

Vilniaus Universitetas
Matematikos ir informatikos fakultetas

Kursinis darbas

**Lietuvos akcijų rinkos gražos priklausomybė nuo
makroekonominių duomenų**

Kursinio vadovas: lektorius dr. Dmitrij Celov

Darbą atliko: Manvydas Sokolovas ir Paulius Kostickis

VILNIUS 2016

Turinys

Įvadas:	2
Teorija:	2
Rizika ir diversifikavimas:	2
Ilgalaikio turto įkainojimo modelis	3
APT taikymo prielaidos:	5
Duomenų transformacijos	6
Duomenų vizualizacija	7
Modeliavimas	10
Literatūra	12
Priedai:	45
Kintamieji	45

Santrauka:

Makroekonominiai rodikliai gali padėti nuspėti ateities verslo ciklą, kuris turi įtakos akcijų pelningumui. Sudarytas „OMX Vilnius“ indekso grąžos prognozavimo modelis naudojant „Arbitražo įkainojimo teorema“ (angl. APT), siekiant išsiaiškinti, kokie Lietuvos makroekonominiai rodikliai tiesiškai paveikia „OMX Vilnius“ indekso grąžas. Sukurta pelninga strategija remiantis Lietuvos makro rodikliais ir palyginta su atsitiktiniu investavimu. Tirti mėnesiniai duomenys nuo 2002 iki 2016 metų. Dalis makroekonominių rodiklių yra reikšmingi vertinant indekso kainos pokyčius.

Darbe naudojami trumpiniai:

kk – kasyba ir karjerų eksploatacija
mp – mažmeninė prekyba
vp – verslo plėtros aktyvumas per 3 mėnesius
ta – turimos akcijos
ul – užsakymų lūkesčiai
dll – darbo lygio lūkesčiai
mhope – mažmeninės prekybos pasitikėjimas
shope – statybų pasitikėjimas
phope – paslaugų pasitikėjimas
vhope – vartotojų pasitikėjimas
pramhope – pramonės pasitikėjimas
gkl – gamintojų kainų lygis
ip – industrinė produkcija

Įvadas:

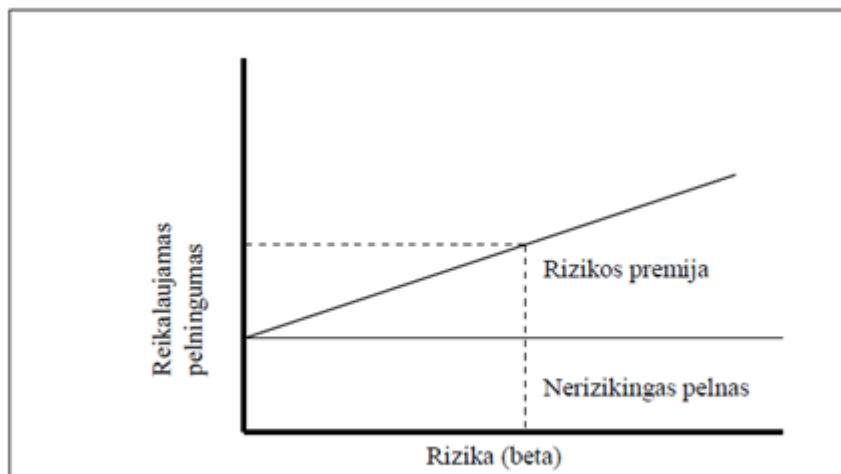
Akcijų pokyčiams yra reikšminga ekonomikos būseną. Jei ekonomika auga, tuomet dauguma įmonių padidina savo pelną ir gamybą. Priešingas procesas, kai ekonomika traukiasi. Egzistuoja makroekonominiai rodikliai tokie kaip infliacija, valiutos kursas, palūkanų dydis, kurie gali paaiškinti akcijų grąžas. Tačiau yra atliktas tyrimas Rusijos rinkoje, remiantis arbitražo įkainojimo teorija, kurio tyrimo autorius nesugebėjo rasti reikšmingos makroekonominių duomenų įtakos akcijų grąžoms. O Šri Lankos ir Indijos rinkoms atlikti empiriniai tyrimai parodė, jog makroekonominiai rodikliai turi įtakos akcijų grąžai. Todėl nutarta ištirti Lietuvos akcijų rinkos grąžos priklausomybę nuo makro rodiklių remiantis Arbitražo įkainojimo teorija (angl. APT) ir iš daugelių rodiklių atrikti svarbiausius. Kylančiose rinkose daug dažniau pasitaiko neteisingai įkainotų finansinių instrumentų nei išvystytose (efektyviose) rinkose. Lietuvos rinką galime priskirti besivystančiosios rinkos kategorijai ir galbūt tyrimo išvadas, naudojant APT, galima bus praktiškai pritaikyti pelningai investuojant.

Teorija:

Rizika ir diversifikavimas:

Investuojant į akcijas, kuo didesnis standartinis nuokrypis, tuo didesnė galimybė prarasti investuotus pinigus. Šiuo atveju rizika bus laikoma standartiniu nuokrypiu. Taigi rizika ir pajamos iš investicijų yra tiesiogiai tarpusavyje susijusios: kuo didesnė rizika, tuo turėtų būti didesnis laukiamas pelnas iš investicijų, kad jos kompensuotų aukštą rizikos lygį. Svarbus tampa reikalaujamas pelningumas – tai mažiausias laukiamas pelnas, kurio yra reikalaujama už investavimą į rizikingą aktyvą. Investuotojas gali rinktis nerizikingą pelną, pirkdamas nerizikingus vertybinius popierius. Ši nerizikinga pelno norma yra minimumas, kurio gali tikėtis investuotojas neprisiimdamas jokios rizikos. Pirkdamas rizikingesnę vertybinę popierių, investuotojas reikalaus rizikos premijos (atpildo už rizikingų vertybinių popierių laikymą. Taigi nerizikinga

pelno norma bei rizikos premija sudaro reikalaujamą pelningumą. Ši priklausomybė pavaizduota paveiksle



Šaltinis: G.Kancerevyčius, 2003, 329p.

apačioje.

H.Markowitz sukurta portfelio teorija leidžia investuotojams įvertinti riziką ir laukiamas pajamas. Galimą riziką sumažinti, o pelną padidinti, jei bus investuojama į skirtingas įmones, kurių akcijų kainos juda skirtingomis kryptimis. Diversifikacija eliminuoja nesisteminę riziką dėl dviejų priežasčių: atskiros įmonės akcijos sudaro nedidelę dalį portfelyje, todėl poveikis (tiek teigiamas, tiek neigiamas) mažai juntamas.

Rizikai ir pelningumui, jų dinamikai prognozuoti kuriamos skirtingos strategijos, taikomos techninės ir fundamentalios analizės žinios, kuriami rizikos ir pelningumo modeliai. Darbe įvertinama rizika, kurią prisiima investuotojai investuodami į Lietuvos akcijų rinką ir taip pat apskaičiuojama kokia premija turėtų būti mokama už šios rizikos prisiėmimą.

Ilgalaikio turto įkainojimo modelis

H.Markowitz sukurta portfelio teorija teigia, kad investuotojui rūpi, kaip gauti kuo daugiau naudos pasirenkant įvairius vertybinius popierius. Toks portfelis turėtų pasižymėti dvejomis savybėmis: 1. mažiau pelningumu priimtina rizika lygiui; 2. mažiausia rizika esant norimam pelningumui. Tai vieno periodo naudos maksimizavimo būdas. (F. K.Reilly, K. C.Brown, 2000) William Sharpe toliau plėtojo šią teoriją. 1964m. sukurta kapitalo (ilgalaikio turto) įkainojimo modelis (toliau CAPM), įvertina pasirinkto vertybinio popieriaus laukiamos grąžos ryšį su rizika. Šis modelis leidžia įvertinti ne tik pačius rizikingiausius, bet ir mažiau rizikingus vertybinius popierius. Taigi CAPM esmė – parodyti, kokia vertybinio popieriaus rizikos dalis gali arba negali būti sumažinta diversifikacijos būdu. (Y.A. Javed, p.2) Autoriai, kurie tiria CAPM (A.R.Strong (2006), L.Gillette (2005) Y.A.Javed bei kiti), teigia, kad CAPM pasižymi tam tikromis prielaidomis: 1. visi investuotojai siekia pelno; 2. investuotojai nėra linkę rizikuoti; 3. grąžos dispersija atitinka rizikos laipsnį; 4. egzistuoja nerizikingas vertybinis popierius, kurį investuotojas gali tiek skolinti, tiek skolintis neribotom sumom su nerizikinga pelno norma; 5. vertybinių popierių kiekiai yra fiksuoti, lengvai platinami rinkoje ir lengvai dalomi; 6. rinkoje informacija visus dalyvius pasiekia vienu metu, tad ji nėra vertinga; 7. laikoma, kad rinkoje nėra mokesčių, sandorio kaštų ar kitų apribojimų; 8. rinkoje veikia tobula konkurencija, t.y. vienas investuotojas negali niekaip paveikti akcijos rinkos kainos; 9. atskirti finansinis ir gamybos sektoriai; 10. nėra infliacijos bei nerizikingos palūkanų normos kitimo. Taikant CAPM praktikoje reikia įvertinti šias prielaidas. Vienos iš jų yra realios (pvz., kai kurie investuotojai gali būti atleisti nuo mokesčių, įmanoma skolinti pinigus už nerizikingą normą), tačiau dauguma iš šiam modeliui taikomų prielaidų laikomos nerealiomis (pvz., infliacijos ir palūkanų normos nėra nekintamos, dažniausiai mokami komisiniai tarpininkas, neįmanoma pasiskolinti už nerizikingas palūkanas ir panašiai). (G.Kancerevyčius, 2003, p. 334) CAPM modelyje rizika skirstoma į sisteminę ir nesisteminę (šios sąvokos pateiktos 1.2.1. skyriuje). Sisteminei rizikai skaičiuoti, kaip jau minėta anksčiau, naudojamas beta koeficientas, kuris parodo vertybinio popieriaus jautrumą su rinka, t.y. lyginamas vertybinio popieriaus

(portfelio) kitimas su rinkos kitimu. CAPM teorijos šalininkai tai pat pritaria portfelio diversifikavimui, nes, pasak jų, tik tokiu būdu galima sumažinti visą nesisteminę riziką. (Y.A. Javed, p.3) Taigi pagal CAPM modelį pageidaujamas pelningumas apskaičiuojamas kaip nerizikingos pelno normos ir rizikos priedo, įvertinančio vertybinio popieriaus sisteminę riziką, suma (R.Norvaišienė, 2004, p.33-34): $RR_i = R_f + \beta_i \cdot (MR - R_f)$ (4) Vertybinio popieriaus ir rizikos premija lygi finansinio instrumento ir beta koeficiento bei rinkos rizikos premijos sandaugai (G.Kancerevyčius, 2003, p.329). Taigi 4 lygybę galima užrašyti taip: $RR_i = R_f + \beta_i \cdot (MR - R_f)$ (5) kur: RR_i – reikalaujamas (planuojamas) pelningumas; R_f – nerizikinga pelno norma; β_i – finansinio instrumento ir beta; $MR - R_f$ – visos rinkos reikalaujamas (planuojamas) pelningumas; $MR - R_f$ yra rinkos rizikos premija, kuri atspindi papildomą pelningumą. Iš šios lygybės matyti, kad išaugus beta koeficientui, padidėja ir laukiamas pelningumas. Taigi 5 lygtis nustato tiesinę betą ir vertybinio popieriaus pelningumo priklausomybę ir grafiškai (4 paveikslas) išreiškia vertybinio popieriaus rinkos liniją (SML – security market line).

Taške R_f rizikos laipsnis lygus 0. Jei vertybinis popierius yra virš SML (taškas A), tai reiškia, kad jis neįvertintas. Ir atvirkščiai, jei vertybinis popierius yra žemiau SML, tai reiškia, kad jis pervertinamas (taškas B). Taigi SML nuolydis atskiram vertybiniam popieriui nesikeičia. SML kreivės nuolydis keičiasi tik tuo atveju, kai keičiasi visos rinkos investuotojų požiūris į rizikos ir pelningumo santykį. (G.Kancerevyčius, 2003, p.329) CAPM modelis patrauklus tuo, jog siūlo puikią galimybę gana patikimai įvertinti riziką bei jos ryšį su laukiamomis pajamomis. Tam naudojamos pagrindinės įgyvendinamos sąlygos: laukiamos pajamos yra tiesiogiai susijusios su vertybinių popierių koeficientais; premija yra teigiama, t.y. laukiamos rinkos portfelio pajamos viršija laukiamas pajamas iš vertybinių popierių, kurių pajamos nekoreliuoja su rinka (E. F.Fama, K. R.French, 2004, p.30) Nors CAPM modelis gana populiarus vertinant vertybinius popierius, tačiau jis turi keletą trūkumų. Visų pirma, šis modelis remiasi keliomis labai griežtomis ir nerealiomis prielaidomis. Antra, CAPM buvo sukurtas kaip nekintantis vertybinių popierių vertinimo modelis. To pasekoje, modelis neatsižvelgia į finansinės rinkos dinamiką. Trečia, CAPM modelis buvo sukurtas stengiantis teoriškai nustatyti ryšį tarp finansinių bei nekilnojamo turto sektorių. (R. J. Torz, 1998)

<http://etalpykla.lituanistikadb.lt/fedora/get/LT-LDB-0001:J.04~2007~1367163822251/DS.002.0>

Pagrindinio kapitalo kainos modelis ir jo prielaidos Pagrindinio kapitalo kainos modelis buvo sukurtas, siekiant pagrįsti skirtingų aktyvų rizikos premijos skirtumus. Pagal CAPM, šiuos skirtumus lemia skirtingas aktyvų generuojamų pajamų rizikingumas. Todėl jama, kad tikslus rizikos matas yra beta, o rizikos premija, tenkanti vienam rizikingumo vienetui, yra 202 tokia pati visiems aktyvų tipams. Jeigu yra žinoma laisva nuo rizikos pelno norma ir tam tikro aktyvo koeficientas beta, tai taikant CAPM gali būti nustatoma laukiamą aktyvo rizikos premija. Pagrindinis CAPM lygtis (Jagannathan, McGrattan, 1995): $ER_i = R_0 + (\beta_i - \beta_m) \cdot (ER_m - R_0)$ (1) 7ia: $i = 1, 2, \dots, n$; ER_i – laukiama i – tojo aktyvo pelno norma; R_0 – laisva nuo rizikos pelno norma (ilgalaikis vyriausybės vertybinis popierio pelno norma); ER_m – laukiama rinkos portfelio pelno norma; β_i – i – tojo aktyvo beta. Pagrindinio kapitalo kainos modelio lygtis rodo, kad visų aktyvų, kurių rizika yra lygi β_i , pageidaujama rizikos premija yra tokia pati – lygi laukiamos pelno normos ir laisvos nuo rizikos pelno normos skirtumui, t.y. $(ER_i - R_0)$. CAPM yra paremtas Markowitz (1952) portfelio teorija. Taria, kad kiekvienas investuotojas diversifikuoja savo aktyvų portfelį, pasirinkdamas jam priimtina rizikos ir pelno normos derinį. Siekiant supaprastinti sudėtingą išorinį aplinką, priimtos papildomos CAPM prielaidos: 1. Investuotojai laukiamą pelno normą ir riziką vertina vienoda tikimybe. Taigi visi investuotojai disponuoja ta pačia informacija apie rinkoje cirkuliuojančius aktyvus; 2. Investavimo perspektyvos – vienas periodas (vieneri metai); 3. Investuotojai vengia rizikos; 4. Visi investuotojai gali skolintis arba skolinti neribotą apimtį piniginių lš nustatytu laisvu nuo rizikos pelno normos tarifu. Šios palikanos nekinta ilgą laikotarpį; 5. Sandorių kaštai yra lygūs nuliui, nėra infliacijos, nėra diferencijuotų mokesčių; 6. Rinkoje veikia daug investuotojų, todėl pavieniai investuotojai, vykdydami pirkimo-pardavimo sandorius, negali turėti tokios aktyvų kainai; 7. Investuotojai siekia maksimaliai padidinti savo laukiamą pelną per vieną investavimo periodą, esant duotam ar žemesniam rizikos lygiui; 8. Kapitalo rinka yra pusiausvyra; 9. Investicijų vertinimui taikomi rodikliai – pelno normos vidurkis ir dispersija. CAPM mokymams padeda suprasti kaip investuotojai vertina potenciali investicini galimybę rizikingumą. Tai didina resursų panaudojimo efektyvumą. Jeigu pagrindinio kapitalo kainos modelis tinkamai vertina investuotojų elgesį, tai analizuojant istorinius duomenis turėtų būti nustatytas teigiamas tiesinis ryšys tarp vidutinės finansinio aktyvo pelno normos ir jos betos. Be to, turėtų neegzistuoti joks kitas matas, kurio pagalba būtų paaiškinti CAPM betos nepagrįsti finansinio aktyvo vidutini pelno normų skirtumai. Aktyvų įkainojimo teorija (APT) CAPM pagrindu buvo kuriami ir tobulinami kiti

aktyvų įkainojimo modeliai, kurie galėjo turėti mažiau apribojimų ir prielaidų, taip pat turėti ir daugiau įtakojančių rodiklių. Vienas iš patobulintų CAPM yra APT. 1966 m. atsirado pirmosios arbitražo įkainojimo teorijos idėjos, kai B. F. King (1966) pradėjo finansinių aktyvų gražos pokyčius aiškinti ekonominiais duomenimis. Tačiau APT teorijos kūrėju yra laikomas S. A. Ross (1976), kuris pateikė teorijai reikiamas prielaidas ir matematiškai pagrindė koncepciją. Šios teorijos pagrindas yra panašus, kaip ir CAPM, t. y. investuotojai reikalauja rizikos premijos už nediversifikuotos (sisteminės) rizikos prisisėmimą. Tačiau arbitražo įkainojimo teorijos naudingumas yra tas, jog galima įtraukti kitus sisteminės rizikos šiame darbe yra tas, jog galime ištirti. Taip pat APT prielaidos yra paprastesnės ir realistiškesnės.

