Regresinės analizės projektinis darbas

Vainius Gataveckas, Matas Gaulia, Dovydas Martinkus

### Duomenų apdorojimas

library(tidyverse)  
library(janitor)  
library(countrycode)  
  
# Natūralaus gyventojų prieaugio duomenys  
pop\_natural <- read\_csv("natural-population-growth.csv") %>%  
 filter(Year == 2017) %>%  
 dplyr::select(1, 4) %>%  
 set\_names(c("country", "natural\_growth")) %>%  
 mutate(country = countryname(country))  
  
# Bendras gyventojų prieaugis iš kurio bus gaunamas migracijos prieaugis  
pop\_total <- read\_csv("population-growth-rates.csv") %>%  
 filter(Year == 2017) %>%  
 dplyr::select(1, 4) %>%  
 set\_names(c("country", "total\_growth")) %>%  
 mutate(country = countryname(country))  
  
  
# UNData duomenys  
country\_stats <- read\_csv("country\_profile\_variables.csv") %>%  
 clean\_names() %>%  
 dplyr::select(-c(2, 3, 4, 5, 6, 7)) %>%  
 mutate(country = countryname(country))  
  
  
# World Happiness Report duomenys  
happiness <- read\_csv("2017.csv") %>%  
 clean\_names() %>%  
 dplyr::select(-c(2), -starts\_with("whisker"), -c("dystopia\_residual", "happiness\_score", "family")) %>%  
 mutate(country = countryname(country))

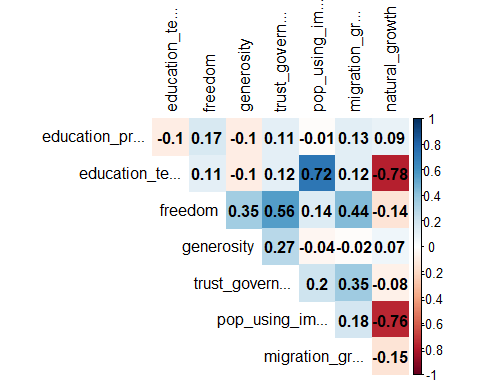
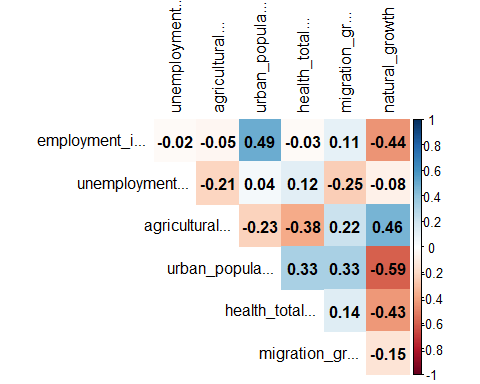
x <- reduce(list(pop\_natural, pop\_total, country\_stats, happiness), inner\_join, by = "country")  
  
  
# Išfiltruojami nenaudinti kintamieji  
x <- x %>%  
 dplyr::select(  
 -starts\_with("gdp"),  
 -starts\_with("labour"),  
 -starts\_with("international"),  
 -starts\_with("balance"),  
 -starts\_with("population"),  
 -starts\_with("fertility"),  
 -starts\_with("net"),  
 -starts\_with("energy\_prod"),  
 -starts\_with("forest"),  
 -starts\_with("threatened"),  
 -starts\_with("seats"),  
 -starts\_with("urban\_population\_growth"),  
 -starts\_with("refugees"),  
 -starts\_with("infant"),  
 -starts\_with("life\_expectancy"),  
 -starts\_with("co2"),  
 -starts\_with("economy"),  
 -starts\_with("education\_government"),  
 -starts\_with("energy"),  
 -health\_physicians\_per\_1000\_pop,  
 -individuals\_using\_the\_internet\_per\_100\_inhabitants,  
 -mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40,  
 -pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent  
 ) %>%  
 mutate(across(everything(), ~ replace(., . %in% c("...", "-99", ".../..."), NA))) %>%  
 mutate(across(starts\_with("education"), ~ str\_split(., "/") %>% map(~ mean(as.numeric(.)))))  
  
  
pop <- x$pop\_using\_improved\_drinking\_water\_urban\_rural\_percent  
f1 <- possibly(~ `[[`(.x, 1), 1)  
x$pop\_using\_improved\_drinking\_water\_urban <- pop %>%  
 str\_split("/") %>%  
 map(f1)  
f2 <- possibly(~ `[[`(.x, 2), 1)  
x$pop\_using\_improved\_drinking\_water\_rural <- pop %>%  
 str\_split("/") %>%  
 map(f2)  
  
  
x <- x %>%  
 dplyr::select(-pop\_using\_improved\_drinking\_water\_urban\_rural\_percent) %>%  
 mutate(across(-country, as.numeric)) %>%  
 mutate(migration\_growth = total\_growth - natural\_growth) %>%  
 drop\_na() %>%  
 dplyr::select(-total\_growth)

library(rsample)  
  
set.seed(123)  
  
# sudaramos kategorijos pagal tai ar migracijos/natūralus prieaugiai yra teigiami ar neigiami  
x <- x %>% mutate(  
 category = factor(case\_when(  
 migration\_growth >= 0 & natural\_growth >= 0 ~ 0, # "P migation, P natural",  
 migration\_growth >= 0 & natural\_growth < 0 ~ 1, # "P migation, N natural",  
 migration\_growth < 0 & natural\_growth >= 0 ~ 2, # "N migation, P natural",  
 TRUE ~ 3 # "N migration, N natural"  
 ))  
)  
  
  
# padalijimas į mokymo ir testavimo aibes  
train\_test\_split <- initial\_split(x, prop = 0.8)  
train <- training(train\_test\_split)  
test <- testing(train\_test\_split)  
  
country\_train <- train$country  
country\_test <- test$country  
  
train <- train %>% dplyr::select(-country)

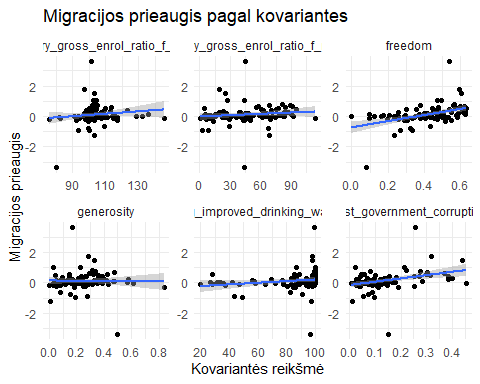
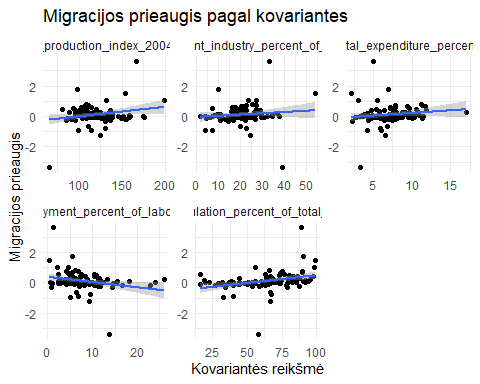
library(recipes)  
  
# iš anksto panaikinami kintamieji, kurie labai stipriai koreliuoja su kitais  
correlated\_recipe <- recipe(natural\_growth ~ ., data = train) %>%  
 add\_role(migration\_growth, new\_role = "outcome") %>%  
 add\_role(category, new\_role = "outcome") %>%  
 step\_corr(all\_numeric\_predictors(), threshold = 0.8) %>%  
 step\_nzv(all\_numeric\_predictors())  
  
  
correlated\_recipe <- prep(correlated\_recipe, training = train)  
  
train <- bake(correlated\_recipe, NULL)

### Regresijos modeliai

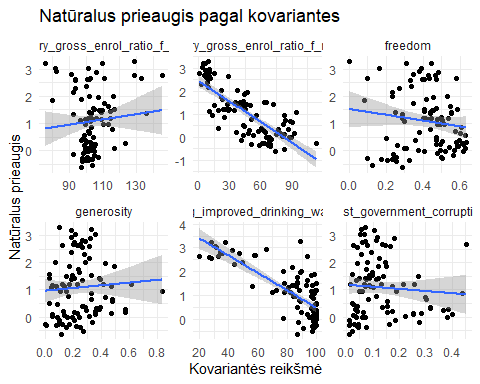
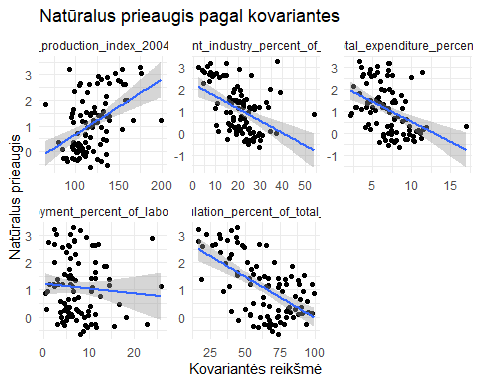
library(corrplot)  
  
# koreliacijų grafikai  
regression\_train <- train %>% dplyr::select(-category)  
  
correlation <- function(name, name2) {  
 correlation\_matrix <- regression\_train %>%  
 dplyr::select(1:5, {{ name }}, {{ name2 }}) %>%  
 set\_names(., str\_trunc(names(.), 15)) %>%  
 cor()  
  
 corrplot(correlation\_matrix, order = "original", method = "color", type = "upper", diag = FALSE, tl.col = "black", addCoef.col = "black")  
  
  
  
 correlation\_matrix <- regression\_train %>%  
 dplyr::select(6:length(regression\_train), {{ name }}, {{ name2 }}) %>%  
 set\_names(., str\_trunc(names(.), 15)) %>%  
 cor()  
  
 corrplot(correlation\_matrix, order = "original", method = "color", type = "upper", diag = FALSE, tl.col = "black", addCoef.col = "black")  
}  
  
  
correlation(migration\_growth, natural\_growth)



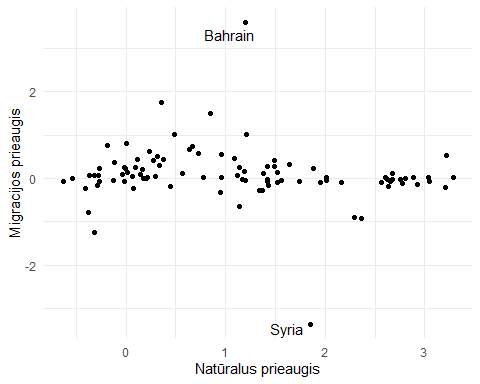
# sklaidos diagramos su kiekviena kovariante  
scatterplot <- function(name, name2, main, ylab) {  
 a <- regression\_train %>%  
 dplyr::select(1:5, {{ name }}, -{{ name2 }}) %>%  
 pivot\_longer(-{{ name }}) %>%  
 ggplot(aes(x = value, y = {{ name }})) +  
 facet\_wrap(vars(name), scales = "free") +  
 geom\_point() +  
 geom\_smooth(method = "lm") +  
 theme\_minimal() +  
 labs(title = main) + xlab("Kovariantės reikšmė") + ylab(ylab)  
  
  
 b <- regression\_train %>%  
 dplyr::select(6:length(regression\_train), {{ name }}, -{{ name2 }}) %>%  
 pivot\_longer(-{{ name }}) %>%  
 ggplot(aes(x = value, y = {{ name }})) +  
 facet\_wrap(vars(name), scales = "free") +  
 geom\_point() +  
 geom\_smooth(method = "lm") +  
 theme\_minimal() +  
 labs(title = main) + xlab("Kovariantės reikšmė") + ylab(ylab)  
  
 plot(a)  
  
 plot(b)  
}  
  
  
scatterplot(migration\_growth, natural\_growth, "Migracijos prieaugis pagal kovariantes","Migracijos prieaugis")



scatterplot(natural\_growth, migration\_growth, "Natūralus prieaugis pagal kovariantes","Natūralus prieaugis")



library(ggrepel)  
# migracijos ir natūralaus prieaugio sklaidos grafikas  
ggplot(regression\_train, aes(natural\_growth, migration\_growth)) +  
 geom\_point() +  
 theme\_minimal() +  
 xlab("Natūralus prieaugis") +  
 ylab("Migracijos prieaugis") + labs(main="Migracijos ir natūralus prieaugis") +  
 geom\_text\_repel(data=(regression\_train %>% cbind(country\_train))[abs(regression\_train$migration\_growth)>2,]  
 ,aes(label=country\_train))



