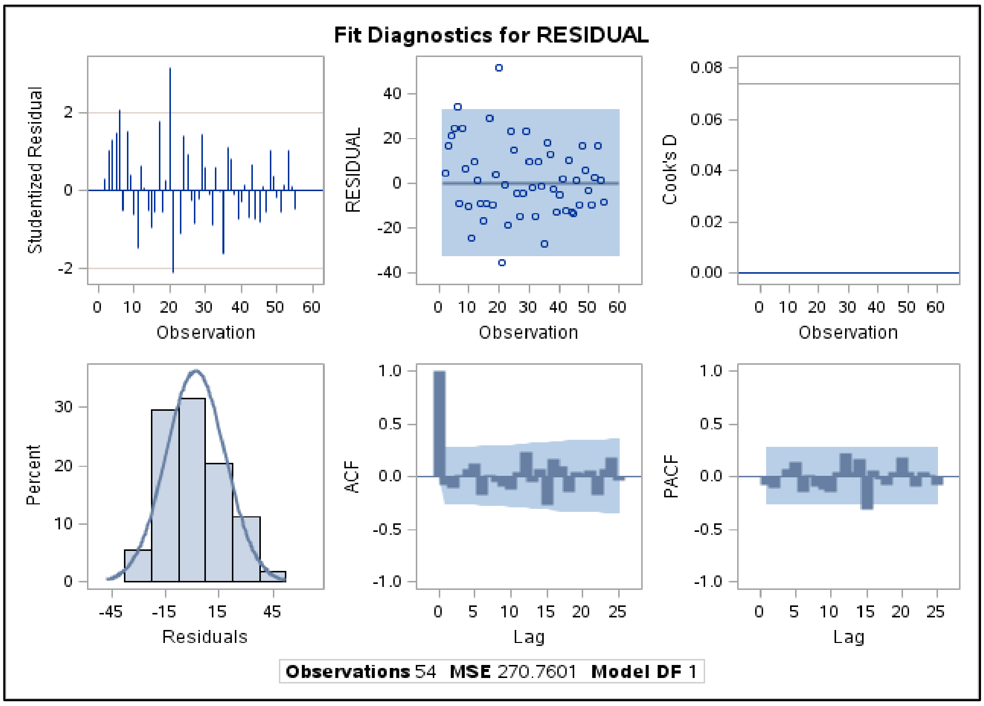
$$Y_t = Y_{t-1} - 1{,}7295\,(Y_{t-1} - Y_{t-2}) + 0{,}984(Y_{t-2} - Y_{t-3}) + \ 1.622\varepsilon_{t-1} - 0{,}785\varepsilon_{t-2} + \varepsilon_t \quad \textbf{(3.9)}$$

3.3.3 lentelė

***ARIMA(2,1,2)* modelio liekanų Box – Ljung testas**

Autocorrelation Check of Residuals									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	4.03	2	0.1336	-0.059	-0.089	0.021	0.079	0.125	-0.178
12	9.69	8	0.2875	0.019	-0.053	-0.095	-0.114	0.039	0.231
18	20.14	14	0.1259	-0.051	0.082	-0.253	0.168	0.111	-0.135
24	27.02	20	0.1347	0.038	0.036	0.060	-0.159	0.072	0.185

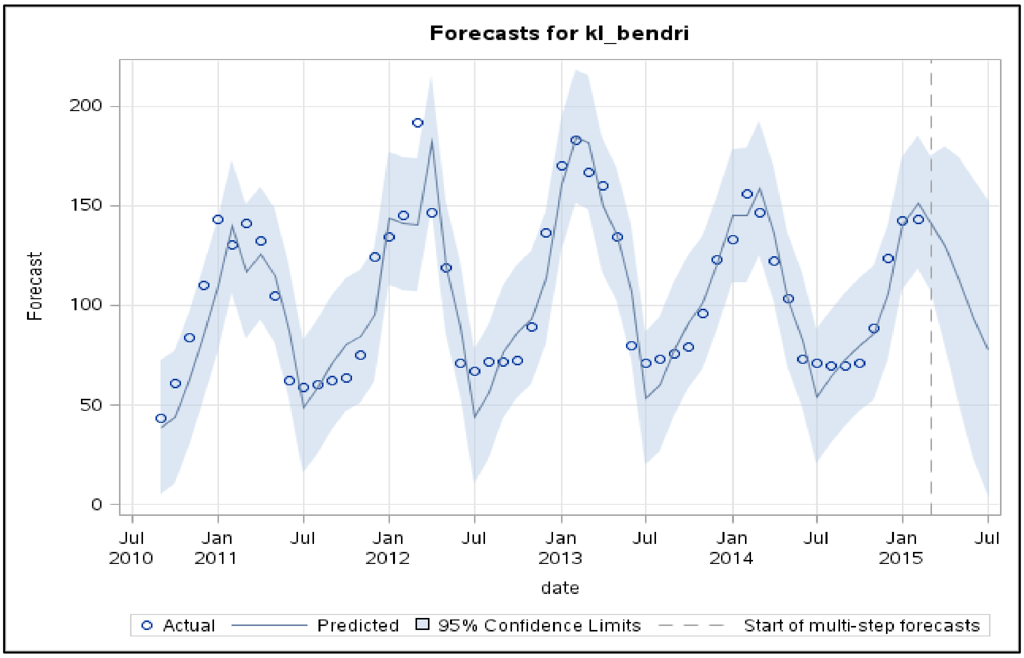
Box – Ljung testo *p* reikšmės didesnės už 0,05, dėl to hipotezė apie liekanų nekoreliuotumą priimama.



3.3.3. pav. Modelio *ARIMA(2,1,2)* modelio liekanų grafinė analizė

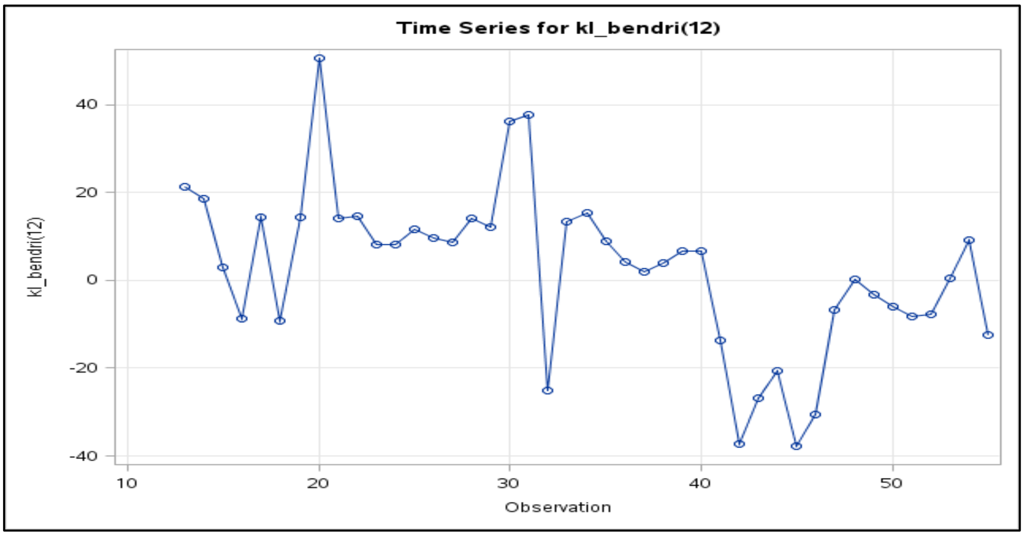
Grafinė modelio liekanų analizė rodo (3.3.3. pav.), kad liekanos išsidėsčiusios apie nulinę tiesę, ACF grafike stulpeliai nekerta kritinės reikšmės ribų. Tai reiškia, kad pasirinkto modelio liekanos yra baltojo triukšmo procesas. ARCH testo liekanoms (žr. 1 priedas) nulinė hipotezė apie ARCH poveikį priimama, todėl galima teigti, kad liekanų dispersija yra pastovi.

Klaipėdos gyventojų išlaidų laiko eilutei atliekame modelio *ARIMA(2,1,2)* prognozę.



3.3.4. pav. Modelio *ARIMA(2,1,2)* prognozė

Modelio vidutinė procentinė absoliutinė paklaida (MAPE) - 13,582%.



3.3.5 pav. Klaipėdos gyventojų išlaidų laiko eilutė integruota sezoniškai vieną kartą

Diferencijavus sezoniškai Klaipėdos gyventojų išlaidų, būsto išlaikymui, laiko eilutę, gaunama nauja eilutė (3.3.5 pav.). Eilutės stacionarumą patikrinsime panaudoję ADF vienetinių šaknų testą, kadangi, iš grafinės analizės tai konstatuoti - sunku.

3.3.4 lentelė

ADF stacionarumo testas sezoniškai diferencijuotai Klaipėdos gyventojų išlaidų laiko eilutei

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-20.0710	0.0008	-3.64	0.0005		
	1	-17.6810	0.0018	-2.98	0.0038		
	2	-8.5568	0.0370	-1.93	0.0522		
Single Mean	0	-20.3030	0.0048	-3.59	0.0098	6.50	0.0116
	1	-17.8739	0.0105	-2.93	0.0503	4.35	0.0779
	2	-8.5959	0.1633	-1.88	0.3374	1.81	0.6198

Išplėstinis Dickey-Fuller stacionarumo testas (3.3.4 lentelė) parodė, kad mūsų gautoji laiko eilutė yra stacionari. Diferencijuotos laiko eilutės ADF testo *p* reikšmė - 0,0098 < 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą atmetama. Mažiausias Akaike, Bajeso ir SSE reikšmes turi modelis *SARIMA (1,0,0)(0,1,0)₁₂*, todėl būtent jį ir sudarome:

$$Y_t = Y_{t-12} - 0{,}5221(Y_{t-1} - Y_{t-13}) + \varepsilon_t \quad \textbf{(3.10)}$$

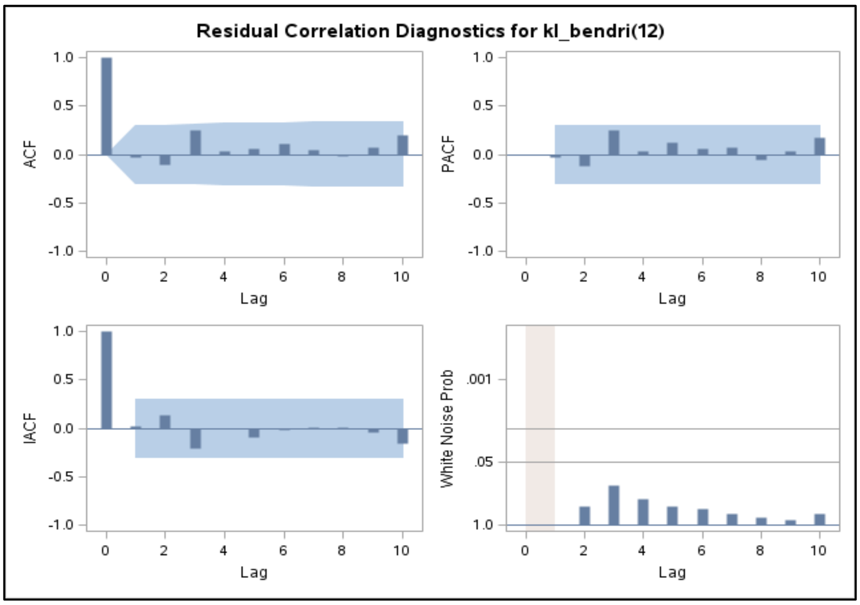
Modelio liekanų grafinė analizė ir Box – Ljung statistikos testas rodo, kad liekanos atitinka keliamus reikalavimus, t. y.

- liekanų reikšmės išsidėsčiusios apie nulinę tiesę (liekanų vidurkis artimas 0);
- ACF grafike stulpeliai nekerta kritinės reikšmės ribų (stacionaru);
- Box – Ljung testo *p* reikšmė > 0,05, todėl hipotezė apie liekanų nekoreliuotumą neatmetama, t. y. liekanos yra nekoreliuotos.

3.3.5 lentelė

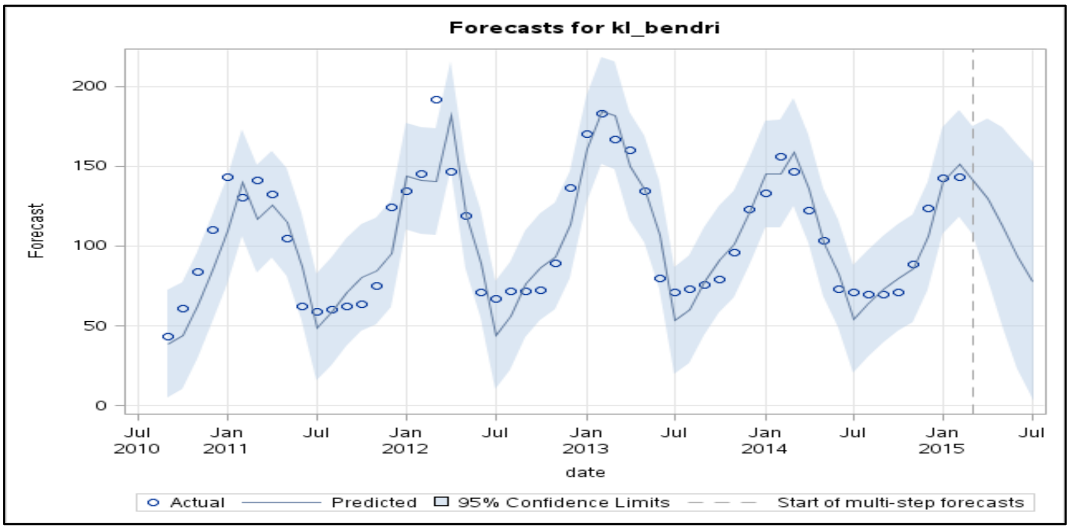
***SARIMA(1,0,0)(0,1,0)$_{12}$* modelio Box – Ljung testas**

Autocorrelation Check of Residuals									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	4.65	5	0.4595	-0.028	-0.109	0.254	0.035	0.065	0.112
12	16.90	11	0.1108	0.054	-0.023	0.066	0.203	0.107	-0.373
18	24.77	17	0.1000	0.107	0.172	-0.265	0.010	0.053	-0.035
24	31.09	23	0.1206	-0.079	0.063	-0.066	-0.233	0.028	-0.015



3.3.6 pav. Modelio *SARIMA(1,0,0)(0,1,0)$_{12}$* modelio liekanų grafinė analizė

ARCH testo liekanoms (žr. 1 priedas) su nulinė hipotezė apie ARCH poveikį priimama, todėl galima teigti, kad liekanų dispersija yra pastovi. Atliekame klaipėdiečių išlaidų, būsto išlaikymui, prognozę.



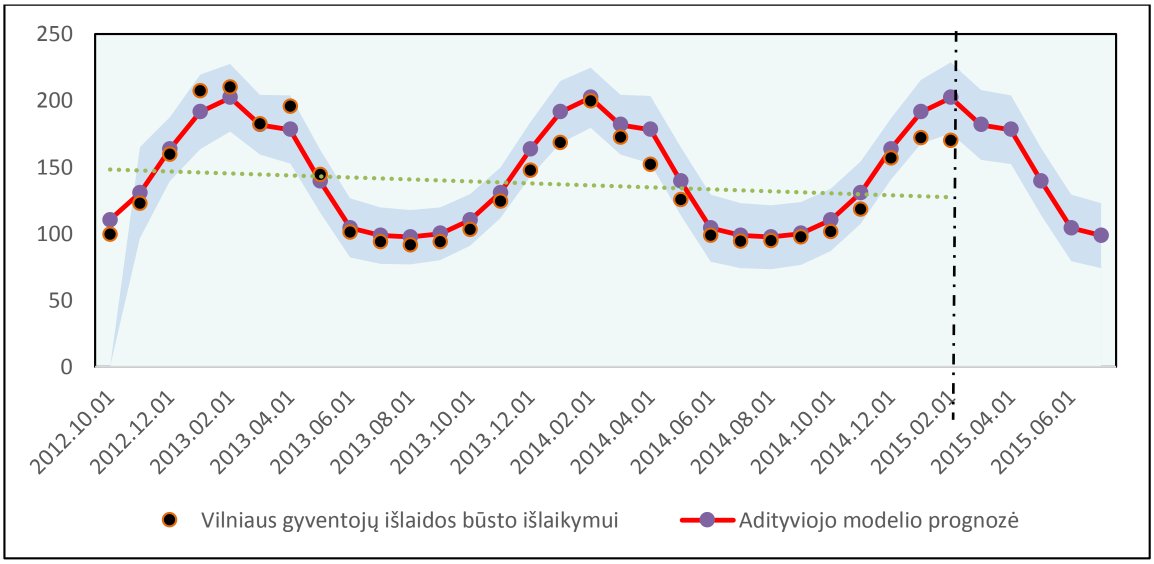
3.3.7 pav. Modelio *SARIMA(1,0,0)(0,1,0)$_{12}$* prognozė

Modelio vidutinė procentinė absoliutinė paklaida (MAPE)- 9,856%.

3.4. ADITYVUSIS LAIKO EILUTĖS REGRESINIS MODELIS VILNIEČIŲ BŪSTO IŠLAIKYMO IŠLAIDOMS

Sudarinėdami Vilniaus gyventojų išlaidų, būsto išlaikymui, sezoninį regresinį modelį, kaip ir kiekvienai sezoninei laiko eilutei pirmiausia turime išskirti trendą. Trendui rasti naudojame mažiausių kvadratų metodą. Šiuo atveju išskyrėme tiesinį trendą, kurio lygtis yra užrašoma tokiu pavidalu:

$$m_t = 1164{,}4 - 0{,}0247 \cdot t. \qquad \textbf{(3.11)}$$

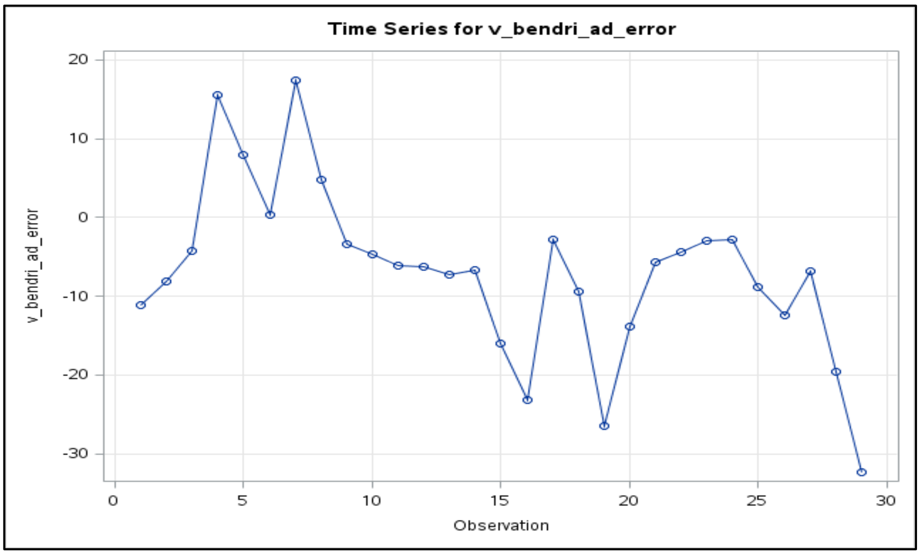


3.4.1 pav. Vilniaus gyventojų išlaidų, būsto išlaikymui, adityvaus regresinio modelio prognozė

Atidžiau pasižiūrėjus į grafiką, galima įžvelgti Vilniaus gyventojų išlaidų, būsto išlaikymui, mažėjimo tendenciją, taip pat, tai matyti iš trendo lygties, kur koeficientas prie kintamojo *t* yra neigiamas. 3.4.1 paveiksle aiškiai galime pamatyti, jog didžiausi skirtumai tarp prognozuojamos reikšmės ir faktiškos yra šaltuoju metų laiku.

Modeliui įvertinti ir palyginti su kitais prognozavimo modeliais apskaičiuosime vidutinę procentinę absoliutinę paklaidą. MAPE= 7,204%.

Tam, kad modelis būtų korektiškas, reikia patikrinti jo paklaidų adekvatumą.