APT taikymo prielaidos:

- 1) finansų rinkos apibūdinamos kaip tobulos ir efektyvios;
- 2) apibrėžtumo atveju investuotojai visada teiks pirmenybę didesnio pelningumo portfeliui;
- 3) egzistuoja tam tikros svarbios sisteminės rizikos, kurios tiesiškai veikia aktyvų pelningumą,
- t. y. aktyvų pelningumus generuojantį stochastinį procesą galima išreikšti kaip n rizikos faktorių ar indeksų tiesinę kombinaciją; be to, investuotojai tas rizikas suvokia ir gali įvertinti aktyvo jautrumą toms rizikoms;
- 4) ekonomikoje yra agresyvių investuotojų, kurie išnaudos aktyvų numatomų pelningumų skirtumus pasinaudodami arbitražu.

Šios prielaidos bus taikomos modelyje.

1 pav. Juoda tiesė B yra S&P 500 indekso vertybinių popierių rinkos tiesė (SML), taškas A yra portfelis, kuris yra aukščiau tiesės. Kai rinka pasiekusi pusiausvyrą, remiantis CAPM tokia situacija neįmanoma, nes galimas tik vienas visiems prieinamas optimalus portfelis. Tačiau pavaizduota situacija yra praktiškai galima.. A portfelio gražos vidurkis gali būti didesnis nei SML gražos vidurkis fiksuotam Beta dydžiui.

Šis aktyvų įkainojimo modelis yra grindžiamas tuo, jog aktyvo pelningumą galima numatyti naudojant analizuojamo aktyvo ir daugelio įprastų rizikos faktorių tarpusavio ryšį. Sukurta S. Ross (1976) arbitražo įkainojimo teorija numato ryšį tarp atskiro aktyvo pelningumo ir portfelio pelningumo pasitelkiant daugelio nepriklausomų kintamųjų (makroekonominių faktorių, tokių kaip infliacija, ekonomikos augimas, tarptautinės gamybos apimtys, palūkanų normos ir t.t.) tiesinę kombinaciją. APT paaiškina aktyvo kainą, kai tikėta, kad aktyvas yra neteisingai įkainotas. Tuo tikslu naudojami rizikingo aktyvo pelningumas ir keleto makroekonominių faktorių rizikos premijos. Taigi investuotojai pasitelkia šią įkainojimo teoriją, siekdami pasipildyti iš neteisingai įvertintų (dažniausiai nepakankamai įvertintų) aktyvų. Tačiau neteisingai įkainoto aktyvo kaina skirsis nuo tos, kuri nustatoma šio modelio dėka. Tokiu būdu investuotojai, norintys pasinaudoti arbitražu ir gauti faktiškai nerizikingą pelną, sieks palaikyti trumpas perversinto aktyvo pozicijas ir kartu laikyti ilgas portfelio (kurio pagrindu yra atliekami APT skaičiavimai) pozicijas.

Jei APT prielaidos patenkinamos, tuomet laukiami pelningumai bus išsidėstę arti SML, o aktyvas turės tiek rizikos charakteristikų, kiek yra faktorių. APT teorijos prielaidos yra artimesnės realybei, nes ne visi investuotojai elgiasi vienodai rinkoje, net jei jų elgsena grindžiama racionalumu, t. y. APT nereikalauja CAPM prielaidų tenkinimo dėl investuotojų naudingumo funkcijų. Be to, ne kiekvienas investuotojas yra linkęs turėti rinkos portfelį kaip vienintelę alternatyvą, t. y. APT nereikalauja CAPM prielaidos tenkinimo dėl rinkos portfelio savybių (kad jis apima visus rizikingus aktyvus ir yra efektyvus vidurkio-dispersijos prasme). Juk rinkoje yra ir kitų rizikingų aktyvų, t. y. kapitalo rinkose daug agresyvių investuotojų. Ši prielaida patvirtinama 1 pav. duomenimis.

Naudodami Lietuvos makro duomenis, matuojame rizikos premijas, kurias gauname už riziką investuojant Lietuvos rinkoje. Makroduomenys iliustruoja Lietuvos ekonominę būklę.

Duomenų transformacijos

Duomenys naudojami nuo 2002 metų iki 2016 metų, ankstesnių duomenų nepavyko išgauti.

OMX Vilnius indeksas transformuojamas į mėnesinius procentinius pokyčius: jei OMX indekso mėnesinė kaina nuo laiko (mėnesio) t žymėsime $X(t)$, tai $r(t) = (\log(X(t)) - \log(X(t-1))) \times 100$ %, čia $r(t)$ yra indekso mėnesinis pokytis procentais. Tokiu pačiu būdu gaunama kitų akcijų indeksų graža logaritmuojant ir diferencijuojant S&P500, S&P350 indeksų kainas. Kasybos ir karjerų eksploatacijos indeksas, mažmeninės prekybos indeksas, industrinės produkcijos indeksas yra taip pat logaritmuojami ir diferencijuojami. Visi logaritmuoti duomenys padauginami iš šimto, kad pokytis būtų interpretuojamas procentais. Euribor yra dalinamas iš 12 ir diferencijuojamas, kad būtų gautas mėnesinis pokytis. Nedarbas dalinamas iš 10 (bedarbių skaičius tenkantis 1000 gyventojų, padalinę iš 10 gauname procentais) ir taip pat yra diferencijuojamas. Dolerio/euro valiutų kursas, pasitikejimo ir lūkesčių rodikliai yra diferencijuojami. Diferencijavimu šiuo atveju vadiname šio mėnesio duomenų atėmimą iš praeito periodo(mėnesio) duomenų.

Tikrinami duomenys po transformacijų ar turi vienetinę šaknį (stacionarumo tikrinimas), atliekamas Dickey-Fuller testas:

	p
OMX	0.01
SP350	0.01
SP500	0.01
kk	0.01
dll	0.01
nedarbas	0.01
infliacija	0.52
mhope	0.01
phope	0.01
pramhope	0.01
shope	0.01
ta	0.01
ul	0.01
vhope	0.01
vp	0.01
mp	0.01
palukanos	0.01
gkl	0.01
ip	0.01
kursas	0.01

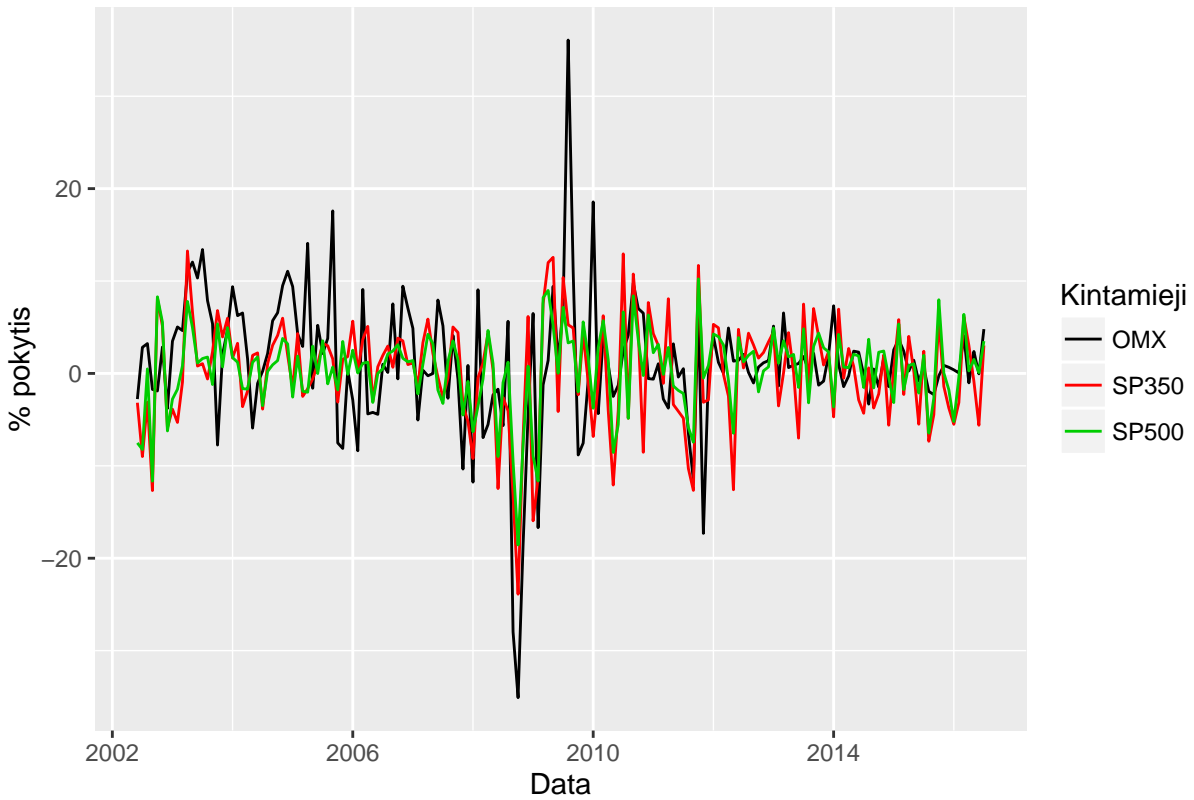
- Visų kintamųjų, išskyrus infliaciją, p - value mažiau už 0.05, galime atmesti H_0 , kad turi vienetinę šaknį, visi kintamieji, išskyrus infliaciją, yra stacionarūs.

Duomenų vizualizacija

Akcijų indeksų mėnesiniai procentiniai pokyčiai:

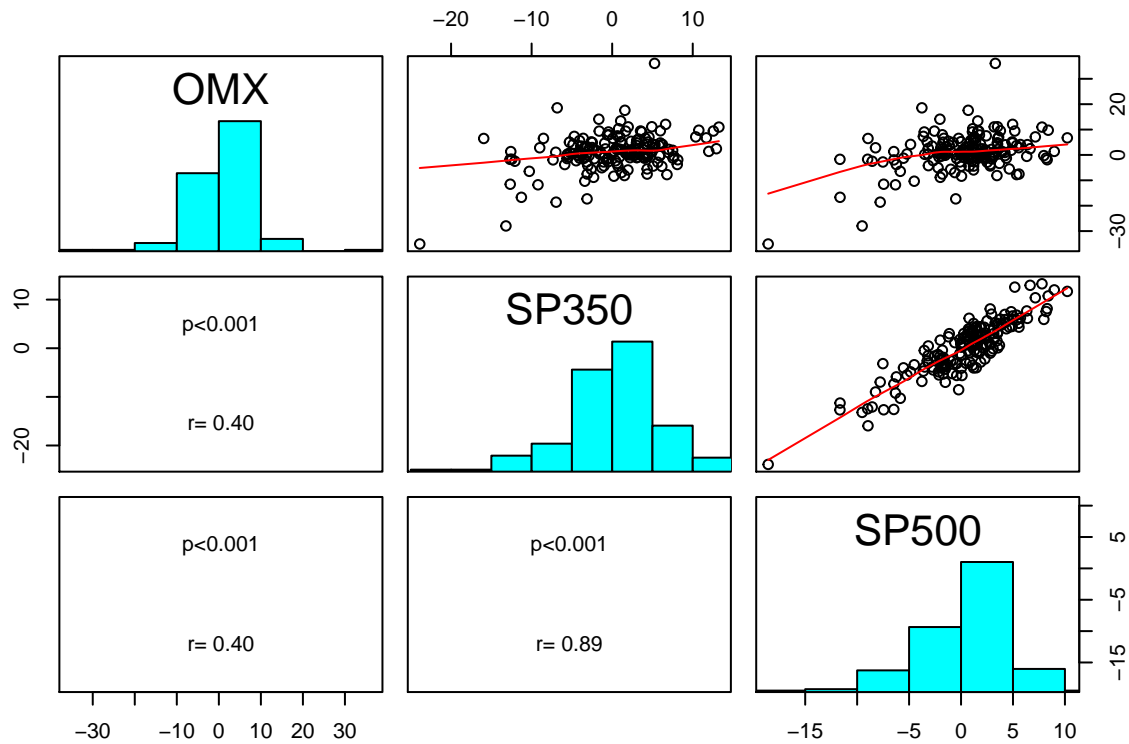
```
## [1] ""
```

```
index<U+0173> m<U+0117>nesiniai gr<U+0105><U+017E><U+0173> poky<U+
```



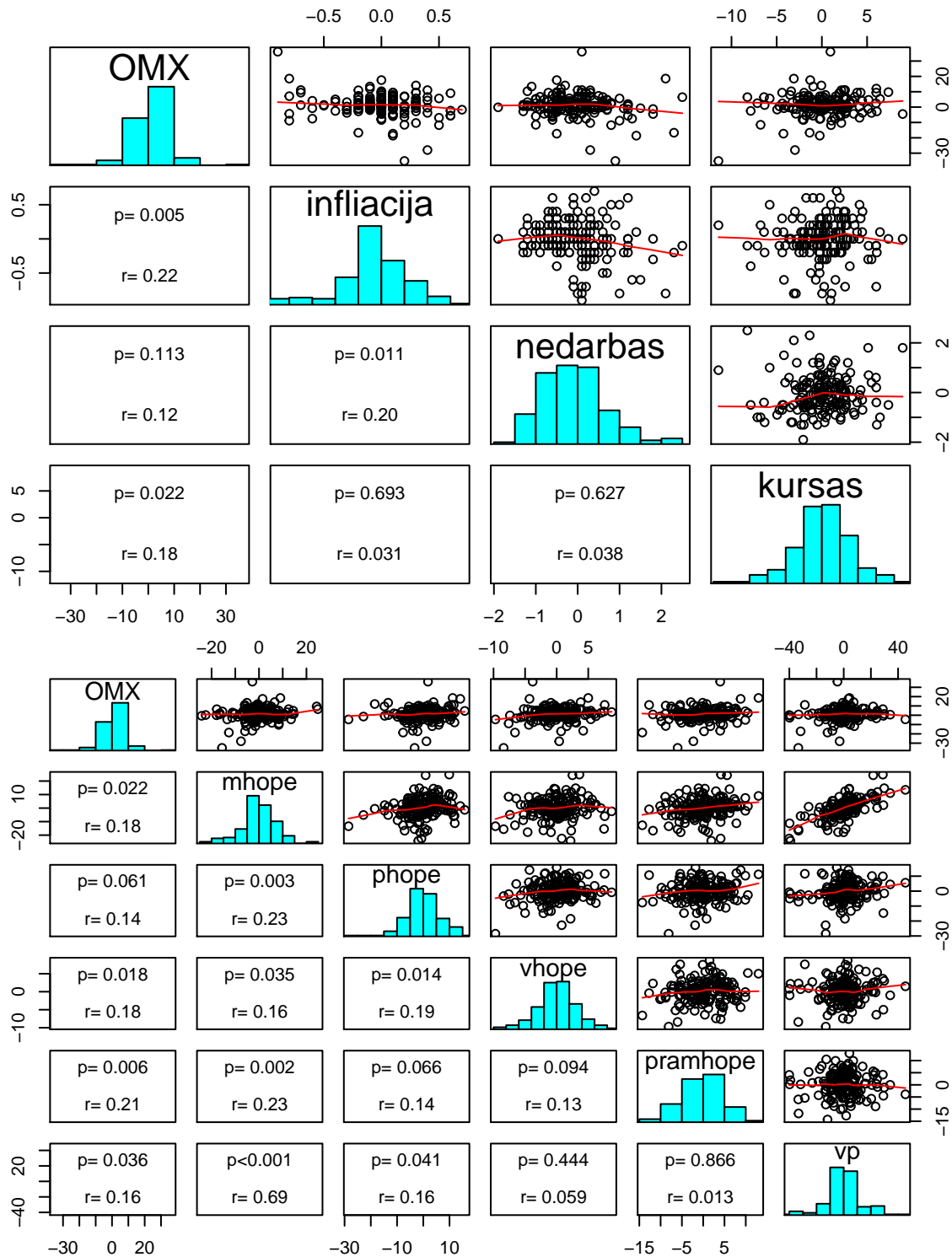
- Tikėtina, jog S&P350 ir S&P500 indeksų svyravimai turėtų būti labai panašūs, taigi pažiūrėjus į grafiką galime matyti, jog jų kreivės juda panašiai, tai yra šie dydžiai stipriai koreliuoti.
- OMX Vilnius kreivės svyravimai panašūs, tačiau nevisai sutampa su S&P indeksų. Todėl galima manyti, jog šie dydžiai yra gana silpnai koreliuoti.

Akcijų rinkų koreliacija:



- S&P350 ir S&P500 indeksų koreliacija reikšminga ir jie stipriai koreliuoja. Europos ir JAV akcijų rinkų svyravimai yra glaudžiai susiję.
- Kaip ir teigta anksčiau OMX Vilnius koreliacija su S&P indeksais yra reikšminga, tačiau nėra didelė. Nesimato stiprios įtakos iš lyderiaujančių pasaulio rinkų.

Kintamųjų koreliacija su OMX Vilnius. r - koreliacijos koeficientas, p - koreliacijos reikšmingumas:



- Kaip ir buvo tikėtasi tarp kintamųjų ir OMX Vilnius rasta nestipri koreliacija, tačiau reikšminga.