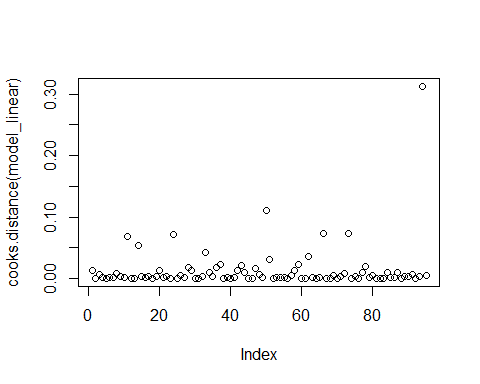
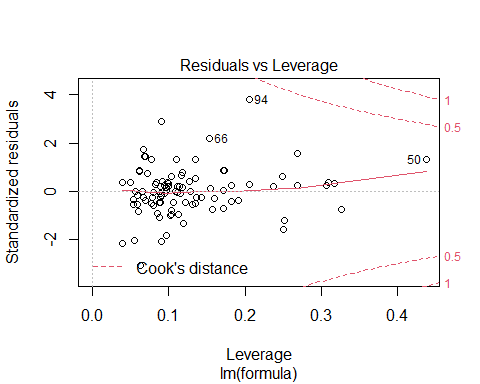
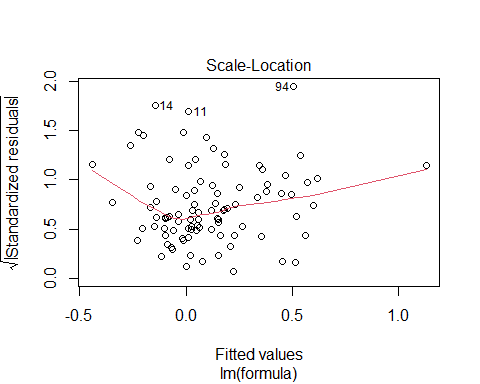
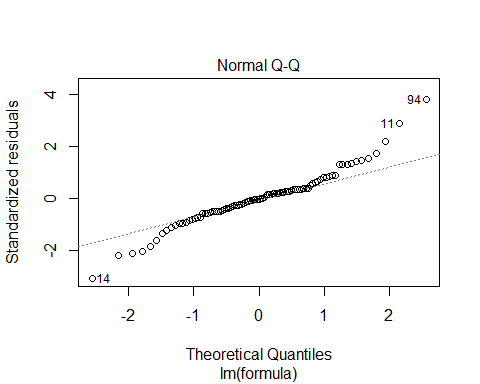
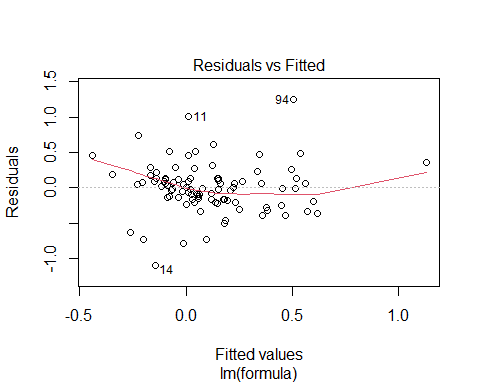
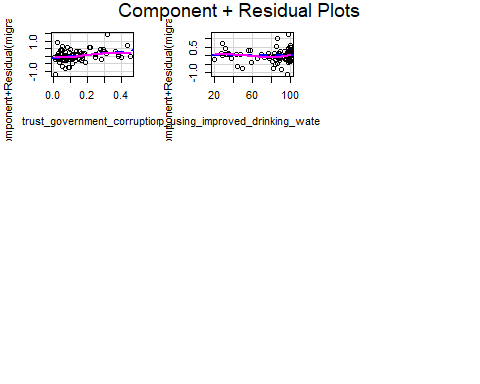
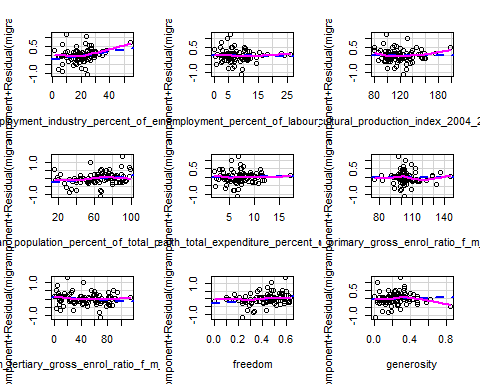
# matomos dvi labai stiprios išskirtys  
outlier\_indices <- regression\_train$migration\_growth %>%  
 abs() %>%  
 order(decreasing = TRUE) %>%  
 `[`(1:2)

library(car)  
library(effects)  
library(lm.beta)  
library(broom)  
  
  
# sudaromas paprastos regresijos modelis, atliekama pažingsninė regresija  
linear\_fit <- function(formula) {  
 model\_linear <- lm(formula, data = data)  
  
 # diagnostiniai grafikai  
 crPlots(model\_linear)  
 plot(model\_linear)  
 plot(cooks.distance(model\_linear))  
  
 # pažingsninė regresija  
 model\_linear\_small <- MASS::stepAIC(model\_linear, direction = "both", trace = 0)  
  
 # ar yra statistiškai reikšmingas skirtumas  
 print(anova(model\_linear, model\_linear\_small))  
  
 # kovariančių efektų grafikas  
 plot(predictorEffects(model\_linear\_small))  
 print(summary(model\_linear\_small))  
  
  
 stand <- lm.beta(model\_linear\_small)  
 # standartizuotų koeficientų grafikas  
 coeff\_plot <- tidy(stand) %>%  
 filter(term != "(Intercept)") %>%  
 ggplot(aes(term, estimate)) +  
 geom\_pointrange(aes(ymin = estimate - std.error, ymax = estimate + std.error), color = "blue") +  
 scale\_x\_discrete() +  
 coord\_flip() +  
 theme\_minimal() +  
 labs(x = "Kovariantė", y = "Standartizuotos koeficientų reikšmės")  
  
 plot(coeff\_plot)  
  
 model\_linear\_small  
}

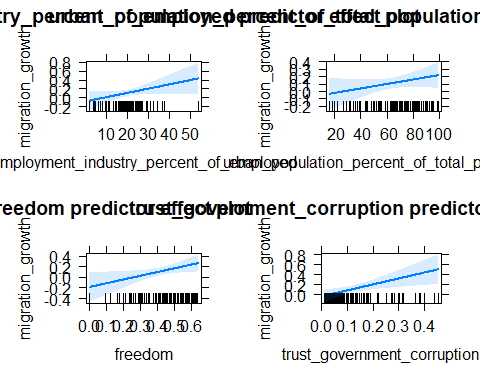
# Atskirai apmokomi modeliai migracijos ir natūraliam prieaugiui  
print("Tiesinės regresijos modelis migracijos prieaugiui")

## [1] "Tiesinės regresijos modelis migracijos prieaugiui"

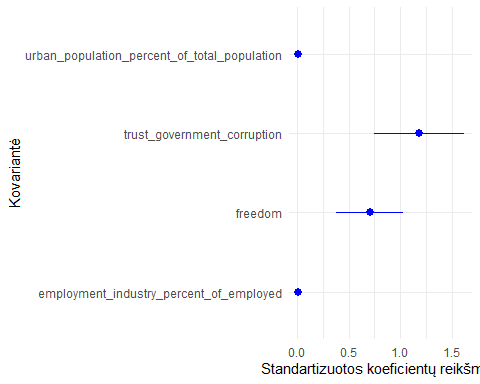
data <- regression\_train %>%  
 dplyr::select(-natural\_growth) %>%  
 slice(-outlier\_indices)  
model\_linear\_migration <- linear\_fit(migration\_growth ~ .)



## Analysis of Variance Table  
##   
## Model 1: migration\_growth ~ employment\_industry\_percent\_of\_employed +   
## unemployment\_percent\_of\_labour\_force + agricultural\_production\_index\_2004\_2006\_100 +   
## urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## freedom + generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban  
## Model 2: migration\_growth ~ employment\_industry\_percent\_of\_employed +   
## urban\_population\_percent\_of\_total\_population + freedom +   
## trust\_government\_corruption  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 83 11.211   
## 2 90 11.356 -7 -0.1452 0.1536 0.9931



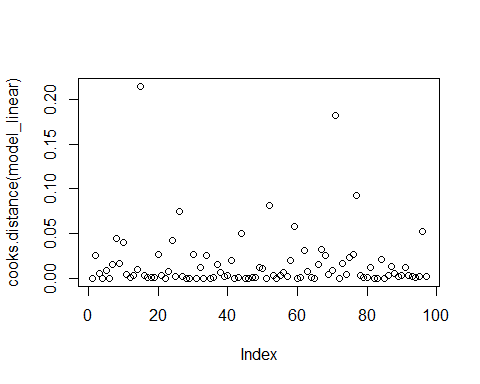
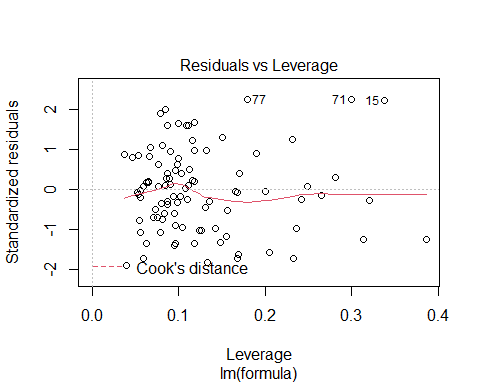
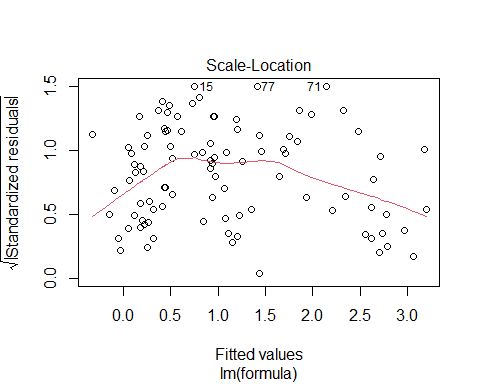
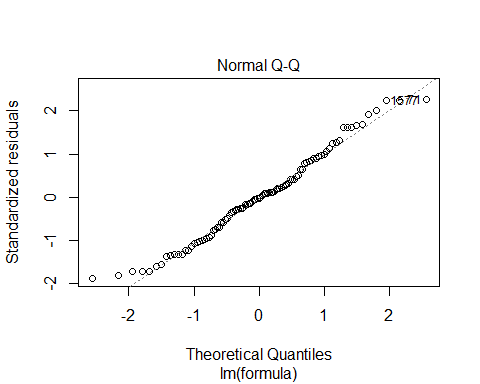
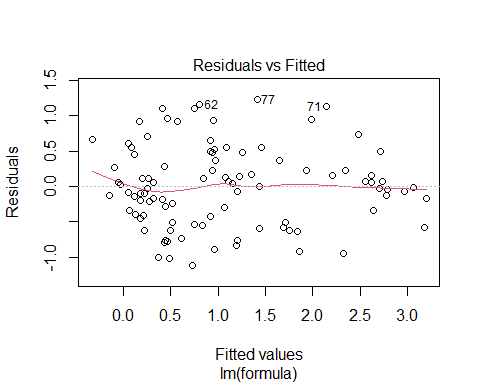
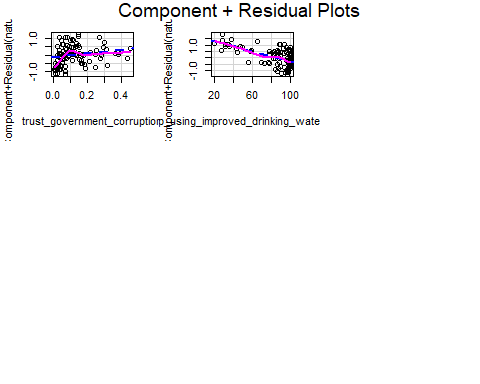
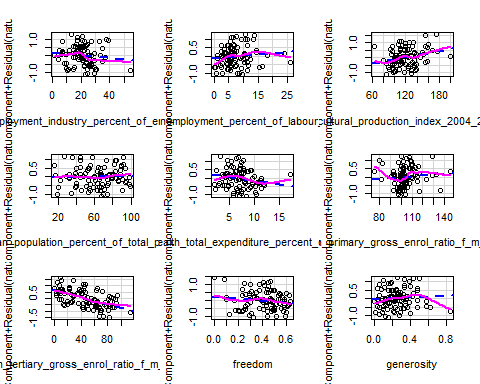
##   
## Call:  
## lm(formula = migration\_growth ~ employment\_industry\_percent\_of\_employed +   
## urban\_population\_percent\_of\_total\_population + freedom +   
## trust\_government\_corruption, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.13140 -0.17003 0.01515 0.13945 1.30339   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) -0.720036 0.160058 -4.499  
## employment\_industry\_percent\_of\_employed 0.009551 0.005220 1.830  
## urban\_population\_percent\_of\_total\_population 0.003024 0.002161 1.399  
## freedom 0.703689 0.323155 2.178  
## trust\_government\_corruption 1.184538 0.434899 2.724  
## Pr(>|t|)   
## (Intercept) 2.04e-05 \*\*\*  
## employment\_industry\_percent\_of\_employed 0.07062 .   
## urban\_population\_percent\_of\_total\_population 0.16517   
## freedom 0.03205 \*   
## trust\_government\_corruption 0.00775 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3552 on 90 degrees of freedom  
## Multiple R-squared: 0.3368, Adjusted R-squared: 0.3074   
## F-statistic: 11.43 on 4 and 90 DF, p-value: 1.518e-07