3.4.2 pav. Vilniaus gyventojų išlaidų, būsto išlaikymui, adityvaus regresinio modelio paklaidos

3.4.1 lentelė

ADF stacionarumo testas Vilniaus gyventojų adityvaus regresinio modelio liekanoms

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-6.5170	0.0687	-1.48	0.1273		
	1	-5.9001	0.0849	-1.19	0.2085		
	2	-0.8062	0.5011	-0.27	0.5774		
Single Mean	0	-10.4262	0.0883	-2.05	0.2668	2.19	0.5269
	1	-11.6470	0.0602	-1.81	0.3686	1.77	0.6294
	2	-3.1081	0.6260	-0.79	0.8066	0.50	0.9503

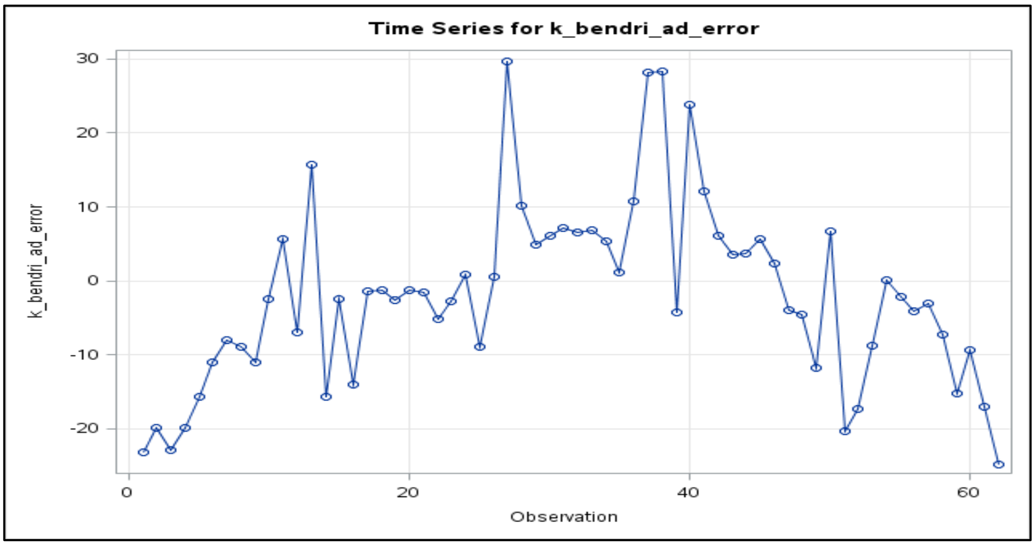
Liekanų laiko eilutės ADF testo p reikšmė $-$ 0,2668 $<$ 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą neatmetama, t. y. eilutė tikrai nėra stacionari. Liekanų vidurkis - 6,785 nėra artimas nuliui, todėl sudaryto adityvaus modelio paklaidos nėra baltasis triukšmas ir modelis taikomas nekorektiškai.

3.5. ADITYVUSIS LAIKO EILUTĖS REGRESINIS MODELIS KAUNIEČIŲ BŪSTO IŠLAIKYMO IŠLAIDOMS

Pirmiausiai išskiriame tiesinį trendą, kurio lygtis yra užrašoma tokiu pavidalu:

$$m_t = 0{,}0044 \cdot t - 68{,}788. \qquad \textbf{(3.12)}$$

Gyventojų išlaidos turi tendencija didėti Kauno mieste.



3.5.1 pav. Kauno gyventojų išlaidų, būsto išlaikymui, adityvaus regresinio modelio paklaidos

Tikriname modelio paklaidų adekvatumą.

3.5.1 lentelė

ADF stacionarumo testas Kauno gyventojų adityviojo regresinio modelio liekanoms

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-26.8095	<.0001	-4.08	<.0001		
	1	-14.1040	0.0070	-2.58	0.0105		
	2	-10.6636	0.0201	-2.22	0.0265		
Single Mean	0	-27.3864	0.0007	-4.11	0.0018	8.43	0.0010
	1	-14.4272	0.0353	-2.59	0.1003	3.36	0.2260
	2	-10.8194	0.0951	-2.21	0.2044	2.45	0.4547

3.5.2 lentelė

Kauno gyventojų adityviojo regresinio modelio liekanų Box – Ljung testas

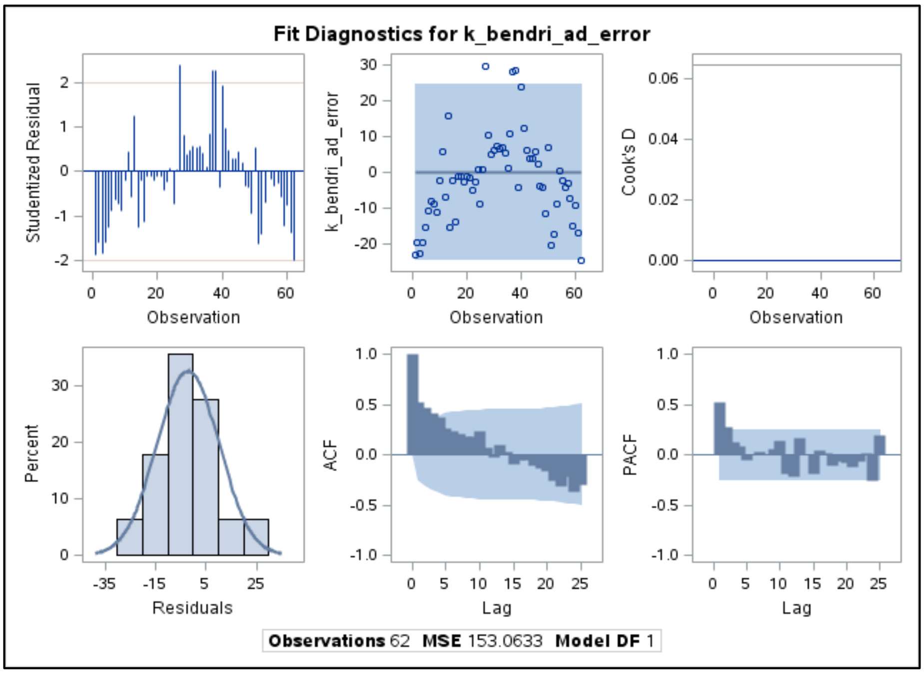
Autocorrelation Check for White Noise									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	61.31	6	<.0001	0.520	0.470	0.408	0.368	0.264	0.230
12	74.06	12	<.0001	0.212	0.177	0.179	0.238	0.073	-0.032

Liekanų laiko eilutės ADF testo *p* reikšmė – 0,0018 < 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą atmetama, t. y. eilutė yra stacionari. 3.5.2 lentelėje matome, kad apskaičiuotos *p* reikšmės visais atvejais yra mažesnės už reikšmingumo lygmenį α*=0.05*, todėl nulinę hipotezę, H_0: paklaidos yra neautokoreliuotos, atmetame ir galime daryti prielaidą, kad tai nėra baltasis triukšmas. ARCH testo liekanoms (žr. 1 priedas) su nulinė hipotezė apie ARCH poveikį atmetama, todėl galima teigti, kad liekanos heteroskedastiškos. Tačiau liekanos autokoreliuotos, todėl taikome autoregresinį procesą liekanoms. Mažiausius modelio tinkamumo įverčius gavome su *AR(2)* procesu:

$$Y_t = m_t + s_t + \varepsilon_t,$$

$$\varepsilon_t = -0{,}40547\varepsilon_{t-1} - 0{,}30245\varepsilon_{t-2} + e_t. \qquad \mathbf{(3.13)}$$

Sudarytojo modelio liekanos tenkina keliamas sąlygas.



3.5.2 pav. Kauno adityvaus regresinio modelio liekanų su *AR(2)* procesu grafinė analizė

3.5.3 lentelė

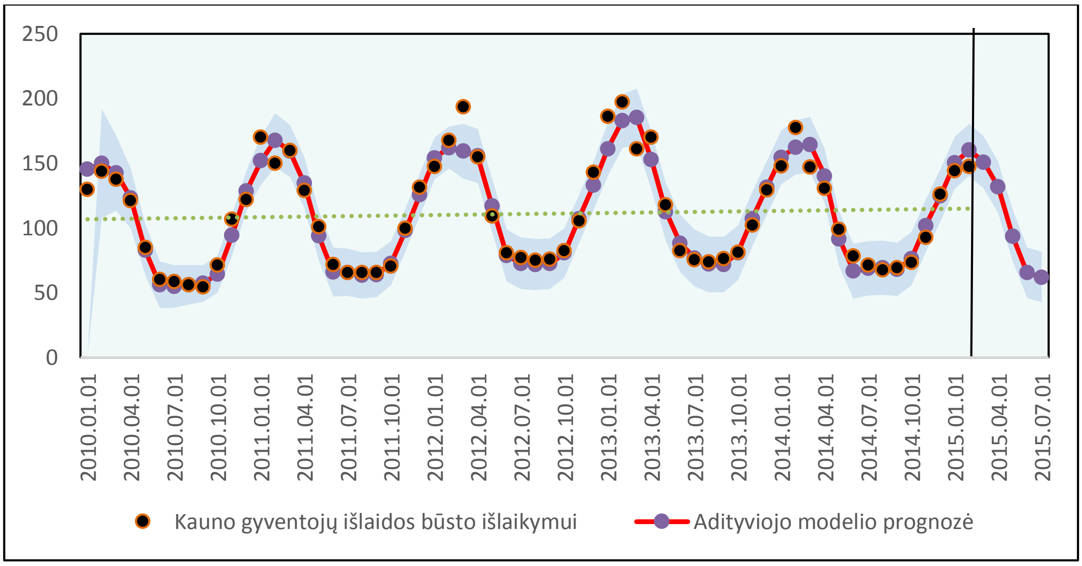
Kauno gyventojų adityviojo regresinio modelio liekanų Box – Ljung testas

Autocorrelation Check of Residuals									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	1.89	4	0.7555	-0.038	-0.089	0.070	0.115	-0.017	-0.011
12	10.62	10	0.3879	0.034	-0.044	0.006	0.254	-0.054	-0.206
18	18.05	16	0.3211	0.186	0.108	-0.181	0.008	0.092	-0.040
24	25.66	22	0.2668	-0.020	0.054	-0.108	-0.147	0.080	-0.180

Box – Ljung testo *p* reikšmė – 0,7555> 0,05, todėl hipotezė apie liekanų nekoreliuotumą neatmetama, t. y. liekanos yra nekoreliuotos. Liekanų vidurkis yra artimas nuliui. Šio modelio liekanų grafinė analizė (3.5.2. pav.) leidžia daryti prielaidą, kad liekanos yra baltasis triukšmas, kadangi liekanos išsibarsčiusios apie nulį, ACF grafike nėra reikšmingumo lygmenį kertančių stulpelių, kas rodo liekanų stacionarumą.

ARCH testo liekanoms nulinė hipotezė apie ARCH poveikį neatmetama, todėl galima teigti, kad liekanų dispersija patovi.

Atliekame adityvaus regresinio modelio su liekanų *AR(2)* procesu prognozę.



3.5.3 pav. Kauno gyventojų išlaidų, būsto išlaikymui, adityvaus regresinio modelio su liekanų AR(2) procesu prognozė

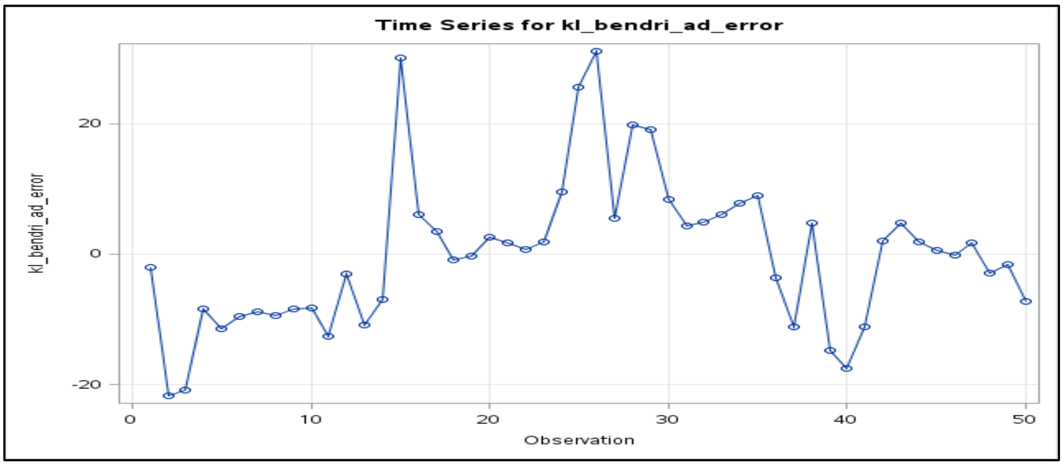
Modelio vidutinė procentinė absoliutinė paklaida (MAPE) - 5,251%.

3.6. ADITYVUSIS LAIKO EILUTĖS REGRESINIS MODELIS KLAIPĖDIEČIŲ BŪSTO IŠLAIKYMO IŠLAIDOMS

Pirmiausiai išskiriame tiesinį trendą, kurio lygtis yra užrašoma tokiu pavidalu:

$$m_t = -0{,}001 \cdot t - 148{,}67. \qquad \textbf{(3.14)}$$

Gyventojų išlaidos turi silpną tendencija mažėti Klaipėdoje.



3.6.1 pav. Klaipėdos gyventojų išlaidų, būsto išlaikymui, adityvaus regresinio modelio paklaidos

Tam, kad modelis būtų korektiškas, reikia patikrinti jo paklaidų adekvatumą.

3.6.1 lentelė

ADF stacionarumo testas Klaipėdos gyventojų adityviojo regresinio modelio liekanoms

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-19.6247	0.0010	-3.45	0.0009		
	1	-17.1200	0.0024	-3.03	0.0032		
	2	-10.9193	0.0178	-2.38	0.0179		
Single Mean	0	-19.6221	0.0068	-3.41	0.0150	5.82	0.0247
	1	-17.1544	0.0145	-3.00	0.0417	4.53	0.0680
	2	-11.0064	0.0866	-2.38	0.1524	2.89	0.3562

3.6.2 lentelė

Klaipėdos gyventojų adityviojo regresinio modelio liekanų Box – Ljung testas

Autocorrelation Check for White Noise									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	61.31	6	<.0001	0.520	0.470	0.408	0.368	0.264	0.230
12	74.06	12	<.0001	0.212	0.177	0.179	0.238	0.073	-0.032

Liekanų laiko eilutės ADF testo *p* reikšmė – 0,0150 < 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą atmetama, t. y. eilutė tikrai yra stacionari. 3.6.2 lentelėje matome, kad apskaičiuotos *p*-reikšmės visais atvejais yra mažesnės už reikšmingumo lygmenį $\alpha=0,05$, todėl nulinę hipotezę, H_0: paklaidos yra neautokoreliuotos, atmetame ir galime daryti prielaidą, kad tai nėra baltasis triukšmas. Galima teigti, kad liekanos heteroskedastiškos (žr.1 priedas). Liekanos autokoreliuotos, todėl pirmiausia taikome autoregresinį procesą liekanoms. Mažiausius modelio AIC, SSE ir BIC gavome su *AR(1)* procesu:

$$Y_t = m_t + s_t + \varepsilon_t,$$

$$\varepsilon_t = -0{,}5995\varepsilon_{t-1} + e_t. \qquad \textbf{(3.15)}$$

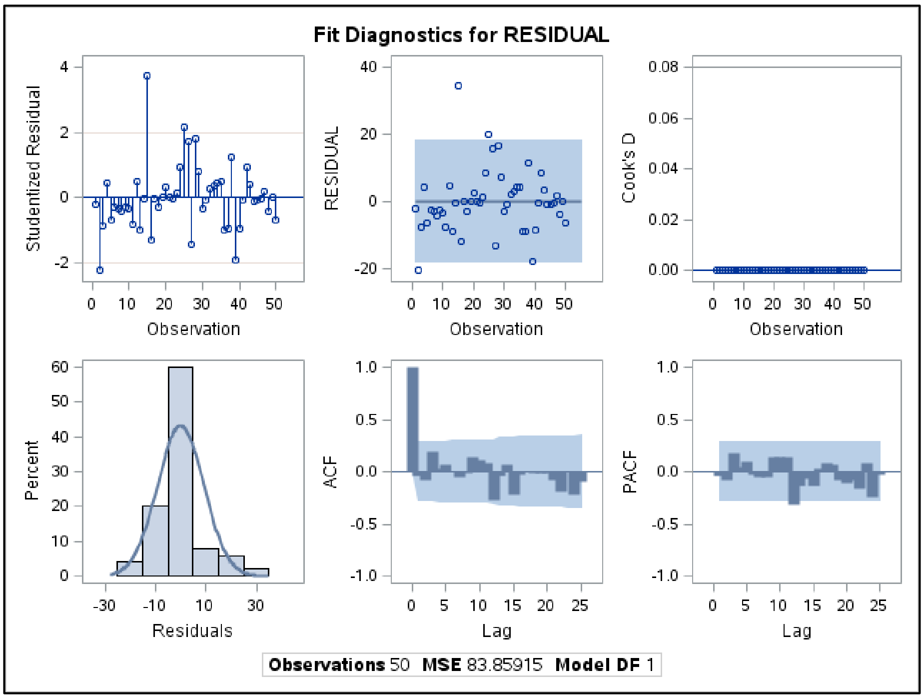
kur e_t- balto triukšmo komponentė.

3.6.3 lentelė

Klaipėdos gyventojų adityviojo regresinio modelio liekanų Box – Ljung testas

Autocorrelation Check of Residuals									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	2.72	5	0.7425	-0.036	-0.081	0.190	0.034	0.066	0.004
12	10.22	11	0.5111	-0.051	-0.011	0.138	0.103	0.081	-0.270
18	14.09	17	0.6608	-0.041	0.066	-0.212	-0.032	0.001	-0.011
24	22.93	23	0.4646	-0.019	-0.020	-0.079	-0.193	-0.004	-0.218

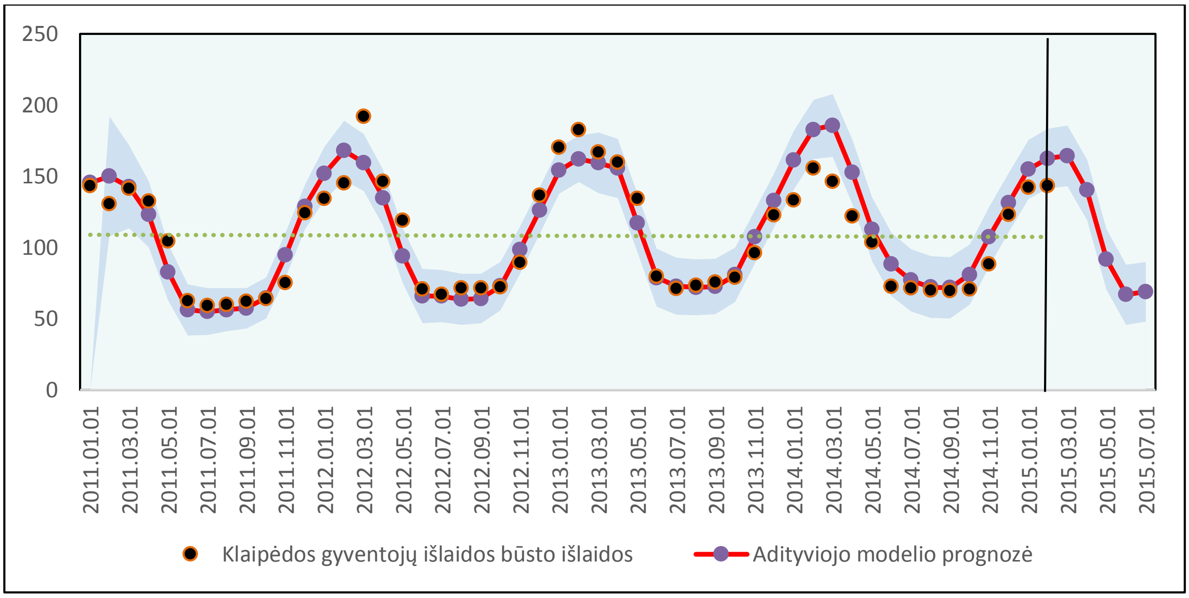
Box – Ljung testo *p* reikšmė - 0,7425 > 0,05, todėl hipotezė apie liekanų nekoreliuotumą neatmetama, t. y. liekanos yra nekoreliuotos. Liekanų vidurkis yra artimas nuliui. Šio modelio liekanų grafinė analizė (3.6.2. pav.) leidžia daryti prielaidą, kad liekanos yra baltasis triukšmas, kadangi liekanos išsibarsčiusios apie nulį, ACF grafike nėra reikšmingumo lygmenį kertančių stulpelių, kas rodo liekanų stacionarumą.