Modeliavimas

```
##
## t test of coefficients:
##
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.087098   0.531421   2.0456 0.042550 *
## lag(SP350, 1)      0.360918   0.124273   2.9042 0.004242 **
## lag(infliacija, 2) -3.898667   2.693478  -1.4474 0.149872
## lag(mhope, 3)      0.146672   0.090940   1.6129 0.108893
## lag(phope, 12)     0.169685   0.094204   1.8012 0.073687 .
## lag(ul, 10)        0.061529   0.029758   2.0677 0.040399 *
## lag(gkl, 9)       -1.276498   0.557982  -2.2877 0.023562 *
## lag(ip, 4)         0.128462   0.086807   1.4799 0.141022
## lag(kursas, 2)     0.380851   0.199093   1.9129 0.057675 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Call:
## lm(formula = OMX ~ lag(SP350, 1) + lag(infliacija, 2) + lag(mhope,
##      3) + lag(phope, 12) + lag(ul, 10) + lag(gkl, 9) + lag(ip,
##      4) + lag(kursas, 2), data = data3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.582  -3.158   0.031   2.654  21.474
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.08710   0.50284   2.162 0.032222 *
## lag(SP350, 1)      0.36092   0.09030   3.997 0.000101 ***
## lag(infliacija, 2) -3.89867   1.95114  -1.998 0.047520 *
## lag(mhope, 3)      0.14667   0.06720   2.183 0.030622 *
## lag(phope, 12)     0.16968   0.07918   2.143 0.033736 *
## lag(ul, 10)        0.06153   0.02444   2.518 0.012867 *
## lag(gkl, 9)       -1.27650   0.48898  -2.611 0.009963 **
## lag(ip, 4)         0.12846   0.08391   1.531 0.127914
## lag(kursas, 2)     0.38085   0.16341   2.331 0.021113 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.187 on 149 degrees of freedom
## (12 observations deleted due to missingness)
## Multiple R-squared:  0.3329, Adjusted R-squared:  0.2971
## F-statistic: 9.293 on 8 and 149 DF,  p-value: 2.389e-10
```

	koeficientai	standartinis nuokrypis	koeficientu t reikmes	p reiksme	R kvadratas
(Intercept)	1.09	0.50	2.16	0.03	0.33
lag(SP350, 1)	0.36	0.09	4.00	0.00	-
lag(infliacija, 2)	-3.90	1.95	-2.00	0.05	-
lag(mhope, 3)	0.15	0.07	2.18	0.03	-

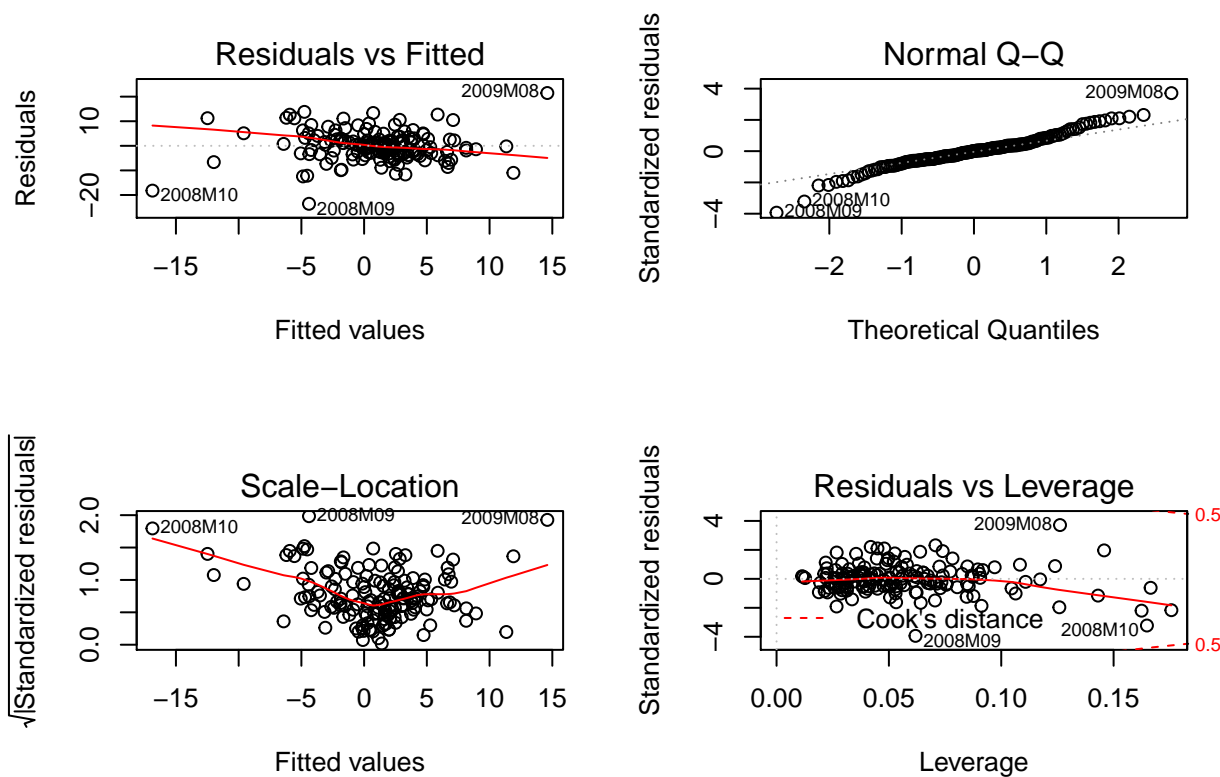
	koeficientai	standartinis nuokrypis	koeficientu t reikmes	p reiksme	R kvadratas
lag(phope, 12)	0.17	0.08	2.14	0.03	-
lag(ul, 10)	0.06	0.02	2.52	0.01	-
lag(gkl, 9)	-1.28	0.49	-2.61	0.01	-
lag(ip, 4)	0.13	0.08	1.53	0.13	-
lag(kursas, 2)	0.38	0.16	2.33	0.02	-

- Sudarytas modelis iš visų kintamųjų, atliktas stepAIC. Didelė dalis makroekonominių duomenų buvo nereikšmingi, tačiau aptikome ir keletą reikšmingų kintamųjų. Modelis paaiškina apie 25% procentus OMX Vilnius indekso grąžos pokyčių.

Literatūra

- 1) „TESTS OF THE ARBITRAGE PRICING THEORY USING MACROECONOMIC VARIABLES IN THE RUSSIAN EQUITY MARKET“ - Mauri Paavola: [link](#)
- 2) „Introductory Econometrics for Finance“ - Chris Brooks
- 3) „Relationship between Macro – Economic Variables and Stock Market Performance of Colombo Stock Exchange“ - Prabath Suranga Morawakage: [link](#)
- 4) „Do macro-economic variables explain stock-market returns? Evidence using a semi-parametric approach“ - Sagarika Mishra: [link](#)

```
par(mfrow = c(2, 2))
plot(modelis_po_aic)
```

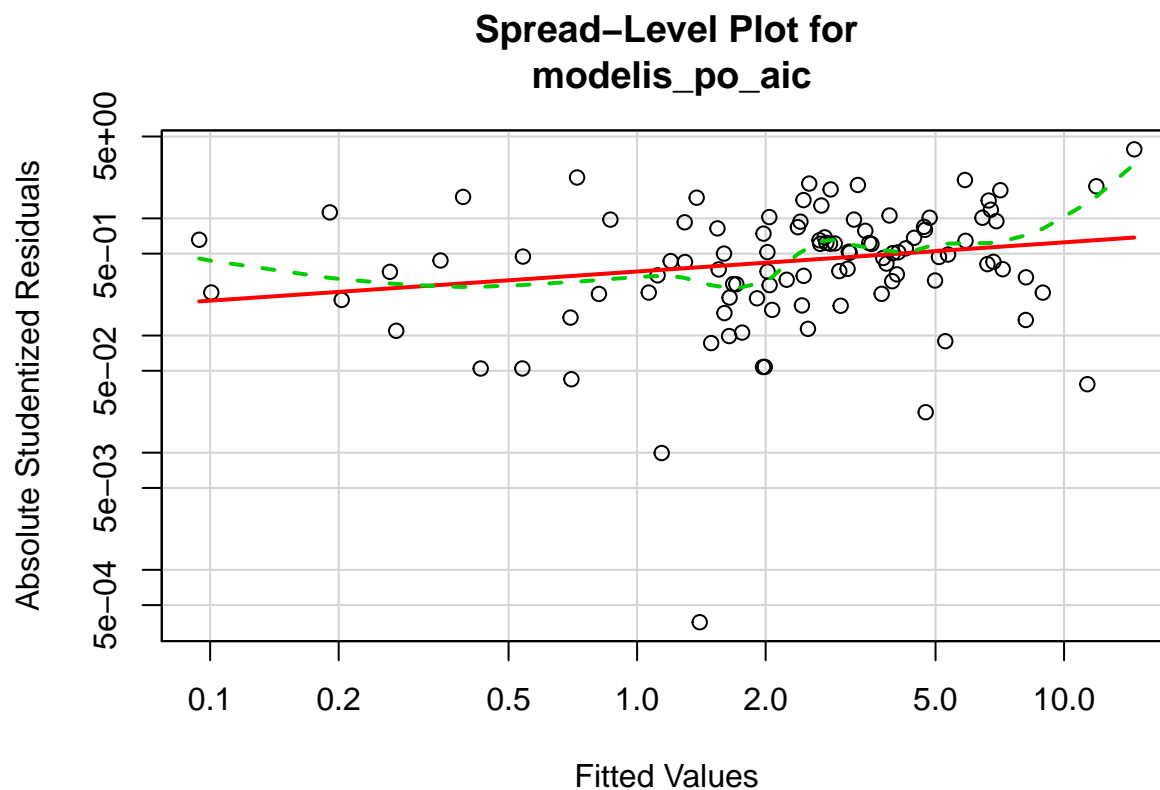


```
res<-bptest(modelis_po_aic)
# p-value < 0.05, tai H0 negalime priimti - duomenys homoskedastiški.
ncvTest(modelis_po_aic)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 7.735989    Df = 1    p = 0.005413089
```

```
# Kadangi p-value > 0.05, tai H0 hipotezė negalime atmesti - duomenys homoskedastiški.
spreadLevelPlot(modelis_po_aic)
```

```
## Warning in spreadLevelPlot.lm(modelis_po_aic): 55 negative fitted values
## removed
```



```
##
## Suggested power transformation: 0.7510512
```

```
vif(modelis)
```

```
##      lag(SP350, 1)      lag(kk, 1)      lag(d11, 1)
##      1.184279      3.354087      1.309610
##      nedarbas lag(infliacija, 2) lag(infliacija, 3)
##      1.472440      12.018637      17.308846
## lag(infliacija, 5) lag(infliacija, 7) lag(infliacija, 8)
##      13.405257      17.392653      12.237699
##      lag(mhope, 3)      lag(phope, 12)      lag(pramhope, 3)
##      1.387610      1.374023      1.512505
##      lag(shope, 4)      lag(ul, 10)      lag(vhope, 4)
##      1.440426      1.207818      1.362711
##      lag(vp, 0)      lag(mp, 1)      lag(gkl, 9)
##      1.452679      1.526044      1.394262
##      lag(ip, 0)      lag(ip, 1)      lag(ip, 4)
##      1.310174      3.182546      1.275043
##      lag(kursas, 2)
##      1.234044
```

Tikrinama autokoreliacija:

```
durbinWatsonTest(modelis_po_aic)
```

```
## lag Autocorrelation D-W Statistic p-value
```

```
##      1      0.07201045      1.844705      0.3
## Alternative hypothesis: rho != 0
```

```
# p-value > 0.05 - liekanos nėra autokoreliuotos.
```

```
Box.test(modelis_po_aic$res, fitdf=0, type="Lj")
```

```
##
## Box-Ljung test
##
## data: modelis_po_aic$res
## X-squared = 0.83497, df = 1, p-value = 0.3608
```

```
#p-value > 0.05, vadinasi H0 neatmetama. Galima teigti, kad liekanos yra baltasis triukšmas.
```

```
shapiro.test(modelis_po_aic$residuals)
```

```
##
## Shapiro-Wilk normality test
##
## data: modelis_po_aic$residuals
## W = 0.96881, p-value = 0.001218
```

```
# p-value > 0.05 - liekanos yra normalios
```

```
pirkti=numeric()

gen=function(pradzia,pirkti){
  b=sample(pradzia:169,length(pirkti),replace=F)
  return(sum(data3[b,"OMX"]))
}

validation=function(pradzia){
  pirkti=numeric()
  for(i in pradzia:168){
    modelis_po_aic <- lm(formula = OMX ~ lag(SP350, 1) + lag(infliacija, 2) + lag(mhope, 3) + lag(phope,1)
    a=predict(modelis_po_aic, data3[1:i+1,])
    a=a[length(a)]
    if(matrix(a)[1]>5){pirkti=append(pirkti,i)}
  }

  graza=sum(data3[pirkti,"OMX"])
  random=replicate(100000,gen(pradzia,pirkti))
  p=mean(random>graza)
  vidurkis_random = mean(random)
  if(p==0){p="nebuvo pirkimu"}
  else{
    p=append(p,graza)
    p=append(p,vidurkis_random)
  }
}
```

```

return(p)
}
validation(90) #kai pradine imtis 70, atsakymo prasme- tikimybe kad geriau uz random, graza strategijos

```

```
## [1] 0.369860 12.426901 8.159706
```

```

modelis = lm(OMX ~ lag(SP350,1) + lag(kk,1) + lag(dll,1) + nedarbas + lag(infliacija,2) + lag(infliacij
modelis_po_aic <- lm(formula = OMX ~ lag(SP350, 1) + lag(infliacija, 2) + lag(mhope, 3) + lag(phope,12)
, data = data3)

coeftest(modelis_po_aic, vcov=vcovHC)

```

```

##
## t test of coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.087098   0.531421   2.0456 0.042550 *
## lag(SP350, 1)      0.360918   0.124273   2.9042 0.004242 **
## lag(infliacija, 2) -3.898667   2.693478  -1.4474 0.149872
## lag(mhope, 3)       0.146672   0.090940   1.6129 0.108893
## lag(phope, 12)      0.169685   0.094204   1.8012 0.073687 .
## lag(ul, 10)         0.061529   0.029758   2.0677 0.040399 *
## lag(gkl, 9)        -1.276498   0.557982  -2.2877 0.023562 *
## lag(ip, 4)          0.128462   0.086807   1.4799 0.141022
## lag(kursas, 2)      0.380851   0.199093   1.9129 0.057675 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
summary(modelis_po_aic)
```

```

##
## Call:
## lm(formula = OMX ~ lag(SP350, 1) + lag(infliacija, 2) + lag(mhope,
##      3) + lag(phope, 12) + lag(ul, 10) + lag(gkl, 9) + lag(ip,
##      4) + lag(kursas, 2), data = data3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.582  -3.158   0.031   2.654  21.474
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.08710   0.50284   2.162 0.032222 *
## lag(SP350, 1)      0.36092   0.09030   3.997 0.000101 ***
## lag(infliacija, 2) -3.89867   1.95114  -1.998 0.047520 *
## lag(mhope, 3)       0.14667   0.06720   2.183 0.030622 *
## lag(phope, 12)      0.16968   0.07918   2.143 0.033736 *
## lag(ul, 10)         0.06153   0.02444   2.518 0.012867 *
## lag(gkl, 9)        -1.27650   0.48898  -2.611 0.009963 **
## lag(ip, 4)          0.12846   0.08391   1.531 0.127914

```



```
## lag(kursas, 2)      0.38085    0.16341    2.331 0.021113 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.187 on 149 degrees of freedom
## (12 observations deleted due to missingness)
## Multiple R-squared:  0.3329, Adjusted R-squared:  0.2971
## F-statistic: 9.293 on 8 and 149 DF,  p-value: 2.389e-10
```

```
ncvTest(modelis_po_aic)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 7.735989    Df = 1    p = 0.005413089
```

```
# Kadangi p-value > 0.05, tai H0 hipotezė priimama - duomenys homoskedastiški.
```

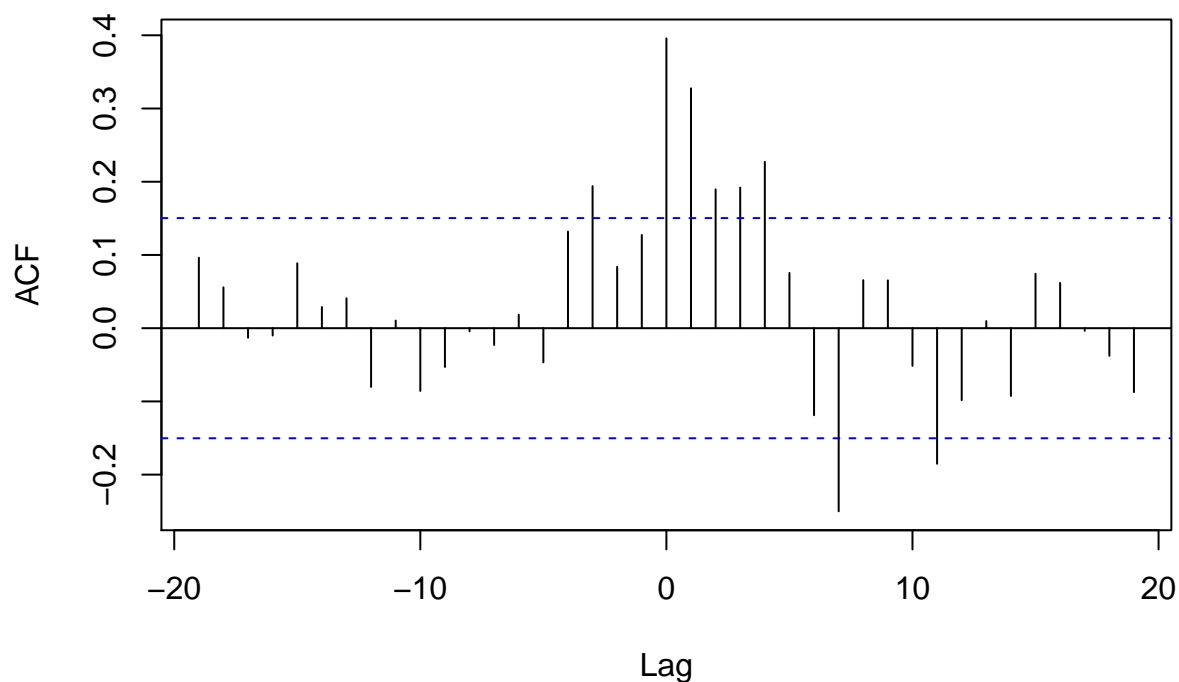
```
names2=c("SP350","SP500","kk","dll","nedarbas","infliacija","mhope","phope","pramhope","shope","ta","ul")

for (i in 3:length(data3[1,])){

  ccfvalues <- ccf(data3$OMX,data3[,i]) #aiskinames laga
  print(names2[i-2])
  print(ccfvalues)

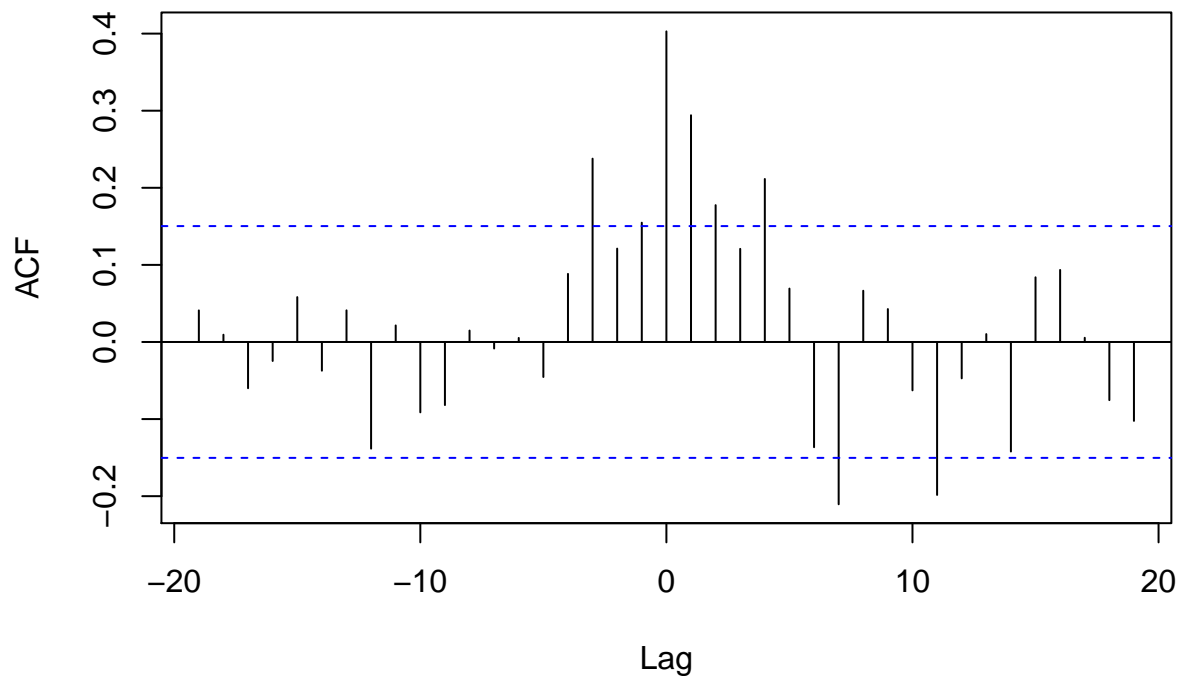
}
```

data3\$OMX & data3[, i]