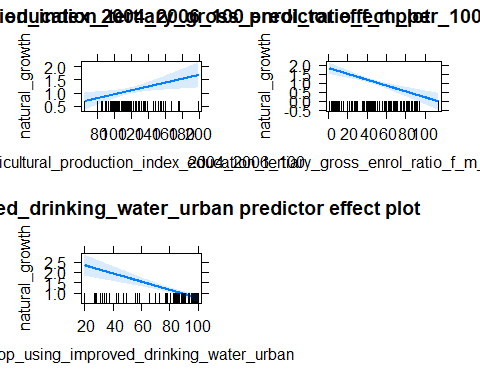
print("Tiesinės regresijos modelis natūraliam prieaugiui")

## [1] "Tiesinės regresijos modelis natūraliam prieaugiui"

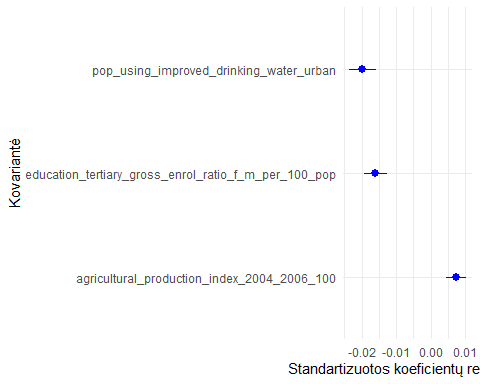
data <- regression\_train %>% dplyr::select(-migration\_growth)  
model\_linear\_natural <- linear\_fit(natural\_growth ~ .)



## Analysis of Variance Table  
##   
## Model 1: natural\_growth ~ employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force +   
## agricultural\_production\_index\_2004\_2006\_100 + urban\_population\_percent\_of\_total\_population +   
## health\_total\_expenditure\_percent\_of\_gdp + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + freedom +   
## generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban  
## Model 2: natural\_growth ~ agricultural\_production\_index\_2004\_2006\_100 +   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + pop\_using\_improved\_drinking\_water\_urban  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 85 30.826   
## 2 93 33.392 -8 -2.5665 0.8846 0.533



##   
## Call:  
## lm(formula = natural\_growth ~ agricultural\_production\_index\_2004\_2006\_100 +   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + pop\_using\_improved\_drinking\_water\_urban,   
## data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.17333 -0.52663 0.02113 0.33529 1.40262   
##   
## Coefficients:  
## Estimate Std. Error  
## (Intercept) 2.579233 0.468106  
## agricultural\_production\_index\_2004\_2006\_100 0.007448 0.002911  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.016088 0.003293  
## pop\_using\_improved\_drinking\_water\_urban -0.019777 0.003939  
## t value Pr(>|t|)   
## (Intercept) 5.510 3.18e-07 \*\*\*  
## agricultural\_production\_index\_2004\_2006\_100 2.559 0.0121 \*   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -4.885 4.29e-06 \*\*\*  
## pop\_using\_improved\_drinking\_water\_urban -5.021 2.47e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.5992 on 93 degrees of freedom  
## Multiple R-squared: 0.7062, Adjusted R-squared: 0.6968   
## F-statistic: 74.52 on 3 and 93 DF, p-value: < 2.2e-16



# Matoma, kad migracijos prieaugiui tiesiniu modeliu gaunami daug prastesni rezultatai negu  
 # natūraliam prieaugiui

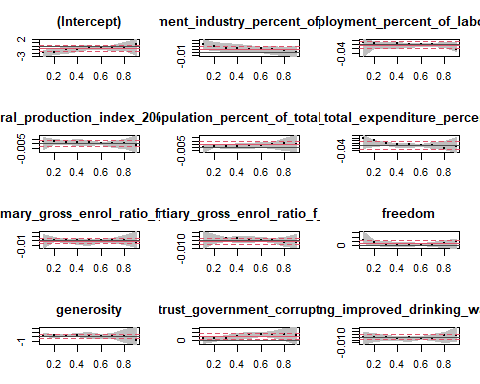
# Kvantilių regresija  
library(quantreg)  
  
quantile\_fit <- function() {  
 model\_quantile <- rq(formula, data = data, tau = tau)  
  
 print(summary(model\_quantile, se = "boot"))  
 plot(summary(model\_quantile))  
 print(anova(model\_quantile, test = "Wald", joint = FALSE))  
  
 model\_quantile  
}

print("Kvantilių regresija migracijos prieaugiui")

## [1] "Kvantilių regresija migracijos prieaugiui"

data <- regression\_train %>%  
 dplyr::select(-natural\_growth) %>%  
 slice(-outlier\_indices)  
tau <- seq(0.1, 0.9, 0.1)  
formula <- migration\_growth ~ .  
  
model\_quantile\_migration <- quantile\_fit()

##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.1  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) -1.92648 0.86451   
## employment\_industry\_percent\_of\_employed 0.02580 0.01321   
## unemployment\_percent\_of\_labour\_force 0.00798 0.01387   
## agricultural\_production\_index\_2004\_2006\_100 0.00187 0.00413   
## urban\_population\_percent\_of\_total\_population -0.00024 0.00471   
## health\_total\_expenditure\_percent\_of\_gdp 0.05893 0.03334   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00062 0.00664   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00165 0.00590   
## freedom 1.14056 0.84652   
## generosity 0.27141 0.46390   
## trust\_government\_corruption 0.61266 0.68705   
## pop\_using\_improved\_drinking\_water\_urban -0.00149 0.00603   
## t value Pr(>|t|)  
## (Intercept) -2.22841 0.02856  
## employment\_industry\_percent\_of\_employed 1.95290 0.05420  
## unemployment\_percent\_of\_labour\_force 0.57562 0.56643  
## agricultural\_production\_index\_2004\_2006\_100 0.45241 0.65216  
## urban\_population\_percent\_of\_total\_population -0.05026 0.96004  
## health\_total\_expenditure\_percent\_of\_gdp 1.76741 0.08084  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.09405 0.92530  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.28049 0.77980  
## freedom 1.34736 0.18153  
## generosity 0.58508 0.56008  
## trust\_government\_corruption 0.89173 0.37512  
## pop\_using\_improved\_drinking\_water\_urban -0.24659 0.80584  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.2  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) -1.67072 0.69183   
## employment\_industry\_percent\_of\_employed 0.01984 0.01037   
## unemployment\_percent\_of\_labour\_force 0.00332 0.01043   
## agricultural\_production\_index\_2004\_2006\_100 0.00257 0.00244   
## urban\_population\_percent\_of\_total\_population 0.00153 0.00271   
## health\_total\_expenditure\_percent\_of\_gdp 0.04143 0.01907   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00100 0.00382   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00087 0.00339   
## freedom 0.65556 0.46671   
## generosity 0.35233 0.30143   
## trust\_government\_corruption 0.50116 0.48716   
## pop\_using\_improved\_drinking\_water\_urban -0.00087 0.00383   
## t value Pr(>|t|)  
## (Intercept) -2.41492 0.01794  
## employment\_industry\_percent\_of\_employed 1.91260 0.05925  
## unemployment\_percent\_of\_labour\_force 0.31797 0.75131  
## agricultural\_production\_index\_2004\_2006\_100 1.05429 0.29481  
## urban\_population\_percent\_of\_total\_population 0.56558 0.57320  
## health\_total\_expenditure\_percent\_of\_gdp 2.17225 0.03269  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.26261 0.79350  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.25549 0.79897  
## freedom 1.40463 0.16386  
## generosity 1.16885 0.24581  
## trust\_government\_corruption 1.02874 0.30659  
## pop\_using\_improved\_drinking\_water\_urban -0.22598 0.82177  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.3  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) -1.05588 0.67939   
## employment\_industry\_percent\_of\_employed 0.01177 0.00946   
## unemployment\_percent\_of\_labour\_force -0.00229 0.00921   
## agricultural\_production\_index\_2004\_2006\_100 0.00174 0.00268   
## urban\_population\_percent\_of\_total\_population 0.00090 0.00266   
## health\_total\_expenditure\_percent\_of\_gdp 0.01587 0.01574   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00186 0.00317   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00212 0.00251   
## freedom 0.21495 0.38890   
## generosity 0.30400 0.28415   
## trust\_government\_corruption 0.53938 0.57532   
## pop\_using\_improved\_drinking\_water\_urban -0.00135 0.00270   
## t value Pr(>|t|)  
## (Intercept) -1.55417 0.12395  
## employment\_industry\_percent\_of\_employed 1.24452 0.21681  
## unemployment\_percent\_of\_labour\_force -0.24877 0.80415  
## agricultural\_production\_index\_2004\_2006\_100 0.64880 0.51826  
## urban\_population\_percent\_of\_total\_population 0.34025 0.73453  
## health\_total\_expenditure\_percent\_of\_gdp 1.00771 0.31652  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.58733 0.55858  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.84494 0.40057  
## freedom 0.55272 0.58194  
## generosity 1.06983 0.28780  
## trust\_government\_corruption 0.93754 0.35120  
## pop\_using\_improved\_drinking\_water\_urban -0.50001 0.61839  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.4  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) -0.78905 0.62453   
## employment\_industry\_percent\_of\_employed 0.01353 0.00834   
## unemployment\_percent\_of\_labour\_force 0.00385 0.00802   
## agricultural\_production\_index\_2004\_2006\_100 0.00171 0.00231   
## urban\_population\_percent\_of\_total\_population 0.00090 0.00278   
## health\_total\_expenditure\_percent\_of\_gdp 0.01508 0.01501   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00002 0.00366   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00193 0.00231   
## freedom 0.18594 0.34368   
## generosity 0.36225 0.31018   
## trust\_government\_corruption 1.05988 0.56770   
## pop\_using\_improved\_drinking\_water\_urban -0.00283 0.00266   
## t value Pr(>|t|)  
## (Intercept) -1.26343 0.20997  
## employment\_industry\_percent\_of\_employed 1.62243 0.10850  
## unemployment\_percent\_of\_labour\_force 0.47925 0.63302  
## agricultural\_production\_index\_2004\_2006\_100 0.73741 0.46295  
## urban\_population\_percent\_of\_total\_population 0.32584 0.74536  
## health\_total\_expenditure\_percent\_of\_gdp 1.00469 0.31796  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00521 0.99586  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.83347 0.40697  
## freedom 0.54102 0.58994  
## generosity 1.16789 0.24619  
## trust\_government\_corruption 1.86699 0.06543  
## pop\_using\_improved\_drinking\_water\_urban -1.06377 0.29052  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.5  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) -0.70883 0.62031   
## employment\_industry\_percent\_of\_employed 0.00868 0.00830   
## unemployment\_percent\_of\_labour\_force -0.00092 0.00800   
## agricultural\_production\_index\_2004\_2006\_100 0.00160 0.00265   
## urban\_population\_percent\_of\_total\_population 0.00243 0.00289   
## health\_total\_expenditure\_percent\_of\_gdp 0.01011 0.01460   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00033 0.00419   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00091 0.00259   
## freedom 0.22736 0.35053   
## generosity 0.15808 0.35722   
## trust\_government\_corruption 1.44830 0.58068   
## pop\_using\_improved\_drinking\_water\_urban -0.00114 0.00282   
## t value Pr(>|t|)  
## (Intercept) -1.14270 0.25645  
## employment\_industry\_percent\_of\_employed 1.04623 0.29849  
## unemployment\_percent\_of\_labour\_force -0.11501 0.90872  
## agricultural\_production\_index\_2004\_2006\_100 0.60567 0.54639  
## urban\_population\_percent\_of\_total\_population 0.83999 0.40333  
## health\_total\_expenditure\_percent\_of\_gdp 0.69240 0.49062  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.07900 0.93722  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.34953 0.72757  
## freedom 0.64862 0.51838  
## generosity 0.44253 0.65925  
## trust\_government\_corruption 2.49414 0.01461  
## pop\_using\_improved\_drinking\_water\_urban -0.40334 0.68774  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.6  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) -0.57342 0.61038   
## employment\_industry\_percent\_of\_employed 0.01109 0.00846   
## unemployment\_percent\_of\_labour\_force -0.00069 0.00785   
## agricultural\_production\_index\_2004\_2006\_100 0.00052 0.00258   
## urban\_population\_percent\_of\_total\_population 0.00361 0.00368   
## health\_total\_expenditure\_percent\_of\_gdp 0.00595 0.01674   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00018 0.00398   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00018 0.00255   
## freedom 0.37166 0.35996   
## generosity 0.12338 0.43834   
## trust\_government\_corruption 1.48083 0.59655   
## pop\_using\_improved\_drinking\_water\_urban -0.00213 0.00304   
## t value Pr(>|t|)  
## (Intercept) -0.93945 0.35022  
## employment\_industry\_percent\_of\_employed 1.31017 0.19375  
## unemployment\_percent\_of\_labour\_force -0.08854 0.92966  
## agricultural\_production\_index\_2004\_2006\_100 0.20257 0.83997  
## urban\_population\_percent\_of\_total\_population 0.98168 0.32911  
## health\_total\_expenditure\_percent\_of\_gdp 0.35552 0.72310  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.04564 0.96370  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.06958 0.94469  
## freedom 1.03248 0.30485  
## generosity 0.28147 0.77905  
## trust\_government\_corruption 2.48234 0.01507  
## pop\_using\_improved\_drinking\_water\_urban -0.70163 0.48487  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.7  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) -0.55428 0.65403   
## employment\_industry\_percent\_of\_employed 0.00963 0.00949   
## unemployment\_percent\_of\_labour\_force 0.00224 0.00890   
## agricultural\_production\_index\_2004\_2006\_100 -0.00001 0.00293   
## urban\_population\_percent\_of\_total\_population 0.00429 0.00480   
## health\_total\_expenditure\_percent\_of\_gdp 0.00480 0.01834   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00052 0.00402   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00023 0.00332   
## freedom 0.25461 0.41303   
## generosity 0.07785 0.50986   
## trust\_government\_corruption 1.65563 0.74975   
## pop\_using\_improved\_drinking\_water\_urban -0.00195 0.00364   
## t value Pr(>|t|)  
## (Intercept) -0.84748 0.39917  
## employment\_industry\_percent\_of\_employed 1.01541 0.31286  
## unemployment\_percent\_of\_labour\_force 0.25143 0.80210  
## agricultural\_production\_index\_2004\_2006\_100 -0.00288 0.99771  
## urban\_population\_percent\_of\_total\_population 0.89337 0.37424  
## health\_total\_expenditure\_percent\_of\_gdp 0.26188 0.79406  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.12935 0.89740  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.07062 0.94387  
## freedom 0.61646 0.53928  
## generosity 0.15270 0.87901  
## trust\_government\_corruption 2.20825 0.02998  
## pop\_using\_improved\_drinking\_water\_urban -0.53485 0.59418  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.8  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) -0.68182 0.86952   
## employment\_industry\_percent\_of\_employed 0.00546 0.01085   
## unemployment\_percent\_of\_labour\_force -0.00259 0.01391   
## agricultural\_production\_index\_2004\_2006\_100 -0.00005 0.00362   
## urban\_population\_percent\_of\_total\_population 0.00592 0.00584   
## health\_total\_expenditure\_percent\_of\_gdp -0.01712 0.02741   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00073 0.00589   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00213 0.00434   
## freedom 0.61071 0.52841   
## generosity 0.12248 0.73599   
## trust\_government\_corruption 1.47967 0.92107   
## pop\_using\_improved\_drinking\_water\_urban 0.00258 0.00501   
## t value Pr(>|t|)  
## (Intercept) -0.78413 0.43520  
## employment\_industry\_percent\_of\_employed 0.50357 0.61590  
## unemployment\_percent\_of\_labour\_force -0.18606 0.85285  
## agricultural\_production\_index\_2004\_2006\_100 -0.01388 0.98896  
## urban\_population\_percent\_of\_total\_population 1.01394 0.31356  
## health\_total\_expenditure\_percent\_of\_gdp -0.62440 0.53408  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.12467 0.90109  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.49109 0.62466  
## freedom 1.15574 0.25110  
## generosity 0.16642 0.86823  
## trust\_government\_corruption 1.60647 0.11197  
## pop\_using\_improved\_drinking\_water\_urban 0.51529 0.60772  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.9  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 0.68367 1.08774   
## employment\_industry\_percent\_of\_employed 0.00037 0.01348   
## unemployment\_percent\_of\_labour\_force -0.02508 0.01941   
## agricultural\_production\_index\_2004\_2006\_100 -0.00229 0.00433   
## urban\_population\_percent\_of\_total\_population 0.00528 0.00728   
## health\_total\_expenditure\_percent\_of\_gdp -0.00231 0.03505   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00417 0.00821   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00289 0.00458   
## freedom 1.04742 0.74337   
## generosity -0.84280 0.90656   
## trust\_government\_corruption 1.21270 1.27195   
## pop\_using\_improved\_drinking\_water\_urban 0.00246 0.00669   
## t value Pr(>|t|)  
## (Intercept) 0.62852 0.53139  
## employment\_industry\_percent\_of\_employed 0.02754 0.97810  
## unemployment\_percent\_of\_labour\_force -1.29218 0.19988  
## agricultural\_production\_index\_2004\_2006\_100 -0.52764 0.59916  
## urban\_population\_percent\_of\_total\_population 0.72559 0.47013  
## health\_total\_expenditure\_percent\_of\_gdp -0.06590 0.94762  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.50830 0.61259  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.62930 0.53088  
## freedom 1.40902 0.16256  
## generosity -0.92967 0.35524  
## trust\_government\_corruption 0.95342 0.34315  
## pop\_using\_improved\_drinking\_water\_urban 0.36836 0.71354