3.6.2 pav. Klaipėdos adityvaus regresinio modelio liekanų su AR(1) procesu grafinė analizė

ARCH testas rodo, kad nulinė hipotezė apie ARCH poveikį neatmetama, todėl galima teigti, kad liekanų dispersija tapo pastovi.

Atliekame adityvaus regresinio modelio su liekanų *AR(1)* procesu prognozę.

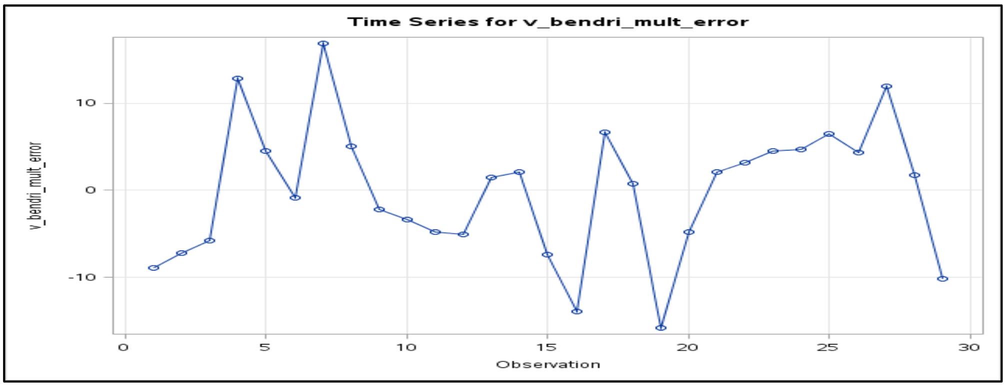


3.6.3 pav. Klaipėdos gyventojų išlaidų, būsto išlaikymui, adityvaus regresinio modelio su liekanų *AR(1)* procesu prognozė

Modeliui įvertinti ir palyginti su kitais prognozavimo modeliais apskaičiuosime vidutinę procentinę absoliutinę paklaidą. MAPE= 5,421%.

3.7. MULTIPLIKATYVUSIS LAIKO EILUTĖS REGRESINIS MODELIS VILNIEČIŲ BŪSTO IŠLAIKYMO IŠLAIDOMS

Sudarinėdami šilumos kainų multiplikatyvųjį modelį, kaip ir adityvaus modelio atveju, turime išskirti trendą. Trendo lygtis užrašoma (3.11) pavidalu.



3.7.1 pav. Vilniaus gyventojų išlaidų, būsto išlaikymui, multiplikatyvaus regresinio modelio paklaidos

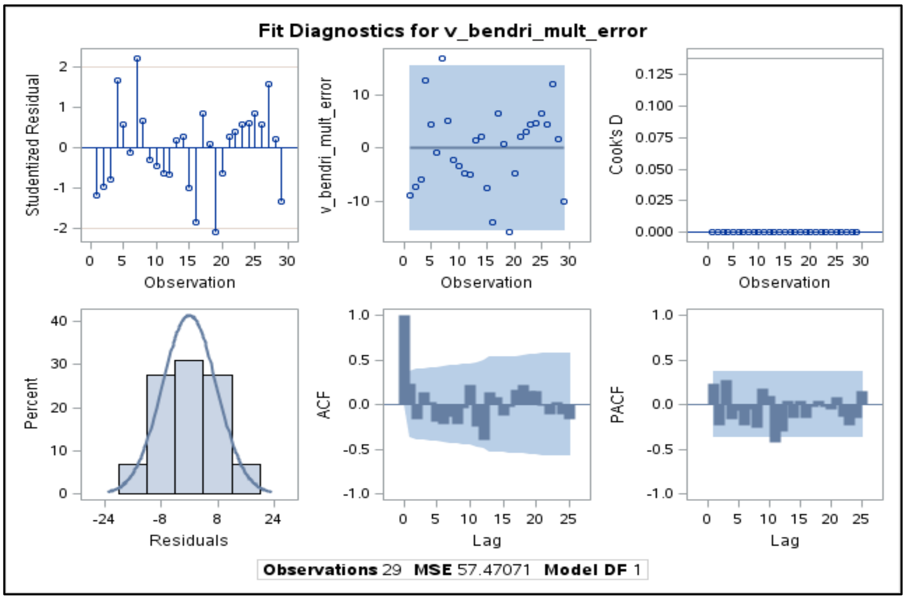
Modeliui įvertinti ir palyginti su kitais prognozavimo modeliais apskaičiuosime vidutinę procentinę absoliutinę paklaidą. MAPE=7,718%.

Patikrinsime sudaryto modelio adekvatumą.

3.7.1 lentelė

Vilniaus gyventojų multiplikatyviojo regresinio modelio liekanų Box – Ljung testas

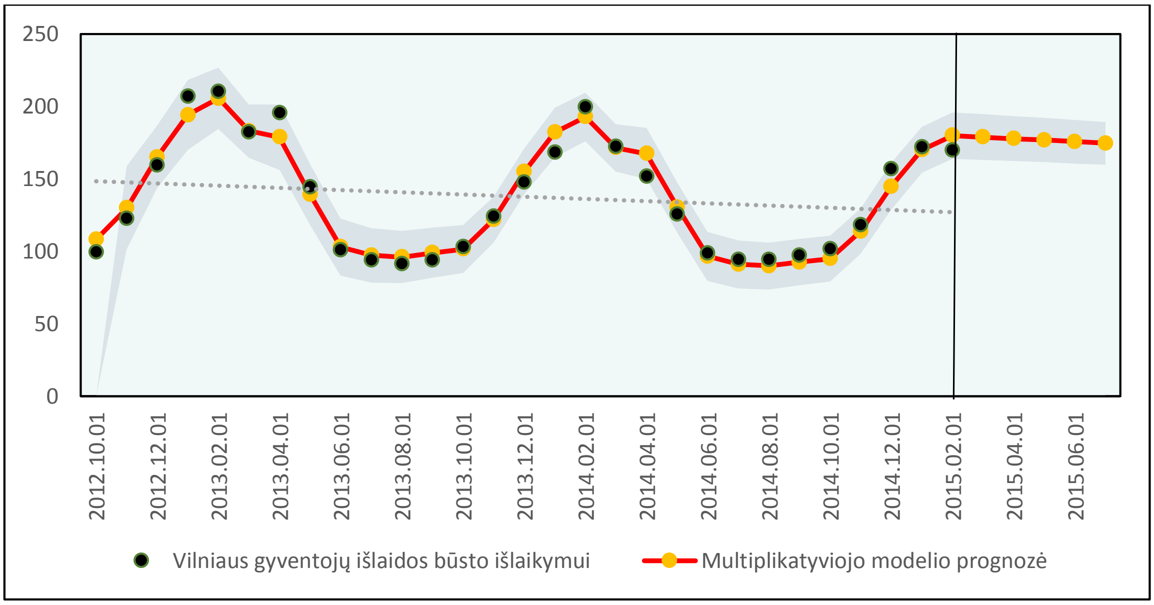
Autocorrelation Check for White Noise									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	6.63	6	0.3569	0.240	-0.162	0.139	0.034	-0.184	-0.222



3.7.2 pav. Vilniaus multiplikatyvaus regresinio modelio modelio liekanų grafinė analizė

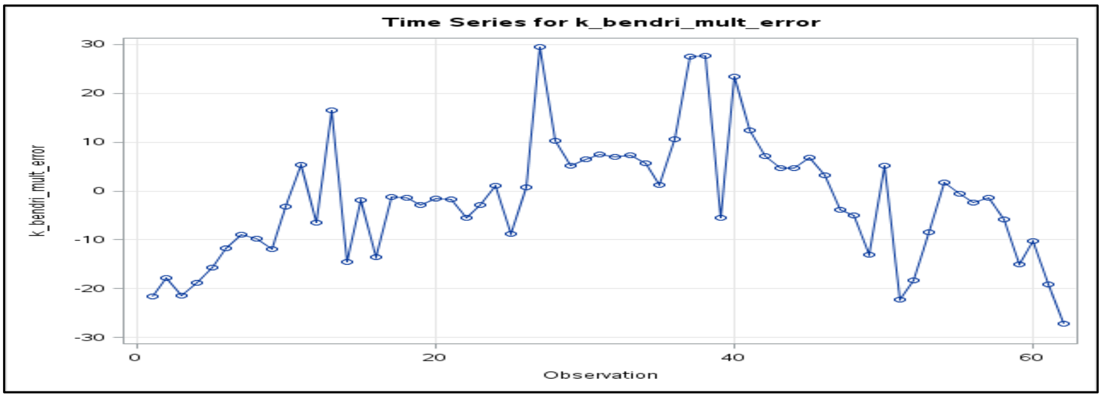
Iš 3.7.2 paveikslo matyti, kad autokoreliacijos nariai yra nereikšminiai, todėl sudarytasis multiplikatyvusis laiko eilutės modelis šiuo aspektu yra korektiškas. Taip pat, patikriname, jog atsitiktinės paklaidos yra baltasis triukšmas, t. y. paklaidos pasiskirstę atsitiktinai.

3.7.1 lentelėje matome, kad apskaičiuota *p* reikšmė yra didesnė už reikšmingumo lygmenį $\alpha=0,05$ $(p>\alpha)$, todėl nulinę hipotezę, H_0: paklaidos yra neautokoreliuotos, priimame ir galime daryti prielaidą, kad tai yra baltasis triukšmas. Multiplikatyvusis laiko eilutės regresinis modelis prognozuoti Vilniaus gyventojų išlaidoms, būsto išlaikymui, taikomas korektiškai.



3.7.3 pav. Vilniaus gyventojų išlaidų, būsto išlaikymui, multiplikatyvaus regresinio modelio prognozė

3.8. MULTIPLIKATYVUSIS LAIKO EILUTĖS REGRESINIS MODELIS KAUNIEČIŲ BŪSTO IŠLAIKYMO IŠLAIDOMS



3.8.1 pav. Kauno gyventojų išlaidų, būsto išlaikymui, multiplikatyvaus regresinio modelio paklaidos

3.8.1 lentelė

ADF stacionarumo testas Kauno gyventojų multiplikatyviojo regresinio modelio liekanoms

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-26.5579	<.0001	-4.00	0.0001		
	1	-13.7010	0.0079	-2.45	0.0149		
	2	-10.2791	0.0226	-2.07	0.0380		
Single Mean	0	-27.0792	0.0008	-4.02	0.0024	8.07	0.0010
	1	-14.0138	0.0396	-2.46	0.1292	3.04	0.3054
	2	-10.4450	0.1052	-2.07	0.2586	2.14	0.5323

3.8.2 lentelė

Kauno gyventojų multiplikatyviojo regresinio modelio liekanų Box – Ljung testas

Autocorrelation Check for White Noise									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	58.32	6	<.0001	0.518	0.462	0.398	0.355	0.247	0.213
12	70.64	12	<.0001	0.198	0.171	0.180	0.240	0.075	-0.031

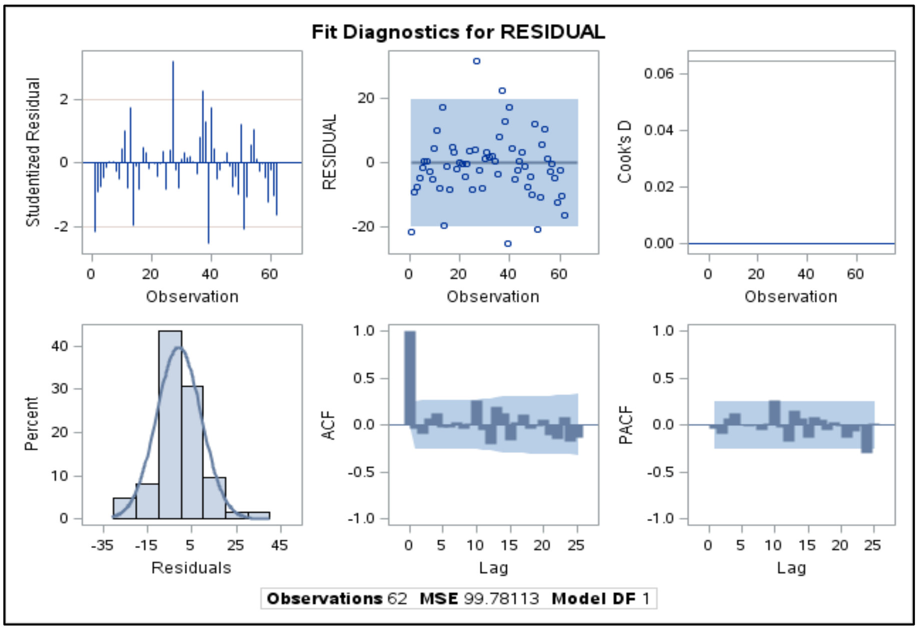
Liekanų laiko eilutės ADF testo p reikšmė – 0,0024< 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą atmetama, t. y. eilutė stacionari. 3.8.2 lentelėje matome, kad apskaičiuotos p reikšmės visais atvejais yra mažesnės už reikšmingumo lygmenį $\alpha=0,05$, todėl nulinę hipotezę, H_0: paklaidos yra neautokoreliuotos, atmetame ir galime daryti prielaidą, kad tai nėra baltasis triukšmas. ARCH testo liekanoms (žr. 1 priedas) su nulinė hipotezė atmetama, todėl galima teigti, kad liekanos heteroskedastiškos. Tačiau liekanos autokoreliuotos, todėl taikome autoregresinį procesą liekanoms. Mažiausius modelio tinkamumo įverčius, kaip ir adityviojo modelio atveju, gavome su *AR(2)* procesu:

$$Y_t = m_t + s_t + \varepsilon_t,$$

$$\varepsilon_t = -0{,}41203\varepsilon_{t-1} - 0{,}30192\varepsilon_{t-2} + e_t. \tag{3.16}$$

e_t- baltojo triukšmo komponentė.

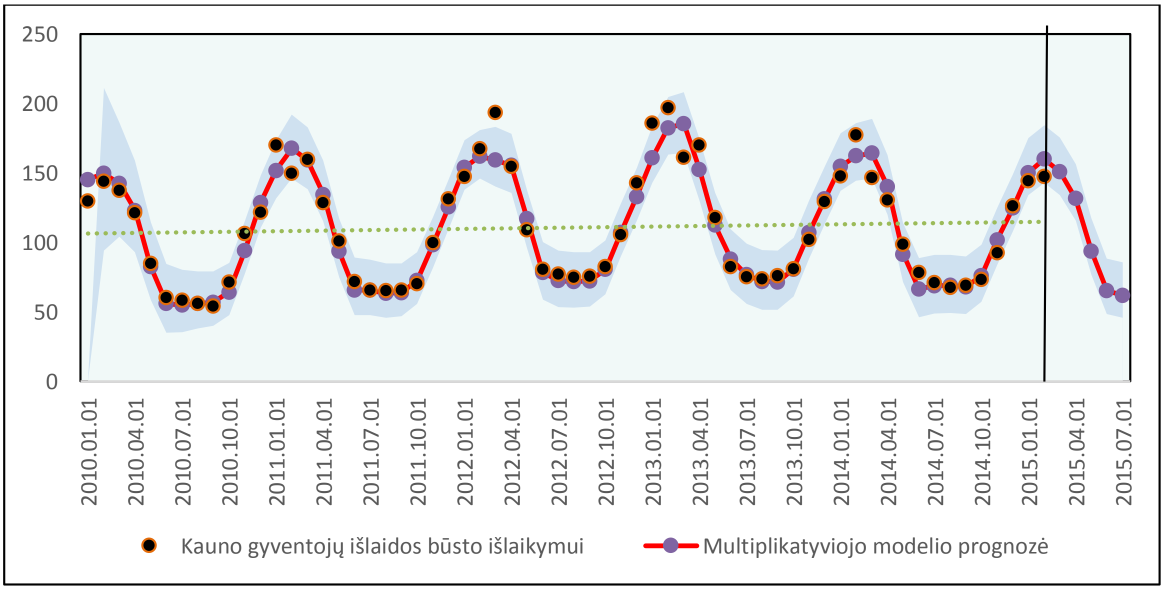
Tikriname, ar sudarytojo modelio liekanos tenkina keliamas sąlygas.



3.8.2 pav. Kauno multiplikatyvaus regresinio modelio modelio liekanų grafinė analizė

Sudarytas modelis yra adekvatus, kadangi liekanos yra baltojo triukšmo procesas. Modelio liekanų grafinė analizė rodo, kad liekanų reikšmės išsidėsčiusios apie nulinę tiesę, ACF grafike stulpeliai nekerta kritinės reikšmės ribų, Box – Ljung statistikos p reikšmės > 0,05. Atlikus ARCH testą liekanoms nulinė hipotezė priimama, todėl galima teigti, kad liekanos nėra heteroskedastiškos (žr. 1 priedas).

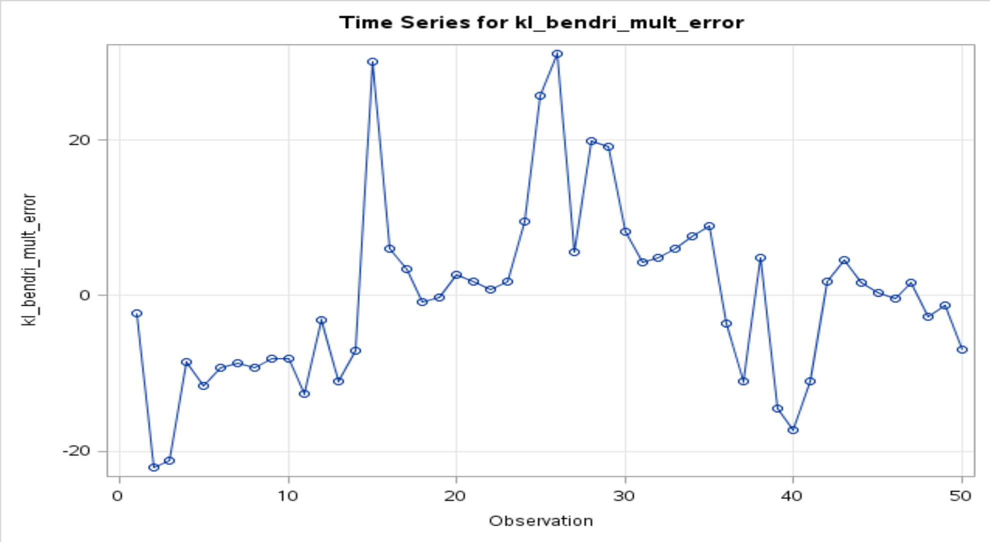
Atliekame multiplikatyvinio regresinio modelio su liekanų *AR(2)* procesu prognozę.



3.8.3 pav. Kauno gyventojų išlaidų, būsto išlaikymui, multiplikatyvaus regresinio modelio su liekanų *AR(2)* procesu prognozė

Modeliui įvertinti ir palyginti su kitais prognozavimo modeliais apskaičiuojame vidutinę procentinę absoliutinę paklaidą. MAPE=4,973%.

3.9. MULTIPLIKATYVUSIS LAIKO EILUTĖS REGRESINIS MODELIS KLAIPĖDIEČIŲ BŪSTO IŠLAIKYMO IŠLAIDOMS



3.9.1 pav. Klaipėdos gyventojų išlaidų, būsto išlaikymui, multiplikatyvaus regresinio modelio paklaidos

Tikriname ar Klaipėdos gyventojų išlaidų, būsto išlaikymui, multiplikatyvaus regresinio modelio paklaidos yra stacionarios.

3.9.1 lentelė

ADF stacionarumo testas Klaipėdos gyventojų multiplikatyviojo regresinio modelio liekanoms

Augmented Dickey-Fuller Unit Root Tests							
Type	**Lags**	**Rho**	**Pr < Rho**	**Tau**	**Pr < Tau**	**F**	**Pr > F**
Zero Mean	0	-19.6225	0.0010	-3.45	0.0009		
	1	-17.1428	0.0024	-3.04	0.0031		
	2	-10.9272	0.0178	-2.40	0.0172		
Single Mean	0	-19.6205	0.0068	-3.41	0.0150	5.82	0.0245
	1	-17.1739	0.0144	-3.01	0.0405	4.57	0.0658
	2	-11.0112	0.0865	-2.40	0.1480	2.93	0.3452

Eilutės ADF testo *p* reikšmė < 0,05, todėl hipotezė apie vienetinės šaknies egzistavimą atmetama, t. y. liekanų laiko eilutė yra stacionari.

