Regresinės analizės projektinis darbas

Vainius Gataveckas, Matas Gaulia, Dovydas Martinkus

### Cleaning the data

library(tidyverse)  
library(janitor)  
  
  
pop\_natural <- read\_csv("natural-population-growth.csv") %>%  
 filter(Year == 2017) %>%  
 select(1, 4) %>%  
 set\_names(c("country", "natural\_growth"))  
  
pop\_total <- read\_csv("population-growth-rates.csv") %>%  
 filter(Year == 2017) %>%  
 select(1, 4) %>%  
 set\_names(c("country", "total\_growth"))  
  
  
country\_stats <- read\_csv("country\_profile\_variables.csv") %>%  
 clean\_names() %>%  
 select(-c(2, 3, 4, 5, 6, 7))  
  
happiness <- read\_csv("2017.csv") %>%  
 clean\_names() %>%  
 select(-c(2), -starts\_with("whisker"), -c("dystopia\_residual", "happiness\_score", "family"))

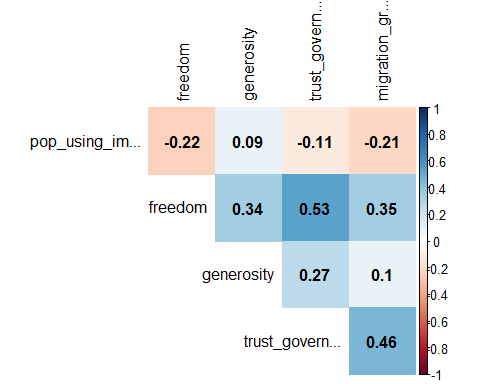
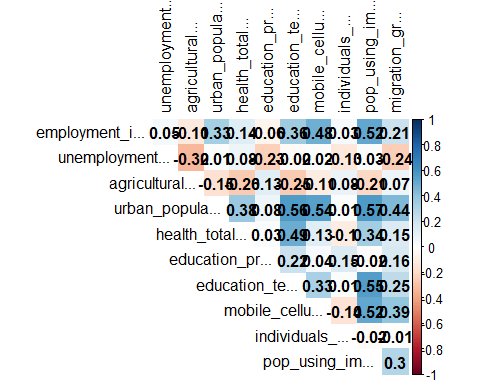
x <- reduce(list(pop\_natural, pop\_total, country\_stats, happiness), left\_join, by = "country")  
  
  
  
country <- x$country  
  
x <- x %>%  
 select(  
 -starts\_with("gdp"),  
 -starts\_with("labour"),  
 -starts\_with("international"),  
 -starts\_with("balance"),  
 -starts\_with("population"),  
 -starts\_with("fertility")  
 ) %>%  
 select(  
 -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,  
 -contains("41")  
 ) %>%  
 mutate(across(everything(), ~ replace(., . %in% c("...", "-99", ".../..."), "0"))) %>%  
 mutate(across(starts\_with("education") | starts\_with("pop"), ~ str\_split(., "/") %>% map(~ mean(as.numeric(.))))) %>%  
 mutate(across(-country, as.numeric)) %>%  
 mutate(migration\_growth = total\_growth - natural\_growth) %>%  
 drop\_na() %>%  
 select(-total\_growth, -country)

library(rsample)  
  
set.seed(123456)  
  
train\_test\_split <- initial\_split(tibble(country = country), prop = 0.8)  
train <- training(train\_test\_split)  
test <- testing(train\_test\_split)  
  
country\_train <- train$country  
country\_test <- test$country  
  
  
classification <- 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"  
 select(-c("migration\_growth", "natural\_growth"))  
  
  
train\_test\_split <- initial\_split(x, prop = 0.8)  
regression\_train <- training(train\_test\_split)  
regression\_test <- testing(train\_test\_split)  
  
  
train\_test\_split <- initial\_split(classification, strata = category, prop = 0.65)  
classification\_train <- training(train\_test\_split)  
classification\_test <- testing(train\_test\_split)

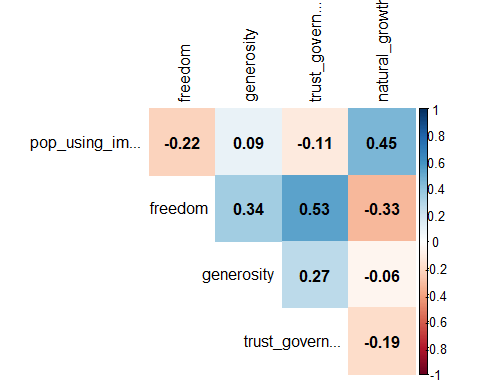
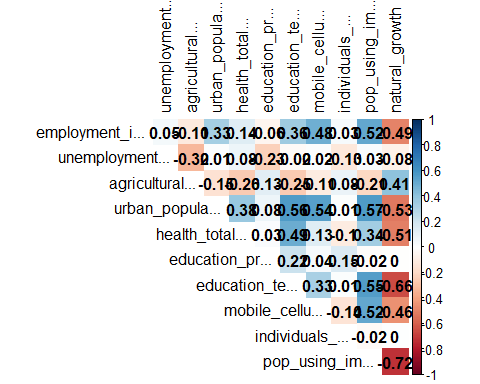
### Regression models

library(recipes)  
  
reg\_recipe <- recipe(natural\_growth ~ .,  
 data = regression\_train  
) %>%  
 add\_role(migration\_growth, new\_role = "outcome") %>%  
 step\_corr(all\_predictors(), threshold = 0.7) %>%  
 step\_nzv(all\_predictors())  
  
  
reg\_recipe <- prep(reg\_recipe, training = regression\_train)  
  
regression\_train <- bake(reg\_recipe, regression\_train)  
regression\_test <- bake(reg\_recipe, regression\_test)

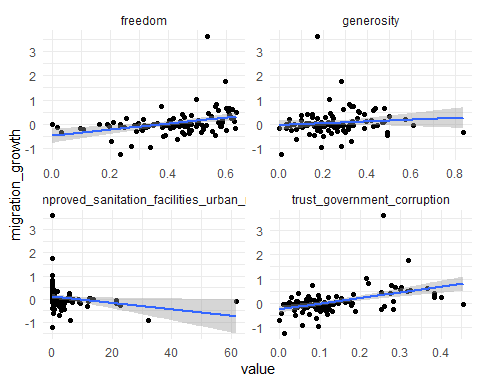
library(corrplot)  
  
  
correlation <- function(name, name2) {  
 correlation\_matrix <- regression\_train %>%  
 select(1:10, {{ name }}, -{{ name2 }}) %>%  
 set\_names(., str\_trunc(names(.), 15)) %>%  
 cor()  
  
 corrplot(correlation\_matrix, order = "original", method = "color", type = "upper", diag = FALSE, tl.col = "black", addCoef.col = "black", )  
  
  
  
 correlation\_matrix <- regression\_train %>%  
 select(11: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)



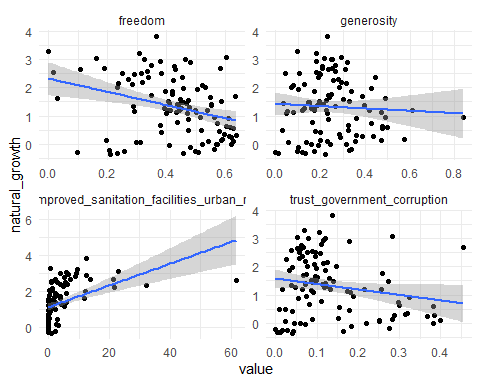
correlation(natural\_growth, migration\_growth)



scatterplot <- function(name, name2) {  
 regression\_train %>%  
 select(1:10, {{ 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()  
  
  
 regression\_train %>%  
 select(11: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()  
}  
  
  
scatterplot(migration\_growth, natural\_growth)



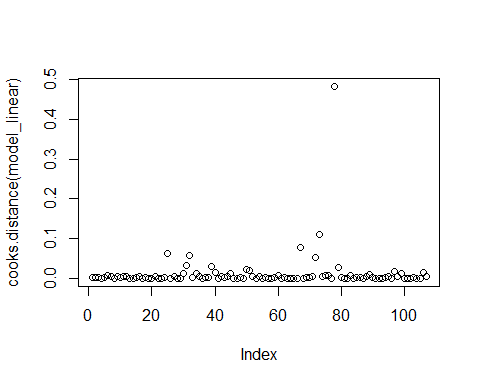
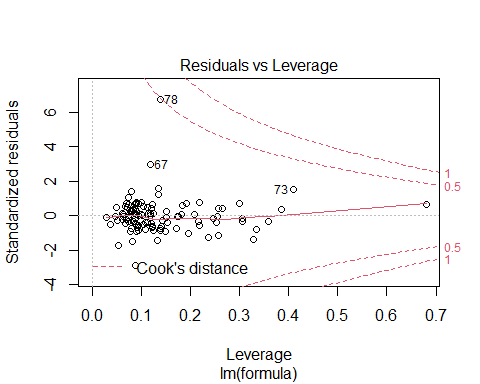
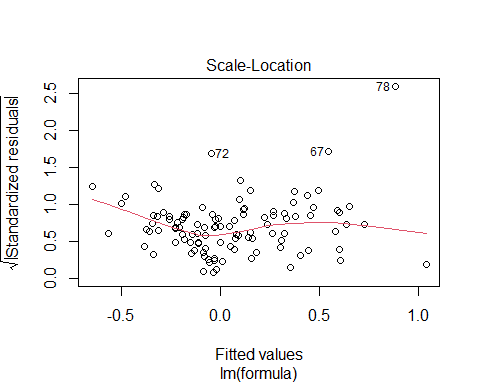
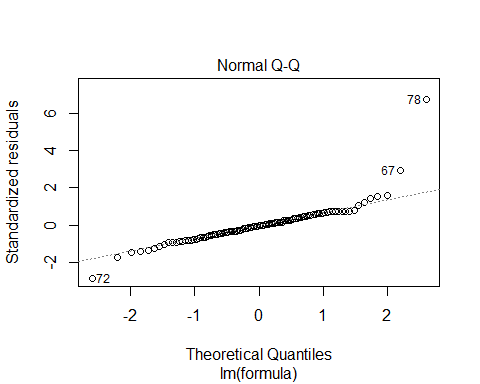
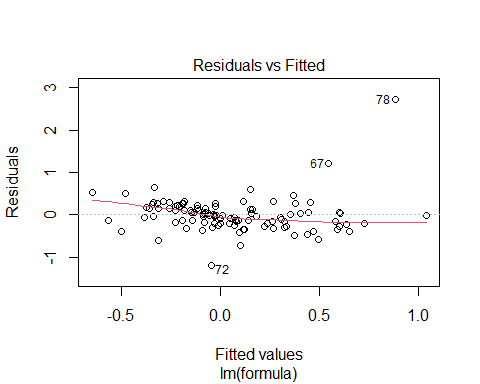
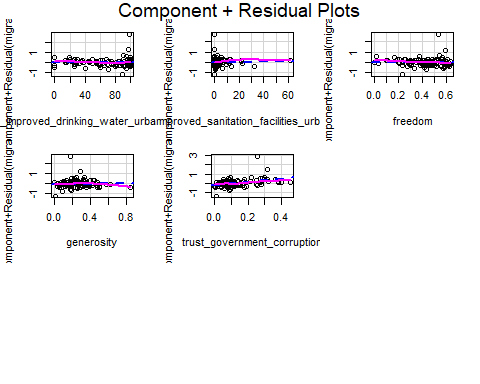
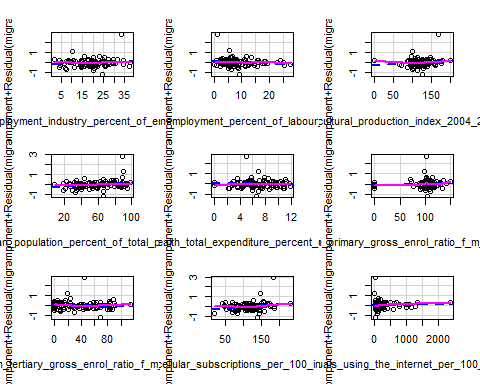
scatterplot(natural\_growth, migration\_growth)