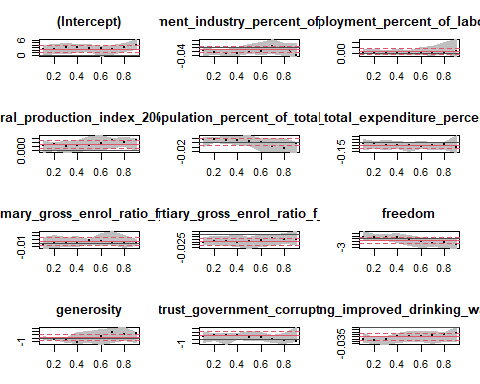
## Quantile Regression Analysis of Deviance Table  
##   
## Model: migration\_growth ~ employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force + agricultural\_production\_index\_2004\_2006\_100 + urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + freedom + generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban  
## Tests of Equality of Distinct Slopes: tau in { 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 }  
##   
## Df Resid Df F value Pr(>F)  
## employment\_industry\_percent\_of\_employed 8 847 0.4387 0.8980  
## unemployment\_percent\_of\_labour\_force 8 847 0.8835 0.5298  
## agricultural\_production\_index\_2004\_2006\_100 8 847 0.3063 0.9638  
## urban\_population\_percent\_of\_total\_population 8 847 0.2722 0.9749  
## health\_total\_expenditure\_percent\_of\_gdp 8 847 0.4797 0.8711  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 8 847 0.2504 0.9808  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 8 847 0.4294 0.9037  
## freedom 8 847 0.5251 0.8382  
## generosity 8 847 1.4714 0.1636  
## trust\_government\_corruption 8 847 0.7046 0.6877  
## pop\_using\_improved\_drinking\_water\_urban 8 847 0.4466 0.8931

print("Kvantilių regresija natūraliam prieaugiui")

## [1] "Kvantilių regresija natūraliam prieaugiui"

data <- regression\_train %>% dplyr::select(-migration\_growth)  
tau <- seq(0.1, 0.9, 0.1)  
formula <- natural\_growth ~ .  
  
model\_quantile\_migration <- quantile\_fit()