3.9.2 lentelė

Klaipėdos gyventojų multiplikatyviojo regresinio modelio liekanų Box – Ljung testas

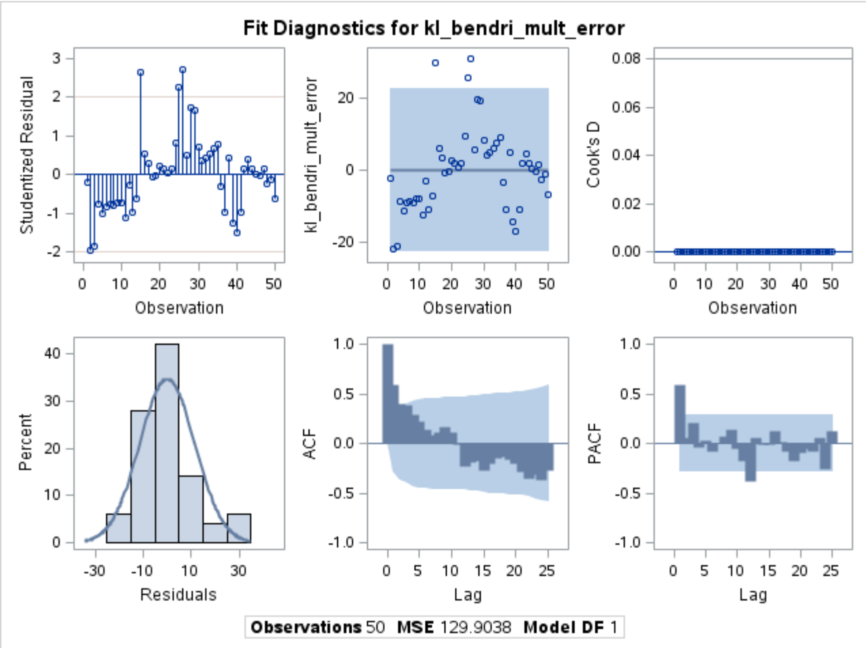
Autocorrelation Check for White Noise									
To Lag	**Chi-Square**	**DF**	**Pr > ChiSq**	**Autocorrelations**					
6	43.76	6	<.0001	0.595	0.394	0.386	0.282	0.216	0.136
12	51.24	12	<.0001	0.084	0.108	0.169	0.117	-0.016	-0.230

Kaip ir adityviosios regresijos modelio atveju, liekanos yra autokoreliuotos. Modelio liekanoms taikysime *AR(1)* procesą, atsižvelgiant į modelio tinkamumo įverčius.

$$Y_t = m_t + s_t + \varepsilon_t,$$
$$\varepsilon_t = -0{,}59954\varepsilon_{t-1} + e_t. \qquad \textbf{(3.17)}$$

e_t - balto triukšmo komponentė.

Atliekame modelio paklaidų adekvatumo analizę, kad patikrintume, ar modelis yra tinkamas.



3.9.2 pav. Klaipėdos multiplikatyvaus regresinio modelio modelio su *AR(1)* liekanų procesu grafinė analizė

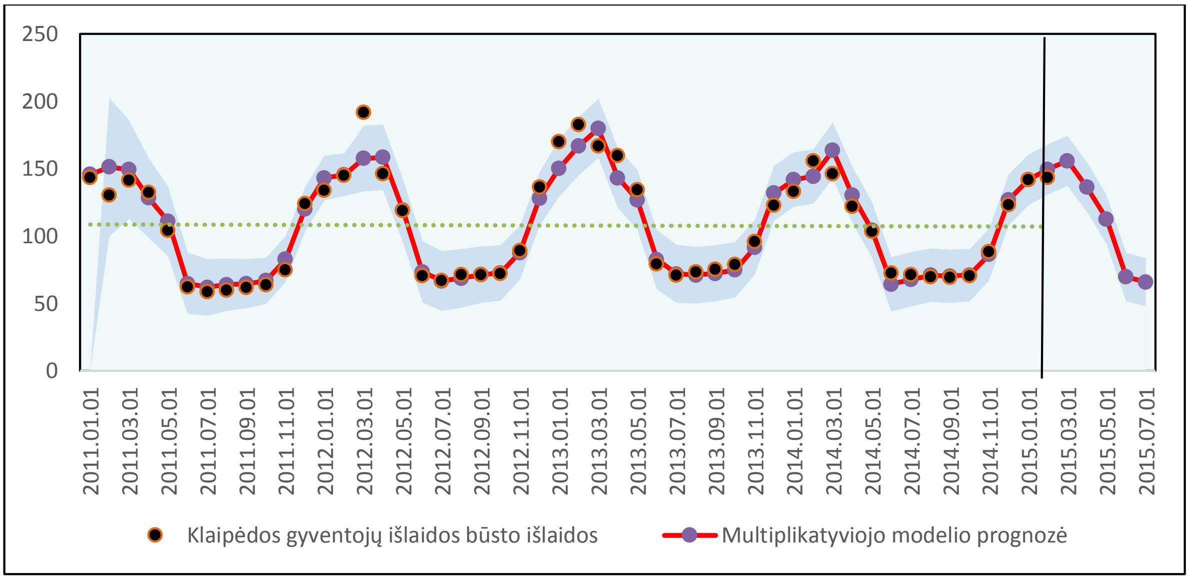
3.9.3 lentelė

Klaipėdos multiplikatyviojo regresinio modelio liekanų *AR(1)* proceso Box – Ljung testas

Autocorrelation Check of Residuals									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	2.75	5	0.7390	-0.036	-0.082	0.190	0.034	0.067	0.008
12	10.35	11	0.4994	-0.049	-0.010	0.138	0.105	0.081	-0.273
18	14.18	17	0.6546	-0.043	0.064	-0.211	-0.033	-0.001	-0.012
24	22.94	23	0.4643	-0.018	-0.019	-0.080	-0.192	-0.004	-0.217

Modelio liekanų grafinė analizė rodo, kad liekanų reikšmės išsidėsčiusios apie nulinę tiesę, ACF grafike stulpeliai nekerta kritinės reikšmės ribų, Box – Ljung statistikos p reikšmės > 0,05. Atlikus ARCH testą liekanoms nulinė hipotezė neatmetama, todėl galima teigti, kad liekanos nėra heteroskedastiškos (žr. 1 priedas). Sudarytas modelis yra adekvatus, kadangi liekanos yra baltojo triukšmo procesas.

Panaudodami Klaipėdos multiplikatyvinį regresinį modelį su liekanų *AR(1)* procesu sudarome prognozę.



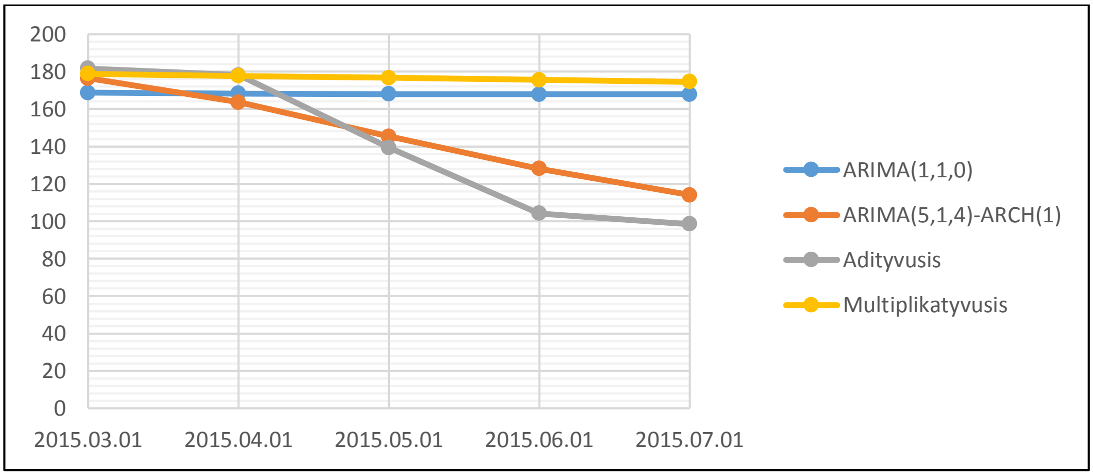
3.9.3 pav. Klaipėdos gyventojų išlaidų, būsto išlaikymui, multiplikatyvaus regresinio modelio su liekanų AR(1) procesu prognozė

Modeliui įvertinti ir palyginti su kitais prognozavimo modeliais apskaičiuosime vidutinę procentinę absoliutinę paklaidą. MAPE=5,401%

.

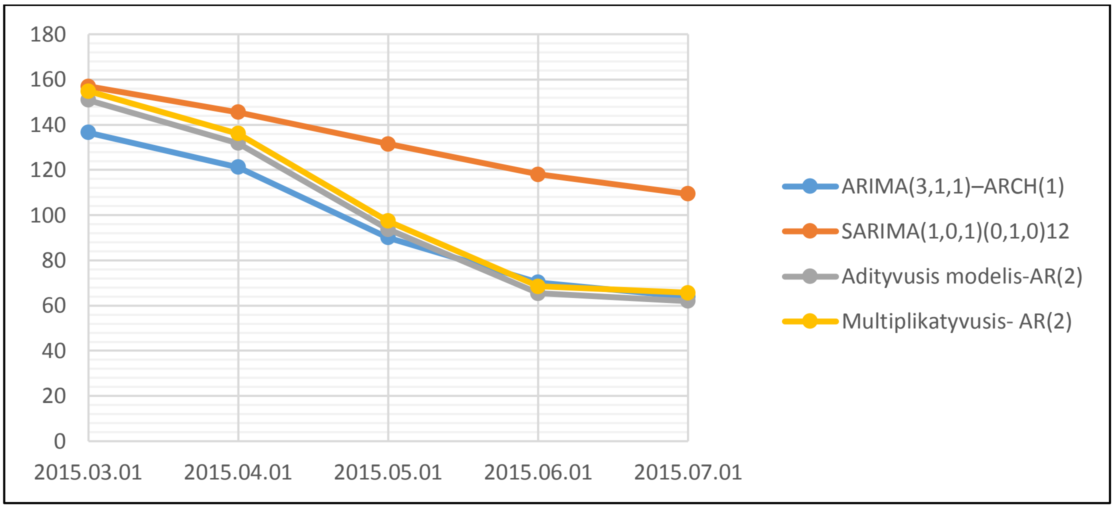
3.10. PROGNOZAVIMO MODELIŲ PALYGINIMAS

3.10.1-3 pav pateikiama kiekvieno miesto visų modelių prognozuojamos išlaidos, būsto išlaikymui, bei kiekvienu atveju apskaičiuotos modelio, visiems istoriniams duomenims, MAPE reikšmės.



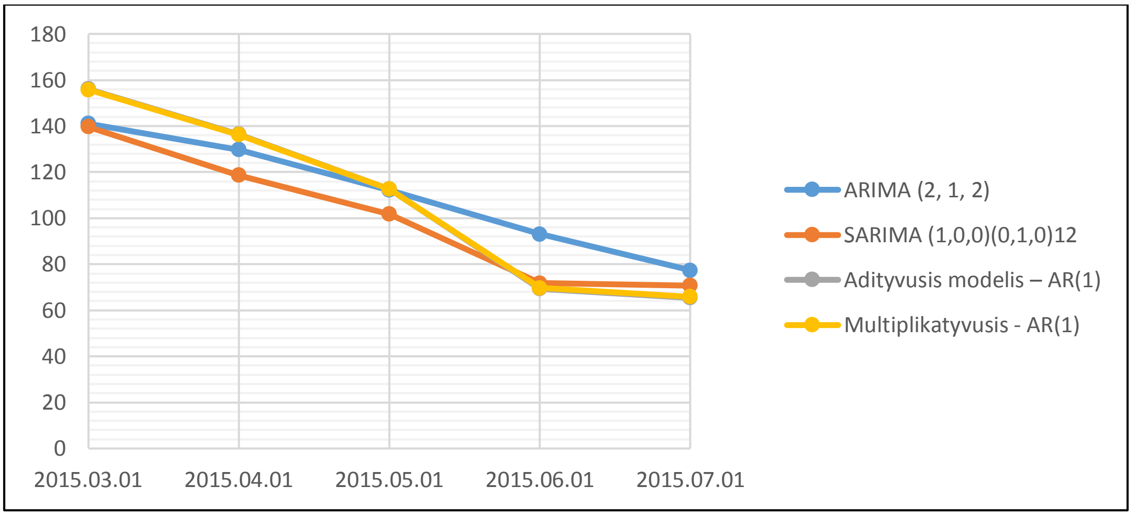
3.10.1 pav. Grafinis Vilniaus gyventojų išlaidų būstui išlaikyti visų prognozavimo modelių palyginimas

Pagal 3.10.1 pav. matome, kad didžiausios išlaidos prognozuojamos multiplikatyviuoju sezoniniu regresijos modeliu, tuo tarpu mažiausios adityviuoju. *ARIMA(1,1,0)* ir multiplikatyviojo modelių prognozės pokyčiai labai maži. Mažiausia MAPE reikšmė (7,204%) (žr. 3.10.1 lentelė) yra adityviojo modelio, tačiau šis modelis neadekvatus.



3.10.2 pav. Grafinis Kauno gyventojų išlaidų būstui išlaikyti visų prognozavimo modelių palyginimas

Iš 3.10.2 pav matyti, mažiausią MAPE (4,973%) turi multiplikatyvusis sezoninis regresijos modelis su *AR(2)* procesu liekanoms, tačiau tiek adityvusis sezoninis regresijos modelis-*AR(*2), tiek *SARIMA(1,0,1)(0,1,0)*$_{12}$ gerai tiktų modeliuoti kauniečių išlaidų, būsto išlaikymui, eilutei. Didžiausias išlaidas prognozuoja *ARIMA(3,1,1)* su procesu liekanoms *ARCH(1)*.



3.10.3 pav. Grafinis Klaipėdos gyventojų išlaidų būstui išlaikyti visų prognozavimo modelių palyginimas

3.10.3 pav. matome, kad visų modelių prognozės mažai skiriasi. Mažiausios MAPE reikšmės (5,421% ir 5,401%) yra sezoninių regresinių su liekanų procesu *AR(1)* (žr. 3.10.1 lentelė) atitinkamai adityviojo ir multiplikatyviojo.

3.10.1 lentelė

Gyventojų išlaidomų, būsto išlaikymui, prognozavimo metodų palyginimas

Miestas	Modelis	MAPE %	Prognozės				
			Kovas	Balandis	Gegužė	Birželis	Liepa
Vilnius	ARIMA(1,1,0)	11,669	168.7569	168.2629	168.018	167.8965	167.8362
	ARIMA(5,1,4)-ARCH(1)	12.228	176.3936	163.6581	145.3218	128.061	114.0397
	Adityvusis modelis	7,204*	181.7115	178.1282	139.4018	104.2195	98.54172
	Multiplikatyvusis modelis	7,718	178.8173	177.7488	176.6803	175.6118	174.5433
Kaunas	ARIMA(3,1,1)–ARCH(1)	21,284	156.9728	145.5789	131.4767	118.072	109.4664
	SARIMA(1,0,1)(0,1,0)$_{12}$	7,582	136.5675	121.2299	90.02521	70.26312	63.90523
	Adityvusis modelis-AR(2)	5,251	150.9976	131.8422	93.77578	65.38224	62.00451
	Multiplikatyvusis modelis - AR(2)	4,973	154.8587	136.1548	97.39905	68.50777	65.64134
Klaipėda	ARIMA (2, 1, 2)	13,423	141.1201	129.7066	112.1829	93.10698	77.35928
	SARIMA (1,0,0)(0,1,0)$_{12}$	9,856	139.6639	118.6595	101.8118	71.76379	70.7872
	Adityvusis modelis – AR(1)	5,421	156.1824	136.4646	112.6777	69.30515	65.4528
	Multiplikatyvusis modelis - AR(1)	5,401	155.8349	136.251	112.6806	69.73234	65.90655

*sudarytas modelis neadekvatus

DISKUSIJA

Norint išsiaiškinti tiksliausius prognozavimo metodus kiekvienai laiko eilutei, kurių prognozės geriausiai atitiktų tikrąsias gyventojų išlaidas, apskaičiavome vidutinės procentines absoliutines paklaidas. Taip pat, gautas prognozes lyginome tarpusavyje ir su tikraisiais duomenimis.

Vilniaus gyventojų išlaidoms, būsto išlaikymui, tiksliausias prognozavimo modelis yra adityvusis (MAPE- 7,204%), tačiau šis modelis yra neadekvatus, todėl būtų renkamasi antrasis pagal tikslumą - multiplikatyvusis (MAPE-7,718). Bet šio modelio prognozės skirtumai labai maži, kaip ir *ARIMA(1,1,0)* modelio atveju. Kitaip sakant, šiais modeliais gautos prognozės yra konstantos, kurios tinka nebent vieno-dviejų žingsnių prognozėms. Paradoksalu, tačiau Vilniaus miesto atveju, tinkamiausias modelis yra *ARIMA(5,1,4)-ARCH(1),* kurio paklaida didžiausia (12.228 %), nors traktuojama kaip tiksli (V.Boguslauskas, 2011).

Kauno gyventojų išlaidoms, būsto išlaikymui, tiksliausias prognozavimo modelis yra multiplikatyvusis su *AR(2)* procesu liekanoms. Pritaikius multiplikatyvųjį regresijos modelį buvo nustatyta, kad šios liekanos autokoreliuotos, tačiau stacionarios, todėl pritaikėme autoregresinį antros eilės procesą.

Klaipėdos gyventojams prognozės, apskaičiuotos mūsų sudarytais modeliais yra optimistiškiausios lyginant su Kauno ir Vilniaus miestais, t. y. jų išlaidos yra prognozuojamos mažiausios. Visi parinkti metodai gana gerai prognozuoja reikšmes. Išskyrus *ARIMA(2,1,2)*, kurio prognozės vasaros mėnesiais smarkiausiai skiriasi nuo tikrosios reikšmės.

Pagal rezultatus galima būtų teigti, jog universaliausias prognozavimo modelis mūsų tiriamoms laiko eilutėms yra multiplikatyvusis sezoninis regresijos su autoregresiniu procesu paklaidoms modelis. Šie modeliai dažnai taikomi nedidelėms laiko eilutėms pvz. prognozuojant šalies bendrąjį vidaus produktą.

Apibendrinant galima pasakyti, kad ne visuomet taikant pačius tiksliausius modelius gausime labiausiai tikrovę atitinkančias prognozes. Nors galime pastebėti tendencija, jog tie modeliai, kurie turėjo didžiausias paklaidas, labiausiai skirdavosi nuo tikrųjų laiko eilutės reikšmių.

IŠVADOS

1. Atlikus modelių analizę bei prognozes buvo nustatyta, kad:
    - Vilniaus gyventojų išlaidoms, būsto išlaikymui, labiausiai tinka *ARIMA(5,1,4)-ARCH(1),* kurio paklaida MAPE=12,228%.
    - Kauno gyventojų išlaidoms, būsto išlaikymui, labiausiai tinka multiplikatyvusis sezoninis regresinis laiko eilutės modelis su autoregresiniu antros eilės procesu liekanoms, kurio paklaida MAPE=4,973%.
    - Klaipėdos gyventojų išlaidoms, būsto išlaikymui, labiausiai tinka multiplikatyvusis sezoninis regresinis laiko eilutės modelis su autoregresiniu antros eilės procesu liekanoms, kurio paklaida MAPE=5,401%, tačiau ir kiti modeliai mažai skiriasi tiek MAPE reikšme, tiek ir prognozėmis.
2. Kadangi visų trijų miestų išlaidų eilučių sezoniniai svyravimai nėra pastovūs, tinkamesnis modelis yra multiplikatyvus.
3. *ARIMA(1,1,0)* ir multiplikatyvus regresinis modeliai nėra tinkami prognozuoti tolimas vilniečių išlaidas, nes prognozių pakyčiai labai maži (prognozė yra konstanta).
4. ARIMA modeliai geriau prognozuoja, sudaryti didesniam kiekiui duomenų.

PADĖKOS

Dėkoju doc. dr. Audriui Kabašinskui už patarimus ir konsultacijas rašant magistro darbą, šeimos nariams už supratingumą, kantrybę ir palaikymą.