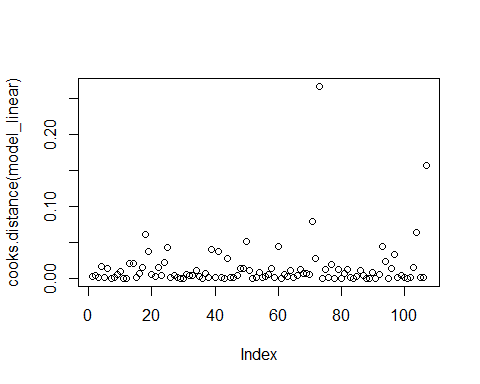
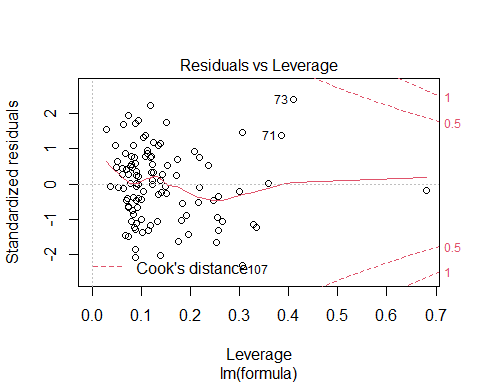
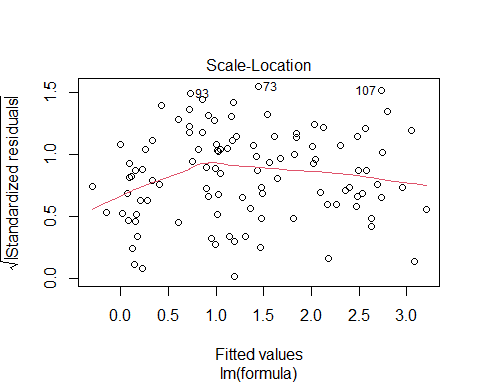
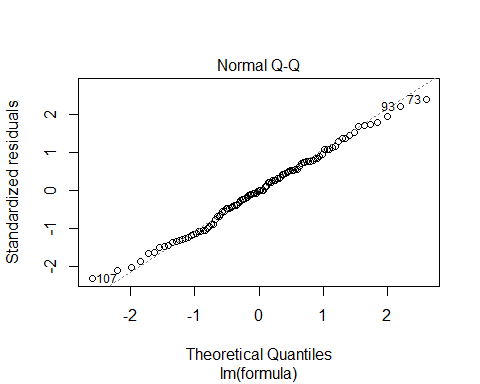
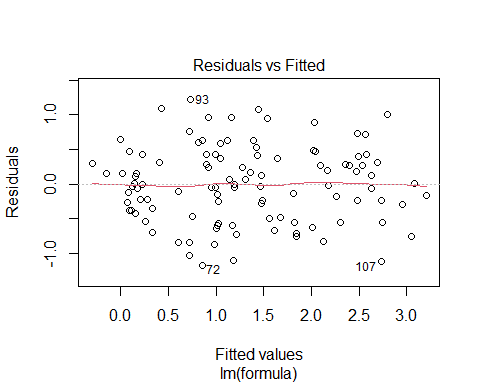
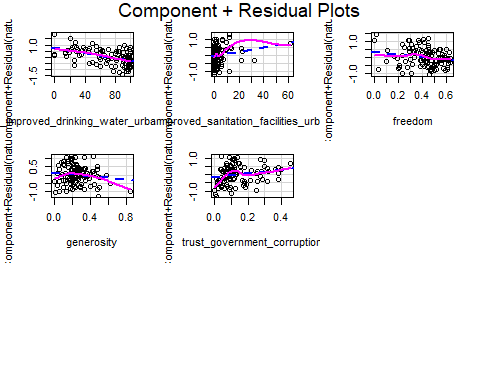
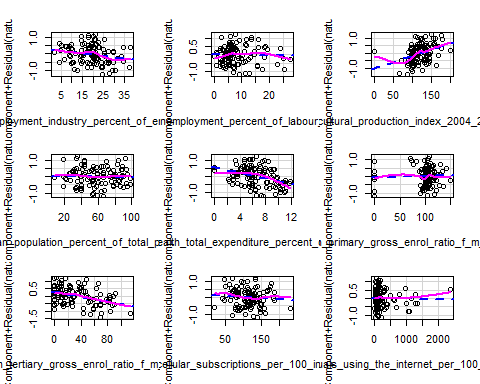
### Regression model

library(car)  
library(effects)  
library(lm.beta)  
  
  
linear\_fit <- function(formula, data) {  
 model\_linear <- lm(formula, data = data)  
  
 crPlots(model\_linear)  
 plot(model\_linear)  
 plot(cooks.distance(model\_linear))  
  
 return(model\_linear)  
  
 model\_linear <- stats::step(model\_linear, direction = "both")  
  
 plot(predictorEffects(model\_linear))  
  
  
 stand\_coeffs <- lm.beta(model\_linear)  
  
 tibble(x = names(stand\_coeffs$standardized.coefficients), y = stand\_coeffs$standardized.coefficients) %>%  
 cbind(confint(stand\_coeffs)) %>%  
 set\_names(c("variable", "coeff", "low", "high")) %>%  
 ggplot(aes(variable, coeff)) +  
 geom\_pointrange(aes(ymin = low, ymax = high), color = "blue") +  
 scale\_x\_discrete() +  
 coord\_flip() +  
 theme\_minimal()  
  
 model\_linear  
}

model\_linear\_migration <- linear\_fit(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 +  
 mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 + individuals\_using\_the\_internet\_per\_100\_inhabitants +  
 pop\_using\_improved\_drinking\_water\_urban\_rural\_percent + pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +  
 freedom + generosity + trust\_government\_corruption, regression\_train %>% select(-natural\_growth))

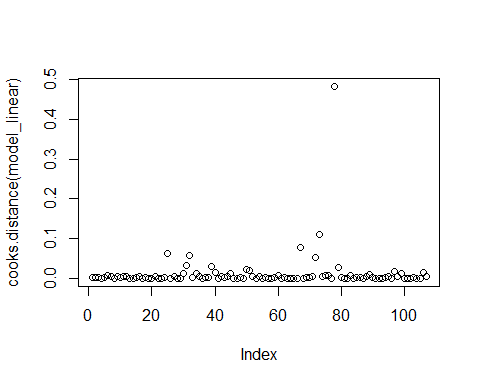
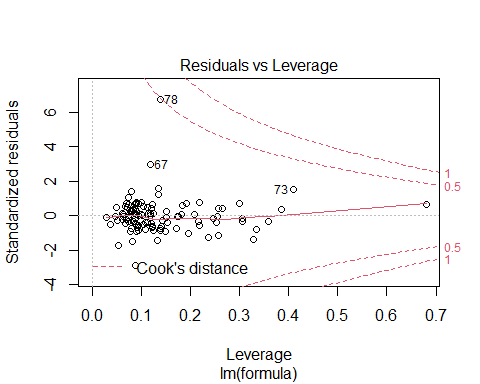
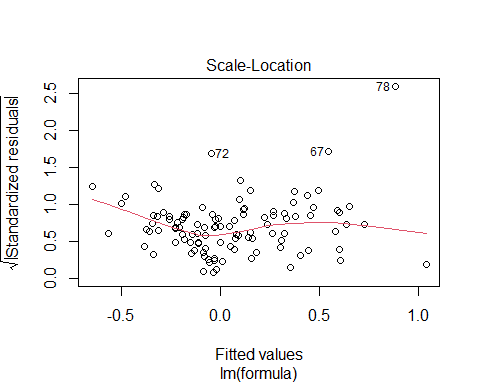
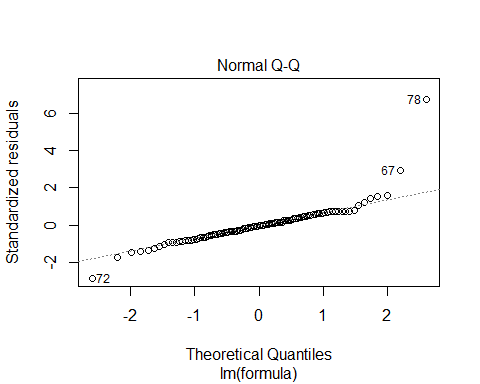
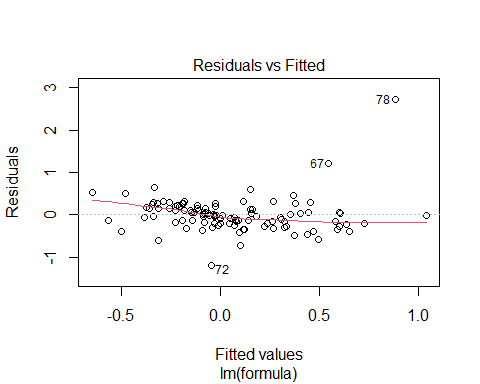
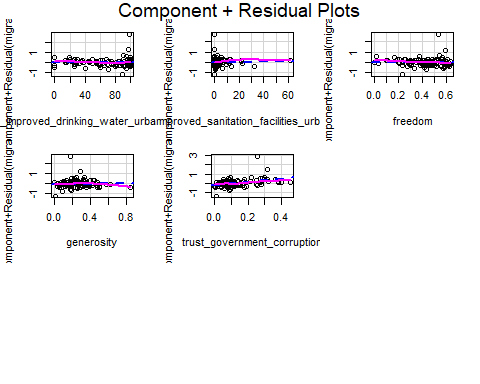
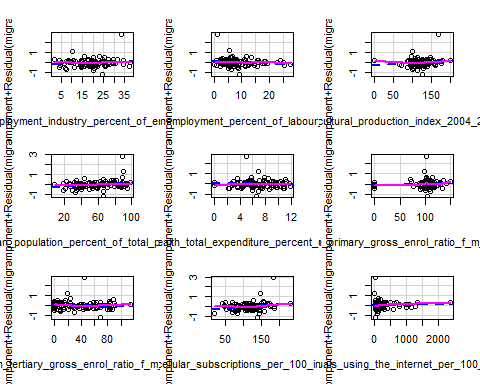


model\_linear\_natural <- linear\_fit(natural\_growth ~ employment\_industry\_percent\_of\_employed +  
 unemployment\_percent\_of\_labour\_force + agricultural\_production\_index\_2004\_2006\_100 +  
 urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +  
 education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +  
 mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 + individuals\_using\_the\_internet\_per\_100\_inhabitants +  
 pop\_using\_improved\_drinking\_water\_urban\_rural\_percent + pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +  
 freedom + generosity + trust\_government\_corruption, regression\_train %>% select(-migration\_growth))