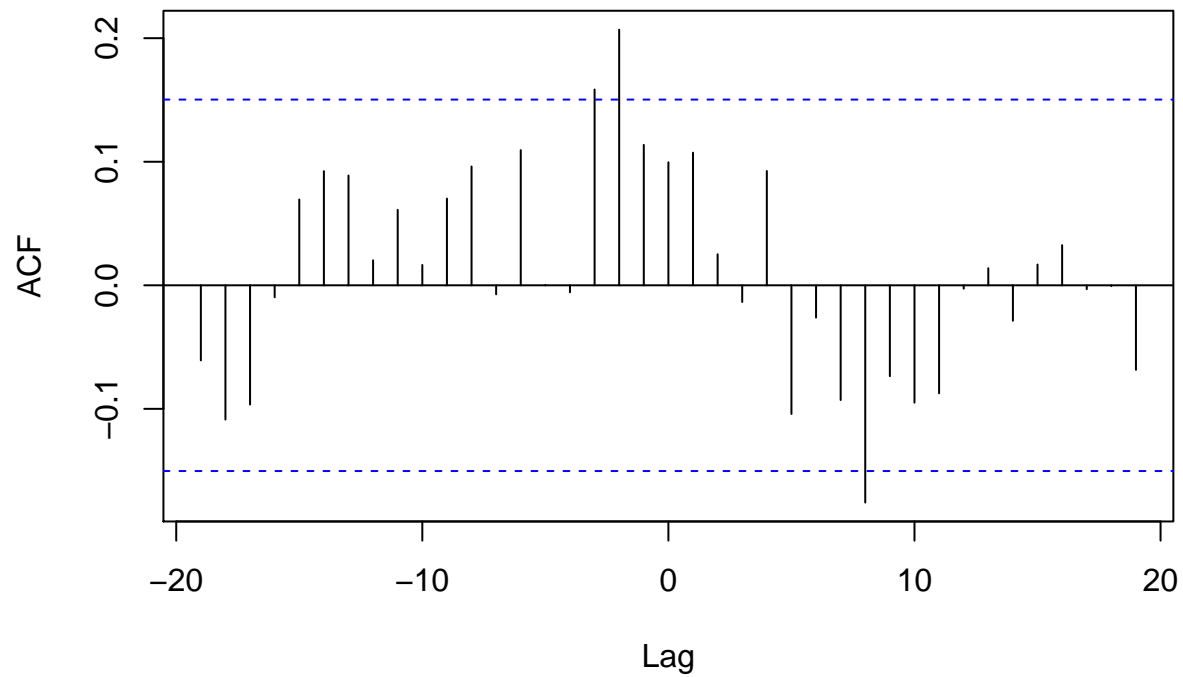
```
## [1] "SP350"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
##  0.096  0.056 -0.013 -0.010  0.089  0.029  0.041 -0.080  0.010 -0.086
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
## -0.053 -0.004 -0.023  0.018 -0.047  0.132  0.194  0.084  0.127  0.396
##     1     2     3     4     5     6     7     8     9    10
##  0.328  0.190  0.192  0.227  0.075 -0.119 -0.250  0.066  0.065 -0.051
##    11    12    13    14    15    16    17    18    19
## -0.185 -0.098  0.010 -0.093  0.074  0.062 -0.004 -0.038 -0.087
```

data3\$OMX & data3[, i]



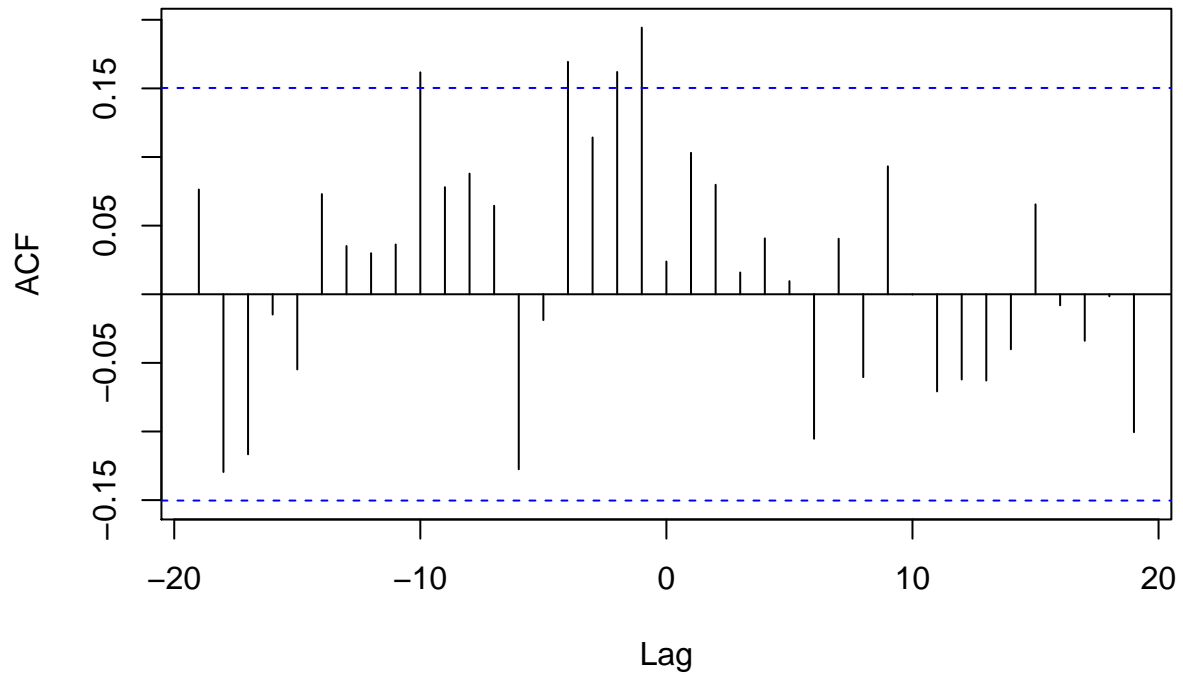
```
## [1] "SP500"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
##  0.041  0.009 -0.060 -0.025  0.058 -0.037  0.041 -0.138  0.021 -0.091
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
## -0.082  0.015 -0.008  0.005 -0.045  0.088  0.238  0.121  0.155  0.403
##     1     2     3     4     5     6     7     8     9    10
##  0.294  0.178  0.121  0.211  0.069 -0.136 -0.211  0.067  0.043 -0.063
##    11    12    13    14    15    16    17    18    19
## -0.198 -0.047  0.010 -0.142  0.084  0.093  0.005 -0.075 -0.102
```

data3\$OMX & data3[, i]



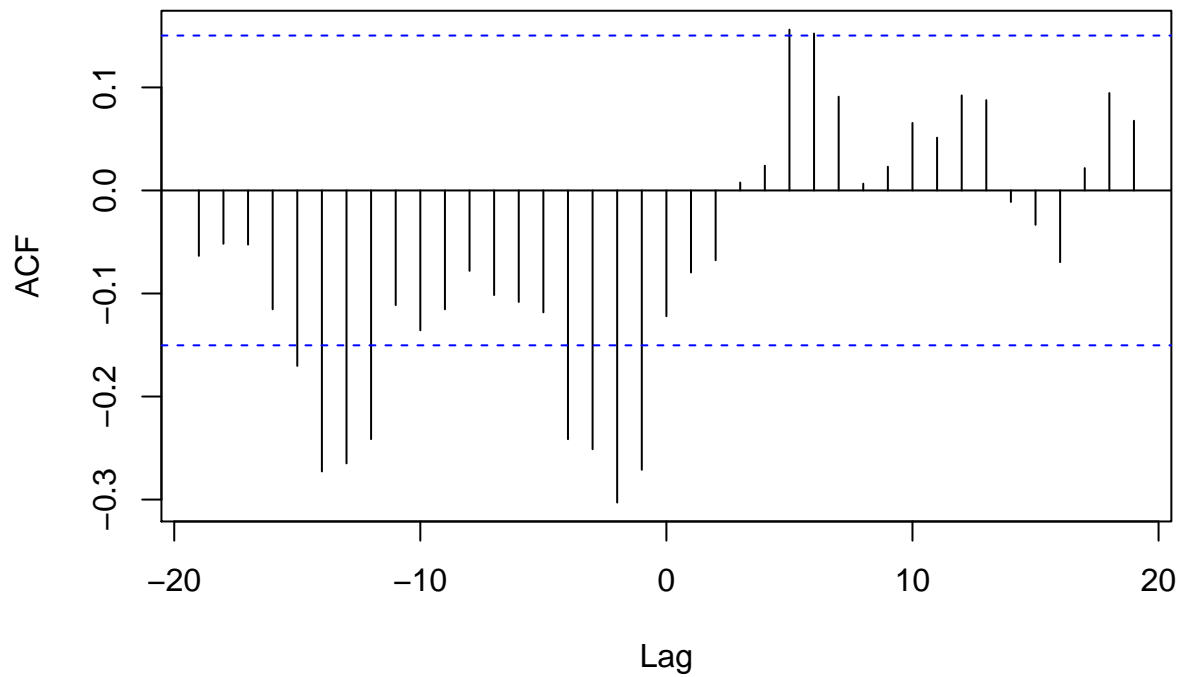
```
## [1] "kk"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
## -0.061 -0.109 -0.097 -0.010  0.069  0.092  0.089  0.020  0.061  0.016
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
##  0.070  0.096 -0.007  0.109  0.000 -0.006  0.158  0.207  0.114  0.100
##     1     2     3     4     5     6     7     8     9    10
##  0.107  0.025 -0.014  0.093 -0.104 -0.026 -0.093 -0.176 -0.074 -0.095
##    11    12    13    14    15    16    17    18    19
## -0.087 -0.003  0.014 -0.029  0.017  0.033 -0.003 -0.001 -0.068
```

data3\$OMX & data3[, i]



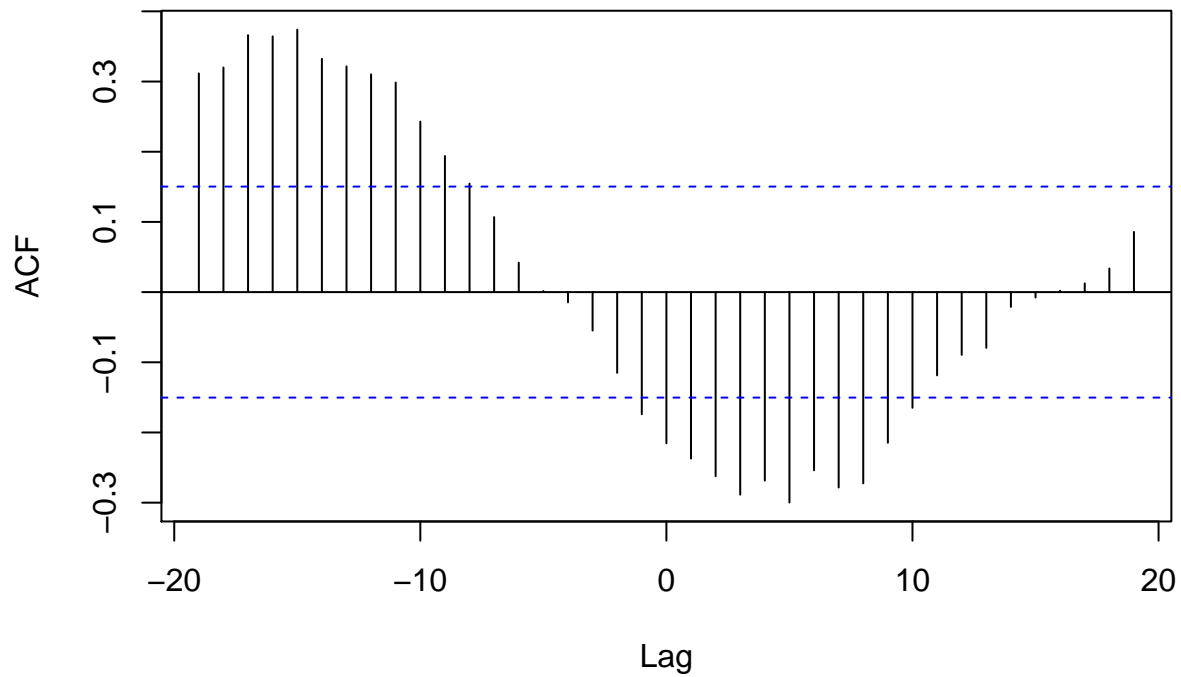
```
## [1] "dl1"
##
## Autocorrelations of series 'X', by lag
##
##      -19      -18      -17      -16      -15      -14      -13      -12      -11      -10
## 0.076 -0.130 -0.117 -0.015 -0.055 0.073 0.035 0.030 0.036 0.162
##      -9       -8       -7       -6       -5       -4       -3       -2       -1        0
## 0.078 0.088 0.064 -0.128 -0.019 0.169 0.114 0.162 0.194 0.024
##      1        2        3        4        5        6        7        8        9       10
## 0.103 0.080 0.016 0.041 0.010 -0.105 0.040 -0.060 0.093 0.000
##     11      12      13      14      15      16      17      18      19
## -0.071 -0.062 -0.063 -0.040 0.065 -0.008 -0.034 -0.001 -0.101
```

data3\$OMX & data3[, i]



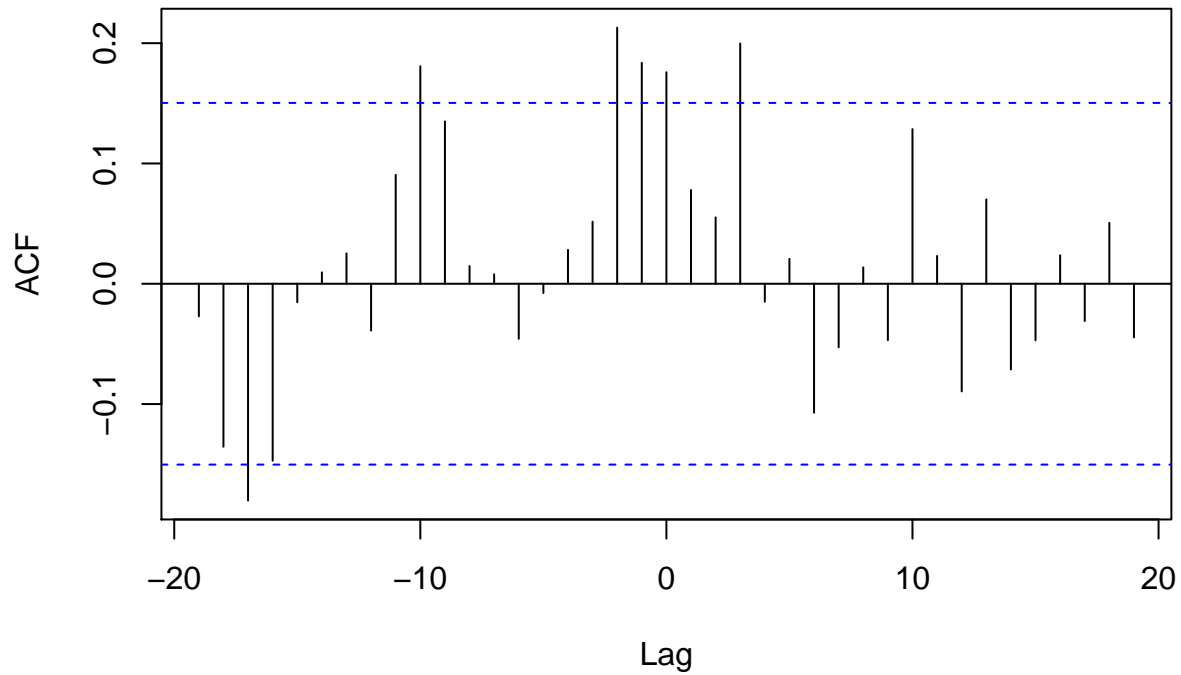
```
## [1] "nedarbas"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
## -0.063 -0.052 -0.053 -0.115 -0.170 -0.273 -0.265 -0.241 -0.111 -0.136
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
## -0.115 -0.078 -0.101 -0.108 -0.118 -0.241 -0.251 -0.303 -0.271 -0.122
##     1     2     3     4     5     6     7     8     9    10
## -0.080 -0.068  0.008  0.024  0.156  0.152  0.091  0.007  0.023  0.065
##    11    12    13    14    15    16    17    18    19
##  0.051  0.092  0.088 -0.011 -0.033 -0.070  0.022  0.095  0.068
```

data3\$OMX & data3[, i]