LITERATŪRA

1. Boguslauskas, V., *Ekonometrikos pagrindai*, Kaunas, 2004.
2. Leipus, R., *Ekonometrija II.*, Vilnius, 2013.
3. Kavaliauskas, M.; Rudzkis, R., *Laiko eilučių analizė*. Paskaitų konspektas, Kaunas, 2015.
4. Karpuškienė, V. *ARMA/ARIMA modeliai*, 2009.
5. Chatfield, C. *The Analysis Of Time Series*. Fifth edition, The University of Bath, UK, 1996.
6. Francq, C.; Zakoian, J. M., *GARCH Models: Structure, Statistical Inference and Financial Applications.*, Wiley, 2010.
7. Eangle, R. F., *Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation* // Econometrica, Vol. 50, 1982, p. 987–1007.
8. Richards, R., *Unit Root Tests* [žiūrėta 2015 04 30]. Prieiga per internetą: http://faculty.washington.edu/ezivot/econ584/notes/unitroot.pdf
9. Bollerslev, T., Generalized Autoregressive Conditional Heteroskedasticity. Journal of Econometrics, 1986, p. 309-328.
10. R. Levulienė „Statistikos taikymai naudojant SAS“, Vilnius, 2009.
11. Nobelio 2003 m. ekonomikos mokslo premijos laureatai [žiūrėta 2015 05 20]. Prieiga per internetą: http://www.lb.lt/nobelio_laureatai_3
12. Samsonovas, O. Šildymo kainų Kauno mieste indeksas ir jo prognozavimas. *Bakalauro darbas*. KTU, 2013.
13. Taikomoji laiko eilučių ekonometrija. Paskaitų konspektas V. Kvedaras, 2005 [žiūrėta 2015 04 14]. Prieiga per internetą: http://web.vu.lt/mif/v.kvedaras/files/2013/09/Konspektas_2005.pdf
14. SAS dokumentacija. Laiko eilučių prognozavimo posistemė [žiūrėta 2015 05 09]. Prieiga per internetą:http://support.sas.com/documentation/cdl/en/etsug/60372/HTML/default/viewer.htm#tfintro_toc.htm
15. Engeneering Handbook of Statistical Methods. Prieiga per internetą: http://www.itl.nist.gov/div898/handbook/pmc/section4/pmc445.htm
16. T. Bollerslev; R. Y. Chou; K. F. Kroner, *ARCH modeling in finance*. Journal of Econometrics, North-Holland, 1992. [žiūrėta 2015 05 13] Prieiga per internetą: http://www-stat.wharton.upenn.edu/~steele/Courses/434/434Context/GARCH/BollerslevReview.pdf
17. SAS ARIMA procedure. Prieiga per internetą: http://www.dms.umontreal.ca/~duchesne/chap7.pdf
18. SAS AUTOREG procedure. Prieiga per internetą: http://www.dms.umontreal.ca/~duchesne/chap8.pdf

PRIEDAI

1 Priedas. Modelių liekanų heteroskedastiškumo tikrinimas ARCH testu

ARCH testas modelio ARIMA(1,1,0) liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	Q	Pr > Q	LM	Pr > LM
1	0.5181	0.4717	0.3782	0.5385
2	0.7281	0.6949	0.4547	0.7967
3	0.8395	0.8400	0.7148	0.8697
4	0.8541	0.9310	0.7211	0.9487
5	1.1364	0.9508	1.0392	0.9593
6	1.3260	0.9702	1.2660	0.9735
7	1.9774	0.9611	2.6350	0.9166
8	2.5085	0.9613	3.6689	0.8857
9	2.7576	0.9731	3.6801	0.9312
10	7.3573	0.6913	4.8337	0.9020
11	7.5606	0.7520	6.3511	0.8489
12	7.6140	0.8145	6.4682	0.8907

ARCH testas modelio ARIMA(5,1,4) liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	Q	Pr > Q	LM	Pr > LM
1	3.8860	0.0487	2.7258	0.0987
2	3.9166	0.1411	3.6505	0.1612
3	6.1247	0.1057	4.8980	0.1794
4	14.9371	0.0048	6.4497	0.1680
5	15.7498	0.0076	7.3836	0.1936
6	15.8083	0.0148	7.5098	0.2763
7	15.8181	0.0268	8.8559	0.2632
8	15.8181	0.0451	9.8845	0.2732
9	15.8792	0.0694	9.9570	0.3540
10	15.8866	0.1029	9.9658	0.4435
11	16.0917	0.1378	10.0439	0.5264
12	16.9820	0.1503	10.2674	0.5925

ARCH testas modelio ARIMA(3,1,1) liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	Q	Pr > Q	LM	Pr > LM
1	7.9196	0.0049	7.0589	0.0079
2	8.6272	0.0134	9.9973	0.0067
3	8.7289	0.0331	11.9139	0.0077
4	8.7585	0.0674	13.6716	0.0084
5	10.4351	0.0638	13.8001	0.0169
6	12.6796	0.0484	14.8275	0.0216
7	15.4874	0.0302	15.5499	0.0296

8	15.5447	0.0494	15.8645	0.0444
9	15.6531	0.0745	15.9807	0.0673
10	15.6718	0.1094	15.9861	0.1000
11	16.7676	0.1149	16.5897	0.1206
12	19.5562	0.0760	16.5979	0.1654

ARCH testas modelio SARIMA(1,0,1) liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	0.0402	0.8411	0.0004	0.9846
2	1.3044	0.5209	0.7483	0.6879
3	1.5320	0.6749	1.1280	0.7703
4	2.9994	0.5579	2.9727	0.5624
5	4.7846	0.4427	4.3586	0.4990
6	7.0660	0.3148	5.7658	0.4499
7	9.7595	0.2026	8.0273	0.3302
8	12.2397	0.1408	11.1256	0.1947
9	13.3414	0.1478	13.8773	0.1268
10	16.5505	0.0849	14.2009	0.1640
11	16.6037	0.1202	15.2398	0.1718
12	23.5148	0.0237	15.7117	0.2048

ARCH testas modelio ARIMA(2,1,2) liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	1.8571	0.1730	1.7360	0.1876
2	2.2868	0.3187	2.3967	0.3017
3	5.0910	0.1653	5.7629	0.1237
4	5.3097	0.2570	5.9715	0.2013
5	5.3262	0.3774	5.9745	0.3087
6	5.3876	0.4951	6.5993	0.3595
7	6.1015	0.5279	7.2813	0.4002
8	6.7728	0.5613	7.6152	0.4719
9	8.4124	0.4932	9.4556	0.3963
10	8.8425	0.5471	10.8532	0.3691
11	9.3144	0.5929	11.0556	0.4386
12	10.4801	0.5739	11.0620	0.5236

ARCH testas modelio SARIMA(1,0,0) liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	2.0814	0.1491	1.6382	0.2006
2	2.5149	0.2844	1.7543	0.4160

3	2.5810	0.4608	1.9075	0.5918
4	4.3448	0.3614	3.0483	0.5498
5	7.3267	0.1975	4.1740	0.5246
6	10.8279	0.0938	5.2141	0.5167
7	12.4813	0.0858	5.5312	0.5954
8	14.1452	0.0781	7.0529	0.5309
9	14.3716	0.1097	7.5323	0.5819
10	17.5394	0.0632	8.5121	0.5789
11	17.5545	0.0925	9.4246	0.5828
12	30.4262	0.0024	14.6712	0.2599

ARCH testas Vilniaus adityvaus regresinio modelio liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	0.8842	0.3471	1.6449	0.1997
2	1.5453	0.4618	2.8278	0.2432
3	2.2599	0.5202	5.7524	0.1243
4	2.2940	0.6819	7.0816	0.1316
5	3.5775	0.6117	7.1433	0.2102
6	4.7449	0.5769	9.2902	0.1579
7	5.7637	0.5676	9.4334	0.2230
8	6.5360	0.5874	9.5203	0.3003
9	7.1485	0.6217	14.9768	0.0916
10	9.9425	0.4455	20.4230	0.0255
11	10.2329	0.5096	22.3429	0.0218
12	10.7221	0.5529	23.5279	0.0236

ARCH testas Kauno adityvaus regresinio modelio liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	6.2434	0.0125	5.5888	0.0181
2	7.7083	0.0212	5.6618	0.0590
3	8.1619	0.0428	5.6891	0.1278
4	8.9911	0.0613	8.0638	0.0893
5	11.2983	0.0458	9.7653	0.0822
6	14.3364	0.0261	11.1013	0.0853
7	16.9532	0.0177	11.9178	0.1033
8	19.2629	0.0135	13.1715	0.1061
9	19.9322	0.0183	13.7151	0.1328
10	22.0900	0.0147	14.9696	0.1332
11	29.0304	0.0022	16.5782	0.1210
12	29.1210	0.0038	22.4083	0.0332

ARCH testas Kauno adityvaus regresinio modelio liekanoms su AR(2) procesu

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	1.1697	0.2795	0.7344	0.3914
2	2.0724	0.3548	1.1279	0.5690
3	2.2630	0.5196	1.6438	0.6495
4	4.7701	0.3117	3.9778	0.4090
5	7.8037	0.1674	5.5789	0.3494
6	11.9124	0.0640	7.5427	0.2736
7	15.7907	0.0271	9.9264	0.1928
8	18.0036	0.0212	11.9588	0.1531
9	19.8027	0.0192	14.9131	0.0934
10	22.2137	0.0141	15.1613	0.1263
11	23.0331	0.0175	15.2886	0.1697
12	33.7084	0.0007	17.7690	0.1229

ARCH testas Kauno multiplikatyvaus regresinio modelio liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	5.4040	0.0201	5.1622	0.0231
2	6.3038	0.0428	5.1933	0.0745
3	6.5813	0.0865	5.2105	0.1570
4	7.5258	0.1106	7.4608	0.1134
5	9.8691	0.0790	9.2332	0.1001
6	12.9544	0.0438	10.7929	0.0950
7	15.4791	0.0303	11.6972	0.1110
8	17.5233	0.0251	12.9022	0.1153
9	18.0444	0.0347	13.4068	0.1450
10	20.7549	0.0229	15.0545	0.1301
11	28.9321	0.0023	17.5396	0.0929
12	29.0502	0.0039	23.0894	0.0270

ARCH testas Kauno multiplikatyvaus regresinio modelio liekanoms su AR(2) procesu

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	1.1151	0.2910	0.7322	0.3922
2	2.0716	0.3549	1.2080	0.5466
3	2.2228	0.5275	1.6363	0.6512
4	4.8704	0.3009	4.0841	0.3947
5	8.1661	0.1473	5.7802	0.3282
6	12.6368	0.0492	7.7487	0.2571
7	16.7568	0.0190	9.9955	0.1888
8	18.8454	0.0157	11.6264	0.1687
9	20.7748	0.0137	14.2960	0.1122
10	23.0903	0.0104	14.6936	0.1436
11	23.9341	0.0130	14.7281	0.1953

12	36.8742	0.0002	18.8448	0.0923

ARCH testas Klaipėdos adityvaus regresinio modelio liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	2.9704	0.0848	2.9541	0.0857
2	3.0414	0.2186	3.0212	0.2208
3	3.1541	0.3685	3.0395	0.3856
4	3.2254	0.5208	3.2969	0.5094
5	4.8359	0.4362	4.6569	0.4592
6	7.0450	0.3167	5.5945	0.4701
7	9.6123	0.2116	6.7863	0.4515
8	11.5578	0.1720	7.5397	0.4797
9	12.4874	0.1872	8.0262	0.5315
10	13.1488	0.2155	8.4747	0.5826
11	18.7875	0.0650	10.4380	0.4915
12	19.0211	0.0880	10.9738	0.5312

ARCH testas Klaipėdos adityvaus regresinio modelio liekanoms su AR(1) procesu

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	0.5176	0.4718	0.4820	0.4875
2	0.5188	0.7715	0.4985	0.7794
3	0.6044	0.8954	0.6084	0.8945
4	1.2701	0.8664	1.1418	0.8876
5	2.7629	0.7365	2.1444	0.8288
6	4.6476	0.5897	3.3312	0.7663
7	6.5341	0.4789	4.6555	0.7019
8	8.1074	0.4231	6.1022	0.6358
9	8.3843	0.4959	6.6919	0.6692
10	10.6386	0.3864	7.2823	0.6986
11	11.6682	0.3891	7.3049	0.7739
12	12.3575	0.4174	7.3487	0.8337

ARCH testas Klaipėdos multiplikatyvaus regresinio modelio liekanoms

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	3.1182	0.0774	3.0943	0.0786
2	3.2051	0.2014	3.1610	0.2059
3	3.3312	0.3433	3.1801	0.3647
4	3.4057	0.4924	3.4616	0.4837
5	5.0205	0.4134	4.8247	0.4376
6	7.2221	0.3008	5.7494	0.4518
7	9.7856	0.2011	6.9284	0.4364
8	11.7240	0.1640	7.6672	0.4666

9	12.6565	0.1788	8.1590	0.5182
10	13.3056	0.2071	8.5954	0.5709
11	18.8417	0.0640	10.4512	0.4903
12	19.1388	0.0852	10.9483	0.5334

ARCH testas Klaipėdos multiplikatyvaus regresinio modelio liekanoms su AR(1) procesu

Tests for ARCH Disturbances Based on OLS Residuals				
Order	**Q**	**Pr > Q**	**LM**	**Pr > LM**
1	0.5125	0.4741	0.4764	0.4901
2	0.5141	0.7733	0.4917	0.7820
3	0.6060	0.8951	0.6111	0.8939
4	1.2560	0.8688	1.1341	0.8888
5	2.7188	0.7432	2.1267	0.8314
6	4.5733	0.5996	3.3066	0.7695
7	6.4306	0.4905	4.6252	0.7056
8	7.9841	0.4350	6.0723	0.6391
9	8.2624	0.5079	6.6758	0.6708
10	10.5155	0.3965	7.2556	0.7011
11	11.5194	0.4008	7.2734	0.7765
12	12.1803	0.4313	7.3082	0.8366

2 Priedas. GARCH modelių parametrų įverčiai

Vilniaus gyventojų išlaidų ARIMA(5,1,4) liekanų GARCH modelių parametrų įverčiai

Obs	_MODEL_	_AH_0	_AH_1	_AH_2	_AH_3	_GH_1
1	arch_1	98.4080	0.73089	.	.	.
2	arch_2	98.4037	0.73081	0	.	.
3	arch_3	76.3457	0.57424	0	0.11635	.
4	garch_1_1	98.4018	0.73074	.	.	0

Kauno gyventojų išlaidų ARIMA(3,1,1) liekanų GARCH modelių parametrų įverčiai

Obs	_MODEL_	_AH_0	_AH_1	_AH_2	_AH_3	_GH_1
1	arch_1	165.927	0.35587	.	.	.
2	arch_2	164.229	0.36009	0	.	.
3	arch_3	151.414	0.35953	0	0.051491	.
4	garch_1_1	165.929	0.35586	.	.	0

3 Priedas. GARCH modelių parinkimo AIC ir BIC kriterijai

Vilniaus gyventojų išlaidų ARIMA(5,1,4) liekanų GARCH modelių parinkimo kriterijai

Obs	Model	BIC	AIC
1	arch_1	236.047767	233.383358
2	arch_2	239.379971	235.383358
3	garch_1_1	239.379971	235.383358
4	arch_3	241.110706	235.781888

Kauno gyventojų išlaidų ARIMA(3,1,1) liekanų GARCH modelių parinkimo kriterijai

Obs	Model	BIC	AIC
1	arch_1	515.92338	511.701633
2	garch_1_1	520.034254	513.701633
3	arch_2	520.035605	513.702984
4	arch_3	524.010238	515.566743

4 Priedas. ARIMA modelių parinkimo AIC, BIC, SSE kriterijai

Vilniaus gyventojų išlaidų ARIMA modelių parinkimo kriterijai

Model	AIC	BIC	SSE
_ARIMA_0_0	258.5591	258.5591	16790.69
_ARIMA_0_1	252.7682	254.1004	12712.42
_ARIMA_0_2	254.7355	257.3999	12697.58
_ARIMA_0_3	254.1043	258.1009	11558.71
_ARIMA_0_4	255.9363	261.2651	11489.56
_ARIMA_0_5	254.1633	260.8243	10041.12
_ARIMA_1_0	252.6567	253.9889	12661.89
_ARIMA_1_1	254.4956	257.16	12589.24
_ARIMA_1_2	256.4564	260.453	12571.64
_ARIMA_1_3	256.0172	261.3461	11522.84
_ARIMA_1_4	256.4581	263.1191	10898.72
_ARIMA_1_5	255.2443	263.2375	9716.911
_ARIMA_2_0	254.4964	257.1608	12589.6
_ARIMA_2_1	255.7288	259.7255	12249.19
_ARIMA_2_2	248.6853	254.0141	8868.236
_ARIMA_2_3	250.6846	257.3456	8868.004
_ARIMA_2_4	258.0685	266.0618	10748.16
_ARIMA_2_5	256.0022	265.3277	9295.302
_ARIMA_3_0	256.3972	260.3939	12545.11
_ARIMA_3_1	257.6065	262.9353	12195.77
_ARIMA_3_2	254.6352	261.2963	10211.81
_ARIMA_3_3	251.5512	259.5444	8516.213
_ARIMA_3_4	259.9131	269.2385	10688.64

_ARIMA_3_5	254.4391	265.0968	8184.612
_ARIMA_4_0	252.5304	257.8592	10173.65
_ARIMA_4_1	249.1237	255.7847	8387.175
_ARIMA_4_2	258.7585	266.7517	11016.28
_ARIMA_4_3	252.3676	261.693	8163.719
_ARIMA_4_4	254.3654	265.0231	8163.098
_ARIMA_4_5	253.8768	265.8666	7468.86
_ARIMA_5_0	252.9168	259.5778	9603.923
_ARIMA_5_1	251.1175	259.1107	8385.319
_ARIMA_5_2	252.8778	262.2032	8313.834
_ARIMA_5_3	253.6647	264.3223	7961.325
_ARIMA_5_4	253.9771	265.9669	7495.664
_ARIMA_5_5	271.754	285.076	13167.95

Kauno gyventojų išlaidų ARIMA modelių parinkimo kriterijai

Model	AIC	BIC	SSE
_ARIMA_0_0	558.2535	558.2535	33681.17
_ARIMA_0_1	548.8998	551.0107	27961.03
_ARIMA_0_2	543.2877	547.5095	24680.77
_ARIMA_0_3	544.2475	550.5802	24263.46
_ARIMA_0_4	545.6477	554.0912	24026.04
_ARIMA_0_5	538.2711	548.8255	20602.74
_ARIMA_0_6	536.1955	548.8607	19271.16
_ARIMA_0_7	537.7664	552.5425	19136.09
_ARIMA_1_0	543.5728	545.6837	25622.84
_ARIMA_1_1	545.4515	549.6733	25571.95
_ARIMA_1_2	544.2871	550.6197	24279.21
_ARIMA_1_3	546.0748	554.5183	24194.84
_ARIMA_1_4	547.6287	558.1831	24018.55
_ARIMA_1_5	537.1643	549.8295	19579.68
_ARIMA_1_6	537.8208	552.5969	19153.15
_ARIMA_1_7	539.4155	556.3025	19026.33
_ARIMA_2_0	545.2965	549.5182	25507.03
_ARIMA_2_1	531.248	537.5807	19606.58
_ARIMA_2_2	523.5699	532.0134	14200.44
_ARIMA_2_3	548.2665	558.8209	24271.03
_ARIMA_2_4	514.8893	527.5545	13589.92
_ARIMA_2_5	551.4745	566.2506	23957.92
_ARIMA_2_6	516.1665	533.0535	12996.65
_ARIMA_2_7	518.0498	537.0476	12971.81
_ARIMA_3_0	541.616	547.9486	23239
_ARIMA_3_1	520.223	528.6665	15836.85
_ARIMA_3_2	514.2257	524.7801	13890.94
_ARIMA_3_3	513.864	526.5292	13363.41