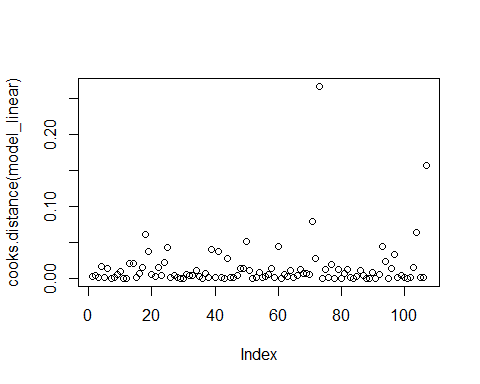
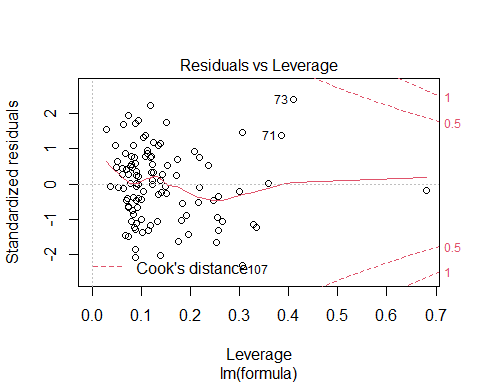
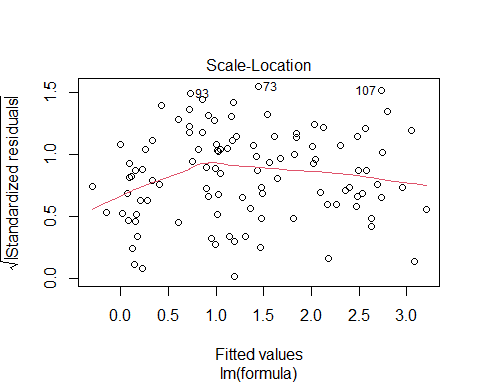
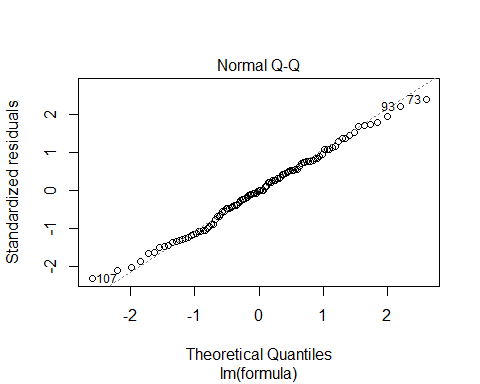
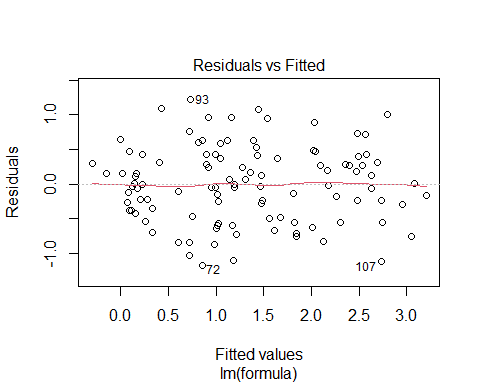
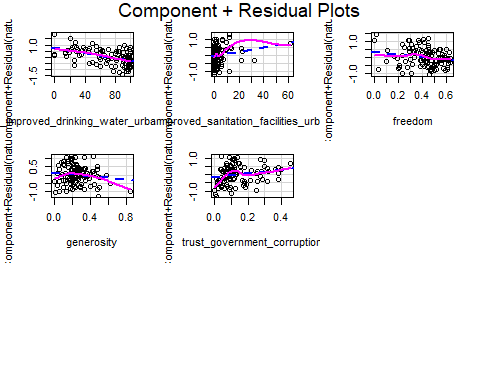
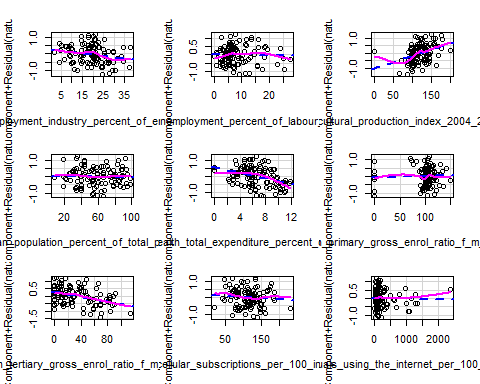


library(quantreg)  
  
fit\_quantile <- function(formula, data, quantiles) {  
 model\_quantile <- rq(formula, data = regression\_train, tau = quantiles)  
  
 summary(model\_quantile, se = "boot")  
 plot(summary(model\_quantile))  
 anova(model\_quantile, test = "Wald", joint = TRUE)  
  
 model\_quantile  
}

model\_linear\_migration <- linear\_fit(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 +  
 mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 + individuals\_using\_the\_internet\_per\_100\_inhabitants +  
 pop\_using\_improved\_drinking\_water\_urban\_rural\_percent + pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +  
 freedom + generosity + trust\_government\_corruption, regression\_train %>% select(-natural\_growth))

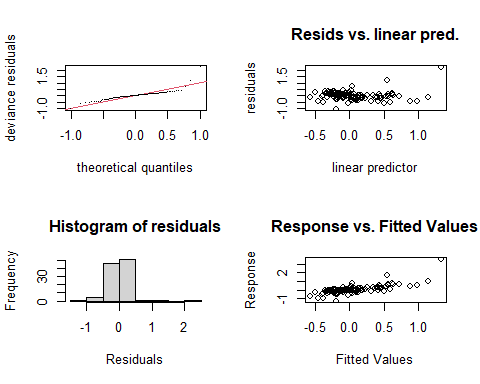


model\_linear\_natural <- linear\_fit(natural\_growth ~ employment\_industry\_percent\_of\_employed +  
 unemployment\_percent\_of\_labour\_force + agricultural\_production\_index\_2004\_2006\_100 +  
 urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +  
 education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +  
 mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 + individuals\_using\_the\_internet\_per\_100\_inhabitants +  
 pop\_using\_improved\_drinking\_water\_urban\_rural\_percent + pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +  
 freedom + generosity + trust\_government\_corruption, regression\_train %>% select(-migration\_growth))



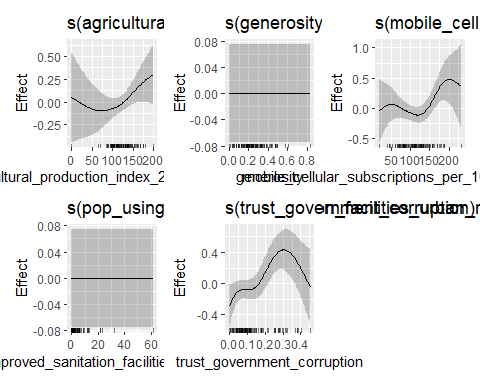
library(mgcv)  
library(gratia)  
  
fit\_gam <- function(formula, data) {  
 model\_gam <- gam(formula, data = data, select = TRUE)  
  
 gam.check(model\_gam)  
 summary(model\_gam)  
 draw(model\_gam)  
 k.check(model\_gam)  
  
 model\_gam  
}

model\_gam\_migration <- fit\_gam(migration\_growth ~ 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 +  
 education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +  
 s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) + individuals\_using\_the\_internet\_per\_100\_inhabitants +  
 pop\_using\_improved\_drinking\_water\_urban\_rural\_percent + s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) +  
 freedom + s(generosity) + s(trust\_government\_corruption), regression\_train %>% select(-natural\_growth))



##   
## Method: GCV Optimizer: magic  
## Smoothing parameter selection converged after 44 iterations.  
## The RMS GCV score gradient at convergence was 3.001327e-08 .  
## The Hessian was positive definite.  
## Model rank = 55 / 55   
##   
## 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'  
## s(agricultural\_production\_index\_2004\_2006\_100) 9.00e+00  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 9.00e+00  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 9.00e+00  
## s(generosity) 9.00e+00  
## s(trust\_government\_corruption) 9.00e+00  
## edf  
## s(agricultural\_production\_index\_2004\_2006\_100) 1.43e+00  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 3.68e+00  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 6.29e-10  
## s(generosity) 7.01e-11  
## s(trust\_government\_corruption) 3.60e+00  
## k-index p-value  
## s(agricultural\_production\_index\_2004\_2006\_100) 1.04 0.64  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 1.07 0.81  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 1.00 0.49  
## s(generosity) 1.10 0.86  
## s(trust\_government\_corruption) 1.21 0.97

draw(model\_gam\_migration)



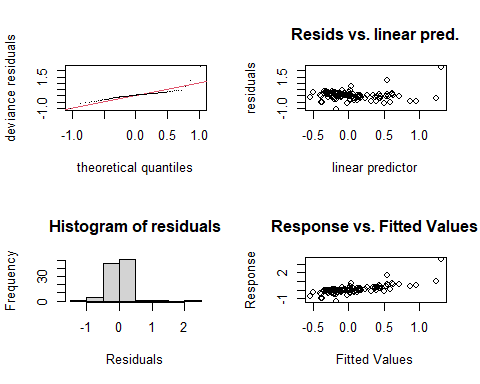
k.check(model\_gam\_migration)