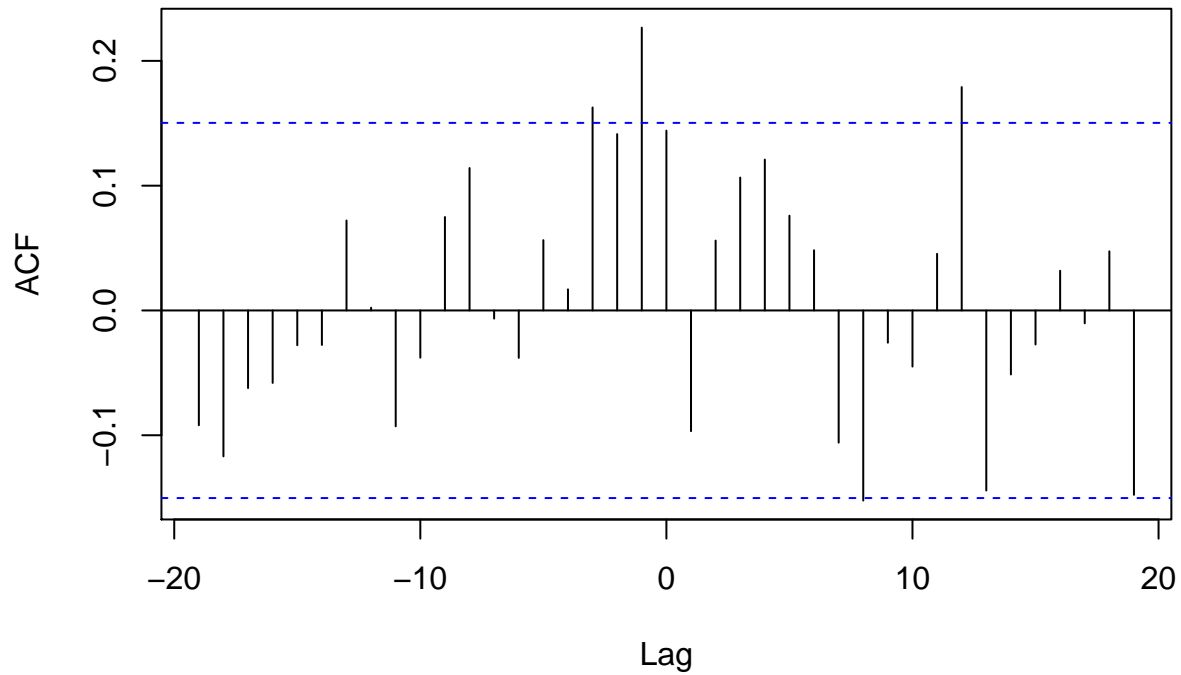
```
## [1] "inflaciija"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
## 0.312 0.320 0.366 0.364 0.374 0.332 0.322 0.310 0.299 0.243
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
## 0.194 0.155 0.107 0.042 0.002 -0.015 -0.055 -0.115 -0.174 -0.215
##     1     2     3     4     5     6     7     8     9    10
## -0.237 -0.262 -0.289 -0.269 -0.300 -0.254 -0.278 -0.272 -0.215 -0.165
##    11    12    13    14    15    16    17    18    19
## -0.119 -0.089 -0.079 -0.021 -0.008 0.002 0.012 0.034 0.086
```

data3\$OMX & data3[, i]



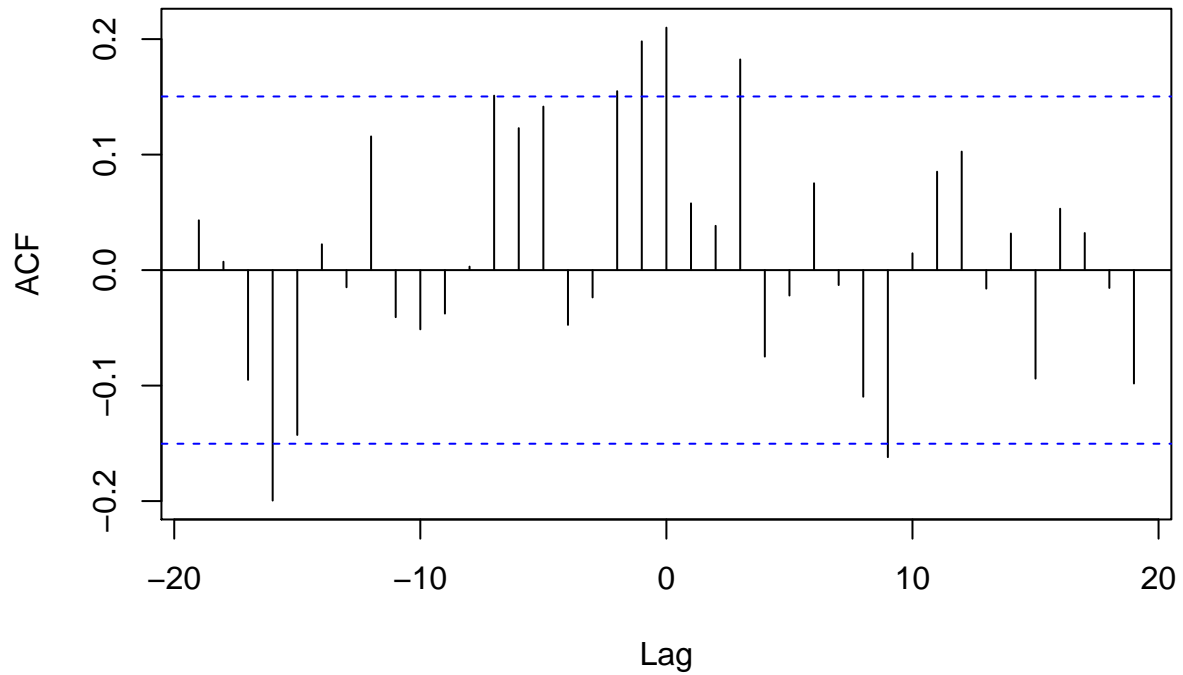
```
## [1] "mhope"
##
## Autocorrelations of series 'X', by lag
##
##      -19      -18      -17      -16      -15      -14      -13      -12      -11      -10
## -0.027 -0.135 -0.180 -0.147 -0.015  0.009  0.025 -0.039  0.091  0.181
##      -9       -8       -7       -6       -5       -4       -3       -2       -1        0
##  0.135  0.015  0.008 -0.046 -0.008  0.028  0.052  0.213  0.184  0.176
##       1        2        3        4        5        6        7        8        9       10
##  0.078  0.055  0.200 -0.015  0.021 -0.107 -0.053  0.014 -0.047  0.129
##      11      12      13      14      15      16      17      18      19
##  0.023 -0.090  0.070 -0.071 -0.047  0.024 -0.031  0.051 -0.045
```

data3\$OMX & data3[, i]



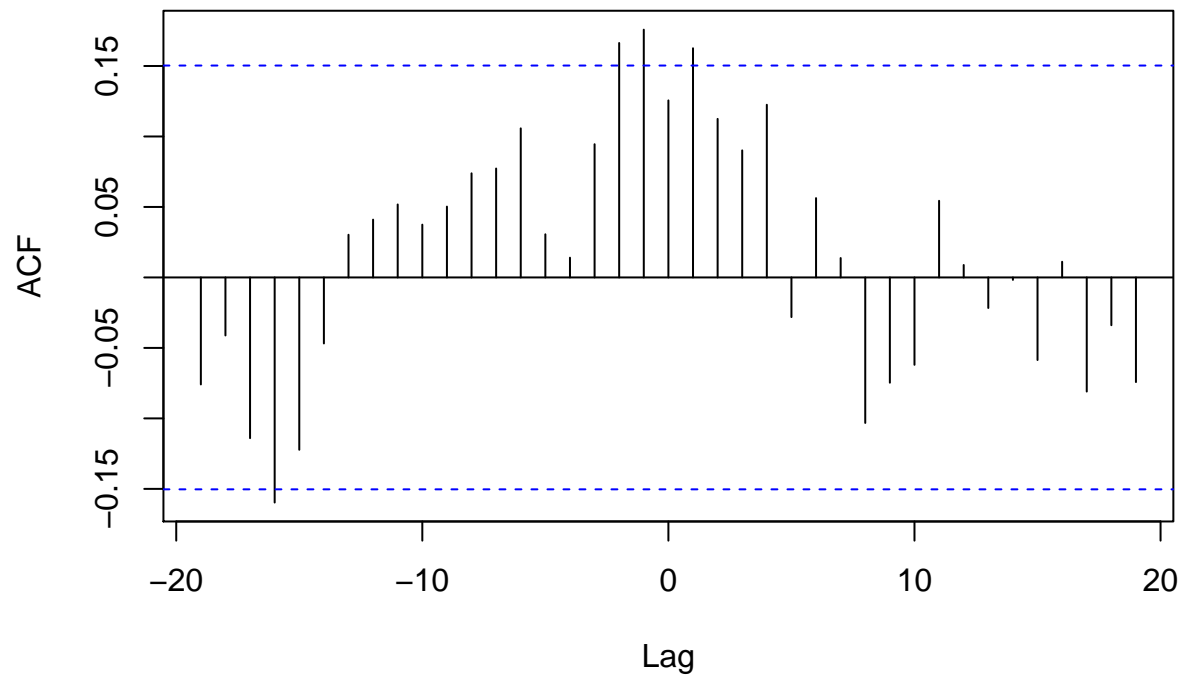
```
## [1] "phope"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
## -0.092 -0.117 -0.062 -0.058 -0.028 -0.028  0.072  0.002 -0.093 -0.038
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
##  0.075  0.114 -0.007 -0.038  0.056  0.017  0.163  0.141  0.227  0.144
##    1     2     3     4     5     6     7     8     9    10
## -0.097  0.056  0.107  0.121  0.076  0.048 -0.106 -0.152 -0.026 -0.045
##   11    12    13    14    15    16    17    18    19
##  0.045  0.179 -0.144 -0.051 -0.027  0.032 -0.010  0.047 -0.148
```


data3\$OMX & data3[, i]



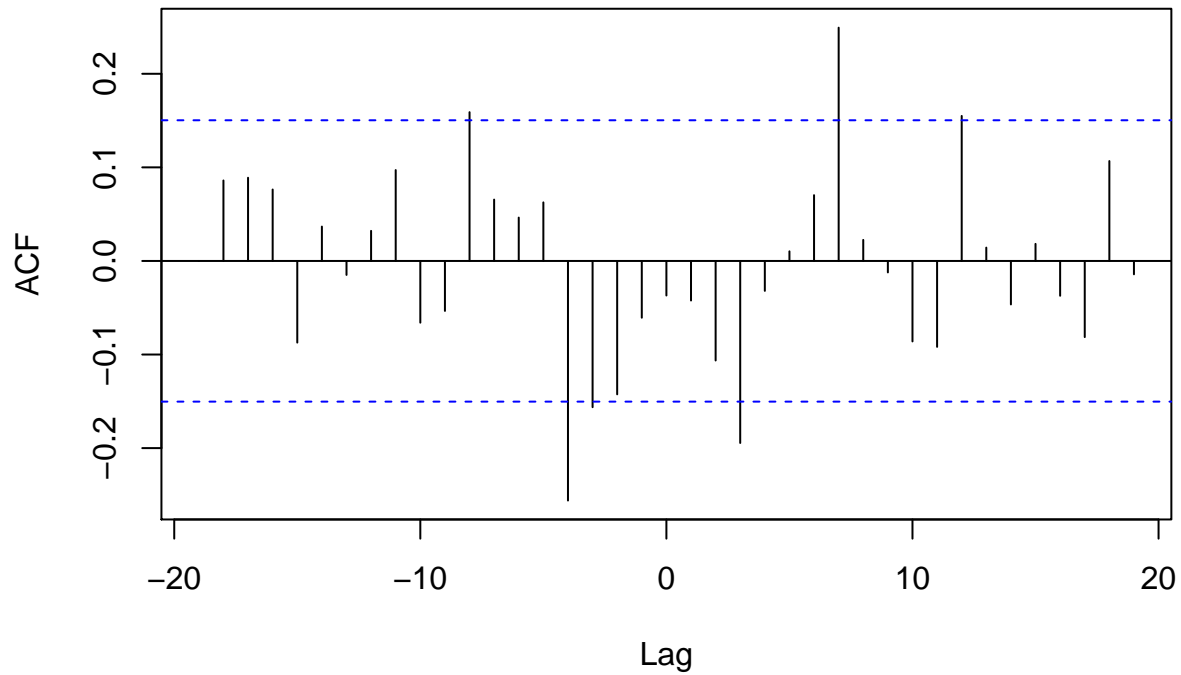
```
## [1] "pramhope"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
## 0.043 0.007 -0.095 -0.200 -0.143 0.022 -0.015 0.116 -0.041 -0.051
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
## -0.038 0.003 0.151 0.123 0.142 -0.047 -0.024 0.155 0.198 0.210
##     1     2     3     4     5     6     7     8     9    10
## 0.058 0.038 0.182 -0.075 -0.022 0.075 -0.013 -0.110 -0.162 0.015
##    11    12    13    14    15    16    17    18    19
## 0.085 0.103 -0.016 0.032 -0.094 0.053 0.032 -0.015 -0.098
```

data3\$OMX & data3[, i]



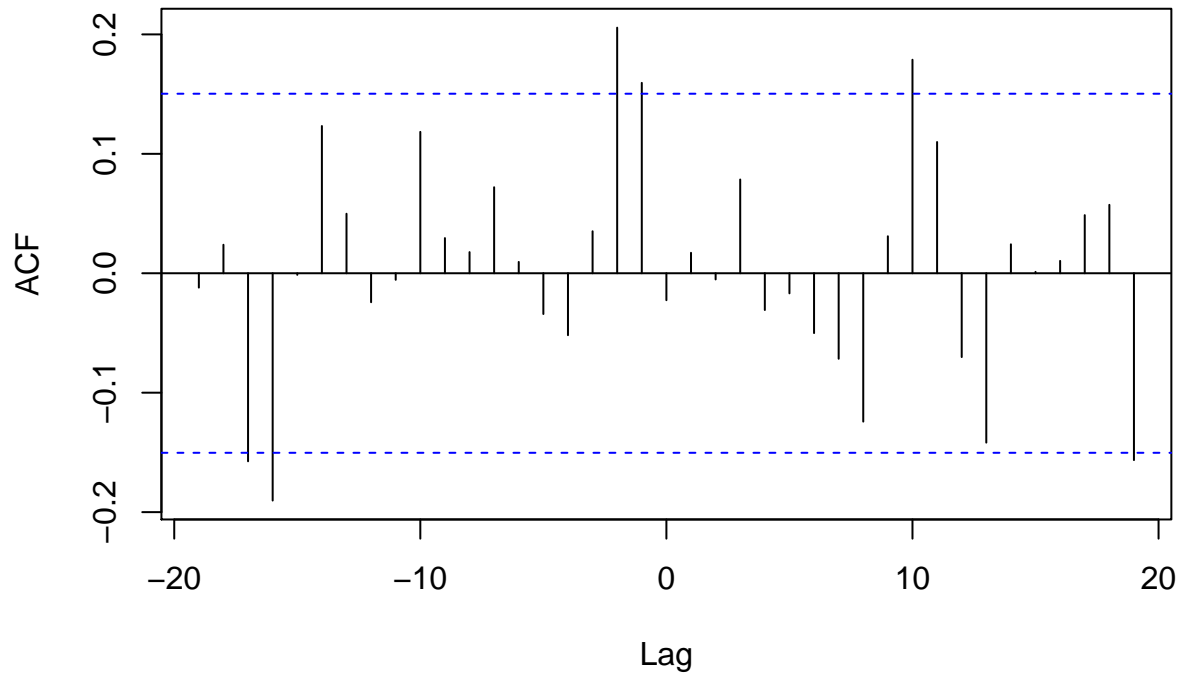
```
## [1] "shope"
##
## Autocorrelations of series 'X', by lag
##
##      -19      -18      -17      -16      -15      -14      -13      -12      -11      -10
## -0.076 -0.041 -0.114 -0.160 -0.122 -0.047  0.030  0.041  0.052  0.037
##      -9       -8       -7       -6       -5       -4       -3       -2       -1       0
##  0.050  0.074  0.077  0.106  0.031  0.014  0.095  0.166  0.176  0.126
##      1       2       3       4       5       6       7       8       9      10
##  0.163  0.112  0.090  0.123 -0.028  0.056  0.014 -0.103 -0.075 -0.062
##     11     12     13     14     15     16     17     18     19
##  0.054  0.009 -0.022 -0.002 -0.059  0.011 -0.081 -0.034 -0.074
```

data3\$OMX & data3[, i]



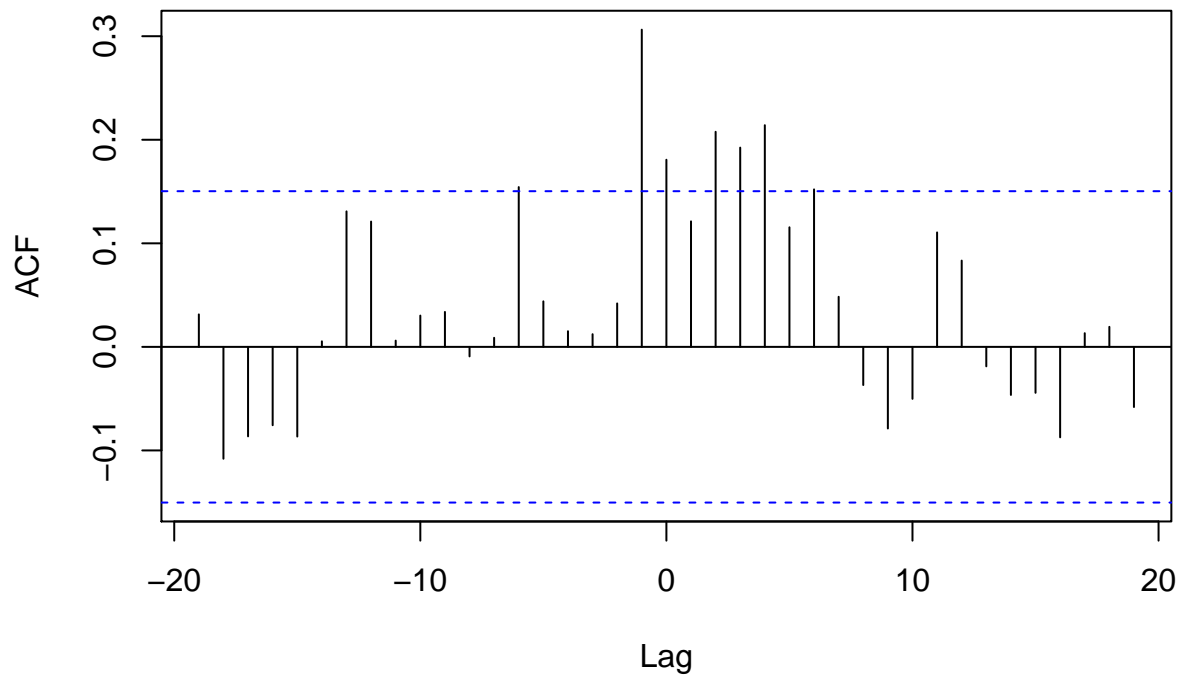
```
## [1] "ta"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
##  0.000  0.086  0.089  0.076 -0.087  0.037 -0.015  0.032  0.097 -0.066
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
## -0.053  0.159  0.066  0.046  0.063 -0.256 -0.156 -0.143 -0.061 -0.037
##     1     2     3     4     5     6     7     8     9    10
## -0.042 -0.106 -0.195 -0.032  0.010  0.070  0.249  0.023 -0.012 -0.086
##    11    12    13    14    15    16    17    18    19
## -0.092  0.155  0.014 -0.046  0.018 -0.037 -0.081  0.107 -0.014
```

data3\$OMX & data3[, i]