##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.1  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 2.50971 1.29872   
## employment\_industry\_percent\_of\_employed -0.01501 0.01689   
## unemployment\_percent\_of\_labour\_force -0.00088 0.02202   
## agricultural\_production\_index\_2004\_2006\_100 0.00645 0.00503   
## urban\_population\_percent\_of\_total\_population 0.00862 0.00653   
## health\_total\_expenditure\_percent\_of\_gdp -0.01162 0.04948   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00344 0.01059   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.01472 0.00763   
## freedom 0.36074 0.78272   
## generosity -0.08716 0.56891   
## trust\_government\_corruption 0.96367 0.99080   
## pop\_using\_improved\_drinking\_water\_urban -0.02608 0.00707   
## t value Pr(>|t|)  
## (Intercept) 1.93244 0.05664  
## employment\_industry\_percent\_of\_employed -0.88842 0.37682  
## unemployment\_percent\_of\_labour\_force -0.04016 0.96806  
## agricultural\_production\_index\_2004\_2006\_100 1.28112 0.20364  
## urban\_population\_percent\_of\_total\_population 1.31967 0.19049  
## health\_total\_expenditure\_percent\_of\_gdp -0.23485 0.81489  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.32493 0.74604  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -1.92997 0.05695  
## freedom 0.46088 0.64606  
## generosity -0.15321 0.87860  
## trust\_government\_corruption 0.97262 0.33350  
## pop\_using\_improved\_drinking\_water\_urban -3.68870 0.00040  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.2  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 2.49146 1.12678   
## employment\_industry\_percent\_of\_employed -0.02191 0.01515   
## unemployment\_percent\_of\_labour\_force 0.00818 0.01766   
## agricultural\_production\_index\_2004\_2006\_100 0.00668 0.00422   
## urban\_population\_percent\_of\_total\_population 0.00742 0.00541   
## health\_total\_expenditure\_percent\_of\_gdp -0.01192 0.04305   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00060 0.00914   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.01397 0.00601   
## freedom 0.44285 0.81340   
## generosity -0.29519 0.61286   
## trust\_government\_corruption 1.02961 0.92154   
## pop\_using\_improved\_drinking\_water\_urban -0.02668 0.00629   
## t value Pr(>|t|)  
## (Intercept) 2.21113 0.02971  
## employment\_industry\_percent\_of\_employed -1.44604 0.15184  
## unemployment\_percent\_of\_labour\_force 0.46318 0.64442  
## agricultural\_production\_index\_2004\_2006\_100 1.58464 0.11676  
## urban\_population\_percent\_of\_total\_population 1.37030 0.17420  
## health\_total\_expenditure\_percent\_of\_gdp -0.27696 0.78249  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.06531 0.94808  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -2.32402 0.02251  
## freedom 0.54445 0.58756  
## generosity -0.48167 0.63128  
## trust\_government\_corruption 1.11727 0.26703  
## pop\_using\_improved\_drinking\_water\_urban -4.24196 0.00006  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.3  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 3.18479 1.15878   
## employment\_industry\_percent\_of\_employed -0.02824 0.01566   
## unemployment\_percent\_of\_labour\_force 0.00925 0.01998   
## agricultural\_production\_index\_2004\_2006\_100 0.00427 0.00437   
## urban\_population\_percent\_of\_total\_population 0.00618 0.00597   
## health\_total\_expenditure\_percent\_of\_gdp -0.04808 0.04404   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00120 0.01006   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.01305 0.00643   
## freedom 0.41716 0.90677   
## generosity -0.50126 0.66770   
## trust\_government\_corruption 1.02811 0.90837   
## pop\_using\_improved\_drinking\_water\_urban -0.02631 0.00676   
## t value Pr(>|t|)  
## (Intercept) 2.74839 0.00731  
## employment\_industry\_percent\_of\_employed -1.80308 0.07492  
## unemployment\_percent\_of\_labour\_force 0.46285 0.64465  
## agricultural\_production\_index\_2004\_2006\_100 0.97719 0.33125  
## urban\_population\_percent\_of\_total\_population 1.03570 0.30328  
## health\_total\_expenditure\_percent\_of\_gdp -1.09181 0.27800  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.11921 0.90539  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -2.02982 0.04550  
## freedom 0.46005 0.64666  
## generosity -0.75073 0.45489  
## trust\_government\_corruption 1.13182 0.26089  
## pop\_using\_improved\_drinking\_water\_urban -3.89476 0.00020  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.4  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 3.22425 1.26827   
## employment\_industry\_percent\_of\_employed -0.03079 0.01845   
## unemployment\_percent\_of\_labour\_force 0.00781 0.02044   
## agricultural\_production\_index\_2004\_2006\_100 0.00564 0.00530   
## urban\_population\_percent\_of\_total\_population 0.00613 0.00643   
## health\_total\_expenditure\_percent\_of\_gdp -0.05890 0.04501   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00084 0.01053   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.01623 0.00623   
## freedom 0.49441 1.07752   
## generosity -1.01588 0.87830   
## trust\_government\_corruption 1.03362 0.93994   
## pop\_using\_improved\_drinking\_water\_urban -0.02019 0.00816   
## t value Pr(>|t|)  
## (Intercept) 2.54225 0.01283  
## employment\_industry\_percent\_of\_employed -1.66902 0.09879  
## unemployment\_percent\_of\_labour\_force 0.38212 0.70333  
## agricultural\_production\_index\_2004\_2006\_100 1.06435 0.29018  
## urban\_population\_percent\_of\_total\_population 0.95404 0.34277  
## health\_total\_expenditure\_percent\_of\_gdp -1.30846 0.19424  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.07942 0.93689  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -2.60435 0.01086  
## freedom 0.45884 0.64752  
## generosity -1.15663 0.25066  
## trust\_government\_corruption 1.09967 0.27458  
## pop\_using\_improved\_drinking\_water\_urban -2.47497 0.01531  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.5  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 2.64220 1.27213   
## employment\_industry\_percent\_of\_employed -0.02361 0.01944   
## unemployment\_percent\_of\_labour\_force 0.00352 0.02022   
## agricultural\_production\_index\_2004\_2006\_100 0.00596 0.00547   
## urban\_population\_percent\_of\_total\_population 0.00228 0.00764   
## health\_total\_expenditure\_percent\_of\_gdp -0.05480 0.05369   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00591 0.01174   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.01463 0.00662   
## freedom -0.27038 1.09466   
## generosity 0.08732 1.10537   
## trust\_government\_corruption 0.54026 0.95658   
## pop\_using\_improved\_drinking\_water\_urban -0.01822 0.00865   
## t value Pr(>|t|)  
## (Intercept) 2.07700 0.04082  
## employment\_industry\_percent\_of\_employed -1.21476 0.22782  
## unemployment\_percent\_of\_labour\_force 0.17418 0.86214  
## agricultural\_production\_index\_2004\_2006\_100 1.09062 0.27852  
## urban\_population\_percent\_of\_total\_population 0.29875 0.76586  
## health\_total\_expenditure\_percent\_of\_gdp -1.02079 0.31025  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.50337 0.61600  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -2.21061 0.02975  
## freedom -0.24700 0.80551  
## generosity 0.07900 0.93722  
## trust\_government\_corruption 0.56478 0.57371  
## pop\_using\_improved\_drinking\_water\_urban -2.10564 0.03819  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.6  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 2.67065 1.34267   
## employment\_industry\_percent\_of\_employed -0.01716 0.02279   
## unemployment\_percent\_of\_labour\_force -0.00052 0.02420   
## agricultural\_production\_index\_2004\_2006\_100 0.00910 0.00607   
## urban\_population\_percent\_of\_total\_population 0.00075 0.00914   
## health\_total\_expenditure\_percent\_of\_gdp -0.06036 0.05025   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00418 0.01081   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.01143 0.00636   
## freedom -0.80734 1.08215   
## generosity 0.55063 1.26643   
## trust\_government\_corruption 0.64507 1.18465   
## pop\_using\_improved\_drinking\_water\_urban -0.01932 0.00922   
## t value Pr(>|t|)  
## (Intercept) 1.98906 0.04991  
## employment\_industry\_percent\_of\_employed -0.75294 0.45357  
## unemployment\_percent\_of\_labour\_force -0.02151 0.98289  
## agricultural\_production\_index\_2004\_2006\_100 1.49968 0.13740  
## urban\_population\_percent\_of\_total\_population 0.08228 0.93462  
## health\_total\_expenditure\_percent\_of\_gdp -1.20126 0.23299  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.38656 0.70005  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -1.79730 0.07584  
## freedom -0.74606 0.45769  
## generosity 0.43479 0.66482  
## trust\_government\_corruption 0.54452 0.58751  
## pop\_using\_improved\_drinking\_water\_urban -2.09568 0.03908  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.7  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 2.49104 1.32608   
## employment\_industry\_percent\_of\_employed 0.00521 0.01985   
## unemployment\_percent\_of\_labour\_force 0.00872 0.02144   
## agricultural\_production\_index\_2004\_2006\_100 0.01234 0.00511   
## urban\_population\_percent\_of\_total\_population -0.00763 0.00929   
## health\_total\_expenditure\_percent\_of\_gdp -0.05014 0.04859   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00250 0.01258   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00892 0.00702   
## freedom -1.53524 0.91369   
## generosity 1.65925 1.17447   
## trust\_government\_corruption 0.04818 1.12163   
## pop\_using\_improved\_drinking\_water\_urban -0.01733 0.00740   
## t value Pr(>|t|)  
## (Intercept) 1.87849 0.06374  
## employment\_industry\_percent\_of\_employed 0.26243 0.79363  
## unemployment\_percent\_of\_labour\_force 0.40676 0.68521  
## agricultural\_production\_index\_2004\_2006\_100 2.41565 0.01785  
## urban\_population\_percent\_of\_total\_population -0.82108 0.41390  
## health\_total\_expenditure\_percent\_of\_gdp -1.03185 0.30507  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.19871 0.84296  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -1.27109 0.20717  
## freedom -1.68026 0.09658  
## generosity 1.41276 0.16138  
## trust\_government\_corruption 0.04296 0.96584  
## pop\_using\_improved\_drinking\_water\_urban -2.34208 0.02151  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.8  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 2.95674 1.32939   
## employment\_industry\_percent\_of\_employed 0.00432 0.02293   
## unemployment\_percent\_of\_labour\_force 0.00196 0.02798   
## agricultural\_production\_index\_2004\_2006\_100 0.00964 0.00569   
## urban\_population\_percent\_of\_total\_population -0.01207 0.00940   
## health\_total\_expenditure\_percent\_of\_gdp -0.03853 0.04307   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00273 0.01013   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.00947 0.00655   
## freedom -1.20774 0.88318   
## generosity 1.17463 1.13971   
## trust\_government\_corruption -0.16246 1.12714   
## pop\_using\_improved\_drinking\_water\_urban -0.01415 0.00714   
## t value Pr(>|t|)  
## (Intercept) 2.22413 0.02879  
## employment\_industry\_percent\_of\_employed 0.18861 0.85085  
## unemployment\_percent\_of\_labour\_force 0.07021 0.94419  
## agricultural\_production\_index\_2004\_2006\_100 1.69388 0.09395  
## urban\_population\_percent\_of\_total\_population -1.28382 0.20269  
## health\_total\_expenditure\_percent\_of\_gdp -0.89460 0.37353  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.26939 0.78829  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -1.44487 0.15217  
## freedom -1.36750 0.17507  
## generosity 1.03064 0.30563  
## trust\_government\_corruption -0.14413 0.88574  
## pop\_using\_improved\_drinking\_water\_urban -1.98202 0.05071  
##   
## Call: rq(formula = formula, tau = tau, data = data)  
##   
## tau: [1] 0.9  
##   
## Coefficients:  
## Value Std. Error  
## (Intercept) 3.89000 1.55280   
## employment\_industry\_percent\_of\_employed -0.03758 0.02328   
## unemployment\_percent\_of\_labour\_force 0.01485 0.03086   
## agricultural\_production\_index\_2004\_2006\_100 0.01209 0.00617   
## urban\_population\_percent\_of\_total\_population -0.00221 0.00831   
## health\_total\_expenditure\_percent\_of\_gdp -0.08659 0.05139   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.00013 0.00836   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.01034 0.00667   
## freedom -2.13962 1.13958   
## generosity 1.38785 0.95999   
## trust\_government\_corruption -0.46356 1.17877   
## pop\_using\_improved\_drinking\_water\_urban -0.01180 0.00850   
## t value Pr(>|t|)  
## (Intercept) 2.50515 0.01415  
## employment\_industry\_percent\_of\_employed -1.61443 0.11014  
## unemployment\_percent\_of\_labour\_force 0.48113 0.63166  
## agricultural\_production\_index\_2004\_2006\_100 1.95935 0.05335  
## urban\_population\_percent\_of\_total\_population -0.26601 0.79088  
## health\_total\_expenditure\_percent\_of\_gdp -1.68501 0.09566  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.01504 0.98804  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -1.55070 0.12469  
## freedom -1.87755 0.06387  
## generosity 1.44570 0.15194  
## trust\_government\_corruption -0.39326 0.69511  
## pop\_using\_improved\_drinking\_water\_urban -1.38755 0.16890



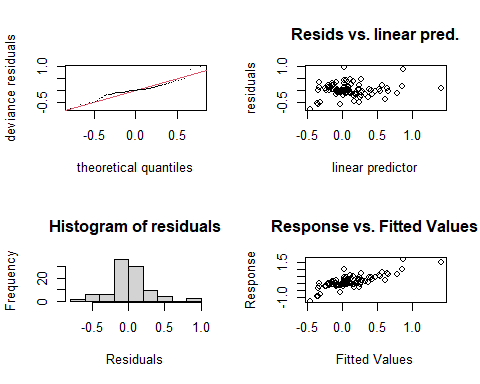
## Quantile Regression Analysis of Deviance Table  
##   
## Model: natural\_growth ~ employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force + agricultural\_production\_index\_2004\_2006\_100 + urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + freedom + generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban  
## Tests of Equality of Distinct Slopes: tau in { 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 }  
##   
## Df Resid Df F value  
## employment\_industry\_percent\_of\_employed 8 865 1.7082  
## unemployment\_percent\_of\_labour\_force 8 865 0.3304  
## agricultural\_production\_index\_2004\_2006\_100 8 865 0.9534  
## urban\_population\_percent\_of\_total\_population 8 865 1.7414  
## health\_total\_expenditure\_percent\_of\_gdp 8 865 0.7544  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 8 865 0.2721  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 8 865 0.5841  
## freedom 8 865 1.7266  
## generosity 8 865 2.7101  
## trust\_government\_corruption 8 865 0.6416  
## pop\_using\_improved\_drinking\_water\_urban 8 865 0.7526  
## Pr(>F)   
## employment\_industry\_percent\_of\_employed 0.092640 .   
## unemployment\_percent\_of\_labour\_force 0.954457   
## agricultural\_production\_index\_2004\_2006\_100 0.471410   
## urban\_population\_percent\_of\_total\_population 0.085268 .   
## health\_total\_expenditure\_percent\_of\_gdp 0.643271   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.974933   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.791577   
## freedom 0.088479 .   
## generosity 0.005959 \*\*  
## trust\_government\_corruption 0.743013   
## pop\_using\_improved\_drinking\_water\_urban 0.644911   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

library(mgcv)  
library(gratia)  
  
# siekiant tiksliau prognozuoti reikšmes naudinga sudaryti apibendrintus adityvius modelius,  
# kuriais galima įtraukti netiesinius sąryšius tarp kovariančių ir atsako  
fit\_gam <- function(formula, data) {  
 model\_gam <- gam(formula, data = data, select = FALSE)  
 gam.check(model\_gam)  
 summary(model\_gam)  
 draw(model\_gam)  
 k.check(model\_gam)  
 model\_gam  
}

print("GAM migracijos prieaugiui")

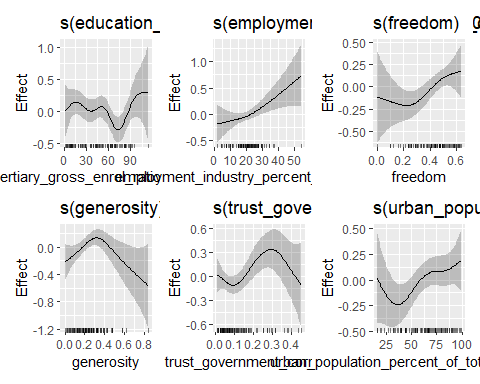
## [1] "GAM migracijos prieaugiui"

data <- regression\_train %>%  
 dplyr::select(-natural\_growth) %>%  
 slice(-outlier\_indices)  
  
# model\_gam\_migration <- fit\_gam(migration\_growth ~ s(employment\_industry\_percent\_of\_employed) +  
# s(unemployment\_percent\_of\_labour\_force) +  
# s(agricultural\_production\_index\_2004\_2006\_100) +  
# s(urban\_population\_percent\_of\_total\_population) +  
# s(health\_total\_expenditure\_percent\_of\_gdp) +  
# s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +  
# s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +  
# s(freedom) +  
# s(generosity) +  
# s(trust\_government\_corruption), data)  
#  
#  
# draw(model\_gam\_migration)  
# k.check(model\_gam\_migration)  
# summary(model\_gam\_migration)  
  
  
  
model\_gam\_migration <- fit\_gam(migration\_growth ~ s(employment\_industry\_percent\_of\_employed) +  
 unemployment\_percent\_of\_labour\_force +  
 agricultural\_production\_index\_2004\_2006\_100 +  
 s(urban\_population\_percent\_of\_total\_population) +  
 health\_total\_expenditure\_percent\_of\_gdp +  
 education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +  
 s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +  
 s(freedom) +  
 s(generosity) +  
 s(trust\_government\_corruption), data)