## k' edf  
## s(agricultural\_production\_index\_2004\_2006\_100) 9 1.425541e+00  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 9 3.677969e+00  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 9 6.290363e-10  
## s(generosity) 9 7.012801e-11  
## s(trust\_government\_corruption) 9 3.602641e+00  
## k-index  
## s(agricultural\_production\_index\_2004\_2006\_100) 1.041585  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 1.072290  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 1.004677  
## s(generosity) 1.097520  
## s(trust\_government\_corruption) 1.210355  
## p-value  
## s(agricultural\_production\_index\_2004\_2006\_100) 0.6425  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 0.7950  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 0.4425  
## s(generosity) 0.8375  
## s(trust\_government\_corruption) 0.9875

summary(model\_gam\_migration)

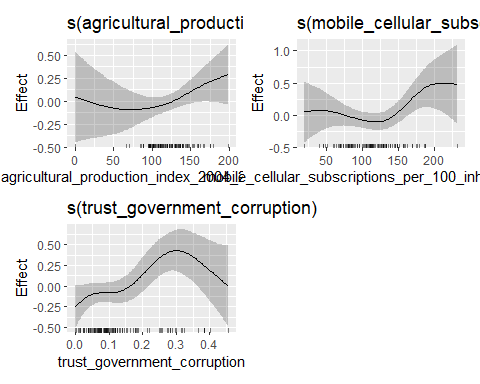
##   
## Family: gaussian   
## Link function: identity   
##   
## Formula:  
## migration\_growth ~ employment\_industry\_percent\_of\_employed +   
## unemployment\_percent\_of\_labour\_force + s(agricultural\_production\_index\_2004\_2006\_100) +   
## urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent +   
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) +   
## freedom + s(generosity) + s(trust\_government\_corruption)  
##   
## Parametric coefficients:  
## Estimate Std. Error  
## (Intercept) -0.7199828 0.3285206  
## employment\_industry\_percent\_of\_employed 0.0083226 0.0068167  
## unemployment\_percent\_of\_labour\_force -0.0062480 0.0081609  
## urban\_population\_percent\_of\_total\_population 0.0036233 0.0026270  
## health\_total\_expenditure\_percent\_of\_gdp 0.0174857 0.0188523  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.0011601 0.0013940  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.0004036 0.0020082  
## individuals\_using\_the\_internet\_per\_100\_inhabitants 0.0000881 0.0001148  
## pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 0.0007992 0.0019160  
## freedom 0.3103805 0.3772297  
## t value Pr(>|t|)   
## (Intercept) -2.192 0.031 \*  
## employment\_industry\_percent\_of\_employed 1.221 0.225   
## unemployment\_percent\_of\_labour\_force -0.766 0.446   
## urban\_population\_percent\_of\_total\_population 1.379 0.171   
## health\_total\_expenditure\_percent\_of\_gdp 0.928 0.356   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.832 0.408   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.201 0.841   
## individuals\_using\_the\_internet\_per\_100\_inhabitants 0.768 0.445   
## pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 0.417 0.678   
## freedom 0.823 0.413   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Approximate significance of smooth terms:  
## edf  
## s(agricultural\_production\_index\_2004\_2006\_100) 1.426e+00  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 3.678e+00  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 6.290e-10  
## s(generosity) 7.013e-11  
## s(trust\_government\_corruption) 3.603e+00  
## Ref.df F  
## s(agricultural\_production\_index\_2004\_2006\_100) 8 0.577  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 9 1.315  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 9 0.000  
## s(generosity) 9 0.000  
## s(trust\_government\_corruption) 9 1.801  
## p-value   
## s(agricultural\_production\_index\_2004\_2006\_100) 0.03357 \*   
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 0.02048 \*   
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 0.86836   
## s(generosity) 0.54235   
## s(trust\_government\_corruption) 0.00201 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.428 Deviance explained = 52.3%  
## GCV = 0.18657 Scale est. = 0.15395 n = 107

model\_gam\_migration <- fit\_gam(migration\_growth ~ 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 +  
 education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +  
 s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) + individuals\_using\_the\_internet\_per\_100\_inhabitants +  
 pop\_using\_improved\_drinking\_water\_urban\_rural\_percent +  
 freedom + s(trust\_government\_corruption), regression\_train %>% select(-natural\_growth))



##   
## Method: GCV Optimizer: magic  
## Smoothing parameter selection converged after 9 iterations.  
## The RMS GCV score gradient at convergence was 4.621315e-08 .  
## The Hessian was positive definite.  
## Model rank = 37 / 37   
##   
## 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(agricultural\_production\_index\_2004\_2006\_100) 9.00 1.43 1.04  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 9.00 3.43 1.07  
## s(trust\_government\_corruption) 9.00 3.49 1.21  
## p-value  
## s(agricultural\_production\_index\_2004\_2006\_100) 0.57  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 0.70  
## s(trust\_government\_corruption) 0.98

draw(model\_gam\_migration)



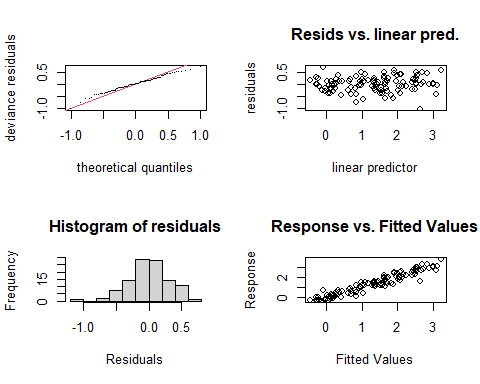
k.check(model\_gam\_migration)

## k' edf k-index  
## s(agricultural\_production\_index\_2004\_2006\_100) 9 1.432640 1.039590  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 9 3.425876 1.065412  
## s(trust\_government\_corruption) 9 3.489154 1.205108  
## p-value  
## s(agricultural\_production\_index\_2004\_2006\_100) 0.6175  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 0.7050  
## s(trust\_government\_corruption) 0.9675

summary(model\_gam\_migration)

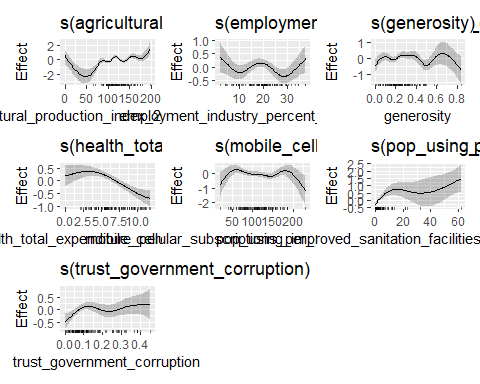
##   
## Family: gaussian   
## Link function: identity   
##   
## Formula:  
## migration\_growth ~ employment\_industry\_percent\_of\_employed +   
## unemployment\_percent\_of\_labour\_force + s(agricultural\_production\_index\_2004\_2006\_100) +   
## urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent +   
## freedom + s(trust\_government\_corruption)  
##   
## Parametric coefficients:  
## Estimate Std. Error  
## (Intercept) -7.330e-01 3.300e-01  
## employment\_industry\_percent\_of\_employed 8.266e-03 6.835e-03  
## unemployment\_percent\_of\_labour\_force -6.083e-03 8.199e-03  
## urban\_population\_percent\_of\_total\_population 3.692e-03 2.626e-03  
## health\_total\_expenditure\_percent\_of\_gdp 1.820e-02 1.893e-02  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 1.226e-03 1.400e-03  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 3.825e-04 2.019e-03  
## individuals\_using\_the\_internet\_per\_100\_inhabitants 8.927e-05 1.153e-04  
## pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 7.058e-04 1.923e-03  
## freedom 3.210e-01 3.791e-01  
## t value Pr(>|t|)   
## (Intercept) -2.221 0.0289 \*  
## employment\_industry\_percent\_of\_employed 1.209 0.2298   
## unemployment\_percent\_of\_labour\_force -0.742 0.4601   
## urban\_population\_percent\_of\_total\_population 1.406 0.1632   
## health\_total\_expenditure\_percent\_of\_gdp 0.962 0.3388   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.876 0.3837   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 0.189 0.8502   
## individuals\_using\_the\_internet\_per\_100\_inhabitants 0.774 0.4409   
## pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 0.367 0.7146   
## freedom 0.847 0.3993   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Approximate significance of smooth terms:  
## edf Ref.df F  
## s(agricultural\_production\_index\_2004\_2006\_100) 1.433 9 0.480  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 3.426 9 1.146  
## s(trust\_government\_corruption) 3.489 9 1.633  
## p-value   
## s(agricultural\_production\_index\_2004\_2006\_100) 0.04096 \*   
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 0.01649 \*   
## s(trust\_government\_corruption) 0.00252 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.422 Deviance explained = 51.6%  
## GCV = 0.1878 Scale est. = 0.1556 n = 107

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 + s(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 +  
 s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) + individuals\_using\_the\_internet\_per\_100\_inhabitants +  
 pop\_using\_improved\_drinking\_water\_urban\_rural\_percent + s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) +  
 freedom + s(generosity) + s(trust\_government\_corruption), regression\_train %>% select(-migration\_growth))



##   
## Method: GCV Optimizer: magic  
## Smoothing parameter selection converged after 100 iterations.  
## The RMS GCV score gradient at convergence was 1.486276e-08 .  
## The Hessian was positive definite.  
## Model rank = 71 / 71   
##   
## 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  
## s(employment\_industry\_percent\_of\_employed) 9.00 4.59  
## s(agricultural\_production\_index\_2004\_2006\_100) 9.00 7.93  
## s(health\_total\_expenditure\_percent\_of\_gdp) 9.00 1.98  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 9.00 5.58  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 9.00 2.99  
## s(generosity) 9.00 7.07  
## s(trust\_government\_corruption) 9.00 3.73  
## k-index p-value  
## s(employment\_industry\_percent\_of\_employed) 0.93 0.24  
## s(agricultural\_production\_index\_2004\_2006\_100) 1.10 0.85  
## s(health\_total\_expenditure\_percent\_of\_gdp) 1.04 0.64  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 1.10 0.78  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 1.15 0.93  
## s(generosity) 1.23 0.99  
## s(trust\_government\_corruption) 1.07 0.69

draw(model\_gam\_natural)



k.check(model\_gam\_natural)

## k' edf  
## s(employment\_industry\_percent\_of\_employed) 9 4.585510  
## s(agricultural\_production\_index\_2004\_2006\_100) 9 7.929461  
## s(health\_total\_expenditure\_percent\_of\_gdp) 9 1.982730  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 9 5.580437  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 9 2.994170  
## s(generosity) 9 7.071169  
## s(trust\_government\_corruption) 9 3.732111  
## k-index  
## s(employment\_industry\_percent\_of\_employed) 0.9346483  
## s(agricultural\_production\_index\_2004\_2006\_100) 1.1034773  
## s(health\_total\_expenditure\_percent\_of\_gdp) 1.0393657  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 1.1014404  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 1.1482162  
## s(generosity) 1.2274681  
## s(trust\_government\_corruption) 1.0685246  
## p-value  
## s(employment\_industry\_percent\_of\_employed) 0.1725  
## s(agricultural\_production\_index\_2004\_2006\_100) 0.8550  
## s(health\_total\_expenditure\_percent\_of\_gdp) 0.6400  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 0.8375  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 0.9350  
## s(generosity) 0.9875  
## s(trust\_government\_corruption) 0.7225

