

Vilniaus Universitetas
Matematikos ir informatikos fakultetas
Ekonometrinės analizės katedra

Kursinis darbas

Įskaitinių eismo įvykių Lietuvos keliuose analizė

Atliko: Linas Šyvis ir Kornelijus Samsonas

Vadovas: prof. Vydas Čekanavičius

VILNIUS 2016

Turinys

Tvarkome duomenis	2
Tvarkom Blaivumo stulpelį	2
Tvarkom Meteo salygų stulp.	3
Įvykių sk. 1000-iui gyventojų didžiausiuose miestuose grafikas.	4
Tikrinam įvykių priklausomybę nuo amžiaus	7
Forecastinam autoįvykius	7
Kaltininku procentas miestuose (Vyrų/Moterys)	7
koreliacijos	7
Literatūra	8

Reikalingi užduotims paketai.

```
library(fpp)
library(dplyr)
```

Nuskaitytume duomenis.

```
rawdata <- read.csv2("2003-2015.csv", header = T)
gyventojai <- read.csv2("gyventojai.csv", header = T, skip = 1)
kor <- read.csv2("koreliacijos.csv", header = T)
```

Tvarkome duomenis

Paliekame tik 5 didžiuosius miestus

```
miestai <- c("Vilniaus m. sav.", "Kauno m. sav.", "Klaipėdos m. sav.", "Siauliu m. sav.",
            "Panevezio m. sav.")
data <- subset(rawdata, Vieta %in% miestai)
```

Turime stulpelius “Gimimo data” ir “Stazas”. Jų reikšmės tikėtina stipriai koreliuoja, patikrinkime. Pertvarkysime stulpelio “Gimimo data” reikšmes taip, kad matytume eismo įvykio dalyvio amžių ir patikrinsime amžiaus koreliaciją su “Stažo” reikšmėmis.

Stulpelį “Gimimo data” pertvarkome į “Amzius”.

```
Amzius <- as.numeric(format(Sys.Date(), format = "%Y")) - as.numeric(format(as.Date(data$Gimimo_data,
                                     format = "%Y"), format = "%Y"))
data <- cbind(data, Amzius)
```

Dabar turime eismo įvykių dalyvių amžių, kuro koreliaciją su stažu galime patikrinti.

Patikrinsime koreliaciją.

```
sum(is.na(data$Stazas))
```

```
## [1] 8184
```

```
sum(is.na(data$Amzius))
```

```
## [1] 1316
```

```
cor(data$Amzius, data$Stazas, use = "pairwise.complete.obs")
```

```
## [1] 0.7957286
```

Tvarkom Blaivumo stulpelį

Matome, kad stulpelį “Blaivumas” sudaro 6 skirtingos reikšmės. Sutrauksime jas visas į “Blaivus” ir “Neblaivus”.

```
data[data[, "Blaivumas"] %in% c("Apsvaiges nuo narkotiniu, psichotropiniu ar kitu psichika veikianciu m
    "Atsisake buti patikrintas", "Neblaivus"), "Blaivumas"] <- "Neblaivus"
data$Blaivumas <- factor(data$Blaivumas)
table(data$Blaivumas)
```

```
##
##    Blaivus Neblaivus
##    21615      2244
```

Tvarkom Meteo salygų stulp.

```
data[data[, "Meteorologines_salygos"] == "Lijundra", "Meteorologines_salygos"] <- "Lietus"
data[data[, "Meteorologines_salygos"] == "Puga", "Meteorologines_salygos"] <- "Sniegas, krusa"
data$Meteorologines_salygos <- factor(data$Meteorologines_salygos)
table(data$Meteorologines_salygos)
```

```
##
##    Apsiniauke      Giedra      Lietus      Rukas Sniegas, krusa
##          8761      12660      2027      141      516
## Stiprus vejas
##          19
```

Kai kurie stulpeliai turi NA reikšmių, reikia jas pašalinti.

```
salinamieji <- c("Lytis", "Apgadintu_transporto_priemoniu_skaicius", "Pilietybe",
    "Neigalumas", "Blaivumas", "Amzius")
ndata <- data[complete.cases(data[salinamieji]), ]
sapply(lapply(ndata, is.na), sum)
```

```
##
##
##          Vieta
##          0
##          Data
##          0
##          Laikas
##          0
##          Dalyviu_skaicius
##          0
##          Zuvusiu_skaicius
##          0
##          Suzeistu_skaicius
##          0
## Dalyvavusiu_transporto_priemoniu_skaicius
##          0
## Apgadintu_transporto_priemoniu_skaicius
##          0
##          Kelio_reiksme
##          20823
##          Dangos_rusis
##          0
```

```
##                                Dangos_bukle
##                                0
##                                Paros_metas
##                                0
##                                Meteorologines_salygos
##                                0
##                                Kelio_kreive
##                                0
##                                Atitvarai
##                                0
##                                Eismo_ivykio_dalyvio_kategorija
##                                0
## KET_pazeidimas_del_kurio_ivyko_eismo_ivykis
##                                0
##                                Lytis
##                                0
##                                Pilietybe
##                                0
##                                Gimimo_data
##                                0
##                                Neigalumas
##                                0
##                                Saugos_dirzas
##                                9588
##                                Oro_pagalviu_suveikimas
##                                15707
##                                Vairuotojo_kvalifikacija
##                                5080
##                                Stazas
##                                6275
##                                Blaivumas
##                                0
##                                Kaltas
##                                0
##                                Amzius
##                                0
```

Pašalinome NA reikšmes (stulpeliuose, kuriuose NA > 5000, NA palikome, kad neprarastume didžiosios dalies duomenų).