_ARIMA_3_4	517.8938	532.6699	13815.56
_ARIMA_3_5	516.1674	533.0544	12996.84
_ARIMA_3_6	577.2179	596.2158	34217.87
_ARIMA_3_7	519.9652	541.0739	12953.84
_ARIMA_4_0	538.888	547.3315	21505.82
_ARIMA_4_1	519.8244	530.3788	15226.22
_ARIMA_4_2	516.1242	528.7894	13867.84
_ARIMA_4_3	529.6114	544.3875	16741.44
_ARIMA_4_4	555.496	572.383	24765.19
_ARIMA_4_5	508.751	527.7489	11137.75
_ARIMA_4_6	518.2729	539.3816	12599.4
_ARIMA_4_7	529.9077	553.1273	14755.23
_ARIMA_5_0	534.3245	544.8789	19311.97
_ARIMA_5_1	518.6724	531.3376	14459.42
_ARIMA_5_2	516.3003	531.0764	13459.32
_ARIMA_5_3	528.098	544.985	15804.44
_ARIMA_5_4	524.9049	543.9027	14514.63
_ARIMA_5_5	527.6002	548.7089	14681.02
_ARIMA_5_6	534.3863	557.6059	15879.31
_ARIMA_5_7	532.7265	558.0569	14954.61
_ARIMA_6_0	529.5847	542.2499	17291.87
_ARIMA_6_1	518.4509	533.2271	13942.32
_ARIMA_6_2	517.2073	534.0943	13220.31
_ARIMA_6_3	510.4352	529.433	11449.54
_ARIMA_6_4	551.5413	572.6501	21737.39
_ARIMA_6_5	519.2238	542.4434	12384.56
_ARIMA_6_6	523.1001	548.4306	12771.43
_ARIMA_6_7	537.3743	564.8157	15618.04
_ARIMA_7_0	525.5385	540.3147	15660.15
_ARIMA_7_1	518.6753	535.5623	13542.33
_ARIMA_7_2	517.3636	536.3615	12826.71
_ARIMA_7_3	510.2668	531.3756	11049.69
_ARIMA_7_4	567.9988	591.2185	27551.09
_ARIMA_7_5	514.0218	539.3523	11005.4
_ARIMA_7_6	517.3063	544.7477	11239.6
_ARIMA_7_7	540.5287	570.081	15916.43

Kauno gyventojų išlaidų SARIMA modelių parinkimo kriterijai

Model	AIC	BIC	SSE
_SARIMA_0_0	428.9917	428.9917	15583.69
_SARIMA_0_1	423.6046	425.5166	13443.32
_SARIMA_0_2	424.6503	428.4744	13189.19
_SARIMA_0_3	424.4254	430.1614	12615.15

_SARIMA_0_4	422.3503	429.9983	11627.77
_SARIMA_0_5	423.985	433.5451	11543.13
_SARIMA_0_6	425.9722	437.4443	11540.17
_SARIMA_0_7	425.9722	437.4443	11540.17
_SARIMA_1_0	421.1116	423.0236	12789.48
_SARIMA_1_1	418.4913	422.3153	11660.61
_SARIMA_1_2	420.3954	426.1314	11638.26
_SARIMA_1_3	421.7366	429.3847	11485.94
_SARIMA_1_4	423.6821	433.2422	11473.42
_SARIMA_1_5	424.9526	436.4247	11307.23
_SARIMA_1_6	426.9426	440.3267	11304.96
_SARIMA_1_7	425.9722	437.4443	11540.17
_SARIMA_2_0	421.7196	425.5436	12438.32
_SARIMA_2_1	420.4289	426.165	11646.08
_SARIMA_2_2	422.1085	429.7566	11571.68
_SARIMA_2_3	423.7196	433.2797	11482.02
_SARIMA_2_4	425.7085	437.1807	11479.48
_SARIMA_2_5	426.9476	440.3317	11306.1
_SARIMA_2_6	427.5899	442.8861	11003.23
_SARIMA_2_7	425.9722	437.4443	11540.17
_SARIMA_3_0	420.2005	425.9366	11592.99
_SARIMA_3_1	421.467	429.1151	11424.17
_SARIMA_3_2	423.4425	433.0026	11418.56
_SARIMA_3_3	420.5287	432.0008	10349.77
_SARIMA_3_4	422.5195	435.9036	10347.87
_SARIMA_3_5	427.1326	442.4287	10903.04
_SARIMA_3_6	427.8333	445.0415	10623.37
_SARIMA_3_7	425.9722	437.4443	11540.17
_SARIMA_4_0	421.3506	428.9987	11397.59
_SARIMA_4_1	423.3314	432.8916	11393.23
_SARIMA_4_2	425.3166	436.7888	11389.86
_SARIMA_4_3	422.2247	435.6089	10287.04
_SARIMA_4_4	424.6034	439.8996	10365.25
_SARIMA_4_5	432.6646	449.8728	11701.09
_SARIMA_4_6	426.4963	445.6166	9937.519
_SARIMA_4_7	425.9722	437.4443	11540.17
_SARIMA_5_0	423.3404	432.9005	11395.26
_SARIMA_5_1	427.6502	439.1224	11934.05
_SARIMA_5_2	426.4224	439.8065	11187.96
_SARIMA_5_3	425.3038	440.5999	10511.46
_SARIMA_5_4	427.2725	444.4807	10504.89
_SARIMA_5_5	420.414	439.5342	8799.278
_SARIMA_5_6	430.275	451.3072	10297.39
_SARIMA_5_7	425.9722	437.4443	11540.17
_SARIMA_6_0	425.2734	436.7455	11380.01

_SARIMA_6_1	427.1984	440.5825	11362.95
_SARIMA_6_2	428.2469	443.543	11148.75
_SARIMA_6_3	431.9793	449.1875	11541.81
_SARIMA_6_4	430.6345	449.7548	10794.98
_SARIMA_6_5	423.3017	444.3339	8956.899
_SARIMA_6_6	433.5097	456.4539	10554.83
_SARIMA_6_7	425.9722	437.4443	11540.17
_SARIMA_7_0	425.2734	436.7455	11380.01
_SARIMA_7_1	427.1984	440.5825	11362.95
_SARIMA_7_2	428.2469	443.543	11148.75
_SARIMA_7_3	431.9793	449.1875	11541.81
_SARIMA_7_4	430.6345	449.7548	10794.98
_SARIMA_7_5	423.3017	444.3339	8956.899
_SARIMA_7_6	433.5097	456.4539	10554.83
_SARIMA_7_7	425.9722	437.4443	11540.17

Klaipėdos gyventojų išlaidų ARIMA modelių parinkimo kriterijai

Model	AIC	BIC	SSE
_ARIMA_0_0	447.2826	447.2826	26427.46
_ARIMA_0_1	440.1873	442.0791	21950.4
_ARIMA_0_2	440.1611	443.9447	21061.24
_ARIMA_0_3	441.8556	447.5311	20930.35
_ARIMA_0_4	439.1361	446.7034	19008.46
_ARIMA_0_5	440.4289	449.888	18736.08
_ARIMA_0_6	437.3292	448.6801	16884.15
_ARIMA_1_0	438.3705	440.2623	21151.45
_ARIMA_1_1	440.361	444.1446	21147.34
_ARIMA_1_2	442.0075	447.6829	20995.32
_ARIMA_1_3	443.173	450.7403	20640.82
_ARIMA_1_4	438.6869	448.146	18081.69
_ARIMA_1_5	438.7287	450.0796	17373.35
_ARIMA_1_6	438.466	451.7087	16589.32
_ARIMA_2_0	440.3552	444.1389	21144.86
_ARIMA_2_1	442.3683	448.0438	21150.51
_ARIMA_2_2	423.9192	431.4865	13934.05
_ARIMA_2_3	444.5052	453.9643	20361.39
_ARIMA_2_4	449.9568	461.3077	21847.4
_ARIMA_2_5	420.8593	434.102	11581.85
_ARIMA_2_6	432.4981	447.6327	14099.65
_ARIMA_3_0	441.3684	447.0438	20723.27
_ARIMA_3_1	430.4164	437.9837	15909.74
_ARIMA_3_2	425.8033	435.2624	13901.13
_ARIMA_3_3	444.286	455.637	19459.83

_ARIMA_3_4	421.0534	434.2962	11627.82
_ARIMA_3_5	441.6042	456.7388	16979.2
_ARIMA_3_6	424.0783	441.1047	11398.71
_ARIMA_4_0	440.73	448.2972	19636.92
_ARIMA_4_1	443.3471	452.8062	19885.8
_ARIMA_4_2	426.1505	437.5014	13440.04
_ARIMA_4_3	428.7516	441.9944	13605.95
_ARIMA_4_4	449.2733	464.4079	19855.88
_ARIMA_4_5	425.155	442.1814	11651.95
_ARIMA_4_6	422.0314	440.9496	10495.11
_ARIMA_5_0	439.8086	449.2677	18500.38
_ARIMA_5_1	428.9167	440.2676	14220.59
_ARIMA_5_2	433.3107	446.5535	14932.65
_ARIMA_5_3	432.9914	448.126	14242.31
_ARIMA_5_4	453.2667	470.2931	20680.31
_ARIMA_5_5	434.8403	453.7585	13630.6
_ARIMA_5_6	423.8106	444.6206	10447.93
_ARIMA_6_0	428.4272	439.7781	14079.25
_ARIMA_6_1	425.453	438.6958	12720.17
_ARIMA_6_2	427.3988	442.5334	12706.11
_ARIMA_6_3	423.8438	440.8702	11344.29
_ARIMA_6_4	420.7265	439.6447	10219.31
_ARIMA_6_5	421.2303	442.0404	9911.995
_ARIMA_6_6	421.5156	444.2174	9571.122

Klaipėdos gyventojų išlaidų SARIMA modelių parinkimo kriterijai

Modelis	AIC	BIC	SSE
_SARIMA_0_0	331.0729	331.0729	13523.1
_SARIMA_0_1	320.5254	322.163	9720.153
_SARIMA_0_2	322.2392	325.5143	9647.211
_SARIMA_0_3	322.4966	327.4094	9214.813
_SARIMA_0_4	322.8734	329.4238	8829.488
_SARIMA_0_5	321.7801	329.968	8139.207
_SARIMA_0_6	323.1791	333.0046	8011.496
_SARIMA_1_0	319.4181	321.0557	9440.992
_SARIMA_1_1	321.1821	324.4573	9382.557
_SARIMA_1_2	321.2873	326.2	8926.17
_SARIMA_1_3	322.0983	328.6486	8651.201
_SARIMA_1_4	324.0983	332.2862	8651.2
_SARIMA_1_5	326.0791	335.9046	8646.835
_SARIMA_1_6	326.9158	338.3789	8386.143
_SARIMA_2_0	321.3985	324.6737	9436.128
_SARIMA_2_1	322.6109	327.5236	9242.558
_SARIMA_2_2	320.1937	326.7441	8228.295

_SARIMA_2_3	324.0983	332.2862	8651.201
_SARIMA_2_4	325.8891	335.7146	8603.709
_SARIMA_2_5	326.7498	338.2129	8349.579
_SARIMA_2_6	329.122	342.2227	8431.773
_SARIMA_3_0	320.3983	325.311	8719.766
_SARIMA_3_1	322.3979	328.9482	8719.676
_SARIMA_3_2	322.2053	330.3932	8230.794
_SARIMA_3_3	323.399	333.2245	8057.982
_SARIMA_3_4	327.99	339.4531	8626.588
_SARIMA_3_5	326.3478	339.4485	7838.139
_SARIMA_3_6	329.9769	344.7152	8181.471
_SARIMA_4_0	322.3981	328.9484	8719.72
_SARIMA_4_1	324.3898	332.5778	8717.831
_SARIMA_4_2	325.6873	335.5128	8558.131
_SARIMA_4_3	324.7843	336.2474	7928.681
_SARIMA_4_4	331.8171	344.9178	9051.492
_SARIMA_4_5	331.2941	346.0324	8470.039
_SARIMA_4_6	335.9357	352.3116	9079.793
_SARIMA_5_0	324.2146	332.4026	8677.73
_SARIMA_5_1	324.4796	334.3051	8290.414
_SARIMA_5_2	329.5739	341.037	8993.75
_SARIMA_5_3	328.723	341.8237	8343.705
_SARIMA_5_4	331.137	345.8753	8435.095
_SARIMA_5_5	348.0849	364.4608	12500.49
_SARIMA_5_6	342.0714	360.0848	10123.76
_SARIMA_6_0	325.9602	335.7857	8619.814
_SARIMA_6_1	328.0111	339.4742	8631.374
_SARIMA_6_2	329.143	342.2437	8436.431
_SARIMA_6_3	330.4391	345.1774	8281.604
_SARIMA_6_4	336.8848	353.2607	9309.434
_SARIMA_6_5	344.3442	362.3577	10747.75
_SARIMA_6_6	348.4583	368.1094	11362.65

5 Priedas. Gyventojų išlaidų prognozavimo modelių ir analizės SAS programos kodas

Vilniaus gyventojų išlaidų, būsto išlaikymui, prognozavimo modelių sudarymas

```
/*           Sudaromas Vilniaus gyventojų išlaidų, būsto išlaikymui, duomenų
failas*/
data vilnius;
           input date yymmdd10. v_bendri 12-20 v_bendri_stdev 21-30 v_bendri_n
31-35
           label v_bendri='Vilniaus gyventojų išlaidos, būsto išlaikymui';
           datalines;
2012.10.01 99.2504    51.750167 219
..................................
;
run;
/*           Grafinės parinktys           */
ODS GRAPHICS ON;
SYMBOL1
           V=DOT C=Blue I=JOIN H=0.6 W=1;
AXIS2 LABEL
           =(ANGLE=90 'Bendri mokesčiai (€)');
AXIS1 LABEL=('Data');
/*           Braižomas Vilniečių išlaidų grafikas*/
proc gplot data = vilnius;
           format date yymmdd10.;
           plot v_bendri*date / overlay frame grid
                      haxis=axis1 vaxis=axis2;
run;
/*           Tiriama laiko eilutė, nustatomas stacionarumas, parenkama
modelio eilė*/
proc arima data=vilnius;
           title1 ' Bendrų mokesčių stacionarumo tikrinimas. Geriausio modelio
parinkimas bei analizė';
           identify var =v_bendri  stationarity=(adf);
           identify var =v_bendri(1) stationarity=(adf) minic perror=( 1:6 )
q=( 1:6 ) scan;
_ARIMA_0_0:
           estimate p=0 q=0 noint outstat=a1;
_ARIMA_0_1:
           estimate p=0 q=1 noint outstat=a2;
_ARIMA_0_2:
           estimate p=0 q=2 noint outstat=a3;
_ARIMA_0_3:
           estimate p=0 q=3 noint outstat=a4;
_ARIMA_0_4:
           estimate p=0 q=4 noint outstat=a5;
_ARIMA_0_5:
           estimate p=0 q=5 noint outstat=a6;
_ARIMA_1_0:
           estimate p=1 q=0 noint outstat=a7;
_ARIMA_1_1:
           estimate p=1 q=1 noint outstat=a8;
_ARIMA_1_2:
           estimate p=1 q=2 noint outstat=a9;
_ARIMA_1_3:
           estimate p=1 q=3 noint outstat=a10;
_ARIMA_1_4:
           estimate p=1 q=4 noint outstat=a11;
```

```
_ARIMA_1_5:
           estimate p=1 q=5 noint outstat=a12;
_ARIMA_2_0:
           estimate p=2 q=0 noint outstat=a13;
_ARIMA_2_1:
           estimate p=2 q=1 noint outstat=a14;
_ARIMA_2_2:
           estimate p=2 q=2 noint outstat=a15;
_ARIMA_2_3:
           estimate p=2 q=3 noint outstat=a16;
_ARIMA_2_4:
           estimate p=2 q=4 noint outstat=a17;
_ARIMA_2_5:
           estimate p=2 q=5 noint outstat=a18;
_ARIMA_3_0:
           estimate p=3 q=0 noint outstat=a19;
_ARIMA_3_1:
           estimate p=3 q=1 noint outstat=a20;
_ARIMA_3_2:
           estimate p=3 q=2 noint outstat=a21;
_ARIMA_3_3:
           estimate p=3 q=3 noint outstat=a22;
_ARIMA_3_4:
           estimate p=3 q=4 noint outstat=a23;
_ARIMA_3_5:
           estimate p=3 q=5 noint outstat=a24;
_ARIMA_4_0:
           estimate p=4 q=0 noint outstat=a25;
_ARIMA_4_1:
           estimate p=4 q=1 noint outstat=a26;
_ARIMA_4_2:
           estimate p=4 q=2 noint outstat=a27;
_ARIMA_4_3:
           estimate p=4 q=3 noint outstat=a28;
_ARIMA_4_4:
           estimate p=4 q=4 noint outstat=a29;
_ARIMA_4_5:
           estimate p=4 q=5 noint outstat=a30;
_ARIMA_5_0:
           estimate p=5 q=0 noint outstat=a31;
_ARIMA_5_1:
           estimate p=5 q=1 noint outstat=a32;
_ARIMA_5_2:
           estimate p=5 q=2 noint outstat=a33;
_ARIMA_5_3:
           estimate p=5 q=3 noint outstat=a34;
_ARIMA_5_4:
           estimate p=5 q=4 noint outstat=a35;
_ARIMA_5_5:
           estimate p=5 q=5 noint outstat=a36;
run;

/*             Sudaroma modelio kriterijų lentelė*/
data aic(keep=_MODLABEL_ AIC);
           set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15 a16 a17 a18
a19
                      a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32 a33
a34 a35 a36;

           if _STAT_='AIC';
           AIC=_VALUE_;
```

```
run;

data bic (keep=_MODLABEL_ BIC);
          set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15 a16 a17 a18
a19
                    a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32 a33
a34 a35 a36;

          if _STAT_='SBC';
          BIC=_VALUE_;
run;

data sse (keep=_MODLABEL_ SSE);
          set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15 a16 a17 a18
a19
                    a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32 a33
a34 a35 a36;

          if _STAT_='SSE';
          SSE=_VALUE_;
run;

data aic_bic_sse;
          merge aic bic sse;
          by _MODLABEL_;
run;
/*            Spausdinama modelio kriterijų lentelė*/
proc print data=aic_bic_sse;
run;

/*           Parenkamas modelis ir atliekama prognozė*/
proc arima data=vilnius   plots=forecast(forecast);
          identify var =v_bendri(1);
          estimate p=1 q=0 noint;
run;

forecast lead=5 interval=month id=date out=Rez_v_bendri1 printall;
run;
/*           Suskaičiuojama MAPE*/
data paklaida1;
          set Rez_v_bendri1;
          MAPE=abs((v_bendri-FORECAST)/v_bendri)*100;
run;

proc summary data=paklaida1;
          var MAPE;
          output out=paklaida1 mean()=;
run;
/*           Spausdinama MAPE*/
proc print data=paklaida1;
          var MAPE;
run;

/*           Atliekama modelio paklaidų analizė, nustatomas
heteroskedastiškumas*/
proc autoreg data=Rez_v_bendri1;
          model RESIDUAL=/ noint archtest;
run;

/*           Parenkamas kitas modelis ir atliekama prognozė*/
proc arima data=vilnius  plots=forecast(forecast);
```

```
          identify var =v_bendri(1);
          estimate p=5 q=4 noint;
run;

forecast lead=5 interval=month id=date out=Rez_v_bendri2 printall;
run;

/*           Suskaičiuojama MAPE*/
data paklaida2;
          set Rez_v_bendri2;
          MAPE=abs((v_bendri-FORECAST)/v_bendri)*100;
run;

proc summary data=paklaida2;
          var MAPE;
          output out=paklaida2 mean()=;
run;
/*           Spausdinama MAPE*/
proc print data=paklaida2;
          var MAPE;
run;
/*                                Paliekami reikalingi kintamieji*/
data Paklaidos(keep= date RESIDUAL STD FORECAST v_bendri);
          set Rez_v_bendri2;
run;

/*           Atliekama modelio paklaidų analizė, nustatomas
heteroskedastiškumas*/
proc autoreg data=Paklaidos;
          model RESIDUAL=/ noint archtest;
run;

/*           Sudaromi GARCH modeliai ir nustatomas geriausias*/
ods output Autoreg.arch_1.FinalModel.Results.FitSummary
          =fitsum_arch_1;
ods output Autoreg.arch_2.FinalModel.Results.FitSummary
          =fitsum_arch_2;
ods output Autoreg.arch_3.FinalModel.Results.FitSummary
          =fitsum_arch_3;
ods output Autoreg.garch_1_1.FinalModel.Results.FitSummary
          =fitsum_garch_1_1;

proc autoreg data=Paklaidos outest=garch_family;
          arch_1 :              model RESIDUAL = / noint dwprob
stationarity=(adf) archtest garch=(q=1);
          arch_2 :              model RESIDUAL = / noint dwprob
stationarity=(adf) archtest garch=(q=2);
          arch_3 :              model RESIDUAL = / noint dwprob
stationarity=(adf) archtest garch=(q=3);
          garch_1_1 :           model RESIDUAL = / noint dwprob
stationarity=(adf) archtest garch=(q=1, p=1);
run;

/*          Sudarytųjų GARCH modelių parametrų lentelė*/
proc print data=garch_family;
          var _MODEL_ _AH_0 _AH_1 _AH_2 _AH_3 _GH_1;
run;

/*          GARCH modelių tinkamumo kriterijų lentelė*/
data sbc_aic;
          set fitsum_arch_1 fitsum_arch_2 fitsum_arch_3 fitsum_garch_1_1;
```

```
            keep Model SBC AIC;

            if Label1="SBC" then
                        do;
                                    SBC=input(cValue1,BEST12.4);
                        end;

            if Label2="SBC" then
                        do;
                                    SBC=input(cValue2,BEST12.4);
                        end;

            if Label1="AIC" then
                        do;
                                    AIC=input(cValue1,BEST12.4);
                        end;

            if Label2="AIC" then
                        do;
                                    AIC=input(cValue2,BEST12.4);
                        end;

            if not (SBC=.) then
                        output;
run;

proc sort data=sbc_aic;
            by AIC;
run;

proc print data=sbc_aic;
            format _NUMERIC_ BEST12.4;
run;

/*             Sudaromas ARCH(1) modelis paklaidoms*/
proc autoreg data=Paklaidos plots=all;
            model RESIDUAL = / noint garch=(q=1) maxit=50;
            output out=a cev=v cpev=pr predicted=as predictedm=ad r=r rm=rm;
run;

/* Apskaičiuojamos prognozuojamos paklaidos*/
data Paklaidos;
            set Paklaidos;
            sigma1=lag(RESIDUAL);
            sigma=sqrt(98.408+0.731*sigma1**2);
            liekana=RESIDUAL/sigma;
run;

/* Apskaičiuojamos prognozuojamos reikšmės*/
data Paklaidos;
            set Paklaidos;
            Prognoze=FORECAST+sigma;
run;

/* Braižomas prognozės grafikas*/
proc gplot data = Paklaidos;
            format date yymmdd10.;
            plot Prognoze*date v_bendri*date/ overlay frame grid;
run;

/*                      Vilniaus gyventojų išlaidų, būsto išlaikymui
```

```
                     regresinių modelių paklaidų tyrimas*/
data vilnius_regresiniai;
input date yymmdd10. v_bendri 12-20 v_bendri_trend 21-32 v_bendri_ad_prog 33-
41 v_bendri_ad_error 42-51 v_bendri_mult_prog 52-63 v_bendri_mult_error 64-
76;
label;
datalines;
2012.10.01  99.2504 148.2784797 110.3074 -11.05693 108.1894459 -8.939019907
............................................................................
;
run;

ODS GRAPHICS ON;/* Grafiniu priemonių įrankio naudojimo nustatymas*/

/*                     Spausdinama regresinių modelių paklaidos*/
proc print data=vilnius_regresiniai;
run;

/*                     Tiriamos adityvaus regresinio modelio paklaidos*/
proc autoreg data=vilnius_regresiniai;
model v_bendri_ad_error= / noint archtest dwprob;
run;
proc arima data=vilnius_regresiniai PLOTS(UNPACK)=Series();
identify var=v_bendri_ad_error stationarity=(adf);
run;

/*                     Tiriamos multiplikatyvaus regresinio modelio paklaidos*/
proc arima data=vilnius_regresiniai PLOTS(UNPACK)=Series();
identify var=v_bendri_mult_error stationarity=(adf);
run;
proc autoreg data=vilnius_regresiniai;
model v_bendri_mult_error= / noint archtest dwprob;
run;
```