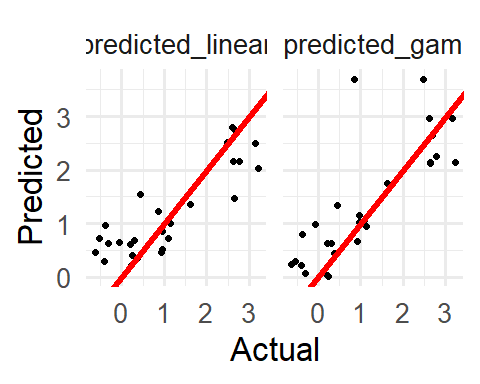
summary(model\_gam\_natural)

##   
## 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 + s(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 +   
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent +   
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) +   
## freedom + s(generosity) + s(trust\_government\_corruption)  
##   
## Parametric coefficients:  
## Estimate Std. Error  
## (Intercept) 1.841e+00 3.936e-01  
## unemployment\_percent\_of\_labour\_force 7.585e-03 9.857e-03  
## urban\_population\_percent\_of\_total\_population 6.996e-04 3.057e-03  
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 1.855e-03 1.714e-03  
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -5.493e-03 2.379e-03  
## individuals\_using\_the\_internet\_per\_100\_inhabitants -3.187e-05 1.318e-04  
## pop\_using\_improved\_drinking\_water\_urban\_rural\_percent -6.161e-03 2.529e-03  
## freedom -4.332e-01 4.793e-01  
## t value Pr(>|t|)   
## (Intercept) 4.677 1.52e-05 \*\*\*  
## unemployment\_percent\_of\_labour\_force 0.769 0.4444   
## urban\_population\_percent\_of\_total\_population 0.229 0.8197   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 1.082 0.2830   
## education\_tertiary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop -2.309 0.0241 \*   
## individuals\_using\_the\_internet\_per\_100\_inhabitants -0.242 0.8097   
## pop\_using\_improved\_drinking\_water\_urban\_rural\_percent -2.436 0.0176 \*   
## freedom -0.904 0.3694   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Approximate significance of smooth terms:  
## edf Ref.df  
## s(employment\_industry\_percent\_of\_employed) 4.586 9  
## s(agricultural\_production\_index\_2004\_2006\_100) 7.929 9  
## s(health\_total\_expenditure\_percent\_of\_gdp) 1.983 9  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 5.580 9  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 2.994 9  
## s(generosity) 7.071 9  
## s(trust\_government\_corruption) 3.732 9  
## F p-value  
## s(employment\_industry\_percent\_of\_employed) 1.846 0.00289  
## s(agricultural\_production\_index\_2004\_2006\_100) 5.343 3.38e-06  
## s(health\_total\_expenditure\_percent\_of\_gdp) 3.433 5.14e-07  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) 1.814 0.00721  
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) 2.524 4.89e-05  
## s(generosity) 1.904 0.01716  
## s(trust\_government\_corruption) 1.489 0.00504  
##   
## s(employment\_industry\_percent\_of\_employed) \*\*   
## s(agricultural\_production\_index\_2004\_2006\_100) \*\*\*  
## s(health\_total\_expenditure\_percent\_of\_gdp) \*\*\*  
## s(mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40) \*\*   
## s(pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent) \*\*\*  
## s(generosity) \*   
## s(trust\_government\_corruption) \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.86 Deviance explained = 91.4%  
## GCV = 0.25218 Scale est. = 0.15349 n = 107

library(yardstick)  
  
  
regression\_eval <- function(column, model\_linear, model\_gam) {  
 AIC(model\_linear)  
 AIC(model\_gam)  
  
  
 regression\_test <- regression\_test %>%  
 mutate(  
 predicted\_linear = predict(model\_linear, regression\_test),  
 predicted\_gam = predict(model\_gam, regression\_test)  
 )  
  
 set <- metric\_set(rmse, mae)  
  
 print(set(regression\_test, {{ column }}, predicted\_linear))  
 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 = "Actual", y = "Predicted") +  
 theme\_minimal(base\_size = 25)  
}

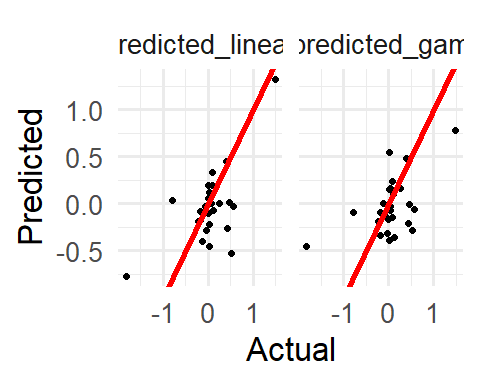
regression\_eval(natural\_growth, model\_linear\_natural, model\_gam\_natural)

## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.685  
## 2 mae standard 0.544  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.803  
## 2 mae standard 0.561



regression\_eval(migration\_growth, model\_linear\_migration, model\_gam\_migration)

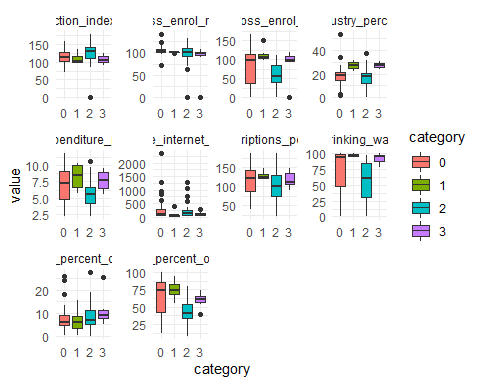
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.415  
## 2 mae standard 0.289  
## # A tibble: 2 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 0.452  
## 2 mae standard 0.322



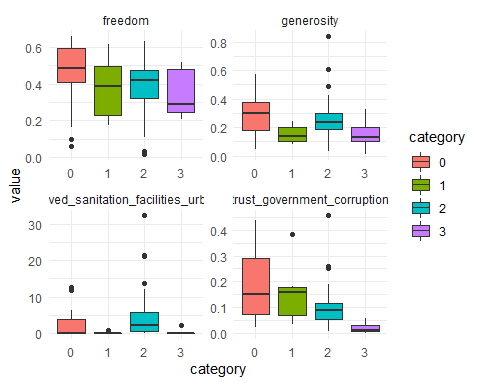
### Classification model

class\_recipe <- recipe(category ~ .,  
 data = classification\_train  
) %>%  
 step\_corr(all\_predictors(), threshold = 0.7) %>%  
 step\_nzv(all\_predictors())  
  
  
class\_recipe <- prep(class\_recipe, training = classification\_train)  
  
classification\_train <- bake(class\_recipe, classification\_train)

classification\_train %>%  
 select(1:10, category) %>%  
 pivot\_longer(-category) %>%  
 ggplot(aes(x = category, y = value, fill = category)) +  
 facet\_wrap(vars(name), scales = "free") +  
 geom\_boxplot() +  
 theme\_minimal()



classification\_train %>%  
 select(11: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()



library(nnet)  
model\_logistic <- nnet::multinom(category ~ ., data = classification\_train, trace = FALSE)

model\_logistic <- stats::step(model\_logistic, direction = "both")