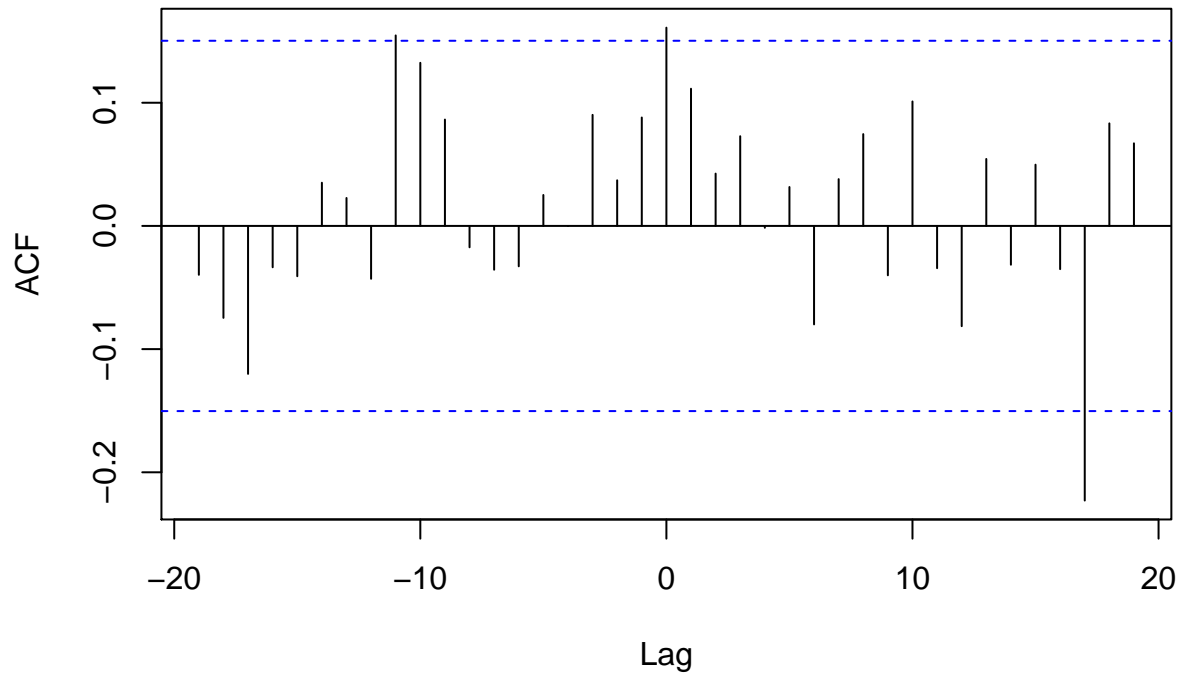
```
## [1] "ul"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
## -0.012  0.024 -0.157 -0.190 -0.001  0.123  0.050 -0.024 -0.005  0.118
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
##  0.029  0.018  0.072  0.010 -0.034 -0.052  0.035  0.206  0.159 -0.023
##     1     2     3     4     5     6     7     8     9    10
##  0.017 -0.005  0.079 -0.031 -0.017 -0.050 -0.072 -0.124  0.031  0.179
##    11    12    13    14    15    16    17    18    19
##  0.110 -0.070 -0.142  0.024  0.001  0.010  0.049  0.057 -0.156
```

data3\$OMX & data3[, i]



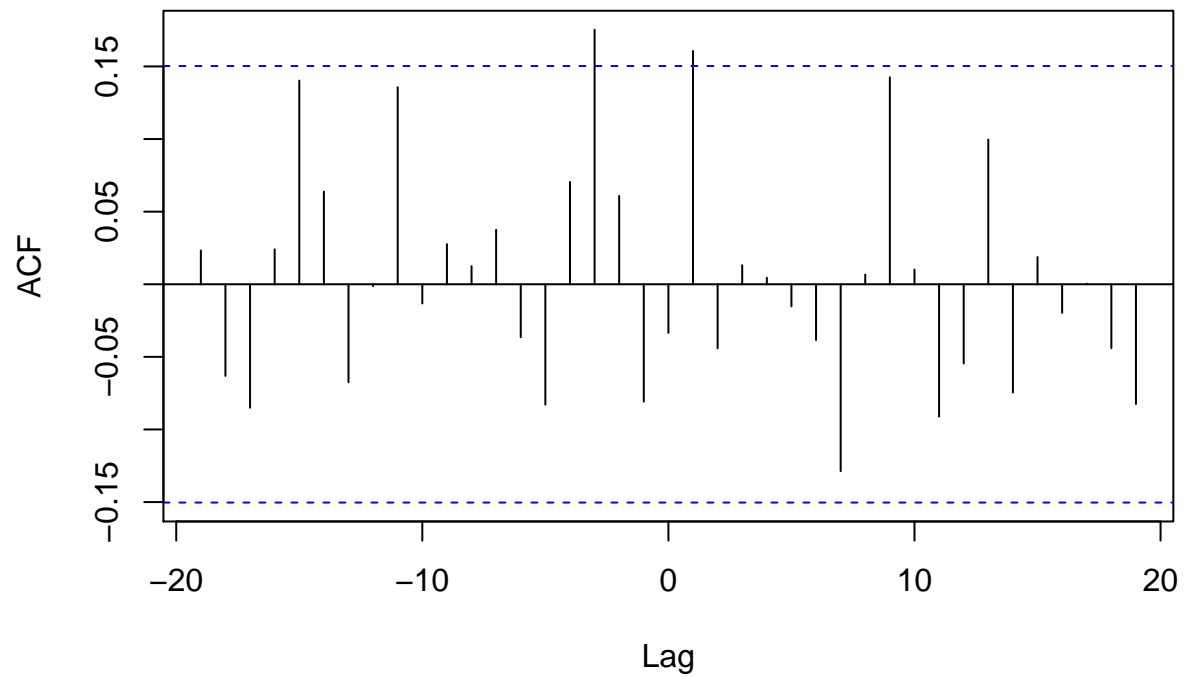
```
## [1] "vhope"
##
## Autocorrelations of series 'X', by lag
##
##      -19      -18      -17      -16      -15      -14      -13      -12      -11      -10
## 0.031 -0.108 -0.086 -0.076 -0.087  0.005  0.131  0.121  0.006  0.030
##      -9       -8       -7       -6       -5       -4       -3       -2       -1        0
## 0.034 -0.009  0.009  0.154  0.044  0.015  0.012  0.042  0.306  0.181
##      1        2        3        4        5        6        7        8        9       10
## 0.121  0.208  0.192  0.214  0.115  0.152  0.048 -0.037 -0.079 -0.050
##     11      12      13      14      15      16      17      18      19
## 0.111  0.083 -0.019 -0.046 -0.044 -0.087  0.013  0.019 -0.058
```

data3\$OMX & data3[, i]



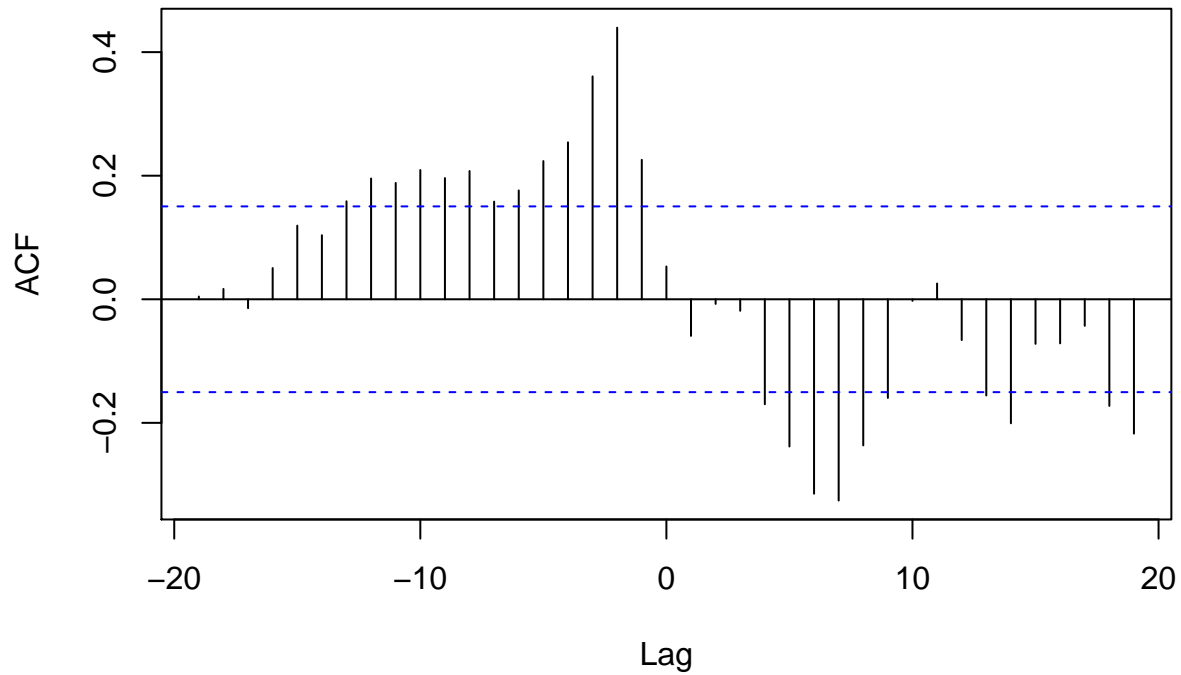
```
## [1] "vp"
##
## Autocorrelations of series 'X', by lag
##
##      -19      -18      -17      -16      -15      -14      -13      -12      -11      -10
## -0.040 -0.075 -0.120 -0.034 -0.041  0.035  0.023 -0.043  0.155  0.132
##      -9       -8       -7       -6       -5       -4       -3       -2       -1        0
##  0.086 -0.017 -0.036 -0.033  0.025  0.000  0.090  0.037  0.088  0.161
##       1        2        3        4        5        6        7        8        9       10
##  0.111  0.042  0.073 -0.002  0.032 -0.080  0.038  0.075 -0.040  0.101
##      11      12      13      14      15      16      17      18      19
## -0.034 -0.081  0.054 -0.032  0.050 -0.035 -0.223  0.083  0.067
```

data3\$OMX & data3[, i]



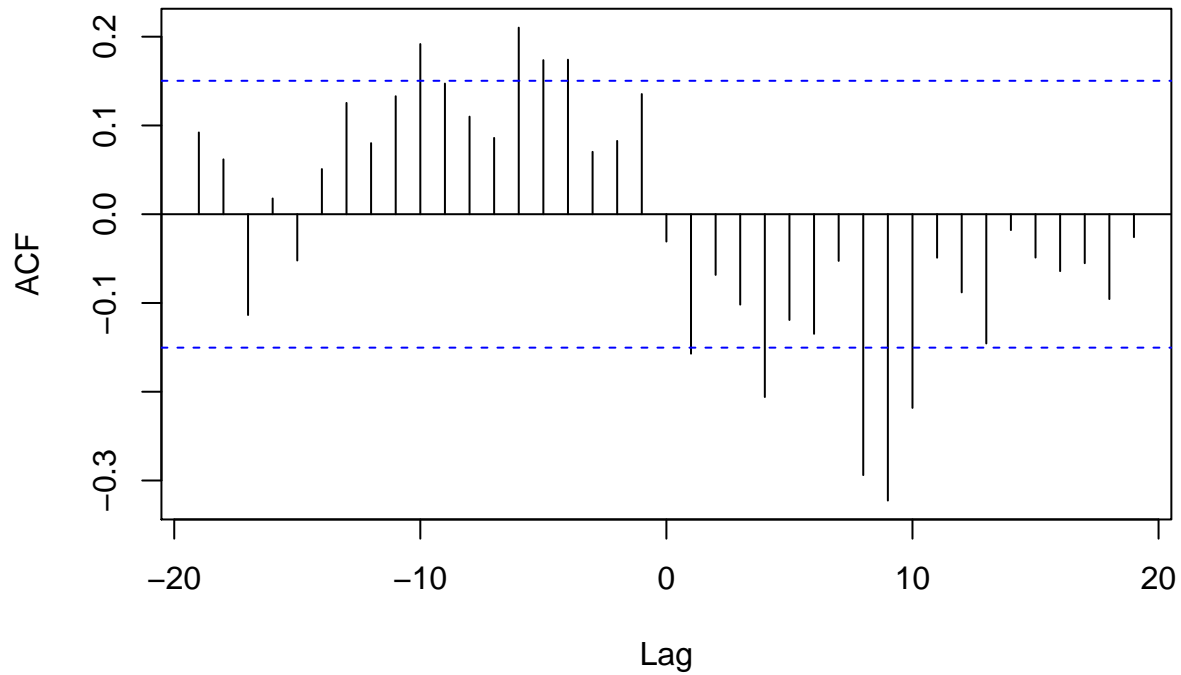
```
## [1] "mp"
##
## Autocorrelations of series 'X', by lag
##
##      -19      -18      -17      -16      -15      -14      -13      -12      -11      -10
## 0.023 -0.063 -0.085  0.024  0.140  0.064 -0.067 -0.001  0.136 -0.013
##      -9       -8       -7       -6       -5       -4       -3       -2       -1        0
## 0.028  0.012  0.037 -0.036 -0.083  0.070  0.175  0.061 -0.081 -0.034
##      1        2        3        4        5        6        7        8        9       10
## 0.161 -0.044  0.013  0.005 -0.015 -0.038 -0.129  0.007  0.143  0.010
##     11      12      13      14      15      16      17      18      19
## -0.091 -0.055  0.100 -0.075  0.019 -0.020  0.000 -0.044 -0.082
```

data3\$OMX & data3[, i]



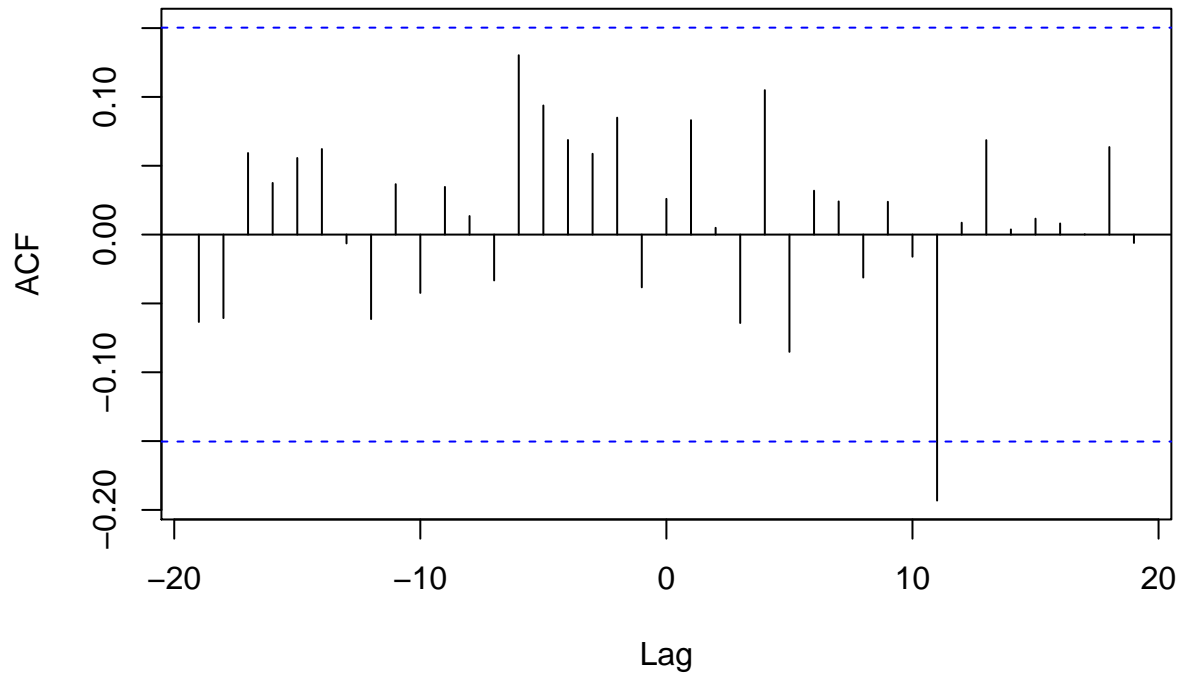
```
## [1] "palukanos"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
## 0.004 0.017 -0.014 0.051 0.119 0.104 0.159 0.195 0.188 0.209
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
## 0.196 0.208 0.158 0.176 0.224 0.254 0.361 0.440 0.226 0.053
##     1     2     3     4     5     6     7     8     9    10
## -0.059 -0.007 -0.019 -0.170 -0.238 -0.315 -0.326 -0.237 -0.160 -0.003
##    11    12    13    14    15    16    17    18    19
## 0.026 -0.066 -0.156 -0.201 -0.072 -0.071 -0.043 -0.173 -0.217
```


data3\$OMX & data3[, i]