5 didžiausius Lietuvos miestus lyginsime pagal įvykius 1000-iui gyventojų.

Sukuriame rodiklį 1000 gyventojų.

```
tukst <- gyventojai[,3]/1000
gyventojai <- cbind(gyventojai, tukst)
```

Įvykių sk. 1000-iui gyventojų didžiausiuose miestuose grafikas.

****Skaičiuojame mėnesinį įvykių skaičių.**

```

by_month <- group_by(data, format(as.Date(data$Data, format = "%Y-%m-%d"), "%Y-%m"),
  Vieta)
agreguoti <- summarise(by_month, sum(Kaltas), sum(Dalyviu_skaicius), sum(Suzeistu_skaicius),
  sum(Zuvusiu_skaicius))
colnames(agreguoti) <- list("Data", "Vieta", "Ivykiai", "Dalyviai", "Suzeisti",
  "Zuve")

```

****Išsaugome didžiųjų miestų įvykių skaičių ir gyventojų laiko eilutes.**

```

ivykiusk <- function(i)
  ts(agreguoti[agreguoti$Vieta == i, "Ivykiai"], start=c(2003,1), frequency = 12)

gyvsk <- function(i)
  ts(rep(gyventojai[gyventojai$Vieta == i, "Visi"], each = 12), start = c(2003,1), frequency = 12)

ivykiai <- sapply(miestai, ivykiusk)
names(ivykiai)<-sprintf(miestai)
gyven <- sapply(miestai, gyvsk)
names(gyven)<-sprintf(miestai)

tukst <- function(i)
  ivykiai[,i] * 1000 / gyven[,i]

tukstgyv <- ts(sapply(miestai, tukst), start = c(2003,1), frequency = 12)
names(tukstgyv)<-sprintf(miestai)

```

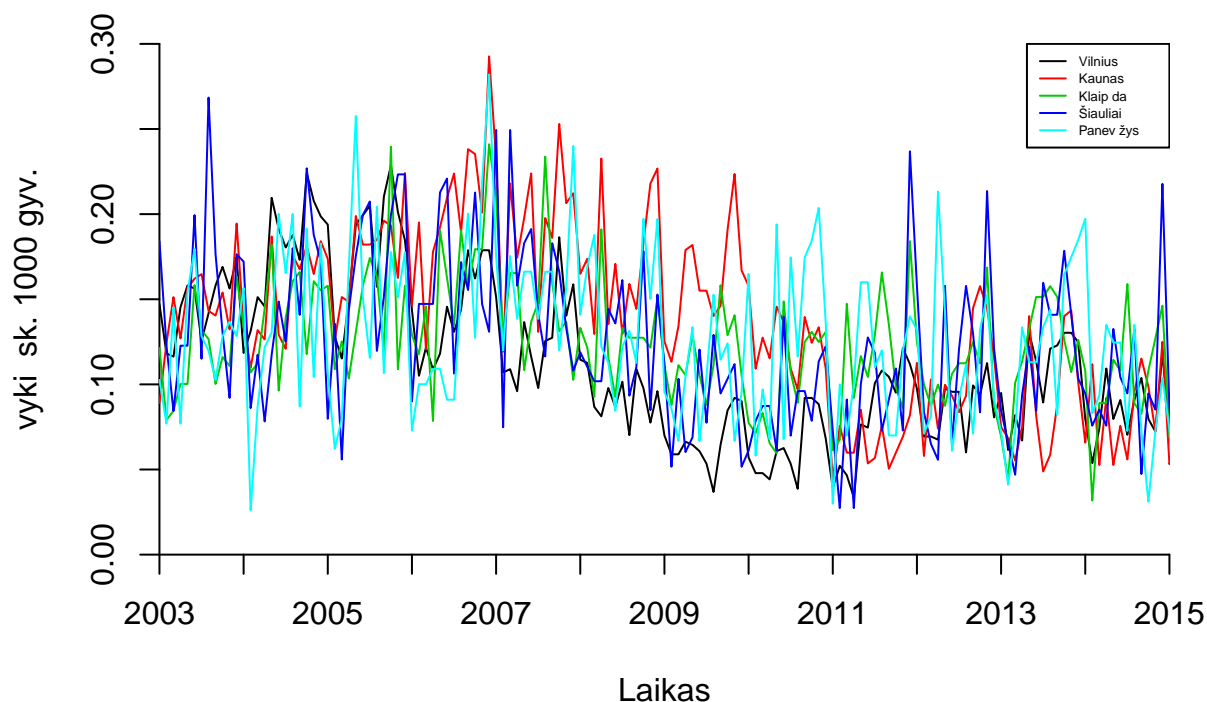
Brėžiame grafiką.

```

plot(0, type = "n", xlim = c(2003,2015), ylim = c(0,0.3), yaxs="i", xaxs="i", main = "Įvykių skaičius 1000 gyventojui")
par(xaxp=c(2003, 2015, 12), yaxp=c(0, 0.3, 6))
axis(1)
axis(2)
for(i in 1:5)
  lines(tukstgyv[,i], col = i, lwd = 1)
legend("topright", col = c(1:5), lty = 1, lwd = 1, legend = c("Vilnius", "Kaunas", "Klaipėda", "Šiauliai", "Utena"))