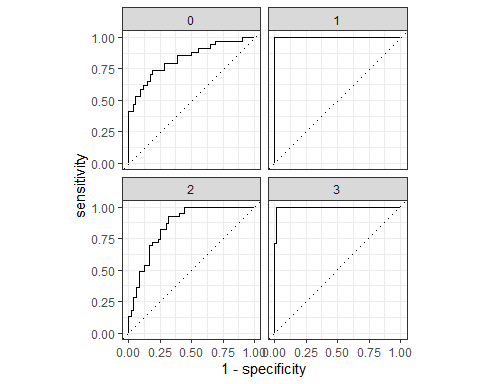
## Start: AIC=173.45  
## category ~ employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force +   
## agricultural\_production\_index\_2004\_2006\_100 + urban\_population\_percent\_of\_total\_population +   
## health\_total\_expenditure\_percent\_of\_gdp + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent +   
## pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +   
## freedom + generosity + trust\_government\_corruption  
##   
## trying - employment\_industry\_percent\_of\_employed   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.411414  
## iter 20 value 77.752722  
## iter 30 value 68.227103  
## iter 40 value 51.793556  
## iter 50 value 48.054899  
## iter 60 value 46.721507  
## iter 70 value 46.023232  
## iter 80 value 45.678438  
## iter 90 value 45.417844  
## iter 100 value 45.391668  
## final value 45.391668   
## stopped after 100 iterations  
## trying - unemployment\_percent\_of\_labour\_force   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.360297  
## iter 20 value 77.242295  
## iter 30 value 69.672274  
## iter 40 value 51.876384  
## iter 50 value 47.123335  
## iter 60 value 45.134458  
## iter 70 value 43.695579  
## iter 80 value 43.074360  
## iter 90 value 42.756337  
## iter 100 value 42.650354  
## final value 42.650354   
## stopped after 100 iterations  
## trying - agricultural\_production\_index\_2004\_2006\_100   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 84.724065  
## iter 20 value 77.822106  
## iter 30 value 67.680806  
## iter 40 value 50.982003  
## iter 50 value 46.859805  
## iter 60 value 45.449849  
## iter 70 value 44.209795  
## iter 80 value 42.742424  
## iter 90 value 42.099857  
## iter 100 value 42.000964  
## final value 42.000964   
## stopped after 100 iterations  
## trying - urban\_population\_percent\_of\_total\_population   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 84.707531  
## iter 20 value 78.978031  
## iter 30 value 69.815840  
## iter 40 value 52.726712  
## iter 50 value 48.848211  
## iter 60 value 46.892750  
## iter 70 value 45.916462  
## iter 80 value 44.430268  
## iter 90 value 43.853241  
## iter 100 value 43.738941  
## final value 43.738941   
## stopped after 100 iterations  
## trying - health\_total\_expenditure\_percent\_of\_gdp   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.002525  
## iter 20 value 77.231792  
## iter 30 value 69.680101  
## iter 40 value 53.112418  
## iter 50 value 50.179673  
## iter 60 value 49.053815  
## iter 70 value 47.817932  
## iter 80 value 46.869680  
## iter 90 value 46.369173  
## iter 100 value 46.119909  
## final value 46.119909   
## stopped after 100 iterations  
## trying - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 84.350667  
## iter 20 value 78.429938  
## iter 30 value 68.978182  
## iter 40 value 53.589707  
## iter 50 value 48.962664  
## iter 60 value 47.689434  
## iter 70 value 45.993488  
## iter 80 value 44.700791  
## iter 90 value 44.079373  
## iter 100 value 43.989691  
## final value 43.989691   
## stopped after 100 iterations  
## trying - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 83.179328  
## iter 20 value 75.971774  
## iter 30 value 66.341939  
## iter 40 value 51.908993  
## iter 50 value 48.813208  
## iter 60 value 47.841006  
## iter 70 value 47.002145  
## iter 80 value 46.265633  
## iter 90 value 45.894905  
## iter 100 value 45.820819  
## final value 45.820819   
## stopped after 100 iterations  
## trying - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.644665  
## iter 20 value 78.813456  
## iter 30 value 67.894699  
## iter 40 value 51.944394  
## iter 50 value 48.547716  
## iter 60 value 47.223355  
## iter 70 value 46.552588  
## iter 80 value 45.811990  
## iter 90 value 45.530377  
## iter 100 value 45.508891  
## final value 45.508891   
## stopped after 100 iterations  
## trying - individuals\_using\_the\_internet\_per\_100\_inhabitants   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 85.165129  
## iter 20 value 82.046953  
## iter 30 value 68.349419  
## iter 40 value 51.017320  
## iter 50 value 47.855720  
## iter 60 value 46.831572  
## iter 70 value 45.921098  
## iter 80 value 45.242080  
## iter 90 value 45.117708  
## iter 100 value 45.100575  
## final value 45.100575   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.725013  
## iter 20 value 75.918929  
## iter 30 value 68.240268  
## iter 40 value 50.713685  
## iter 50 value 46.783906  
## iter 60 value 45.004250  
## iter 70 value 43.853652  
## iter 80 value 42.078854  
## iter 90 value 41.705745  
## iter 100 value 41.503531  
## final value 41.503531   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.914422  
## iter 20 value 78.047132  
## iter 30 value 70.495218  
## iter 40 value 51.236087  
## iter 50 value 47.090688  
## iter 60 value 44.753027  
## iter 70 value 43.853243  
## iter 80 value 43.430578  
## iter 90 value 43.129940  
## iter 100 value 43.083196  
## final value 43.083196   
## stopped after 100 iterations  
## trying - freedom   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.056403  
## iter 20 value 75.779135  
## iter 30 value 70.697529  
## iter 40 value 53.984000  
## iter 50 value 50.003696  
## iter 60 value 49.395150  
## iter 70 value 49.061130  
## iter 80 value 48.841715  
## iter 90 value 48.747216  
## iter 100 value 48.730118  
## final value 48.730118   
## stopped after 100 iterations  
## trying - generosity   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.056844  
## iter 20 value 75.781707  
## iter 30 value 70.806771  
## iter 40 value 55.787721  
## iter 50 value 50.181791  
## iter 60 value 48.472658  
## iter 70 value 48.142095  
## iter 80 value 47.925850  
## iter 90 value 47.873939  
## iter 100 value 47.869403  
## final value 47.869403   
## stopped after 100 iterations  
## trying - trust\_government\_corruption   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.056953  
## iter 20 value 75.781650  
## iter 30 value 70.740618  
## iter 40 value 55.718221  
## iter 50 value 52.929259  
## iter 60 value 52.450807  
## iter 70 value 52.017568  
## iter 80 value 51.728650  
## iter 90 value 51.602550  
## iter 100 value 51.584463  
## final value 51.584463   
## stopped after 100 iterations  
## Df AIC  
## - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 42 167.0071  
## - agricultural\_production\_index\_2004\_2006\_100 42 168.0019  
## - unemployment\_percent\_of\_labour\_force 42 169.3007  
## - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent 42 170.1664  
## - urban\_population\_percent\_of\_total\_population 42 171.4779  
## - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 42 171.9794  
## <none> 45 173.4547  
## - individuals\_using\_the\_internet\_per\_100\_inhabitants 42 174.2012  
## - employment\_industry\_percent\_of\_employed 42 174.7833  
## - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 42 175.0178  
## - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 42 175.6416  
## - health\_total\_expenditure\_percent\_of\_gdp 42 176.2398  
## - generosity 42 179.7388  
## - freedom 42 181.4602  
## - trust\_government\_corruption 42 187.1689  
##   
## Step: AIC=167.01  
## category ~ 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\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +   
## freedom + generosity + trust\_government\_corruption  
##   
## trying - employment\_industry\_percent\_of\_employed   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 83.722997  
## iter 20 value 78.345253  
## iter 30 value 66.043816  
## iter 40 value 50.635496  
## iter 50 value 48.233559  
## iter 60 value 47.212346  
## iter 70 value 46.623261  
## iter 80 value 46.134793  
## iter 90 value 46.088691  
## iter 100 value 46.062466  
## final value 46.062466   
## stopped after 100 iterations  
## trying - unemployment\_percent\_of\_labour\_force   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 82.976243  
## iter 20 value 76.826518  
## iter 30 value 65.301824  
## iter 40 value 51.241440  
## iter 50 value 48.385060  
## iter 60 value 46.441741  
## iter 70 value 45.398436  
## iter 80 value 45.018739  
## iter 90 value 44.983064  
## iter 100 value 44.930404  
## final value 44.930404   
## stopped after 100 iterations  
## trying - agricultural\_production\_index\_2004\_2006\_100   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 90.346821  
## iter 20 value 81.741981  
## iter 30 value 64.251331  
## iter 40 value 49.187236  
## iter 50 value 46.257954  
## iter 60 value 45.136599  
## iter 70 value 44.288209  
## iter 80 value 43.647227  
## iter 90 value 43.595046  
## iter 100 value 43.489307  
## final value 43.489307   
## stopped after 100 iterations  
## trying - urban\_population\_percent\_of\_total\_population   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 88.295907  
## iter 20 value 79.786199  
## iter 30 value 64.192531  
## iter 40 value 51.983906  
## iter 50 value 48.855284  
## iter 60 value 47.700481  
## iter 70 value 46.475540  
## iter 80 value 45.970504  
## iter 90 value 45.877121  
## iter 100 value 45.775819  
## final value 45.775819   
## stopped after 100 iterations  
## trying - health\_total\_expenditure\_percent\_of\_gdp   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 82.819060  
## iter 20 value 77.908360  
## iter 30 value 68.048625  
## iter 40 value 54.427986  
## iter 50 value 52.811391  
## iter 60 value 52.107250  
## iter 70 value 51.724076  
## iter 80 value 51.438715  
## iter 90 value 51.418255  
## iter 100 value 51.379652  
## final value 51.379652   
## stopped after 100 iterations  
## trying - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 85.157052  
## iter 20 value 79.877984  
## iter 30 value 64.656703  
## iter 40 value 51.874505  
## iter 50 value 48.339382  
## iter 60 value 47.272045  
## iter 70 value 45.944511  
## iter 80 value 44.584042  
## iter 90 value 44.458919  
## iter 100 value 44.387648  
## final value 44.387648   
## stopped after 100 iterations  
## trying - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 84.460665  
## iter 20 value 77.900845  
## iter 30 value 65.198130  
## iter 40 value 51.202459  
## iter 50 value 49.045626  
## iter 60 value 48.491611  
## iter 70 value 47.493217  
## iter 80 value 46.626495  
## iter 90 value 46.573925  
## iter 100 value 46.523696  
## final value 46.523696   
## stopped after 100 iterations  
## trying - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 85.850910  
## iter 20 value 80.888109  
## iter 30 value 65.469036  
## iter 40 value 51.050583  
## iter 50 value 48.439386  
## iter 60 value 47.549708  
## iter 70 value 46.559648  
## iter 80 value 46.201316  
## iter 90 value 46.179808  
## iter 100 value 46.165455  
## final value 46.165455   
## stopped after 100 iterations  
## trying - individuals\_using\_the\_internet\_per\_100\_inhabitants   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 88.464822  
## iter 20 value 85.001014  
## iter 30 value 64.897727  
## iter 40 value 50.213122  
## iter 50 value 47.666820  
## iter 60 value 46.585433  
## iter 70 value 45.851012  
## iter 80 value 45.553492  
## iter 90 value 45.521586  
## iter 100 value 45.454587  
## final value 45.454587   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 83.134257  
## iter 20 value 78.073186  
## iter 30 value 66.504721  
## iter 40 value 49.883489  
## iter 50 value 46.901165  
## iter 60 value 45.184150  
## iter 70 value 44.509502  
## iter 80 value 44.064557  
## iter 90 value 44.030667  
## iter 100 value 43.994210  
## final value 43.994210   
## stopped after 100 iterations  
## trying - freedom   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 82.725232  
## iter 20 value 75.925273  
## iter 30 value 68.652039  
## iter 40 value 52.339411  
## iter 50 value 49.959711  
## iter 60 value 49.437237  
## iter 70 value 49.162718  
## iter 80 value 48.999674  
## iter 90 value 48.986640  
## iter 100 value 48.981894  
## final value 48.981894   
## stopped after 100 iterations  
## trying - generosity   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 82.725311  
## iter 20 value 75.927728  
## iter 30 value 68.906014  
## iter 40 value 52.820990  
## iter 50 value 49.405410  
## iter 60 value 48.493443  
## iter 70 value 48.096099  
## iter 80 value 48.074087  
## iter 90 value 48.072939  
## iter 100 value 48.071508  
## final value 48.071508   
## stopped after 100 iterations  
## trying - trust\_government\_corruption   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 82.725424  
## iter 20 value 75.927505  
## iter 30 value 68.807711  
## iter 40 value 54.592777  
## iter 50 value 52.947687  
## iter 60 value 52.467898  
## iter 70 value 52.383853  
## iter 80 value 52.353564  
## iter 90 value 52.351830  
## iter 100 value 52.350603  
## final value 52.350603   
## stopped after 100 iterations  
## trying + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent   
## # weights: 64 (45 variable)  
## initial value 119.221315   
## iter 10 value 82.056669  
## iter 20 value 75.774662  
## iter 30 value 70.553132  
## iter 40 value 52.487549  
## iter 50 value 47.091392  
## iter 60 value 45.152812  
## iter 70 value 43.735974  
## iter 80 value 42.752797  
## iter 90 value 41.936116  
## iter 100 value 41.727360  
## final value 41.727360   
## stopped after 100 iterations  
## Df AIC  
## - agricultural\_production\_index\_2004\_2006\_100 39 164.9786  
## - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent 39 165.9884  
## - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 39 166.7753  
## <none> 42 167.0071  
## - unemployment\_percent\_of\_labour\_force 39 167.8608  
## - individuals\_using\_the\_internet\_per\_100\_inhabitants 39 168.9092  
## - urban\_population\_percent\_of\_total\_population 39 169.5516  
## - employment\_industry\_percent\_of\_employed 39 170.1249  
## - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 39 170.3309  
## - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 39 171.0474  
## + +pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 45 173.4547  
## - generosity 39 174.1430  
## - freedom 39 175.9638  
## - health\_total\_expenditure\_percent\_of\_gdp 39 180.7593  
## - trust\_government\_corruption 39 182.7012  
##   
## Step: AIC=164.98  
## category ~ employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force +   
## urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 + individuals\_using\_the\_internet\_per\_100\_inhabitants +   
## pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +   
## freedom + generosity + trust\_government\_corruption  
##   
## trying - employment\_industry\_percent\_of\_employed   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 87.827487  
## iter 20 value 84.322806  
## iter 30 value 57.518627  
## iter 40 value 50.264288  
## iter 50 value 48.766524  
## iter 60 value 47.899417  
## iter 70 value 47.165926  
## iter 80 value 47.034609  
## iter 90 value 46.998063  
## iter 100 value 46.845444  
## final value 46.845444   
## stopped after 100 iterations  
## trying - unemployment\_percent\_of\_labour\_force   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 91.033838  
## iter 20 value 83.183889  
## iter 30 value 57.504825  
## iter 40 value 50.026337  
## iter 50 value 48.167505  
## iter 60 value 46.959276  
## iter 70 value 46.357760  
## iter 80 value 46.072418  
## iter 90 value 46.016643  
## iter 100 value 45.668642  
## final value 45.668642   
## stopped after 100 iterations  
## trying - urban\_population\_percent\_of\_total\_population   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 91.194882  
## iter 20 value 83.562232  
## iter 30 value 59.521357  
## iter 40 value 51.420741  
## iter 50 value 48.529044  
## iter 60 value 47.803885  
## iter 70 value 46.406830  
## iter 80 value 45.830567  
## iter 90 value 45.777615  
## iter 100 value 45.589344  
## final value 45.589344   
## stopped after 100 iterations  
## trying - health\_total\_expenditure\_percent\_of\_gdp   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 90.448627  
## iter 20 value 83.091915  
## iter 30 value 60.232889  
## iter 40 value 54.611194  
## iter 50 value 53.065310  
## iter 60 value 52.365447  
## iter 70 value 52.177887  
## iter 80 value 52.028942  
## iter 90 value 52.011621  
## iter 100 value 51.917170  
## final value 51.917170   
## stopped after 100 iterations  
## trying - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 88.341254  
## iter 20 value 82.739179  
## iter 30 value 58.781357  
## iter 40 value 50.653368  
## iter 50 value 48.296787  
## iter 60 value 47.511549  
## iter 70 value 45.711570  
## iter 80 value 45.120890  
## iter 90 value 45.012492  
## iter 100 value 44.665803  
## final value 44.665803   
## stopped after 100 iterations  
## trying - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 88.785094  
## iter 20 value 81.067016  
## iter 30 value 57.681981  
## iter 40 value 50.369554  
## iter 50 value 49.099668  
## iter 60 value 48.477164  
## iter 70 value 47.712263  
## iter 80 value 47.505916  
## iter 90 value 47.477104  
## iter 100 value 47.301616  
## final value 47.301616   
## stopped after 100 iterations  
## trying - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 87.599257  
## iter 20 value 83.119008  
## iter 30 value 56.752646  
## iter 40 value 50.597008  
## iter 50 value 49.148376  
## iter 60 value 48.488079  
## iter 70 value 48.064552  
## iter 80 value 47.987343  
## iter 90 value 47.971257  
## iter 100 value 47.934780  
## final value 47.934780   
## stopped after 100 iterations  
## trying - individuals\_using\_the\_internet\_per\_100\_inhabitants   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 83.541738  
## iter 20 value 79.818251  
## iter 30 value 56.845092  
## iter 40 value 49.681524  
## iter 50 value 47.515757  
## iter 60 value 46.301308  
## iter 70 value 45.756706  
## iter 80 value 45.674721  
## iter 90 value 45.662608  
## iter 100 value 45.602194  
## final value 45.602194   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 91.114281  
## iter 20 value 85.222600  
## iter 30 value 56.288354  
## iter 40 value 49.028600  
## iter 50 value 46.801323  
## iter 60 value 46.089002  
## iter 70 value 45.727258  
## iter 80 value 45.448920  
## iter 90 value 45.341396  
## iter 100 value 45.217111  
## final value 45.217111   
## stopped after 100 iterations  
## trying - freedom   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 90.347684  
## iter 20 value 81.774841  
## iter 30 value 65.819725  
## iter 40 value 52.012722  
## iter 50 value 50.749258  
## iter 60 value 50.329767  
## iter 70 value 50.113813  
## iter 80 value 50.058780  
## iter 90 value 50.045950  
## iter 100 value 50.036624  
## final value 50.036624   
## stopped after 100 iterations  
## trying - generosity   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 90.347996  
## iter 20 value 81.769516  
## iter 30 value 64.929466  
## iter 40 value 51.957831  
## iter 50 value 49.388731  
## iter 60 value 48.886353  
## iter 70 value 48.581634  
## iter 80 value 48.535286  
## iter 90 value 48.533375  
## iter 100 value 48.528058  
## final value 48.528058   
## stopped after 100 iterations  
## trying - trust\_government\_corruption   
## # weights: 52 (36 variable)  
## initial value 119.221315   
## iter 10 value 90.347346  
## iter 20 value 81.755607  
## iter 30 value 65.554511  
## iter 40 value 55.086153  
## iter 50 value 54.169685  
## iter 60 value 53.842577  
## iter 70 value 53.690851  
## iter 80 value 53.667775  
## iter 90 value 53.665751  
## iter 100 value 53.663324  
## final value 53.663324   
## stopped after 100 iterations  
## trying + agricultural\_production\_index\_2004\_2006\_100   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 82.725013  
## iter 20 value 75.918929  
## iter 30 value 68.240268  
## iter 40 value 50.713685  
## iter 50 value 46.783906  
## iter 60 value 45.004250  
## iter 70 value 43.853652  
## iter 80 value 42.078854  
## iter 90 value 41.705745  
## iter 100 value 41.503531  
## final value 41.503531   
## stopped after 100 iterations  
## trying + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent   
## # weights: 60 (42 variable)  
## initial value 119.221315   
## iter 10 value 84.724065  
## iter 20 value 77.822106  
## iter 30 value 67.680806  
## iter 40 value 50.982003  
## iter 50 value 46.859805  
## iter 60 value 45.449849  
## iter 70 value 44.209795  
## iter 80 value 42.742424  
## iter 90 value 42.099857  
## iter 100 value 42.000964  
## final value 42.000964   
## stopped after 100 iterations  
## Df AIC  
## - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 36 161.3316  
## - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent 36 162.4342  
## - urban\_population\_percent\_of\_total\_population 36 163.1787  
## - individuals\_using\_the\_internet\_per\_100\_inhabitants 36 163.2044  
## - unemployment\_percent\_of\_labour\_force 36 163.3373  
## <none> 39 164.9786  
## - employment\_industry\_percent\_of\_employed 36 165.6909  
## - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 36 166.6032  
## + +agricultural\_production\_index\_2004\_2006\_100 42 167.0071  
## - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 36 167.8696  
## + +pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 42 168.0019  
## - generosity 36 169.0561  
## - freedom 36 172.0732  
## - health\_total\_expenditure\_percent\_of\_gdp 36 175.8343  
## - trust\_government\_corruption 36 179.3266  
##   
## Step: AIC=161.33  
## category ~ employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force +   
## urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +   
## education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +   
## freedom + generosity + trust\_government\_corruption  
##   
## trying - employment\_industry\_percent\_of\_employed   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 89.567343  
## iter 20 value 79.668835  
## iter 30 value 56.885414  
## iter 40 value 52.497371  
## iter 50 value 50.768870  
## iter 60 value 50.080569  
## iter 70 value 49.767864  
## iter 80 value 49.747885  
## iter 90 value 49.658955  
## iter 100 value 49.446562  
## final value 49.446562   
## stopped after 100 iterations  
## trying - unemployment\_percent\_of\_labour\_force   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 89.041084  
## iter 20 value 80.939086  
## iter 30 value 57.365047  
## iter 40 value 52.005849  
## iter 50 value 50.842791  
## iter 60 value 49.790940  
## iter 70 value 49.399847  
## iter 80 value 49.390091  
## iter 90 value 49.177563  
## iter 100 value 49.075827  
## final value 49.075827   
## stopped after 100 iterations  
## trying - urban\_population\_percent\_of\_total\_population   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 91.532647  
## iter 20 value 77.252178  
## iter 30 value 57.111215  
## iter 40 value 52.114793  
## iter 50 value 50.502538  
## iter 60 value 49.391933  
## iter 70 value 48.566118  
## iter 80 value 48.501441  
## iter 90 value 48.380233  
## iter 100 value 48.133370  
## final value 48.133370   
## stopped after 100 iterations  
## trying - health\_total\_expenditure\_percent\_of\_gdp   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 88.307440  
## iter 20 value 81.253197  
## iter 30 value 59.190531  
## iter 40 value 55.209091  
## iter 50 value 54.158612  
## iter 60 value 53.805309  
## iter 70 value 53.495384  
## iter 80 value 53.472452  
## iter 90 value 53.363510  
## iter 100 value 53.189283  
## final value 53.189283   
## stopped after 100 iterations  
## trying - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 84.011251  
## iter 20 value 71.592075  
## iter 30 value 55.732638  
## iter 40 value 51.866859  
## iter 50 value 50.980385  
## iter 60 value 49.890996  
## iter 70 value 49.580626  
## iter 80 value 49.572459  
## iter 90 value 49.499988  
## iter 100 value 49.435412  
## final value 49.435412   
## stopped after 100 iterations  
## trying - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 89.169639  
## iter 20 value 75.617170  
## iter 30 value 56.358229  
## iter 40 value 51.866153  
## iter 50 value 50.538110  
## iter 60 value 50.040054  
## iter 70 value 49.888867  
## iter 80 value 49.878641  
## iter 90 value 49.850977  
## iter 100 value 49.609061  
## final value 49.609061   
## stopped after 100 iterations  
## trying - individuals\_using\_the\_internet\_per\_100\_inhabitants   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 86.106759  
## iter 20 value 75.395150  
## iter 30 value 56.709496  
## iter 40 value 51.707277  
## iter 50 value 50.711069  
## iter 60 value 50.401339  
## iter 70 value 49.985081  
## iter 80 value 49.969470  
## iter 90 value 49.898386  
## iter 100 value 49.842050  
## final value 49.842050   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 92.700552  
## iter 20 value 80.606935  
## iter 30 value 56.418623  
## iter 40 value 50.914455  
## iter 50 value 49.354745  
## iter 60 value 49.062183  
## iter 70 value 48.905298  
## iter 80 value 48.885912  
## iter 90 value 48.856692  
## iter 100 value 48.762590  
## final value 48.762590   
## stopped after 100 iterations  
## trying - freedom   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 88.343022  
## iter 20 value 82.871608  
## iter 30 value 59.501726  
## iter 40 value 53.557547  
## iter 50 value 52.642775  
## iter 60 value 52.390091  
## iter 70 value 52.208038  
## iter 80 value 52.177313  
## iter 90 value 52.157838  
## iter 100 value 52.100256  
## final value 52.100256   
## stopped after 100 iterations  
## trying - generosity   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 88.343367  
## iter 20 value 82.840697  
## iter 30 value 59.815952  
## iter 40 value 52.696075  
## iter 50 value 51.258867  
## iter 60 value 50.998691  
## iter 70 value 50.718020  
## iter 80 value 50.693870  
## iter 90 value 50.665615  
## iter 100 value 50.616808  
## final value 50.616808   
## stopped after 100 iterations  
## trying - trust\_government\_corruption   
## # weights: 48 (33 variable)  
## initial value 119.221315   
## iter 10 value 88.342227  
## iter 20 value 82.800920  
## iter 30 value 61.669548  
## iter 40 value 58.358524  
## iter 50 value 57.409243  
## iter 60 value 57.226102  
## iter 70 value 57.178469  
## iter 80 value 57.177743  
## iter 90 value 57.176302  
## iter 100 value 57.169007  
## final value 57.169007   
## stopped after 100 iterations  
## trying + agricultural\_production\_index\_2004\_2006\_100   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 85.157052  
## iter 20 value 79.877984  
## iter 30 value 64.656703  
## iter 40 value 51.874505  
## iter 50 value 48.339382  
## iter 60 value 47.272045  
## iter 70 value 45.944511  
## iter 80 value 44.584042  
## iter 90 value 44.458920  
## iter 100 value 44.387648  
## final value 44.387648   
## stopped after 100 iterations  
## trying + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 90.346821  
## iter 20 value 81.741981  
## iter 30 value 64.251331  
## iter 40 value 49.187236  
## iter 50 value 46.257954  
## iter 60 value 45.136599  
## iter 70 value 44.288209  
## iter 80 value 43.647227  
## iter 90 value 43.595046  
## iter 100 value 43.489307  
## final value 43.489307   
## stopped after 100 iterations  
## trying + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent   
## # weights: 56 (39 variable)  
## initial value 119.221315   
## iter 10 value 86.875456  
## iter 20 value 80.285409  
## iter 30 value 65.453699  
## iter 40 value 52.257771  
## iter 50 value 48.881406  
## iter 60 value 47.945644  
## iter 70 value 46.624589  
## iter 80 value 45.428168  
## iter 90 value 45.334859  
## iter 100 value 44.983869  
## final value 44.983869   
## stopped after 100 iterations  
## Df AIC  
## <none> 36 161.3316  
## - urban\_population\_percent\_of\_total\_population 33 162.2667  
## - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent 33 163.5252  
## - unemployment\_percent\_of\_labour\_force 33 164.1517  
## - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 33 164.8708  
## - employment\_industry\_percent\_of\_employed 33 164.8931  
## + +education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 39 164.9786  
## - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 33 165.2181  
## - individuals\_using\_the\_internet\_per\_100\_inhabitants 33 165.6841  
## + +agricultural\_production\_index\_2004\_2006\_100 39 166.7753  
## - generosity 33 167.2336  
## + +pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 39 167.9677  
## - freedom 33 170.2005  
## - health\_total\_expenditure\_percent\_of\_gdp 33 172.3786  
## - trust\_government\_corruption 33 180.3380