Kauno gyventojų išlaidų, būsto išlaikymui, prognozavimo modelių sudarymas

```
/*            Sudaromas Kauno gyventojų išlaidų, būsto išlaikymui, duomenų
failas*/
data kaunas;
           input date yymmdd10. k_bendri 12-20 k_bendri_stdev 21-31 k_bendri_n
32-37;
           label k_bendri='Kauno gyventojų išlaidos būsto išlaikymui';
           datalines;
2010.01.01 129.7638 47.0521665 3714
...................................
;
run;

/*            Grafinės parinktys           */
ODS GRAPHICS ON;
SYMBOL1
           V=DOT C=Blue I=JOIN H=0.6 W=1;
AXIS2 LABEL
           =(ANGLE=90 'Bendri mokesčiai (€)');
AXIS1 LABEL=('Data');

/*            Braižomas Kauniečių išlaidų grafikas*/
proc gplot data = Kaunas;
           format date yymmdd10.;
           plot k_bendri*date / overlay frame grid
```

```
                        haxis=axis1 vaxis=axis2;
run;

/*            Tiriama laiko eilutė, nustatomas stacionarumas, parenkama
modelio eilė*/
proc arima data=kaunas            plots=forecast(forecast);
           title1 ' Bendrų mokesčių stacionarumo tikrinimas. Geriausio modelio
parinkimas bei analizė';
           identify var =k_bendri  stationarity=(adf);
           identify var =k_bendri(1) stationarity=(adf) minic scan;

           _ARIMA_0_0:
           estimate p=0 q=0 noint outstat=a1;
           _ARIMA_0_1:
           estimate p=0 q=1 noint outstat=a2;
           _ARIMA_0_2:
           estimate p=0 q=2 noint outstat=a3;
           _ARIMA_0_3:
           estimate p=0 q=3 noint outstat=a4;
           _ARIMA_0_4:
           estimate p=0 q=4 noint outstat=a5;
           _ARIMA_0_5:
           estimate p=0 q=5 noint outstat=a6;
           _ARIMA_0_6:
           estimate p=0 q=6 noint outstat=a7;
           _ARIMA_1_0:
           estimate p=1 q=0 noint outstat=a8;
           _ARIMA_1_1:
           estimate p=1 q=1 noint outstat=a9;
           _ARIMA_1_2:
           estimate p=1 q=2 noint outstat=a10;
           _ARIMA_1_3:
           estimate p=1 q=3 noint outstat=a11;
           _ARIMA_1_4:
           estimate p=1 q=4 noint outstat=a12;
           _ARIMA_1_5:
           estimate p=1 q=5 noint outstat=a13;
           _ARIMA_1_6:
           estimate p=1 q=6 noint outstat=a14;
           _ARIMA_2_0:
           estimate p=2 q=0 noint outstat=a15;
           _ARIMA_2_1:
           estimate p=2 q=1 noint outstat=a16;
           _ARIMA_2_2:
           estimate p=2 q=2 noint outstat=a17;
           _ARIMA_2_3:
           estimate p=2 q=3 noint outstat=a18;
           _ARIMA_2_4:
           estimate p=2 q=4 noint outstat=a19;
           _ARIMA_2_5:
           estimate p=2 q=5 noint outstat=a20;
           _ARIMA_2_6:
           estimate p=2 q=6 noint outstat=a21;
           _ARIMA_3_0:
           estimate p=3 q=0 noint outstat=a22;
           _ARIMA_3_1:
           estimate p=3 q=1 noint outstat=a23;
           _ARIMA_3_2:
           estimate p=3 q=2 noint outstat=a24;
           _ARIMA_3_3:
           estimate p=3 q=3 noint outstat=a25;
```

```
          _ARIMA_3_4:
          estimate p=3 q=4 noint outstat=a26;
          _ARIMA_3_5:
          estimate p=3 q=5 noint outstat=a27;
          _ARIMA_3_6:
          estimate p=3 q=6 noint outstat=a28;
          _ARIMA_4_0:
          estimate p=4 q=0 noint outstat=a29;
          _ARIMA_4_1:
          estimate p=4 q=1 noint outstat=a30;
          _ARIMA_4_2:
          estimate p=4 q=2 noint outstat=a31;
          _ARIMA_4_3:
          estimate p=4 q=3 noint outstat=a32;
          _ARIMA_4_4:
          estimate p=4 q=4 noint outstat=a33;
          _ARIMA_4_5:
          estimate p=4 q=5 noint outstat=a34;
          _ARIMA_4_6:
          estimate p=4 q=6 noint outstat=a35;
          _ARIMA_5_0:
          estimate p=5 q=0 noint outstat=a36;
          _ARIMA_5_1:
          estimate p=5 q=1 noint outstat=a37;
          _ARIMA_5_2:
          estimate p=5 q=2 noint outstat=a38;
          _ARIMA_5_3:
          estimate p=5 q=3 noint outstat=a39;
          _ARIMA_5_4:
          estimate p=5 q=4 noint outstat=a40;
          _ARIMA_5_5:
          estimate p=5 q=5 noint outstat=a41;
          _ARIMA_5_6:
          estimate p=5 q=6 noint outstat=a42;
          _ARIMA_6_0:
          estimate p=6 q=0 noint outstat=a43;
          _ARIMA_6_1:
          estimate p=6 q=1 noint outstat=a44;
          _ARIMA_6_2:
          estimate p=6 q=2 noint outstat=a45;
          _ARIMA_6_3:
          estimate p=6 q=3 noint outstat=a46;
          _ARIMA_6_4:
          estimate p=6 q=4 noint outstat=a47;
          _ARIMA_6_5:
          estimate p=6 q=5 noint outstat=a48;
          _ARIMA_6_6:
          estimate p=6 q=6 noint outstat=a49;

          ;
          run;

/*        GARCH modelių tinkamumo kriterijų lentelė*/
          data aic(keep=_MODLABEL_ AIC);
          set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15
          a16 a17 a18 a19 a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32
          a33 a34 a35 a36 a37 a38 a39 a40 a41 a42 a43 a44 a45 a46 a47 a48 a49
          ;
          if _STAT_='AIC';
                     AIC=_VALUE_;
          run;
```

```
        data bic (keep=_MODLABEL_ BIC);
        set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15
        a16 a17 a18 a19 a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32
        a33 a34 a35 a36 a37 a38 a39 a40 a41 a42 a43 a44 a45 a46 a47 a48 a49
        ;
        if _STAT_='SBC';
                BIC=_VALUE_;
        run;

        data sse (keep=_MODLABEL_ SSE);
        set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15
        a16 a17 a18 a19 a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32
        a33 a34 a35 a36 a37 a38 a39 a40 a41 a42 a43 a44 a45 a46 a47 a48 a49
        ;
        if _STAT_='SSE';
                SSE=_VALUE_;
        run;

        data aic_bic_sse;
        merge aic bic sse;
        by _MODLABEL_;
        run;

/*          Spausdinama modelių kriterijų lentelė*/
        proc print data=aic_bic_sse;
        run;

/*         Parenkamas modelis ir atliekama prognozė*/
        proc arima data=kaunas            plots=forecast(forecast);
        identify var =k_bendri(1) stationarity=(adf) minic scan;
        estimate p=3 q=1 noint;
        forecast lead=5 interval=month id=date out=Rez_k_bendri1 printall;
        run;

/*         Suskaičiuojama MAPE*/
        data paklaida1;
        set Rez_k_bendri1;
        MAPE=abs((k_bendri-FORECAST)/k_bendri)*100;
        run;

        proc summary data=paklaida1;
        var MAPE;
        output out=paklaida1 mean()=;
        run;

/*         Spausdinama MAPE*/
        proc print data=paklaida1;
        var MAPE;
        run;
/*         Atliekama modelio paklaidų analizė, nustatomas
heteroskedastiškumas*/
        proc autoreg data=Rez_k_bendri1;
        model RESIDUAL=/ noint archtest;
        run;

/*      Paliekami reikalingi kintamieji*/
        data Paklaidos(keep= date RESIDUAL STD FORECAST k_bendri);
        set Rez_k_bendri1;
        run;
```

```
/*          Sudaromi GARCH modeliai ir nustatomas geriausias*/
           ods output Autoreg.arch_1.FinalModel.Results.FitSummary
                    =fitsum_arch_1;
           ods output Autoreg.arch_2.FinalModel.Results.FitSummary
                    =fitsum_arch_2;
           ods output Autoreg.arch_3.FinalModel.Results.FitSummary
                    =fitsum_arch_3;
           ods output Autoreg.garch_1_1.FinalModel.Results.FitSummary
                    =fitsum_garch_1_1;

           proc autoreg data=Paklaidos outest=garch_family;
           arch_1 :             model RESIDUAL = / noint dwprob
stationarity=(adf) archtest garch=(q=1);
           arch_2 :             model RESIDUAL = / noint dwprob
stationarity=(adf) archtest garch=(q=2);
           arch_3 :             model RESIDUAL = / noint dwprob
stationarity=(adf) archtest garch=(q=3);
           garch_1_1 :          model RESIDUAL = / noint dwprob
stationarity=(adf) archtest garch=(q=1, p=1);
           run;

/*          Sudarytųjų GARCH modelių parametrų lentelė*/
           proc print data=garch_family;
            var _MODEL_ _AH_0 _AH_1 _AH_2 _AH_3 _GH_1  ;
           run;

/*          GARCH modelių tinkamumo kriterijų lentelė*/
           data sbc_aic;
            set fitsum_arch_1 fitsum_arch_2 fitsum_arch_3 fitsum_garch_1_1 ;
            keep Model SBC AIC;
            if Label1="SBC" then do; SBC=input(cValue1,BEST12.4); end;
            if Label2="SBC" then do; SBC=input(cValue2,BEST12.4); end;
            if Label1="AIC" then do; AIC=input(cValue1,BEST12.4); end;
            if Label2="AIC" then do; AIC=input(cValue2,BEST12.4); end;
            if not (SBC=.) then output;
           run;

           proc sort data=sbc_aic;
            by AIC;
           run;

/*          Spausdinama GARCH modelių tinkamumo kriterijai*/
           proc print data=sbc_aic;
            format _NUMERIC_ BEST12.4;
           run;

/*            Sudaromas ARCH(1) modelis paklaidoms*/
           proc autoreg data=Paklaidos plots=all ;
           model RESIDUAL = / noint garch=(q=1) maxit=50;
           output out=a cev=v cpev=pr predicted=as predictedm=ad r=r rm=rm;
           run;
/* Apskaičiuojamos prognozuojamos paklaidos*/
           data Paklaidos;
           set Paklaidos;
           sigma1=lag(RESIDUAL);
           sigma=sqrt(165.927+0.356*sigma1**2);
           liekana=RESIDUAL/sigma;
           run;

/* Apskaičiuojamos prognozuojamos reikšmės*/
           data Paklaidos;
```

```
          set Paklaidos;
          Prognoze=FORECAST+sigma;
          run;

/* Braizomas prognozes grafikas*/
          proc gplot data = Paklaidos;
          format date yymmdd10.;
          plot Prognoze*date k_bendri*date/ overlay frame grid legend ;
          run;

/*            Tiriama SARIMA modeliai, nustatomas stacionarumas, parenkama
modelio eilė*/
          proc arima data=kaunas  plots=forecast(forecast) ;
          identify var =k_bendri(12) stationarity=(adf) minic scan ;

          _SARIMA_0_0:
          estimate p=0 q=0 noint outstat=b1;
          _SARIMA_0_1:
          estimate p=0 q=1 noint outstat=b2;
          _SARIMA_0_2:
          estimate p=0 q=2 noint outstat=b3;
          _SARIMA_0_3:
          estimate p=0 q=3 noint outstat=b4;
          _SARIMA_0_4:
          estimate p=0 q=4 noint outstat=b5;
          _SARIMA_0_5:
          estimate p=0 q=5 noint outstat=b6;
          _SARIMA_0_6:
          estimate p=0 q=6 noint outstat=b7;
          _SARIMA_0_7:
          estimate p=0 q=7 noint outstat=b8;
          _SARIMA_1_0:
          estimate p=1 q=0 noint outstat=b9;
          _SARIMA_1_1:
          estimate p=1 q=1 noint outstat=b10;
          _SARIMA_1_2:
          estimate p=1 q=2 noint outstat=b11;
          _SARIMA_1_3:
          estimate p=1 q=3 noint outstat=b12;
          _SARIMA_1_4:
          estimate p=1 q=4 noint outstat=b13;
          _SARIMA_1_5:
          estimate p=1 q=5 noint outstat=b14;
          _SARIMA_1_6:
          estimate p=1 q=6 noint outstat=b15;
          _SARIMA_1_7:
          estimate p=1 q=7 noint outstat=b16;
          _SARIMA_2_0:
          estimate p=2 q=0 noint outstat=b17;
          _SARIMA_2_1:
          estimate p=2 q=1 noint outstat=b18;
          _SARIMA_2_2:
          estimate p=2 q=2 noint outstat=b19;
          _SARIMA_2_3:
          estimate p=2 q=3 noint outstat=b20;
          _SARIMA_2_4:
          estimate p=2 q=4 noint outstat=b21;
          _SARIMA_2_5:
          estimate p=2 q=5 noint outstat=b22;
          _SARIMA_2_6:
          estimate p=2 q=6 noint outstat=b23;
```

```
_SARIMA_2_7:
estimate p=2 q=7 noint outstat=b24;
_SARIMA_3_0:
estimate p=3 q=0 noint outstat=b25;
_SARIMA_3_1:
estimate p=3 q=1 noint outstat=b26;
_SARIMA_3_2:
estimate p=3 q=2 noint outstat=b27;
_SARIMA_3_3:
estimate p=3 q=3 noint outstat=b28;
_SARIMA_3_4:
estimate p=3 q=4 noint outstat=b29;
_SARIMA_3_5:
estimate p=3 q=5 noint outstat=b30;
_SARIMA_3_6:
estimate p=3 q=6 noint outstat=b31;
_SARIMA_3_7:
estimate p=3 q=7 noint outstat=b32;
_SARIMA_4_0:
estimate p=4 q=0 noint outstat=b33;
_SARIMA_4_1:
estimate p=4 q=1 noint outstat=b34;
_SARIMA_4_2:
estimate p=4 q=2 noint outstat=b35;
_SARIMA_4_3:
estimate p=4 q=3 noint outstat=b36;
_SARIMA_4_4:
estimate p=4 q=4 noint outstat=b37;
_SARIMA_4_5:
estimate p=4 q=5 noint outstat=b38;
_SARIMA_4_6:
estimate p=4 q=6 noint outstat=b39;
_SARIMA_4_7:
estimate p=0 q=6 noint outstat=b40;
_SARIMA_5_0:
estimate p=5 q=0 noint outstat=b41;
_SARIMA_5_1:
estimate p=5 q=1 noint outstat=b42;
_SARIMA_5_2:
estimate p=5 q=2 noint outstat=b43;
_SARIMA_5_3:
estimate p=5 q=3 noint outstat=b44;
_SARIMA_5_4:
estimate p=5 q=4 noint outstat=b45;
_SARIMA_5_5:
estimate p=5 q=5 noint outstat=b46;
_SARIMA_5_6:
estimate p=5 q=6 noint outstat=b47;
_SARIMA_5_7:
estimate p=5 q=7 noint outstat=b48;
_SARIMA_6_0:
estimate p=6 q=0 noint outstat=b49;
_SARIMA_6_1:
estimate p=6 q=1 noint outstat=b50;
_SARIMA_6_2:
estimate p=6 q=2 noint outstat=b51;
_SARIMA_6_3:
estimate p=6 q=3 noint outstat=b52;
_SARIMA_6_4:
estimate p=6 q=4 noint outstat=b53;
_SARIMA_6_5:
```

```
            estimate p=6 q=5 noint outstat=b54;
            _SARIMA_6_6:
            estimate p=6 q=6 noint outstat=b55;
            _SARIMA_6_7:
            estimate p=6 q=7 noint outstat=b56;
            _SARIMA_7_0:
            estimate p=6 q=0 noint outstat=b57;
            _SARIMA_7_1:
            estimate p=6 q=1 noint outstat=b58;
            _SARIMA_7_2:
            estimate p=6 q=2 noint outstat=b59;
            _SARIMA_7_3:
            estimate p=6 q=3 noint outstat=b60;
            _SARIMA_7_4:
            estimate p=6 q=4 noint outstat=b61;
            _SARIMA_7_5:
            estimate p=6 q=5 noint outstat=b62;
            _SARIMA_7_6:
            estimate p=6 q=6 noint outstat=b63;
            _SARIMA_7_7:
            estimate p=7 q=7 noint outstat=b64;
run;

/*            Sudaroma modelio kriterijų lentelė*/
data aic1(keep=_MODLABEL_ AIC);
            set b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15
            b16 b17 b18 b19 b20 b21 b22 b23 b24 b25 b26 b27 b28 b29 b30 b31 b32
            b33 b34 b35 b36 b37 b38 b39 b40 b41 b42 b43 b44 b45 b46 b47 b48 b49
            b50 b51 b52 b53 b54 b55 b56 b57 b58 b59 b60 b61 b62 b63 b64
            ;
            if _STAT_='AIC';
                       AIC=_VALUE_;
            run;

            data bic1 (keep=_MODLABEL_ BIC);
            set b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15
            b16 b17 b18 b19 b20 b21 b22 b23 b24 b25 b26 b27 b28 b29 b30 b31 b32
            b33 b34 b35 b36 b37 b38 b39 b40 b41 b42 b43 b44 b45 b46 b47 b48 b49
            b50 b51 b52 b53 b54 b55 b56 b57 b58 b59 b60 b61 b62 b63 b64
            ;
            if _STAT_='SBC';
                       BIC=_VALUE_;
            run;

            data sse1 (keep=_MODLABEL_ SSE);
            set b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15
            b16 b17 b18 b19 b20 b21 b22 b23 b24 b25 b26 b27 b28 b29 b30 b31 b32
            b33 b34 b35 b36 b37 b38 b39 b40 b41 b42 b43 b44 b45 b46 b47 b48 b49
            b50 b51 b52 b53 b54 b55 b56 b57 b58 b59 b60 b61 b62 b63 b64
            ;

            if _STAT_='SSE';
                       SSE=_VALUE_;
            run;

            data aic_bic_sse1;
            merge aic1 bic1 sse1;
            by _MODLABEL_;
            run;

/*           Parenkamas modelis ir atliekama prognozė*/
```

```
                      proc arima data=kaunas  plots=forecast(forecast) ;
           identify var =k_bendri(12) stationarity=(adf) minic scan ;
           estimate p=1 q=1 noint;
           forecast lead=5 interval=month id=date out=Rez_k_bendri2 printall;
           run;

/*          Suskaičiuojama MAPE*/
           data paklaida2;
           set Rez_k_bendri2;
           MAPE=abs((k_bendri-FORECAST)/k_bendri)*100;
           run;

           proc summary data=paklaida2;
           var MAPE;
           output out=paklaida2 mean()=;
           run;

/*          Spausdinama MAPE*/
           proc print data=paklaida2;
           var MAPE;
           run;

/*          Atliekama modelio paklaidų analizė, nustatomas
heteroskedastiškumas*/
           proc autoreg data=Rez_k_bendri2;
           model RESIDUAL=/ noint archtest;
           run;

/*          Kauno gyventojų išlaidų, būsto išlaikymui
           regresinių modelių paklaidų tyrimas*/
data kaunas_regresiniai;
           input date yymmdd10. k_bendri 12-20 k_bendri_trend 21-32
k_bendri_ad_prog 33-41 k_bendri_ad_error 42-51
                      k_bendri_mult_prog 52-63 k_bendri_mult_error 64-76;
           label;
           datalines;
2010.01.01 129.7638 106.4254993 152.9200 -23.15622 151.3837606 -21.61993138
.............................................................................
;
run;

ODS GRAPHICS ON; /* Grafiniu priemonių įrankio naudojimo nustatymas*/

/*          Spausdinama regresinių modelių paklaidos*/
proc print data=kaunas_regresiniai;
run;

/*          Tiriamos adityvaus regresinio modelio paklaidos*/
proc autoreg data=kaunas_regresiniai;
           model k_bendri_ad_error= / noint archtest dwprob;
run;

/*          Nustatomas ir sudaromas AR procesas adityvaus
                      regresinio modelio paklaidoms*/
proc arima data=kaunas_regresiniai PLOTS(UNPACK)=Series();
           identify var=k_bendri_ad_error stationarity=(adf) minic q=(0:0)
scan;
           estimate p=2 noint;
           forecast lead=5 interval=month id=date out=Rez1 printall;
run;
```

```
/*          Tiriamos sudaryto modelio paklaidos*/
proc autoreg data=Rez1;
          model RESIDUAL= / noint archtest dwprob;
run;

/*          Tiriamos multiplikatyvaus regresinio modelio paklaidos*/
proc autoreg data=kaunas_regresiniai;
          model k_bendri_mult_error= / noint archtest dwprob;
run;

/*          Nustatomas ir sudaromas AR procesas adityvaus
                     regresinio modelio paklaidoms*/
proc arima data=kaunas_regresiniai PLOTS(UNPACK)=Series();
          identify var=k_bendri_mult_error stationarity=(adf)minic q=(0:0)
scan;
          estimate p=2 noint;
          forecast lead=5 interval=month id=date out=Rez2 printall;
run;

/*          Tiriamos sudaryto modelio paklaidos*/
proc autoreg data=Rez2;
          model RESIDUAL= / noint archtest dwprob;
run;
```