##   
## Method: GCV Optimizer: magic  
## Smoothing parameter selection converged after 7 iterations.  
## The RMS GCV score gradient at convergence was 6.213085e-07 .  
## The Hessian was positive definite.  
## Model rank = 59 / 59   
##   
## Basis dimension (k) checking results. Low p-value (k-index<1) may  
## indicate that k is too low, especially if edf is close to k'.  
##   
## k' edf k-index  
## s(employment\_industry\_percent\_of\_employed) 9.00 1.82 0.99  
## s(urban\_population\_percent\_of\_total\_population) 9.00 3.62 1.00  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 9.00 6.24 0.97  
## s(freedom) 9.00 2.55 0.84  
## s(generosity) 9.00 3.20 1.03  
## s(trust\_government\_corruption) 9.00 3.57 1.02  
## p-value   
## s(employment\_industry\_percent\_of\_employed) 0.370   
## s(urban\_population\_percent\_of\_total\_population) 0.525   
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 0.385   
## s(freedom) 0.055 .  
## s(generosity) 0.640   
## s(trust\_government\_corruption) 0.560   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

draw(model\_gam\_migration)



k.check(model\_gam\_migration)

## k' edf k-index  
## s(employment\_industry\_percent\_of\_employed) 9 1.816777 0.9868366  
## s(urban\_population\_percent\_of\_total\_population) 9 3.617499 0.9980236  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 9 6.243436 0.9668621  
## s(freedom) 9 2.549298 0.8395376  
## s(generosity) 9 3.197427 1.0346404  
## s(trust\_government\_corruption) 9 3.565093 1.0191396  
## p-value  
## s(employment\_industry\_percent\_of\_employed) 0.4075  
## s(urban\_population\_percent\_of\_total\_population) 0.4275  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 0.3300  
## s(freedom) 0.0600  
## s(generosity) 0.6125  
## s(trust\_government\_corruption) 0.5575

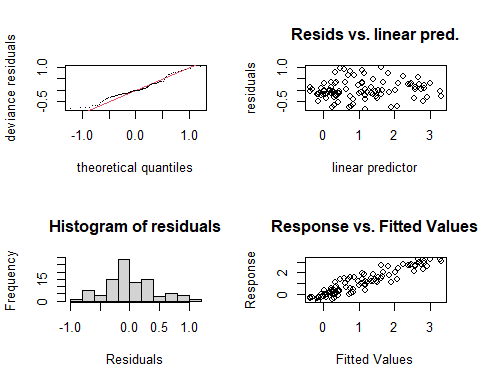
summary(model\_gam\_migration)

##   
## Family: gaussian   
## Link function: identity   
##   
## Formula:  
## migration\_growth ~ s(employment\_industry\_percent\_of\_employed) +   
## unemployment\_percent\_of\_labour\_force + agricultural\_production\_index\_2004\_2006\_100 +   
## s(urban\_population\_percent\_of\_total\_population) + health\_total\_expenditure\_percent\_of\_gdp +   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +   
## s(freedom) + s(generosity) + s(trust\_government\_corruption)  
##   
## Parametric coefficients:  
## Estimate Std. Error  
## (Intercept) 0.3740994 0.4853808  
## unemployment\_percent\_of\_labour\_force -0.0038517 0.0084051  
## agricultural\_production\_index\_2004\_2006\_100 -0.0001339 0.0019064  
## health\_total\_expenditure\_percent\_of\_gdp -0.0040872 0.0170518  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.0018611 0.0040183  
## t value Pr(>|t|)  
## (Intercept) 0.771 0.443  
## unemployment\_percent\_of\_labour\_force -0.458 0.648  
## agricultural\_production\_index\_2004\_2006\_100 -0.070 0.944  
## health\_total\_expenditure\_percent\_of\_gdp -0.240 0.811  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -0.463 0.645  
##   
## Approximate significance of smooth terms:  
## edf Ref.df F  
## s(employment\_industry\_percent\_of\_employed) 1.817 2.302 3.959  
## s(urban\_population\_percent\_of\_total\_population) 3.617 4.454 1.942  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 6.243 7.357 1.702  
## s(freedom) 2.549 3.196 2.817  
## s(generosity) 3.197 3.944 2.867  
## s(trust\_government\_corruption) 3.565 4.370 2.615  
## p-value   
## s(employment\_industry\_percent\_of\_employed) 0.0262 \*  
## s(urban\_population\_percent\_of\_total\_population) 0.1076   
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 0.1230   
## s(freedom) 0.0366 \*  
## s(generosity) 0.0310 \*  
## s(trust\_government\_corruption) 0.0392 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.472 Deviance explained = 61.2%  
## GCV = 0.13236 Scale est. = 0.096153 n = 95

print("GAM natūraliam prieaugiui")

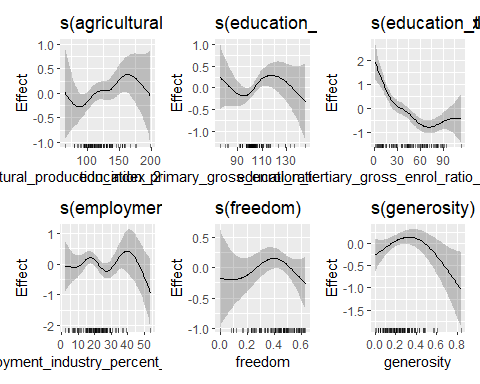
## [1] "GAM natūraliam prieaugiui"

# model\_gam\_natural <- fit\_gam(natural\_growth ~ s(employment\_industry\_percent\_of\_employed) +  
# s(unemployment\_percent\_of\_labour\_force) +  
# s(agricultural\_production\_index\_2004\_2006\_100) +  
# s(urban\_population\_percent\_of\_total\_population) +  
# s(health\_total\_expenditure\_percent\_of\_gdp) +  
# s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +  
# s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +  
# s(freedom) +  
# s(generosity) +  
# s(trust\_government\_corruption), regression\_train %>% dplyr::select(-migration\_growth))  
#  
#  
# draw(model\_gam\_natural)  
# k.check(model\_gam\_natural)  
# summary(model\_gam\_natural)  
  
  
model\_gam\_natural <- fit\_gam(natural\_growth ~ s(employment\_industry\_percent\_of\_employed) +  
 unemployment\_percent\_of\_labour\_force +  
 s(agricultural\_production\_index\_2004\_2006\_100) +  
 urban\_population\_percent\_of\_total\_population +  
 health\_total\_expenditure\_percent\_of\_gdp +  
 s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +  
 s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +  
 s(freedom) +  
 s(generosity) +  
 trust\_government\_corruption, regression\_train %>% dplyr::select(-migration\_growth))



##   
## Method: GCV Optimizer: magic  
## Smoothing parameter selection converged after 7 iterations.  
## The RMS GCV score gradient at convergence was 1.371315e-07 .  
## The Hessian was positive definite.  
## Model rank = 59 / 59   
##   
## Basis dimension (k) checking results. Low p-value (k-index<1) may  
## indicate that k is too low, especially if edf is close to k'.  
##   
## k' edf k-index  
## s(employment\_industry\_percent\_of\_employed) 9.00 5.16 0.96  
## s(agricultural\_production\_index\_2004\_2006\_100) 9.00 4.12 1.00  
## s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 9.00 3.34 1.07  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 9.00 4.83 1.33  
## s(freedom) 9.00 2.68 1.05  
## s(generosity) 9.00 2.41 0.99  
## p-value  
## s(employment\_industry\_percent\_of\_employed) 0.31  
## s(agricultural\_production\_index\_2004\_2006\_100) 0.38  
## s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 0.73  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 1.00  
## s(freedom) 0.66  
## s(generosity) 0.42

draw(model\_gam\_natural)



k.check(model\_gam\_natural)

## k' edf k-index  
## s(employment\_industry\_percent\_of\_employed) 9 5.159902 0.9649547  
## s(agricultural\_production\_index\_2004\_2006\_100) 9 4.123495 0.9987737  
## s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 9 3.340290 1.0692858  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 9 4.832763 1.3262088  
## s(freedom) 9 2.682792 1.0504460  
## s(generosity) 9 2.413316 0.9858652  
## p-value  
## s(employment\_industry\_percent\_of\_employed) 0.3300  
## s(agricultural\_production\_index\_2004\_2006\_100) 0.4350  
## s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 0.7600  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 0.9975  
## s(freedom) 0.6775  
## s(generosity) 0.4150

summary(model\_gam\_natural)

##   
## Family: gaussian   
## Link function: identity   
##   
## Formula:  
## natural\_growth ~ s(employment\_industry\_percent\_of\_employed) +   
## unemployment\_percent\_of\_labour\_force + s(agricultural\_production\_index\_2004\_2006\_100) +   
## urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +   
## s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +   
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) +   
## s(freedom) + s(generosity) + trust\_government\_corruption  
##   
## Parametric coefficients:  
## Estimate Std. Error t value  
## (Intercept) 1.253958 0.311162 4.030  
## unemployment\_percent\_of\_labour\_force 0.001227 0.013146 0.093  
## urban\_population\_percent\_of\_total\_population 0.001981 0.004159 0.476  
## health\_total\_expenditure\_percent\_of\_gdp -0.049611 0.028255 -1.756  
## trust\_government\_corruption 0.316856 0.779622 0.406  
## Pr(>|t|)   
## (Intercept) 0.000141 \*\*\*  
## unemployment\_percent\_of\_labour\_force 0.925931   
## urban\_population\_percent\_of\_total\_population 0.635351   
## health\_total\_expenditure\_percent\_of\_gdp 0.083527 .   
## trust\_government\_corruption 0.685681   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Approximate significance of smooth terms:  
## edf Ref.df F  
## s(employment\_industry\_percent\_of\_employed) 5.160 6.148 2.391  
## s(agricultural\_production\_index\_2004\_2006\_100) 4.123 5.082 1.848  
## s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 3.340 4.130 2.309  
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 4.833 5.874 8.740  
## s(freedom) 2.683 3.333 1.310  
## s(generosity) 2.413 3.019 3.228  
## p-value   
## s(employment\_industry\_percent\_of\_employed) 0.0363 \*   
## s(agricultural\_production\_index\_2004\_2006\_100) 0.1225   
## s(education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 0.0636 .   
## s(education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop) 9.15e-07 \*\*\*  
## s(freedom) 0.2198   
## s(generosity) 0.0276 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.808 Deviance explained = 86.1%  
## GCV = 0.31706 Scale est. = 0.227 n = 97

library(yardstick)  
  
# regresijos modelių įvertinimas  
regression\_test <- function(column, model\_linear, model\_gam, data, title) {  
 print(AIC(model\_linear))  
 print(AIC(model\_gam))  
  
  
 regression\_test <- data %>%  
 mutate(  
 predicted\_linear = predict(model\_linear, data),  
 predicted\_gam = predict(model\_gam, data)  
 )  
  
 set <- metric\_set(rmse, mae)  
  
 print("Tiesinis modelis")  
 print(set(regression\_test, {{ column }}, predicted\_linear))  
 print("GAM modelis")  
 print(set(regression\_test, {{ column }}, predicted\_gam))  
  
 regression\_test %>%  
 pivot\_longer(c(predicted\_gam, predicted\_linear)) %>%  
 mutate(name = factor(name, levels = c("predicted\_linear", "predicted\_gam"))) %>%  
 ggplot(aes({{ column }}, value)) +  
 geom\_point(size = 2) +  
 facet\_wrap(vars(name)) +  
 geom\_abline(color = "red", size = 2.25) +  
 labs(  
 x = "Tikros reikšmės", y = "Prognozuotos reikmšės",  
 title = title  
 ) +  
 theme\_minimal()  
}

# GAM modeliu gaunami nežymiai geresni rezultatai su mokymo duomeninis  
# , tačiau naudojant testavimo aibe pagerėjimo negaunama  
# Apskritai abu modeliai netinkami prognozuoti migracijos prieaugį  
print("Regresija migracijos prieaugiui")

## [1] "Regresija migracijos prieaugiui"

AIC(model\_linear\_migration)

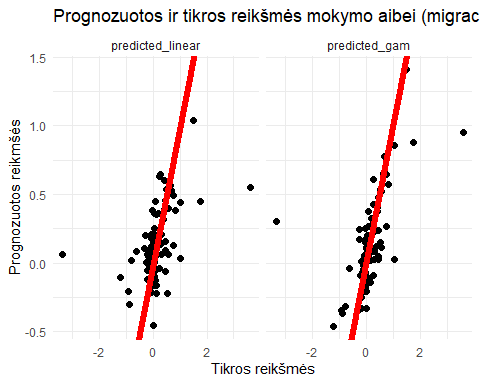
## [1] 79.80556

AIC(model\_gam\_migration)