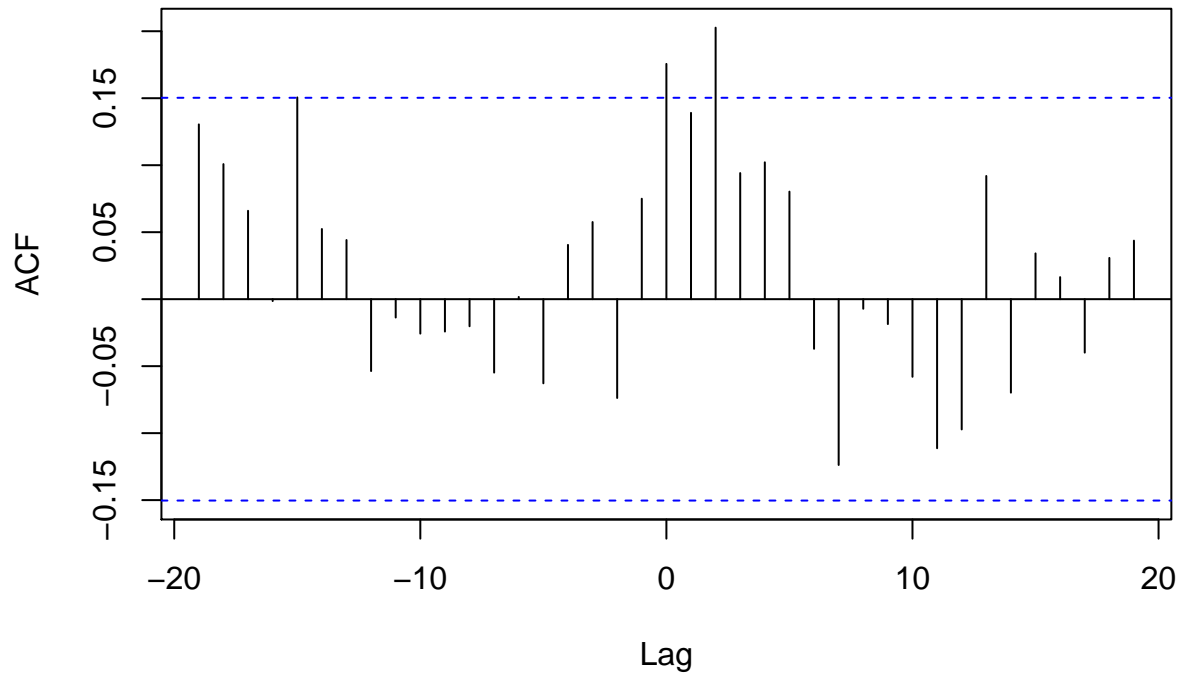
```
## [1] "gkl"
##
## Autocorrelations of series 'X', by lag
##
##   -19   -18   -17   -16   -15   -14   -13   -12   -11   -10
##  0.092  0.062 -0.114  0.018 -0.052  0.051  0.125  0.080  0.133  0.192
##    -9    -8    -7    -6    -5    -4    -3    -2    -1     0
##  0.147  0.110  0.086  0.210  0.174  0.174  0.070  0.083  0.135 -0.031
##     1     2     3     4     5     6     7     8     9    10
## -0.157 -0.068 -0.102 -0.206 -0.119 -0.135 -0.053 -0.294 -0.323 -0.218
##    11    12    13    14    15    16    17    18    19
## -0.049 -0.088 -0.146 -0.018 -0.049 -0.064 -0.055 -0.096 -0.026
```

data3\$OMX & data3[, i]



```
## [1] "ip"
##
## Autocorrelations of series 'X', by lag
##
##      -19      -18      -17      -16      -15      -14      -13      -12      -11      -10
## -0.063 -0.061  0.059  0.038  0.056  0.062 -0.006 -0.061  0.037 -0.042
##      -9       -8       -7       -6       -5       -4       -3       -2       -1        0
##  0.035  0.013 -0.033  0.130  0.094  0.069  0.059  0.085 -0.038  0.026
##       1        2        3        4        5        6        7        8        9       10
##  0.083  0.005 -0.064  0.105 -0.085  0.032  0.024 -0.031  0.024 -0.016
##      11      12      13      14      15      16      17      18      19
## -0.193  0.009  0.069  0.004  0.012  0.008  0.000  0.064 -0.006
```

data3\$OMX & data3[, i]



```
## [1] "kursas"
##
## Autocorrelations of series 'X', by lag
##
##      -19      -18      -17      -16      -15      -14      -13      -12      -11      -10
##  0.130  0.101  0.066 -0.001  0.151  0.052  0.044 -0.054 -0.014 -0.026
##      -9       -8       -7       -6       -5       -4       -3       -2       -1        0
## -0.024 -0.020 -0.055  0.002 -0.063  0.041  0.058 -0.074  0.075  0.176
##       1        2        3        4        5        6        7        8        9       10
##  0.139  0.203  0.094  0.102  0.080 -0.037 -0.124 -0.007 -0.019 -0.058
##      11      12      13      14      15      16      17      18      19
## -0.111 -0.097  0.092 -0.070  0.034  0.016 -0.040  0.031  0.044
```

```
mod_inflaciija = lm(ip ~ lag(kk,0) + lag(dll,0) + nedarbas + lag(SP350,0) + lag(inflaciija,0) + lag(phope,0) +
stepAIC(mod_inflaciija)
```

```
## Start:  AIC=456.14
## ip ~ lag(kk, 0) + lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(inflaciija,
##      0) + lag(phope, 0) + lag(pramhope, 0) + lag(shope, 0) + ta +
##      lag(ul, 0) + lag(vhope, 0) + lag(vp, 0) + lag(mp, 0) + lag(gkl,
##      0) + lag(mhope, 0) + lag(kursas, 0)
##
##              Df Sum of Sq  RSS   AIC
## - lag(vhope, 0)    1      0.2 2036.7 454.16
## - lag(inflaciija, 0) 1      0.4 2036.8 454.17
## - lag(kursas, 0)    1      0.4 2036.9 454.17
## - lag(mhope, 0)    1      3.3 2039.7 454.41
```

```

## - ta                1          4.0 2040.5 454.47
## - lag(SP350, 0)      1          4.9 2041.3 454.54
## - lag(vp, 0)         1          4.9 2041.3 454.54
## - lag(pramhope, 0)   1          6.4 2042.9 454.67
## - lag(dll, 0)        1          6.6 2043.1 454.69
## - lag(ul, 0)         1         13.2 2049.6 455.24
## <none>                2036.5 456.14
## - lag(phope, 0)      1         36.8 2073.2 457.18
## - lag(mp, 0)         1         52.8 2089.3 458.49
## - nedarbas           1        107.7 2144.1 462.90
## - lag(shope, 0)      1        139.2 2175.6 465.38
## - lag(gkl, 0)        1        160.5 2197.0 467.04
## - lag(kk, 0)         1       3935.5 5972.0 637.04
##
## Step: AIC=454.16
## ip ~ lag(kk, 0) + lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(infliacija,
##      0) + lag(phope, 0) + lag(pramhope, 0) + lag(shope, 0) + ta +
##      lag(ul, 0) + lag(vp, 0) + lag(mp, 0) + lag(gkl, 0) + lag(mhope,
##      0) + lag(kursas, 0)
##
##
##      Df Sum of Sq    RSS    AIC
## - lag(infliacija, 0)  1         0.3 2037.0 452.18
## - lag(kursas, 0)      1         0.4 2037.1 452.19
## - lag(mhope, 0)       1         3.4 2040.1 452.44
## - ta                  1         4.4 2041.1 452.53
## - lag(SP350, 0)       1         4.9 2041.6 452.57
## - lag(vp, 0)          1         5.1 2041.8 452.58
## - lag(pramhope, 0)    1         6.4 2043.1 452.69
## - lag(dll, 0)         1         6.4 2043.1 452.69
## - lag(ul, 0)          1        13.0 2049.7 453.24
## <none>                2036.7 454.16
## - lag(phope, 0)      1        37.3 2074.0 455.24
## - lag(mp, 0)          1        53.6 2090.3 456.57
## - nedarbas           1        107.8 2144.5 460.93
## - lag(shope, 0)      1        141.5 2178.2 463.58
## - lag(gkl, 0)        1        161.2 2197.9 465.11
## - lag(kk, 0)         1       3971.5 6008.2 636.06
##
## Step: AIC=452.18
## ip ~ lag(kk, 0) + lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(phope,
##      0) + lag(pramhope, 0) + lag(shope, 0) + ta + lag(ul, 0) +
##      lag(vp, 0) + lag(mp, 0) + lag(gkl, 0) + lag(mhope, 0) + lag(kursas,
##      0)
##
##
##      Df Sum of Sq    RSS    AIC
## - lag(kursas, 0)      1         0.3 2037.3 450.21
## - lag(mhope, 0)       1         3.4 2040.4 450.47
## - ta                  1         4.2 2041.2 450.53
## - lag(SP350, 0)       1         4.7 2041.7 450.57
## - lag(vp, 0)          1         5.2 2042.2 450.62
## - lag(pramhope, 0)    1         6.2 2043.1 450.69
## - lag(dll, 0)         1         6.7 2043.7 450.74
## - lag(ul, 0)          1        13.2 2050.1 451.28
## <none>                2037.0 452.18

```

```

## - lag(phope, 0)      1      37.0 2074.0 453.24
## - lag(mp, 0)         1      54.3 2091.2 454.65
## - nedarbas           1     118.7 2155.7 459.81
## - lag(shope, 0)      1     142.4 2179.4 461.67
## - lag(gkl, 0)        1     187.0 2224.0 465.11
## - lag(kk, 0)         1    3981.8 6018.8 634.36
##
## Step: AIC=450.21
## ip ~ lag(kk, 0) + lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(phope,
##      0) + lag(pramhope, 0) + lag(shope, 0) + ta + lag(ul, 0) +
##      lag(vp, 0) + lag(mp, 0) + lag(gkl, 0) + lag(mhope, 0)
##
##      Df Sum of Sq    RSS    AIC
## - lag(mhope, 0)      1      3.2 2040.5 448.48
## - ta                 1      4.1 2041.5 448.56
## - lag(vp, 0)         1      5.0 2042.3 448.63
## - lag(SP350, 0)      1      5.0 2042.4 448.63
## - lag(pramhope, 0)   1      6.1 2043.4 448.72
## - lag(dll, 0)        1      7.1 2044.4 448.80
## - lag(ul, 0)         1     15.2 2052.5 449.47
## <none>                2037.3 450.21
## - lag(phope, 0)      1     38.3 2075.6 451.37
## - lag(mp, 0)         1     54.1 2091.4 452.67
## - nedarbas           1    121.1 2158.4 458.03
## - lag(shope, 0)      1    145.3 2182.6 459.92
## - lag(gkl, 0)        1    186.7 2224.0 463.12
## - lag(kk, 0)         1   3983.1 6020.4 632.41
##
## Step: AIC=448.48
## ip ~ lag(kk, 0) + lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(phope,
##      0) + lag(pramhope, 0) + lag(shope, 0) + ta + lag(ul, 0) +
##      lag(vp, 0) + lag(mp, 0) + lag(gkl, 0)
##
##      Df Sum of Sq    RSS    AIC
## - lag(vp, 0)         1      1.8 2042.3 446.63
## - lag(SP350, 0)      1      4.6 2045.1 446.86
## - lag(dll, 0)        1      6.0 2046.5 446.97
## - lag(pramhope, 0)   1      9.8 2050.3 447.29
## - ta                 1     11.8 2052.3 447.46
## <none>                2040.5 448.48
## - lag(ul, 0)         1     25.9 2066.4 448.62
## - lag(phope, 0)      1     41.0 2081.5 449.86
## - lag(mp, 0)         1     55.0 2095.5 451.00
## - nedarbas           1    118.7 2159.2 456.09
## - lag(shope, 0)      1    149.3 2189.8 458.48
## - lag(gkl, 0)        1    184.6 2225.1 461.20
## - lag(kk, 0)         1   4009.7 6050.2 631.25
##
## Step: AIC=446.63
## ip ~ lag(kk, 0) + lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(phope,
##      0) + lag(pramhope, 0) + lag(shope, 0) + ta + lag(ul, 0) +
##      lag(mp, 0) + lag(gkl, 0)
##
##      Df Sum of Sq    RSS    AIC

```

```

## - lag(SP350, 0)      1      3.9 2046.2 444.95
## - lag(dll, 0)        1      7.5 2049.8 445.25
## - lag(pramhope, 0)   1     10.4 2052.7 445.49
## - ta                 1     11.1 2053.4 445.55
## <none>                2042.3 446.63
## - lag(ul, 0)         1     27.7 2070.0 446.92
## - lag(phope, 0)      1     40.1 2082.4 447.93
## - lag(mp, 0)         1     57.8 2100.1 449.37
## - nedarbas           1    117.8 2160.1 454.16
## - lag(shope, 0)      1    148.2 2190.5 456.53
## - lag(gkl, 0)        1    183.7 2226.0 459.27
## - lag(kk, 0)         1   4014.2 6056.5 629.43
##
## Step:  AIC=444.95
## ip ~ lag(kk, 0) + lag(dll, 0) + nedarbas + lag(phope, 0) + lag(pramhope,
##      0) + lag(shope, 0) + ta + lag(ul, 0) + lag(mp, 0) + lag(gkl,
##      0)
##
##           Df Sum of Sq    RSS    AIC
## - lag(dll, 0)      1      7.2 2053.4 443.55
## - lag(pramhope, 0) 1     10.7 2056.9 443.84
## - ta               1     14.1 2060.3 444.12
## <none>              2046.2 444.95
## - lag(ul, 0)       1     26.2 2072.4 445.11
## - lag(phope, 0)    1     39.6 2085.8 446.21
## - lag(mp, 0)       1     58.3 2104.5 447.73
## - nedarbas         1    121.6 2167.7 452.76
## - lag(shope, 0)    1    151.5 2197.7 455.09
## - lag(gkl, 0)      1    179.9 2226.0 457.27
## - lag(kk, 0)       1   4030.8 6077.0 628.00
##
## Step:  AIC=443.55
## ip ~ lag(kk, 0) + nedarbas + lag(phope, 0) + lag(pramhope, 0) +
##      lag(shope, 0) + ta + lag(ul, 0) + lag(mp, 0) + lag(gkl, 0)
##
##           Df Sum of Sq    RSS    AIC
## - lag(pramhope, 0) 1     12.0 2065.4 442.54
## - ta               1     13.9 2067.3 442.69
## - lag(ul, 0)       1     19.4 2072.8 443.15
## <none>              2053.4 443.55
## - lag(phope, 0)    1     34.3 2087.7 444.37
## - lag(mp, 0)       1     51.3 2104.7 445.74
## - nedarbas         1    116.0 2169.4 450.89
## - lag(shope, 0)    1    149.9 2203.3 453.53
## - lag(gkl, 0)      1    176.2 2229.7 455.55
## - lag(kk, 0)       1   4053.7 6107.1 626.84
##
## Step:  AIC=442.54
## ip ~ lag(kk, 0) + nedarbas + lag(phope, 0) + lag(shope, 0) +
##      ta + lag(ul, 0) + lag(mp, 0) + lag(gkl, 0)
##
##           Df Sum of Sq    RSS    AIC
## - ta               1     13.3 2078.7 441.63
## <none>              2065.4 442.54