```

vyki skaičius 1000-ii gyventoj



[Reikalingi komentarai] Pagalbiniai stulpeliai

```
diena <- ndata$Paros_metas == "Diena"
diena <- diena * 1
moteris <- ndata$Lytis == "Moteris"
moteris <- moteris * 1
asfaltas <- ndata$Dangos_rusis == "Asfaltbetonis, cementbetonis"
asfaltas <- asfaltas * 1
zvyras <- ndata$Dangos_rusis == "zvyrkelis"
zvyras <- zvyras * 1
slapia <- ndata$Dangos_bukle == "slapia"
slapia <- slapia * 1
giedra <- ndata$Meteorologines_salygos == "Giedra"
giedra <- giedra * 1

ndata <- cbind(ndata, diena, moteris, asfaltas, zvyras, slapia, giedra)
```

Sukuriam miestų lenteles

```
by_year <- group_by(ndata, format(as.Date(ndata$Data, format="%Y-%m-%d"), "%Y"), Vieta)
metiniai <- summarise(by_year, sum(Kaltas), sum(data$Blaivumas == "Neblaivus", na.rm=T), sum(diena)/sum(
colnames(metiniai) <- list("Data", "Vieta", "Ivykiai", "Neblaivus", "Diena", "Moteris", "Asfaltas", "Zvyras", "Slapia", "Giedra"))

vilnius <- data.frame(metiniai[metiniai$Vieta == "Vilniaus m. sav.", 5:10])
kaunas <- data.frame(metiniai[metiniai$Vieta == "Kauno m. sav.", 5:10])
siauliai <- metiniai[metiniai$Vieta == "Siauliu m. sav.", 5:10]
```

```
panevezys <- metiniai[metiniai$Vieta == "Panevezio m. sav.", 5:10]
klaipeda <- metiniai[metiniai$Vieta == "Klaipedos m. sav.", 5:10]
```

```
chisq.test(vilnius[,i], kaunas[,i])
```

```
## Warning in chisq.test(vilnius[, i], kaunas[, i]): Chi-squared approximation
## may be incorrect
```

```
##
## Pearson's Chi-squared test
##
## data: vilnius[, i] and kaunas[, i]
## X-squared = 156, df = 144, p-value = 0.2335
```

Matome, kad $p\text{-value} < 0.05$, vadinasi H_0 atmetam. Proporcijos statistiškai reikšmingai skiriasi. Patikrinkime tarp kurių miestų proporcijos yra statistiškai reikšmingai skirtingos.

Proporcijos statistiškai reikšmingai skiriasi tarp Šiaulių m. sav. ir Panevėžio m. sav. ir tarp Vilniaus m. sav. ir Panevėžio m. sav. Patikrinkime, ar Šiauliuose/Vilniuje įvykių dėl neblaivių vairuotojų kaltės yra statistiškai reikšmingai daugiau negu Panevėžyje.

Paaiškėjo, kad Šiauliuose/Vilniuje įvykių dėl neblaivių vairuotojų kaltės yra statistiškai reikšmingai daugiau negu Panevėžyje.

Tikrinam įvykių priklausomybę nuo amžiaus

Forecastinam autoįvykius

Tikrinsime hipoteze ar Vilniaus mieste eismo įvykių kaltininkai dažniau yra Vyrų nei moterų.

$H_0 = V \text{ sk. } =$

Kaltininkų procentas miestuose (Vyrų/Moterų)

Kaltininkų procentas Vilniaus mieste.

Kaltininkų procentas Kauno mieste.

Kaltininkų procentas Šiaulių mieste.

Kaltininkų procentas Klaipėdos mieste.

Kaltininkų procentas Panevėžio mieste.

Tikrinsime hipoteze ar Vilniaus mieste eismo įvykių kaltininkai dažniau yra Vyrų nei moterų.

koreliacijos

Palyginti zuvusiu/sužeistu menesinius vidurkius tarp laikotarpiu ir vietu. Ar skiriasi įvykių skaičius nakti ir diena?

Literatūra

- <https://www.epolicija.lt/atviri-duomenys>
- <http://osp.stat.gov.lt>
- <http://www.lakd.lt>
- <http://www.lkpt.lt>