## [1] 70.74101

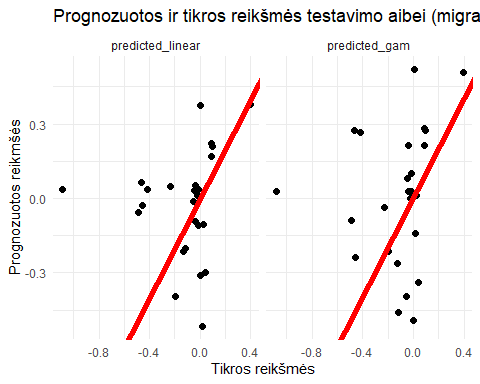
regression\_test(  
 migration\_growth, model\_linear\_migration, model\_gam\_migration, regression\_train,  
 "Prognozuotos ir tikros reikšmės mokymo aibei (migracijos prieaugis)"  
)

## [1] 79.80556  
## [1] 70.74101  
## [1] "Tiesinis modelis"  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.578  
## 2 mae standard 0.303  
## [1] "GAM modelis"  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.529  
## 2 mae standard 0.240



regression\_test(  
 migration\_growth, model\_linear\_migration, model\_gam\_migration, test,  
 "Prognozuotos ir tikros reikšmės testavimo aibei (migracijos prieaugis)"  
)

## [1] 79.80556  
## [1] 70.74101  
## [1] "Tiesinis modelis"  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.345  
## 2 mae standard 0.244  
## [1] "GAM modelis"  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.381  
## 2 mae standard 0.279



# Prognozuojant natūralų prieaugi gaunami geresni rezultatai negu prognozuojant migracijos prieaugį  
# mokymo aibėje matomas stiprus GAM modeliu gautas rezultatų pagerėjimas, tačiau testavimo aibėje skirtumai  
 # tik minimalūs  
print("Regresija natūraliam prieaugiui")

## [1] "Regresija natūraliam prieaugiui"

AIC(model\_linear\_natural)

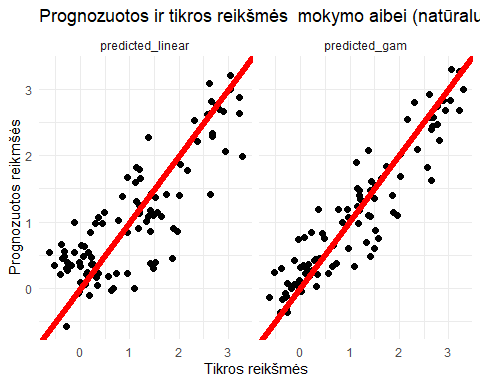
## [1] 181.8344

AIC(model\_gam\_natural)

## [1] 156.1355

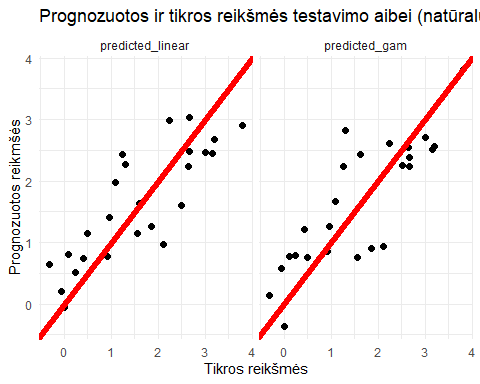
regression\_test(  
 natural\_growth, model\_linear\_natural, model\_gam\_natural, regression\_train,  
 "Prognozuotos ir tikros reikšmės mokymo aibei (natūralus prieaugis)"  
)

## [1] 181.8344  
## [1] 156.1355  
## [1] "Tiesinis modelis"  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.587  
## 2 mae standard 0.470  
## [1] "GAM modelis"  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.403  
## 2 mae standard 0.309



regression\_test(  
 natural\_growth, model\_linear\_natural, model\_gam\_natural, test,  
 "Prognozuotos ir tikros reikšmės testavimo aibei (natūralus prieaugis)"  
)

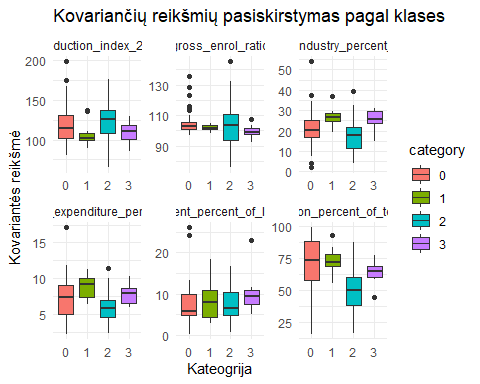
## [1] 181.8344  
## [1] 156.1355  
## [1] "Tiesinis modelis"  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.660  
## 2 mae standard 0.574  
## [1] "GAM modelis"  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.662  
## 2 mae standard 0.561



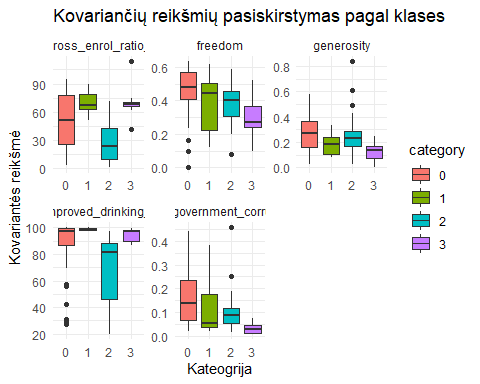
### Multinominė logistinė regresija

classification\_train <- train %>% dplyr::select(-migration\_growth, -natural\_growth)

# Kadangi gautos prastos migracijos prieaugio prognozės, vietoje tikslios   
 # prieaugio reikšmės prognozuojama tik ar tam tikro tipo gyventojų prieaugis teigiamas  
 # ar neigiamas (naudojamos prieš tai sudarytos klasės)  
  
# Stačiakampės diagramos pagal kiekvieną kovariantę  
classification\_train %>%  
 dplyr::select(1:6, category) %>%  
 pivot\_longer(-category) %>%  
 ggplot(aes(x = category, y = value, fill = category)) +  
 facet\_wrap(vars(name), scales = "free") +  
 geom\_boxplot() +  
 theme\_minimal() +  
 labs(title = "Kovariančių reikšmių pasiskirstymas pagal klases") +   
 xlab("Kateogrija") + ylab("Kovariantės reikšmė")



classification\_train %>%  
 dplyr::select(7:length(classification\_train), category) %>%  
 pivot\_longer(-category) %>%  
 ggplot(aes(x = category, y = value, fill = category)) +  
 facet\_wrap(vars(name), scales = "free") +  
 geom\_boxplot() +  
 theme\_minimal() +  
 labs(title = "Kovariančių reikšmių pasiskirstymas pagal klases") +  
 xlab("Kateogrija") + ylab("Kovariantės reikšmė")



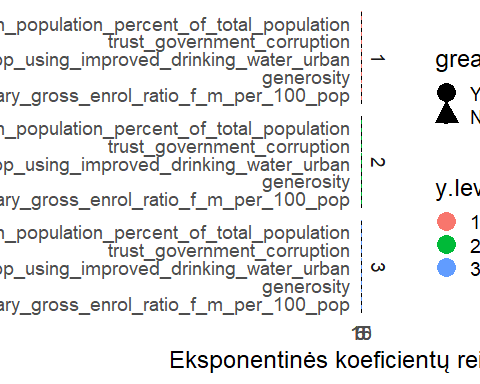
# Pažingsnine regresija sumažintas modelis statistiškai reikšmingai nesiskiria   
anova(model\_logistic, model\_logistic\_small)

## Likelihood ratio tests of Multinomial Models  
##   
## Response: category  
## Model  
## 1 urban\_population\_percent\_of\_total\_population + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban  
## 2 employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force + agricultural\_production\_index\_2004\_2006\_100 + urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + freedom + generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban  
## Resid. df Resid. Dev Test Df LR stat. Pr(Chi)  
## 1 273 145.8441   
## 2 255 125.9399 1 vs 2 18 19.90425 0.3382363

summary(model\_logistic\_small)

## Call:  
## nnet::multinom(formula = category ~ urban\_population\_percent\_of\_total\_population +   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + generosity +   
## trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban,   
## data = classification\_train, trace = FALSE)  
##   
## Coefficients:  
## (Intercept) urban\_population\_percent\_of\_total\_population  
## 1 -39.8819145 -0.04607011  
## 2 2.2575147 -0.01676782  
## 3 -0.7546367 -0.20816219  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop generosity  
## 1 0.03716542 -8.362663  
## 2 -0.03755632 -1.258363  
## 3 0.14190074 -11.776205  
## trust\_government\_corruption pop\_using\_improved\_drinking\_water\_urban  
## 1 -2.163034 0.427085643  
## 2 -4.428862 0.007482486  
## 3 -41.363126 0.086422684  
##   
## Std. Errors:  
## (Intercept) urban\_population\_percent\_of\_total\_population  
## 1 0.2959585 0.03887555  
## 2 1.1680728 0.01799239  
## 3 7.7821840 0.09635350  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop generosity  
## 1 0.02967933 4.347576  
## 2 0.01890919 1.815050  
## 3 0.06040160 6.360179  
## trust\_government\_corruption pop\_using\_improved\_drinking\_water\_urban  
## 1 4.2367679 0.02636595  
## 2 3.1372343 0.01702100  
## 3 0.2589815 0.08187579  
##   
## Residual Deviance: 145.8441   
## AIC: 181.8441

# multinominės logistinės regresijos modelio koeficientų grafikas  
plot\_coefficients <- function(model) {  
 tidy(model) %>%  
 filter(term != "(Intercept)") %>%  
 mutate(greater\_than\_one = if\_else(estimate > 0, "Yes", "No")) %>%  
 ggplot(aes(term, exp(estimate), color = y.level, shape = greater\_than\_one)) +  
 geom\_pointrange(aes(ymin = exp(estimate - std.error), ymax = exp(estimate + std.error)), size=1.5) +  
 scale\_x\_discrete() +  
 coord\_flip() +  
 theme\_minimal(base\_size = 18) +  
 geom\_hline(yintercept = 1, linetype = "dashed") +  
 scale\_y\_continuous(oob = scales::squish, limits = c(-1, 16)) +  
 facet\_grid(rows = vars(y.level), scales = "free") +  
 labs(x = "Kovariantė", y = "Eksponentinės koeficientų reikšmės")  
}  
  
plot\_coefficients(model\_logistic\_small)

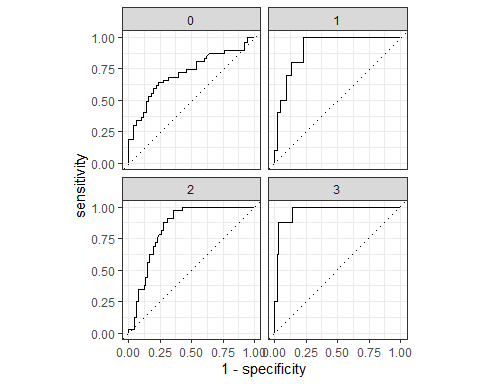


# pačių sudaryta modelio kokybės metrika, kuri "pataiso" bendrą tikslumą  
# priskirdama 0.5 - jeigu teisingai prognozuotas vieno tipo prieaugis  
# 1 - jeigu teisingi abu prieaugiai  
# 0 - jeigu abiejų tipų prieaugiai neteisingi  
  
custom\_metric <- function(y\_true, y\_pred) {  
 c(  
 "custom\_metric", "multiclass",  
 case\_when(  
 y\_true %in% c(0, 3) & y\_pred %in% c(1, 2) ~ 0.5,  
 y\_true %in% c(1, 2) & y\_pred %in% c(0, 3) ~ 0.5,  
 y\_true == y\_pred ~ 1,  
 TRUE ~ 0  
 ) %>%  
 mean()  
 )  
}  
  
# Multinominės logistinės regresijos modelio įvertinimas  
classification\_eval <- function(model, data) {  
 df\_pred\_truth <- tibble(  
 predicted = factor(predict(model, data)),  
 truth = data$category  
 ) %>%  
 cbind(as.data.frame(model$fitted.values))  
  
  
  
 classification\_metrics <- metric\_set(accuracy, j\_index, f\_meas)  
  
 print("Maišos matrica")  
 print(conf\_mat(df\_pred\_truth,  
 truth = truth,  
 estimate = predicted  
 ))  
  
 print("Modelio kokybės metrikos")  
 print(classification\_metrics(df\_pred\_truth,  
 truth = truth,  
 estimate = predicted  
 ) %>%  
 rbind(custom\_metric(df\_pred\_truth$truth, df\_pred\_truth$predicted)))  
  