Klaipėdos gyventojų išlaidų, būsto išlaikymui, prognozavimo modelių sudarymas

```
/*            Sudaromas Klaipėdos gyventojų išlaidų, būsto išlaikymui, duomenų
failas*/
data klaipeda;
input date yymmdd10. kl_bendri 12-20 kl_bendri_stdev 21-30 kl_bendri_n 31-35;
          label kl_bendri='Klaipėdos gyventojų išlaidos būsto išlaikymui';

datalines;
2011.01.01 143.3979   56.25437  2475
...................................
;
run;

/*            Grafinės parinktys            */
ODS GRAPHICS ON;
SYMBOL1
          V=DOT C=Blue I=JOIN H=0.6 W=1;
AXIS2 LABEL
=(ANGLE=90 'Bendri mokesčiai (€)');
AXIS1 LABEL=('Data') order=('01jan2011'd to '01feb2015'd by month);

/*            Braižomas Klaipėdiečių išlaidų grafikas*/
proc gplot data = klaipeda;
format date yymmdd10.;
plot kl_bendri*date / frame grid
                                haxis=axis1 vaxis=axis2;
run;
/*            Tiriama laiko eilutė, nustatomas stacionarumas, parenkama
modelio eilė*/
proc arima data=klaipeda        ;
          identify var =kl_bendri  stationarity=(adf);
          identify var =kl_bendri(1) stationarity=(adf) minic perror=( 0:6 )
q=( 0:6 ) scan ;
          _ARIMA_0_0:
          estimate p=0 q=0 noint outstat=a1;
          _ARIMA_0_1:
```

```
            estimate p=0 q=1 noint outstat=a2;
            _ARIMA_0_2:
            estimate p=0 q=2 noint outstat=a3;
            _ARIMA_0_3:
            estimate p=0 q=3 noint outstat=a4;
            _ARIMA_0_4:
            estimate p=0 q=4 noint outstat=a5;
            _ARIMA_0_5:
            estimate p=0 q=5 noint outstat=a6;
            _ARIMA_0_6:
            estimate p=0 q=6 noint outstat=a7;
            _ARIMA_1_0:
            estimate p=1 q=0 noint outstat=a8;
            _ARIMA_1_1:
            estimate p=1 q=1 noint outstat=a9;
            _ARIMA_1_2:
            estimate p=1 q=2 noint outstat=a10;
            _ARIMA_1_3:
            estimate p=1 q=3 noint outstat=a11;
            _ARIMA_1_4:
            estimate p=1 q=4 noint outstat=a12;
            _ARIMA_1_5:
            estimate p=1 q=5 noint outstat=a13;
            _ARIMA_1_6:
            estimate p=1 q=6 noint outstat=a14;
            _ARIMA_2_0:
            estimate p=2 q=0 noint outstat=a15;
            _ARIMA_2_1:
            estimate p=2 q=1 noint outstat=a16;
            _ARIMA_2_2:
            estimate p=2 q=2 noint outstat=a17;
            _ARIMA_2_3:
            estimate p=2 q=3 noint outstat=a18;
            _ARIMA_2_4:
            estimate p=2 q=4 noint outstat=a19;
            _ARIMA_2_5:
            estimate p=2 q=5 noint outstat=a20;
            _ARIMA_2_6:
            estimate p=2 q=6 noint outstat=a21;
            _ARIMA_3_0:
            estimate p=3 q=0 noint outstat=a22;
            _ARIMA_3_1:
            estimate p=3 q=1 noint outstat=a23;
            _ARIMA_3_2:
            estimate p=3 q=2 noint outstat=a24;
            _ARIMA_3_3:
            estimate p=3 q=3 noint outstat=a25;
            _ARIMA_3_4:
            estimate p=3 q=4 noint outstat=a26;
            _ARIMA_3_5:
            estimate p=3 q=5 noint outstat=a27;
            _ARIMA_3_6:
            estimate p=3 q=6 noint outstat=a28;
            _ARIMA_4_0:
            estimate p=4 q=0 noint outstat=a29;
            _ARIMA_4_1:
            estimate p=4 q=1 noint outstat=a30;
            _ARIMA_4_2:
            estimate p=4 q=2 noint outstat=a31;
            _ARIMA_4_3:
            estimate p=4 q=3 noint outstat=a32;
```

```
            _ARIMA_4_4:
            estimate p=4 q=4 noint outstat=a33;
            _ARIMA_4_5:
            estimate p=4 q=5 noint outstat=a34;
            _ARIMA_4_6:
            estimate p=4 q=6 noint outstat=a35;
            _ARIMA_5_0:
            estimate p=5 q=0 noint outstat=a36;
            _ARIMA_5_1:
            estimate p=5 q=1 noint outstat=a37;
            _ARIMA_5_2:
            estimate p=5 q=2 noint outstat=a38;
            _ARIMA_5_3:
            estimate p=5 q=3 noint outstat=a39;
            _ARIMA_5_4:
            estimate p=5 q=4 noint outstat=a40;
            _ARIMA_5_5:
            estimate p=5 q=5 noint outstat=a41;
            _ARIMA_5_6:
            estimate p=5 q=6 noint outstat=a42;
            _ARIMA_6_0:
            estimate p=6 q=0 noint outstat=a43;
            _ARIMA_6_1:
            estimate p=6 q=1 noint outstat=a44;
            _ARIMA_6_2:
            estimate p=6 q=2 noint outstat=a45;
            _ARIMA_6_3:
            estimate p=6 q=3 noint outstat=a46;
            _ARIMA_6_4:
            estimate p=6 q=4 noint outstat=a47;
            _ARIMA_6_5:
            estimate p=6 q=5 noint outstat=a48;
            _ARIMA_6_6:
            estimate p=6 q=6 noint outstat=a49;
run;

/*          ARIMA modelių tinkamumo kriterijų lentelė*/
            data aic(keep=_MODLABEL_ AIC);
            set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15
            a16 a17 a18 a19 a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32
            a33 a34 a35 a36 a37 a38 a39 a40 a41 a42 a43 a44 a45 a46 a47 a48 a49
            ;

            if _STAT_='AIC';
                        AIC=_VALUE_;
            run;

            data bic (keep=_MODLABEL_ BIC);
            set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15
            a16 a17 a18 a19 a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32
            a33 a34 a35 a36 a37 a38 a39 a40 a41 a42 a43 a44 a45 a46 a47 a48 a49
            ;

            if _STAT_='SBC';
                        BIC=_VALUE_;
            run;

            data sse (keep=_MODLABEL_ SSE);
            set a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12 a13 a14 a15
            a16 a17 a18 a19 a20 a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32
            a33 a34 a35 a36 a37 a38 a39 a40 a41 a42 a43 a44 a45 a46 a47 a48 a49
```

```
            ;

            if _STAT_='SSE';
                        SSE=_VALUE_;
            run;

            data aic_bic_sse;
            merge aic bic sse;
            by _MODLABEL_;
            run;

/*           Spausdinama modelių kriterijų lentelė*/
            proc print data=aic_bic_sse;
            run;

/*           Parenkamas modelis ir atliekama prognozė*/
            proc arima data=klaipeda plots=forecast(forecast);
            identify var =kl_bendri(1);
            estimate p=2 q=2 noint;
            forecast lead=5 interval=month id=date out=Rez_kl_bendri1 printall;
run;

/*           Suskaičiuojama MAPE*/
data paklaida1;
            set Rez_kl_bendri1;
            MAPE=abs((kl_bendri-FORECAST)/kl_bendri)*100;
run;

proc summary data=paklaida1;
            var MAPE;
            output out=paklaida1 mean()=;
run;

/*           Spausdinama MAPE*/
proc print data=paklaida1;
            var MAPE;
            title 'ARIMA(2,1,2) modelio paklaida';
run;

/*           Atliekama modelio paklaidų analizė, nustatomas
heteroskedastiškumas*/
proc autoreg data=Rez_kl_bendri1;
model RESIDUAL=/ noint archtest;
run;

/*            Tiriama SARIMA modeliai, nustatomas stacionarumas, parenkama
modelio eilė*/
proc arima data=klaipeda  ;
            identify var =kl_bendri  stationarity=(adf);
                        identify var =kl_bendri(12) stationarity=(adf) minic
scan ;
            _SARIMA_0_0:
            estimate p=0 q=0 noint outstat=b1;
            _SARIMA_0_1:
            estimate p=0 q=1 noint outstat=b2;
            _SARIMA_0_2:
            estimate p=0 q=2 noint outstat=b3;
            _SARIMA_0_3:
            estimate p=0 q=3 noint outstat=b4;
            _SARIMA_0_4:
            estimate p=0 q=4 noint outstat=b5;
```

```
_SARIMA_0_5:
estimate p=0 q=5 noint outstat=b6;
_SARIMA_0_6:
estimate p=0 q=6 noint outstat=b7;
_SARIMA_1_0:
estimate p=1 q=0 noint outstat=b8;
_SARIMA_1_1:
estimate p=1 q=1 noint outstat=b9;
_SARIMA_1_2:
estimate p=1 q=2 noint outstat=b10;
_SARIMA_1_3:
estimate p=1 q=3 noint outstat=b11;
_SARIMA_1_4:
estimate p=1 q=4 noint outstat=b12;
_SARIMA_1_5:
estimate p=1 q=5 noint outstat=b13;
_SARIMA_1_6:
estimate p=1 q=6 noint outstat=b14;
_SARIMA_2_0:
estimate p=2 q=0 noint outstat=b15;
_SARIMA_2_1:
estimate p=2 q=1 noint outstat=b16;
_SARIMA_2_2:
estimate p=2 q=2 noint outstat=b17;
_SARIMA_2_3:
estimate p=2 q=3 noint outstat=b18;
_SARIMA_2_4:
estimate p=2 q=4 noint outstat=b19;
_SARIMA_2_5:
estimate p=2 q=5 noint outstat=b20;
_SARIMA_2_6:
estimate p=2 q=6 noint outstat=b21;
_SARIMA_3_0:
estimate p=3 q=0 noint outstat=b22;
_SARIMA_3_1:
estimate p=3 q=1 noint outstat=b23;
_SARIMA_3_2:
estimate p=3 q=2 noint outstat=b24;
_SARIMA_3_3:
estimate p=3 q=3 noint outstat=b25;
_SARIMA_3_4:
estimate p=3 q=4 noint outstat=b26;
_SARIMA_3_5:
estimate p=3 q=5 noint outstat=b27;
_SARIMA_3_6:
estimate p=3 q=6 noint outstat=b28;
_SARIMA_4_0:
estimate p=4 q=0 noint outstat=b29;
_SARIMA_4_1:
estimate p=4 q=1 noint outstat=b30;
_SARIMA_4_2:
estimate p=4 q=2 noint outstat=b31;
_SARIMA_4_3:
estimate p=4 q=3 noint outstat=b32;
_SARIMA_4_4:
estimate p=4 q=4 noint outstat=b33;
_SARIMA_4_5:
estimate p=4 q=5 noint outstat=b34;
_SARIMA_4_6:
estimate p=4 q=6 noint outstat=b35;
_SARIMA_5_0:
```

```
            estimate p=5 q=0 noint outstat=b36;
            _SARIMA_5_1:
            estimate p=5 q=1 noint outstat=b37;
            _SARIMA_5_2:
            estimate p=5 q=2 noint outstat=b38;
            _SARIMA_5_3:
            estimate p=5 q=3 noint outstat=b39;
            _SARIMA_5_4:
            estimate p=5 q=4 noint outstat=b40;
            _SARIMA_5_5:
            estimate p=5 q=5 noint outstat=b41;
            _SARIMA_5_6:
            estimate p=5 q=6 noint outstat=b42;
            _SARIMA_6_0:
            estimate p=6 q=0 noint outstat=b43;
            _SARIMA_6_1:
            estimate p=6 q=1 noint outstat=b44;
            _SARIMA_6_2:
            estimate p=6 q=2 noint outstat=b45;
            _SARIMA_6_3:
            estimate p=6 q=3 noint outstat=b46;
            _SARIMA_6_4:
            estimate p=6 q=4 noint outstat=b47;
            _SARIMA_6_5:
            estimate p=6 q=5 noint outstat=b48;
            _SARIMA_6_6:
            estimate p=6 q=6 noint outstat=b49;
run;

/*             Sudaroma modelio kriterijų lentelė*/
data aic1(keep=_MODLABEL_ AIC);
            set b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15
            b16 b17 b18 b19 b20 b21 b22 b23 b24 b25 b26 b27 b28 b29 b30 b31 b32
            b33 b34 b35 b36 b37 b38 b39 b40 b41 b42 b43 b44 b45 b46 b47 b48 b49
            ;

            if _STAT_='AIC';
                       AIC=_VALUE_;
            run;

            data bic1 (keep=_MODLABEL_ BIC);
            set b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15
            b16 b17 b18 b19 b20 b21 b22 b23 b24 b25 b26 b27 b28 b29 b30 b31 b32
            b33 b34 b35 b36 b37 b38 b39 b40 b41 b42 b43 b44 b45 b46 b47 b48 b49
            ;

            if _STAT_='SBC';
                       BIC=_VALUE_;
            run;

            data sse1 (keep=_MODLABEL_ SSE);
            set b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15
            b16 b17 b18 b19 b20 b21 b22 b23 b24 b25 b26 b27 b28 b29 b30 b31 b32
            b33 b34 b35 b36 b37 b38 b39 b40 b41 b42 b43 b44 b45 b46 b47 b48 b49
            ;

            if _STAT_='SSE';
                       SSE=_VALUE_;
            run;

            data aic_bic_sse;
```

```
          merge aic1 bic1 sse1;
          by _MODLABEL_;
          run;

/*         Parenkamas modelis ir atliekama prognozė*/
proc arima data=klaipeda        plots=forecast(forecast) ;
                     identify var =kl_bendri(12) stationarity=(adf) minic
scan ;
          estimate p=1 q=0 noint;
          forecast lead=5 interval=month id=date out=Rez_kl_bendri2 printall;
run;

/*         Atliekama modelio paklaidų analizė, nustatomas
heteroskedastiškumas*/
proc autoreg data=Rez_kl_bendri2;
model RESIDUAL=/ noint archtest;
run;

/*           Suskaičiuojama MAPE*/
data paklaida2;
          set Rez_kl_bendri2;
          MAPE=abs((kl_bendri-FORECAST)/kl_bendri)*100;
run;

proc summary data=paklaida2;
          var MAPE;
          output out=paklaida2 mean()=;
run;

/*           Spausdinama MAPE*/
proc print data=paklaida2;
          var MAPE;
run;

/*                     Klaipėdos gyventojų išlaidų, būsto išlaikymui
regresinių modelių paklaidų tyrimas*/

data klaipeda_regresiniai;
          input date yymmdd10. kl_bendri 12-20 kl_bendri_trend 21-32
kl_bendri_ad_prog 33-41 kl_bendri_ad_error 42-51
                     kl_bendri_mult_prog 52-63 kl_bendri_mult_error 64-76;
          datalines;
2011.01.01 143.3979 108.7993366 145.4235  -2.02560 145.7176265 -2.319741528
...........................................................................
;
run;

/*         Spausdinama regresinių modelių paklaidos*/
proc print data=klaipeda_regresiniai;
run;

ODS GRAPHICS ON; /* Grafiniu priemonių įrankio naudojimo nustatymas*/

/*         Tiriamos adityvaus regresinio modelio paklaidos*/
proc autoreg data=klaipeda_regresiniai;
          model kl_bendri_ad_error= / noint archtest dwprob;
run;

/*         Nustatomas ir sudaromas AR procesas adityvaus
                     regresinio modelio paklaidoms*/
proc arima data=klaipeda_regresiniai;
```

```
            identify var=kl_bendri_ad_error stationarity=(adf) minic q=(0:0)
scan;
            estimate q=0 p=1 noint;
            forecast lead=5 interval=month id=date out=Rez1 printall;
run;

/*          Tiriamos sudaryto modelio paklaidos*/
proc autoreg data=Rez1;
            model RESIDUAL= / noint archtest dwprob;
run;

/*          Tiriamos multiplikatyvaus regresinio modelio paklaidos*/
proc autoreg data=klaipeda_regresiniai;
            model kl_bendri_mult_error= / noint archtest dwprob;
run;

/*          Nustatomas ir sudaromas AR procesas adityvaus
            regresinio modelio paklaidoms*/
proc arima data=klaipeda_regresiniai;
            identify var=kl_bendri_mult_error stationarity=(adf) minic q=(0:0)
scan;
run;

estimate q=0 p=1 noint;
;
forecast lead=5 interval=month id=date out=Rez2  printall;
run;

/*          Tiriamos sudaryto modelio paklaidos*/
proc autoreg data=Rez2;
            model RESIDUAL= / noint archtest dwprob;
run;
```