```

```

## - lag(ul, 0)      1      28.5 2093.9 442.87
## - lag(phope, 0)   1      33.9 2099.3 443.30
## - lag(mp, 0)      1      44.3 2109.7 444.15
## - nedarbas        1     110.5 2175.9 449.40
## - lag(gkl, 0)     1     191.8 2257.2 455.64
## - lag(shope, 0)   1     193.0 2258.5 455.73
## - lag(kk, 0)      1    4043.8 6109.2 624.90
##
## Step: AIC=441.63
## ip ~ lag(kk, 0) + nedarbas + lag(phope, 0) + lag(shope, 0) +
##      lag(ul, 0) + lag(mp, 0) + lag(gkl, 0)
##
##              Df Sum of Sq    RSS    AIC
## <none>                2078.7 441.63
## - lag(phope, 0)   1      34.8 2113.5 442.45
## - lag(ul, 0)      1      40.0 2118.7 442.87
## - lag(mp, 0)      1      52.0 2130.7 443.83
## - nedarbas        1     114.9 2193.6 448.78
## - lag(shope, 0)   1     194.6 2273.3 454.84
## - lag(gkl, 0)     1     201.3 2280.0 455.34
## - lag(kk, 0)      1    4076.5 6155.2 624.17

##
## Call:
## lm(formula = ip ~ lag(kk, 0) + nedarbas + lag(phope, 0) + lag(shope,
##      0) + lag(ul, 0) + lag(mp, 0) + lag(gkl, 0), data = data3)
##
## Coefficients:
## (Intercept)      lag(kk, 0)      nedarbas lag(phope, 0) lag(shope, 0)
##      0.07204      0.80189      1.18602     -0.07356     -0.15544
##      lag(ul, 0)      lag(mp, 0)      lag(gkl, 0)
##     -0.02448     -0.05849     -1.02599

mod_po_aic_inflaciija = lm(ip ~ lag(kk, 0) + nedarbas + lag(phope, 0) + lag(shope,
0) + lag(ul, 0) + lag(mp, 0) + lag(gkl, 0), data=data3)

coeftest(mod_po_aic_inflaciija, vcov=vcovHC)

##
## t test of coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.072045   0.302235   0.2384 0.8118930
## lag(kk, 0)    0.801895   0.046921 17.0904 < 2.2e-16 ***
## nedarbas      1.186021   0.499197   2.3759 0.0186766 *
## lag(phope, 0) -0.073560   0.047677  -1.5429 0.1248151
## lag(shope, 0) -0.155436   0.045908  -3.3858 0.0008905 ***
## lag(ul, 0)    -0.024476   0.014536  -1.6838 0.0941437 .
## lag(mp, 0)    -0.058487   0.029257  -1.9991 0.0472735 *
## lag(gkl, 0)   -1.025988   0.297829  -3.4449 0.0007275 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
summary(mod_po_aic_infliacija)
```

```
##
## Call:
## lm(formula = ip ~ lag(kk, 0) + nedarbas + lag(phope, 0) + lag(shope,
##      0) + lag(ul, 0) + lag(mp, 0) + lag(gkl, 0), data = data3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.3952  -2.0030   0.2558   1.8773   9.3490
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.07204    0.27969   0.258 0.797053
## lag(kk, 0)      0.80189    0.04499  17.824 < 2e-16 ***
## nedarbas        1.18602    0.39632   2.993 0.003199 **
## lag(phope, 0)  -0.07356    0.04469  -1.646 0.101694
## lag(shope, 0)  -0.15544    0.03991  -3.894 0.000144 ***
## lag(ul, 0)     -0.02448    0.01387  -1.765 0.079394 .
## lag(mp, 0)     -0.05849    0.02905  -2.013 0.045726 *
## lag(gkl, 0)    -1.02599    0.25907  -3.960 0.000112 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.582 on 162 degrees of freedom
## Multiple R-squared:  0.6886, Adjusted R-squared:  0.6752
## F-statistic: 51.18 on 7 and 162 DF,  p-value: < 2.2e-16
```

```
mod_infliacija = lm(infliacija ~ lag(kk,0) + lag(dll,0) + nedarbas + lag(SP350,0) +lag(mhope,0) + lag(p
stepAIC(mod_infliacija)
```

```
## Start:  AIC=-440.93
## infliacija ~ lag(kk, 0) + lag(dll, 0) + nedarbas + lag(SP350,
##      0) + lag(mhope, 0) + lag(phope, 0) + lag(pramhope, 0) + lag(shope,
##      0) + ta + lag(ul, 0) + lag(vhope, 0) + lag(vp, 0) + lag(mp,
##      0) + lag(gkl, 0) + lag(ip, 0) + lag(kursas, 0)
##
##              Df Sum of Sq  RSS    AIC
## - lag(kk, 0)   1  0.00009 10.403 -442.93
## - lag(ip, 0)   1  0.00190 10.405 -442.90
## - lag(mp, 0)   1  0.00517 10.408 -442.84
## - lag(ul, 0)   1  0.00642 10.410 -442.82
## - lag(mhope, 0) 1  0.00831 10.412 -442.79
## - lag(dll, 0)   1  0.03545 10.439 -442.35
## - lag(vp, 0)    1  0.04614 10.449 -442.17
## - lag(phope, 0) 1  0.08205 10.485 -441.59
## - ta           1  0.11774 10.521 -441.01
## <none>          10.403 -440.93
## - lag(shope, 0) 1  0.14021 10.543 -440.65
## - lag(SP350, 0) 1  0.17257 10.576 -440.13
## - lag(kursas, 0) 1  0.20792 10.611 -439.56
```



```

## - lag(pramhope, 0) 1 0.21256 10.616 -439.49
## - lag(vhope, 0) 1 0.33490 10.738 -437.54
## - nedarbas 1 0.68278 11.086 -432.12
## - lag(gkl, 0) 1 1.20159 11.605 -424.35
##
## Step: AIC=-442.93
## infliacija ~ lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(mhope,
## 0) + lag(phope, 0) + lag(pramhope, 0) + lag(shope, 0) + ta +
## lag(ul, 0) + lag(vhope, 0) + lag(vp, 0) + lag(mp, 0) + lag(gkl,
## 0) + lag(ip, 0) + lag(kursas, 0)
##
##
## Df Sum of Sq RSS AIC
## - lag(mp, 0) 1 0.00543 10.409 -444.84
## - lag(ul, 0) 1 0.00633 10.410 -444.82
## - lag(ip, 0) 1 0.00775 10.411 -444.80
## - lag(mhope, 0) 1 0.00822 10.412 -444.79
## - lag(dll, 0) 1 0.03562 10.439 -444.34
## - lag(vp, 0) 1 0.04611 10.449 -444.17
## - lag(phope, 0) 1 0.08478 10.488 -443.55
## - ta 1 0.11767 10.521 -443.01
## <none> 10.403 -442.93
## - lag(shope, 0) 1 0.15625 10.559 -442.39
## - lag(SP350, 0) 1 0.17433 10.578 -442.10
## - lag(kursas, 0) 1 0.20791 10.611 -441.56
## - lag(pramhope, 0) 1 0.21313 10.616 -441.48
## - lag(vhope, 0) 1 0.33734 10.741 -439.50
## - nedarbas 1 0.74680 11.150 -433.14
## - lag(gkl, 0) 1 1.36697 11.770 -423.94
##
## Step: AIC=-444.84
## infliacija ~ lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(mhope,
## 0) + lag(phope, 0) + lag(pramhope, 0) + lag(shope, 0) + ta +
## lag(ul, 0) + lag(vhope, 0) + lag(vp, 0) + lag(gkl, 0) + lag(ip,
## 0) + lag(kursas, 0)
##
##
## Df Sum of Sq RSS AIC
## - lag(ip, 0) 1 0.00501 10.414 -446.76
## - lag(ul, 0) 1 0.00582 10.415 -446.74
## - lag(mhope, 0) 1 0.00927 10.418 -446.69
## - lag(dll, 0) 1 0.03062 10.439 -446.34
## - lag(vp, 0) 1 0.05121 10.460 -446.00
## - lag(phope, 0) 1 0.09678 10.505 -445.26
## - ta 1 0.11369 10.522 -444.99
## <none> 10.409 -444.84
## - lag(shope, 0) 1 0.15254 10.561 -444.36
## - lag(SP350, 0) 1 0.17136 10.580 -444.06
## - lag(kursas, 0) 1 0.20830 10.617 -443.47
## - lag(pramhope, 0) 1 0.22899 10.638 -443.14
## - lag(vhope, 0) 1 0.36787 10.777 -440.93
## - nedarbas 1 0.74638 11.155 -435.06
## - lag(gkl, 0) 1 1.36358 11.772 -425.91
##
## Step: AIC=-446.76
## infliacija ~ lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(mhope,

```

```

##      0) + lag(phope, 0) + lag(pramhope, 0) + lag(shope, 0) + ta +
##      lag(ul, 0) + lag(vhope, 0) + lag(vp, 0) + lag(gkl, 0) + lag(kursas,
##      0)
##
##
##      Df Sum of Sq    RSS    AIC
## - lag(ul, 0)      1  0.00572 10.419 -448.66
## - lag(mhope, 0)   1  0.00821 10.422 -448.62
## - lag(dll, 0)     1  0.03023 10.444 -448.26
## - lag(vp, 0)      1  0.04937 10.463 -447.95
## - lag(phope, 0)   1  0.09405 10.508 -447.23
## - ta              1  0.10987 10.524 -446.97
## <none>                                10.414 -446.76
## - lag(shope, 0)   1  0.15268 10.566 -446.28
## - lag(SP350, 0)   1  0.17459 10.588 -445.93
## - lag(kursas, 0)  1  0.20684 10.620 -445.41
## - lag(pramhope, 0) 1  0.22474 10.638 -445.13
## - lag(vhope, 0)   1  0.37117 10.785 -442.80
## - nedarbas        1  0.74416 11.158 -437.02
## - lag(gkl, 0)     1  1.37008 11.784 -427.74
##
## Step: AIC=-448.66
## infliacija ~ lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(mhope,
##      0) + lag(phope, 0) + lag(pramhope, 0) + lag(shope, 0) + ta +
##      lag(vhope, 0) + lag(vp, 0) + lag(gkl, 0) + lag(kursas, 0)
##
##
##      Df Sum of Sq    RSS    AIC
## - lag(mhope, 0)   1  0.01943 10.439 -450.35
## - lag(dll, 0)     1  0.02485 10.444 -450.26
## - lag(vp, 0)      1  0.07676 10.496 -449.41
## - lag(phope, 0)   1  0.09452 10.514 -449.13
## - ta              1  0.11138 10.531 -448.85
## <none>                                10.419 -448.66
## - lag(shope, 0)   1  0.15694 10.576 -448.12
## - lag(SP350, 0)   1  0.16940 10.589 -447.92
## - lag(kursas, 0)  1  0.20290 10.622 -447.38
## - lag(pramhope, 0) 1  0.22013 10.639 -447.11
## - lag(vhope, 0)   1  0.38305 10.803 -444.52
## - nedarbas        1  0.75620 11.176 -438.75
## - lag(gkl, 0)     1  1.37385 11.793 -429.61
##
## Step: AIC=-450.35
## infliacija ~ lag(dll, 0) + nedarbas + lag(SP350, 0) + lag(phope,
##      0) + lag(pramhope, 0) + lag(shope, 0) + ta + lag(vhope, 0) +
##      lag(vp, 0) + lag(gkl, 0) + lag(kursas, 0)
##
##
##      Df Sum of Sq    RSS    AIC
## - lag(dll, 0)     1  0.01400 10.453 -452.12
## - lag(vp, 0)      1  0.06673 10.506 -451.26
## - lag(phope, 0)   1  0.08614 10.525 -450.95
## - ta              1  0.09395 10.533 -450.82
## <none>                                10.439 -450.35
## - lag(shope, 0)   1  0.14825 10.587 -449.95
## - lag(SP350, 0)   1  0.18035 10.619 -449.43
## - lag(pramhope, 0) 1  0.20150 10.640 -449.09

```

```

## - lag(kursas, 0)      1    0.21062 10.649 -448.95
## - lag(vhope, 0)      1    0.38168 10.820 -446.24
## - nedarbas           1    0.75253 11.191 -440.51
## - lag(gkl, 0)        1    1.36911 11.808 -431.39
##
## Step: AIC=-452.12
## infliacija ~ nedarbas + lag(SP350, 0) + lag(phope, 0) + lag(pramhope,
##      0) + lag(shope, 0) + ta + lag(vhope, 0) + lag(vp, 0) + lag(gkl,
##      0) + lag(kursas, 0)
##
##              Df Sum of Sq    RSS    AIC
## - lag(vp, 0)      1    0.07537 10.528 -452.90
## - lag(phope, 0)    1    0.10174 10.555 -452.47
## - ta              1    0.10514 10.558 -452.42
## <none>                        10.453 -452.12
## - lag(shope, 0)    1    0.14675 10.600 -451.75
## - lag(SP350, 0)    1    0.17543 10.628 -451.29
## - lag(pramhope, 0) 1    0.20011 10.653 -450.89
## - lag(kursas, 0)   1    0.20385 10.657 -450.83
## - lag(vhope, 0)    1    0.38262 10.835 -448.01
## - nedarbas         1    0.73869 11.191 -442.51
## - lag(gkl, 0)      1    1.38714 11.840 -432.93
##
## Step: AIC=-452.9
## infliacija ~ nedarbas + lag(SP350, 0) + lag(phope, 0) + lag(pramhope,
##      0) + lag(shope, 0) + ta + lag(vhope, 0) + lag(gkl, 0) + lag(kursas,
##      0)
##
##              Df Sum of Sq    RSS    AIC
## - ta              1    0.11803 10.646 -453.00
## <none>                        10.528 -452.90
## - lag(phope, 0)    1    0.12576 10.654 -452.88
## - lag(pramhope, 0) 1    0.18195 10.710 -451.98
## - lag(kursas, 0)   1    0.19012 10.718 -451.85
## - lag(shope, 0)    1    0.19432 10.723 -451.79
## - lag(SP350, 0)    1    0.19752 10.726 -451.74
## - lag(vhope, 0)    1    0.37561 10.904 -448.94
## - nedarbas         1    0.71516 11.243 -443.72
## - lag(gkl, 0)      1    1.35737 11.886 -434.28
##
## Step: AIC=-453
## infliacija ~ nedarbas + lag(SP350, 0) + lag(phope, 0) + lag(pramhope,
##      0) + lag(shope, 0) + lag(vhope, 0) + lag(gkl, 0) + lag(kursas,
##      0)
##
##              Df Sum of Sq    RSS    AIC
## <none>                        10.646 -453.00
## - lag(phope, 0)    1    0.12811 10.774 -452.97
## - lag(kursas, 0)    1    0.16338 10.810 -452.41
## - lag(pramhope, 0) 1    0.17991 10.826 -452.15
## - lag(shope, 0)    1    0.18711 10.833 -452.04
## - lag(SP350, 0)    1    0.22461 10.871 -451.45
## - lag(vhope, 0)    1    0.48081 11.127 -447.49
## - nedarbas         1    0.67710 11.323 -444.52

```

```
## - lag(gkl, 0)      1    1.31790 11.964 -435.16

##
## Call:
## lm(formula = infliacija ~ nedarbas + lag(SP350, 0) + lag(phope,
##      0) + lag(pramhope, 0) + lag(shope, 0) + lag(vhope, 0) + lag(gkl,
##      0) + lag(kursas, 0), data = data3)
##
## Coefficients:
##      (Intercept)      nedarbas      lag(SP350, 0)      lag(phope, 0)
##      -0.030444      -0.089851      -0.008028      -0.004406
## lag(pramhope, 0)      lag(shope, 0)      lag(vhope, 0)      lag(gkl, 0)
##      -0.007458      -0.005009      -0.018008      0.082371
## lag(kursas, 0)
##      0.012768

mod_po_aic_infliacija = lm(infliacija ~ nedarbas + lag(SP350, 0) + lag(phope, 0) + lag(pramhope,
0) + lag(shope, 0) + lag(vhope, 0) + lag(gkl, 0) + lag(kursas,
0) , data=data3)

coeftest(mod_po_aic_infliacija, vcov=vcovHC)

##
## t test of coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.0304436  0.0207888 -1.4644 0.1450290
## nedarbas      -0.0898510  0.0301417 -2.9810 0.0033198 **
## lag(SP350, 0) -0.0080283  0.0049812 -1.6117 0.1089838
## lag(phope, 0) -0.0044064  0.0031657 -1.3919 0.1658636
## lag(pramhope, 0) -0.0074583  0.0046434 -1.6062 0.1101908
## lag(shope, 0) -0.0050085  0.0025857 -1.9370 0.0544931 .
## lag(vhope, 0) -0.0180075  0.0085969 -2.0946 0.0377686 *
## lag(gkl, 0)    0.0823709  0.0212751  3.8717 0.0001568 ***
## lag(kursas, 0)  0.0127683  0.0091151  1.4008 0.1632015
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

summary(mod_po_aic_infliacija)

##
## Call:
## lm(formula = infliacija ~ nedarbas + lag(SP350, 0) + lag(phope,
##      0) + lag(pramhope, 0) + lag(shope, 0) + lag(vhope, 0) + lag(gkl,
##      0) + lag(kursas, 0), data = data3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82693 -0.11478  0.00718  0.13500  0.58181
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)      -0.030444    0.020065   -1.517   0.13117
## nedarbas         -0.089851    0.028079   -3.200   0.00166 **
## lag(SP350, 0)    -0.008028    0.004356   -1.843   0.06716 .
## lag(phope, 0)    -0.004406    0.003166   -1.392   0.16588
## lag(pramhope, 0) -0.007458    0.004522   -1.649   0.10101
## lag(shope, 0)    -0.005009    0.002978   -1.682   0.09448 .
## lag(vhope, 0)    -0.018008    0.006678   -2.697   0.00775 **
## lag(gkl, 0)       0.082371    0.018451    4.464   1.5e-05 ***
## lag(kursas, 0)    0.012768    0.008123    1.572   0.11795
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2571 on 161 degrees of freedom
## Multiple R-squared:  0.2458, Adjusted R-squared:  0.2084
## F-statistic:  6.56 on 8 and 161 DF,  p-value: 2.134e-07
```

```
mySummary <- function(modelis_po_aic, VCOV) {
  print(coeftest(modelis_po_aic, vcov. = VCOV))
  print(waldtest(modelis_po_aic, vcov = VCOV))
}
```

```
mySummary(modelis_po_aic,vcovHC)
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.087098   0.531421   2.0456 0.042550 *
## lag(SP350, 1)      0.360918   0.124273   2.9042 0.004242 **
## lag(infliacija, 2) -3.898667   2.693478  -1.4474 0.149872
## lag(mhope, 3)       0.146672   0.090940   1.6129 0.108893
## lag(phope, 12)      0.169685   0.094204   1.8012 0.073687 .
## lag(ul, 10)         0.061529   0.029758   2.0677 0.040399 *
## lag(gkl, 9)        -1.276498   0.557982  -2.2877 0.023562 *
## lag(ip, 4)          0.128462   0.086807   1.4799 0.141022
## lag(kursas, 2)      0.380851   0.199093   1.9129 0.057675 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Wald test
##
## Model 1: OMX ~ lag(SP350, 1) + lag(infliacija, 2) + lag(mhope, 3) + lag(phope,
##      12) + lag(ul, 10) + lag(gkl, 9) + lag(ip, 4) + lag(kursas,
##      2)
## Model 2: OMX ~ 1
##   Res.Df Df    F   Pr(>F)
## 1     149
## 2     157 -8 3.013 0.003654 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Priedai:

Kintamieji

Darbe naudojami mėnesiniai duomenys nuo 2002 metų birželio iki 2016 metų liepos. Duomenys gauti iš Eurostat ir Europos Centrinio Banko svetainių. Susiduriama su mėnesinių rodiklių trūkumu: BVP, gyventojų pajamos, pinigų paklausa yra ketvirtiniai duomenys. Taip pat žvelgiant pasauliniu mastu, Vilniaus akcijų birža yra smulki, o akcijų likvidumas čia taip pat žemas. Nepaisant šių problemų, buvo rasta koreliacija tarp kai kurių makroekonominių duomenų ir OMX Vilnius indekso gražų.

- S&P 350 Europe - tai indeksas sudarytas iš 350 didžiausių Europos imonių akcijų.
- S&P 500 - tai indeksas sudarytas iš 500 didžiausių Amerikos imonių akcijų.
- Euribor 3 mėnesių - palūkanų norma už kurią Europos bankai skolina pinigus vieni kitiems trijų mėnesių laikotarpiui.
- JAV dolerio ir Euro valiutos kursas.
- Nedarbas - darbingų nedarbančių žmonių skaičius 1000 žmonių.
- Pramonės gamintojų kainos - matuoja vidutinę kainų raidą, visų produktų bei paslaugų iš pramonės sektoriaus ir parduotų vidinėje rinkoje. Procentinis pokytis nuo praeito mėnesio.
- Infliacija - 12 mėnesių infliacija apskaičiuota pagal mažmeninės prekybos indeksą.
- Industrinė produkcija - indeksas matuoja produkcijos kiekį nuo gamybos, kasybos, elektros ir dujų pramonės. Atskaitiniai metai 2010 (2010m. indeksas = 100).
- Darbo lygio ir užsakymų lūkesčiai - mėnesinės apklausos, teigiamų ir neigiamų atsakymų santykis.
- Pasitikėjimo indikatoriai - pramonės, mažmeninės prekybos, vartotojų, paslaugų, statybų. Tai yra apklausų teigiamų ir neigiamų atsakymų santykis.
- Mažmeninė prekyba, išskyrus variklinių transporto priemonių ir motociklų prekybą- indeksuotas rodiklis (2010m indeksas = 100).