  
  
 print(roc\_auc(df\_pred\_truth, truth = truth, c("0", "1", "2", "3"), estimator = "macro"))  
  
 roc\_curve(df\_pred\_truth, truth = truth, c("0", "1", "2", "3")) %>%  
 autoplot()  
}  
  
print("Pradinis multinominės logistinės regresijos modelis")

## [1] "Pradinis multinominės logistinės regresijos modelis"

classification\_eval(model\_logistic\_small, classification\_train)

## [1] "Maišos matrica"  
## Truth  
## Prediction 0 1 2 3  
## 0 32 4 10 1  
## 1 2 4 0 0  
## 2 12 0 22 0  
## 3 1 2 0 7  
## [1] "Modelio kokybės metrikos"  
## # A tibble: 4 x 3  
## .metric .estimator .estimate   
## <chr> <chr> <chr>   
## 1 accuracy multiclass 0.670103092783505  
## 2 j\_index macro 0.525509827074684  
## 3 f\_meas macro 0.656323877068558  
## 4 custom\_metric multiclass 0.824742268041237  
## # A tibble: 1 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 roc\_auc macro 0.856



# Turimas ne itin ryškus klasių išbalasavimas  
# (daugumos klasės stebėjimų beveik 6 kartus daugiau nei mažiausios)   
# Todėl rezultatai gali pagerėti sugeneravus dirbtinius papildomus stebėjimus  
classification\_train %>% count(category)

## # A tibble: 4 x 2  
## category n  
## <fct> <int>  
## 1 0 47  
## 2 1 10  
## 3 2 32  
## 4 3 8

library(themis)  
  
smote\_recipe <- recipe(category ~ .,  
 data = classification\_train  
) %>%  
 step\_smote(category, over\_ratio = 1)  
  
  
smote\_recipe <- prep(smote\_recipe, training = classification\_train)  
  
classification\_train2 <- bake(smote\_recipe, NULL)

model\_logistic2 <- nnet::multinom(category ~ ., data = classification\_train2, trace = FALSE)  
  
model\_logistic2\_small <- stats::step(model\_logistic2, direction = "both")

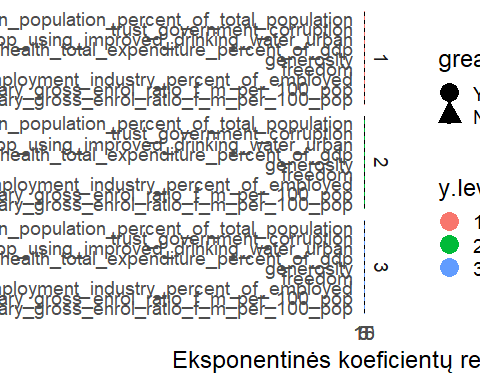
anova(model\_logistic2, model\_logistic2\_small)

## Likelihood ratio tests of Multinomial Models  
##   
## Response: category  
## Model  
## 1 employment\_industry\_percent\_of\_employed + urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + freedom + generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban  
## 2 employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force + agricultural\_production\_index\_2004\_2006\_100 + urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + freedom + generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban  
## Resid. df Resid. Dev Test Df LR stat. Pr(Chi)  
## 1 534 180.0295   
## 2 528 175.6071 1 vs 2 6 4.422416 0.6197099

summary(model\_logistic2\_small)

## Call:  
## nnet::multinom(formula = category ~ employment\_industry\_percent\_of\_employed +   
## urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## freedom + generosity + trust\_government\_corruption + pop\_using\_improved\_drinking\_water\_urban,   
## data = classification\_train2, trace = FALSE)  
##   
## Coefficients:  
## (Intercept) employment\_industry\_percent\_of\_employed  
## 1 -78.04724 0.451424798  
## 2 5.48757 -0.006804232  
## 3 72.40568 -0.178757218  
## urban\_population\_percent\_of\_total\_population  
## 1 -0.08797960  
## 2 -0.02502958  
## 3 -0.45658276  
## health\_total\_expenditure\_percent\_of\_gdp  
## 1 0.78370798  
## 2 -0.05718385  
## 3 0.11190683  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop  
## 1 -0.01023564  
## 2 -0.01370988  
## 3 -0.85938214  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop freedom generosity  
## 1 0.13119162 -13.79913 -19.607247  
## 2 -0.04115902 -1.31000 -2.897483  
## 3 0.34508959 13.68574 -30.026863  
## trust\_government\_corruption pop\_using\_improved\_drinking\_water\_urban  
## 1 13.593242 0.67752809  
## 2 -3.378663 0.01374586  
## 3 -70.597268 0.31405698  
##   
## Std. Errors:  
## (Intercept) employment\_industry\_percent\_of\_employed  
## 1 1.023145 0.11970460  
## 2 3.008252 0.04503703  
## 3 2.535592 0.11119875  
## urban\_population\_percent\_of\_total\_population  
## 1 0.05021285  
## 2 0.01974761  
## 3 0.09355286  
## health\_total\_expenditure\_percent\_of\_gdp  
## 1 0.2771792  
## 2 0.1376968  
## 3 0.3893734  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop  
## 1 0.08360149  
## 2 0.02435611  
## 3 0.10017359  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop freedom generosity  
## 1 0.04422784 6.075000 6.032812  
## 2 0.01960441 2.566533 1.975537  
## 3 0.06706318 6.035300 8.744504  
## trust\_government\_corruption pop\_using\_improved\_drinking\_water\_urban  
## 1 7.632834 0.11627297  
## 2 3.271378 0.01832338  
## 3 1.929764 0.11536189  
##   
## Residual Deviance: 180.0295   
## AIC: 240.0295

plot\_coefficients(model\_logistic2\_small)

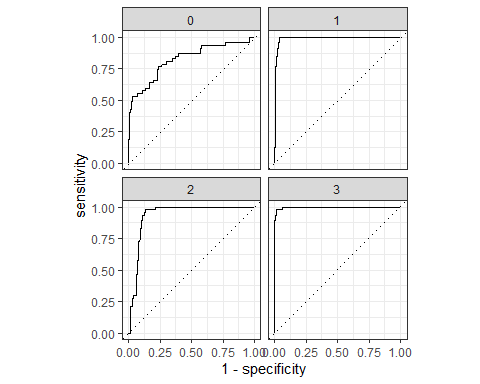


print("Multinominės logistinės regresijos modelis su SMOTE")

## [1] "Multinominės logistinės regresijos modelis su SMOTE"

classification\_eval(model\_logistic2\_small, classification\_train2)

## [1] "Maišos matrica"  
## Truth  
## Prediction 0 1 2 3  
## 0 25 0 9 1  
## 1 6 47 0 0  
## 2 14 0 38 0  
## 3 2 0 0 46  
## [1] "Modelio kokybės metrikos"  
## # A tibble: 4 x 3  
## .metric .estimator .estimate   
## <chr> <chr> <chr>   
## 1 accuracy multiclass 0.829787234042553  
## 2 j\_index macro 0.773049645390071  
## 3 f\_meas macro 0.821463479467331  
## 4 custom\_metric multiclass 0.906914893617021  
## # A tibble: 1 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 roc\_auc macro 0.936



# apskaičiuoja kiek kartų teisingai prognozuotas kiekvieno tipo prieaugis  
# (lengviau interpretuoti negu maišos matricą)  
custom\_confusion <- function(y\_true, y\_pred) {  
 case\_when(  
 y\_true == y\_pred ~ "Correct both",  
 (y\_true %in% c(0, 2) & y\_pred %in% c(0, 2)) | (y\_true %in% c(1, 3) & y\_pred %in% c(1, 3)) ~ "Correct natural",  
 (y\_true %in% c(0, 1) & y\_pred %in% c(0, 1)) | (y\_true %in% c(2, 3) & y\_pred %in% c(2, 3)) ~ "Correct migration",  
 TRUE ~ "Correct none"  
 ) %>%  
 tibble(results = .) %>%  
 count(results)  
}  
  
  
# palyginimui jeigu butų naudojami prieš tai sudaryti 2 regresijos modeliai prognozuoti klases  
class\_predictions <- function() {  
 tibble(  
 migration\_growth = predict(model\_gam\_migration, test),  
 natural\_growth = predict(model\_gam\_natural, test)  
 ) %>%  
 mutate(category = factor(case\_when(  
 migration\_growth >= 0 & natural\_growth >= 0 ~ 0, # "P migration, P natural",  
 migration\_growth >= 0 & natural\_growth < 0 ~ 1, # "P migration, N natural",  
 migration\_growth < 0 & natural\_growth >= 0 ~ 2, # "N migration, P natural",  
 TRUE ~ 3  
 ))) %>%  
 pull(category)  
}  
  
  
classification\_test <- function(model, data, name) {  
 df\_pred\_truth <- tibble(truth = data$category)  
 classification\_metrics <- metric\_set(accuracy, j\_index, f\_meas)  
  
 if (name == "Naudojant du regresijos modelius") {  
 df\_pred\_truth$predicted <- factor(class\_predictions(), levels = c(0, 1, 2, 3))  
 } else {  
 df\_pred\_truth$predicted <- factor(predict(model, test), levels = c(0, 1, 2, 3))  
 }  
  
  
 print("Maišos matrica")  
 conf\_mat(df\_pred\_truth,  
 truth = truth,  
 estimate = predicted  
 ) %>% print()  
  
  
 print(custom\_confusion(df\_pred\_truth$truth, df\_pred\_truth$predicted))  
  
 print("Modelio kokybės metrikos")  
 classification\_metrics(df\_pred\_truth, truth, estimate = predicted) %>%  
 rbind(custom\_metric(df\_pred\_truth$truth, df\_pred\_truth$predicted)) %>%  
 print()  
  
 cat("\n\n")  
}

# Naudojant testavimo aibę.  
# Geriausi rezultatai gauti su modeliu, kuriam naudotas SMOTE algoritmas  
# blogiausi - panaudojus regresijos modelius  
  
# Vėl matoma, kad geresni rezultatai gaunami prognozuojant natūralų gyventojų prieaugį  
print("Naudojant pradinį multinominės logistinės regresijos modelį")

## [1] "Naudojant pradinį multinominės logistinės regresijos modelį"

classification\_test(model\_logistic, test, "Pradinis")

## [1] "Maišos matrica"  
## Truth  
## Prediction 0 1 2 3  
## 0 2 1 2 0  
## 1 0 0 0 0  
## 2 5 0 12 0  
## 3 0 1 1 1  
## # A tibble: 3 x 2  
## results n  
## <chr> <int>  
## 1 Correct both 15  
## 2 Correct migration 2  
## 3 Correct natural 8  
## [1] "Modelio kokybės metrikos"  
## # A tibble: 4 x 3  
## .metric .estimator .estimate   
## <chr> <chr> <chr>   
## 1 accuracy multiclass 0.6   
## 2 j\_index macro 0.333928571428571  
## 3 f\_meas macro 0.527777777777778  
## 4 custom\_metric multiclass 0.8

print("Naudojant multinomės logistinės regresijos modelį su SMOTE")

## [1] "Naudojant multinomės logistinės regresijos modelį su SMOTE"

classification\_test(model\_logistic2\_small, test, "SMOTE")

## [1] "Maišos matrica"  
## Truth  
## Prediction 0 1 2 3  
## 0 2 0 2 0  
## 1 0 1 0 0  
## 2 5 0 13 0  
## 3 0 1 0 1  
## # A tibble: 2 x 2  
## results n  
## <chr> <int>  
## 1 Correct both 17  
## 2 Correct natural 8  
## [1] "Modelio kokybės metrikos"  
## # A tibble: 4 x 3  
## .metric .estimator .estimate   
## <chr> <chr> <chr>   
## 1 accuracy multiclass 0.68   
## 2 j\_index macro 0.499900793650794  
## 3 f\_meas macro 0.621212121212121  
## 4 custom\_metric multiclass 0.84

classification\_test(model\_logistic, test, "Naudojant du regresijos modelius")

## [1] "Maišos matrica"  
## Truth  
## Prediction 0 1 2 3  
## 0 4 1 8 1  
## 1 0 1 0 0  
## 2 3 0 7 0  
## 3 0 0 0 0  
## # A tibble: 4 x 2  
## results n  
## <chr> <int>  
## 1 Correct both 12  
## 2 Correct migration 1  
## 3 Correct natural 11  
## 4 Correct none 1  
## [1] "Modelio kokybės metrikos"  
## # A tibble: 4 x 3  
## .metric .estimator .estimate   
## <chr> <chr> <chr>   
## 1 accuracy multiclass 0.48   
## 2 j\_index macro 0.170634920634921  
## 3 f\_meas macro 0.535873015873016  
## 4 custom\_metric multiclass 0.72

u