plot(predictorEffects(model\_logistic))



eval\_classification <- function(model, classification\_train) {  
 df\_pred\_truth <- tibble(  
 predicted = factor(predict(model, classification\_train)),  
 truth = classification\_train$category  
 ) %>% cbind(as.data.frame(model$fitted.values))  
  
  
 classification\_metrics <- metric\_set(accuracy, mcc, f\_meas)  
  
  
 print(conf\_mat(df\_pred\_truth,  
 truth = truth,  
 estimate = predicted  
 ))  
  
 print(classification\_metrics(df\_pred\_truth,  
 truth = truth,  
 estimate = predicted  
 ))  
  
 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()  
}  
  
eval\_classification(model\_logistic, classification\_train)

## Truth  
## Prediction 0 1 2 3  
## 0 20 0 6 0  
## 1 0 6 0 0  
## 2 13 0 33 0  
## 3 1 0 0 7  
## # A tibble: 3 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 accuracy multiclass 0.767  
## 2 mcc multiclass 0.636  
## 3 f\_meas macro 0.844

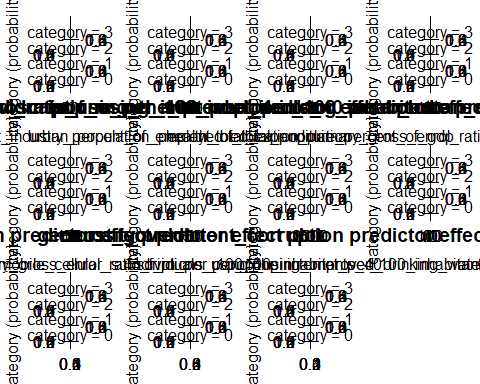


library(themis)  
  
class\_recipe2 <- recipe(category ~ .,  
 data = classification\_train  
) %>%  
 step\_corr(all\_predictors(), threshold = 0.7) %>%  
 step\_nzv(all\_predictors()) %>%  
 step\_smote(category)  
  
  
class\_recipe2 <- prep(class\_recipe2, training = classification\_train)  
  
classification\_train2 <- bake(class\_recipe2, NULL)

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

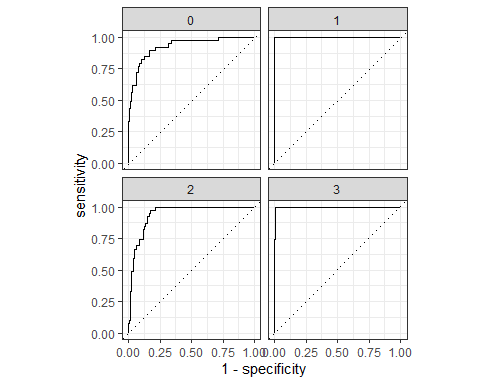
## Start: AIC=181.58  
## category ~ employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force +   
## agricultural\_production\_index\_2004\_2006\_100 + urban\_population\_percent\_of\_total\_population +   
## health\_total\_expenditure\_percent\_of\_gdp + education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent +   
## pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +   
## freedom + generosity + trust\_government\_corruption  
##   
## trying - employment\_industry\_percent\_of\_employed   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 158.589468  
## iter 20 value 147.682159  
## iter 30 value 131.466846  
## iter 40 value 67.133868  
## iter 50 value 57.311741  
## iter 60 value 54.746689  
## iter 70 value 53.495011  
## iter 80 value 52.537084  
## iter 90 value 52.171893  
## iter 100 value 52.141501  
## final value 52.141501   
## stopped after 100 iterations  
## trying - unemployment\_percent\_of\_labour\_force   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 158.776040  
## iter 20 value 141.544503  
## iter 30 value 134.119798  
## iter 40 value 66.558978  
## iter 50 value 56.639508  
## iter 60 value 53.680096  
## iter 70 value 49.987062  
## iter 80 value 48.219705  
## iter 90 value 47.643508  
## iter 100 value 47.514592  
## final value 47.514592   
## stopped after 100 iterations  
## trying - agricultural\_production\_index\_2004\_2006\_100   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 158.215518  
## iter 20 value 138.645668  
## iter 30 value 120.779362  
## iter 40 value 62.740196  
## iter 50 value 54.206825  
## iter 60 value 51.840731  
## iter 70 value 50.162992  
## iter 80 value 48.673380  
## iter 90 value 47.472260  
## iter 100 value 47.298269  
## final value 47.298269   
## stopped after 100 iterations  
## trying - urban\_population\_percent\_of\_total\_population   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 167.076548  
## iter 20 value 148.514149  
## iter 30 value 119.849895  
## iter 40 value 63.788025  
## iter 50 value 56.238393  
## iter 60 value 53.322884  
## iter 70 value 52.013426  
## iter 80 value 50.728090  
## iter 90 value 49.833833  
## iter 100 value 49.608807  
## final value 49.608807   
## stopped after 100 iterations  
## trying - health\_total\_expenditure\_percent\_of\_gdp   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 155.642510  
## iter 20 value 140.667126  
## iter 30 value 121.784573  
## iter 40 value 67.417017  
## iter 50 value 62.109485  
## iter 60 value 57.239842  
## iter 70 value 55.265143  
## iter 80 value 53.217402  
## iter 90 value 52.315550  
## iter 100 value 52.169566  
## final value 52.169566   
## stopped after 100 iterations  
## trying - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 159.194482  
## iter 20 value 140.886055  
## iter 30 value 126.616856  
## iter 40 value 68.541412  
## iter 50 value 57.602150  
## iter 60 value 55.331727  
## iter 70 value 53.932359  
## iter 80 value 51.242593  
## iter 90 value 50.168616  
## iter 100 value 49.956663  
## final value 49.956663   
## stopped after 100 iterations  
## trying - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 158.646310  
## iter 20 value 141.273116  
## iter 30 value 119.364260  
## iter 40 value 68.758351  
## iter 50 value 58.950730  
## iter 60 value 56.996269  
## iter 70 value 55.737484  
## iter 80 value 54.190660  
## iter 90 value 53.728568  
## iter 100 value 53.656762  
## final value 53.656762   
## stopped after 100 iterations  
## trying - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 169.763364  
## iter 20 value 145.853727  
## iter 30 value 118.548374  
## iter 40 value 67.028474  
## iter 50 value 59.256265  
## iter 60 value 55.857963  
## iter 70 value 54.635140  
## iter 80 value 53.960256  
## iter 90 value 53.691931  
## iter 100 value 53.664395  
## final value 53.664395   
## stopped after 100 iterations  
## trying - individuals\_using\_the\_internet\_per\_100\_inhabitants   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 137.200705  
## iter 20 value 130.655488  
## iter 30 value 109.819350  
## iter 40 value 63.595893  
## iter 50 value 56.294284  
## iter 60 value 54.133996  
## iter 70 value 52.283568  
## iter 80 value 51.382965  
## iter 90 value 51.002559  
## iter 100 value 50.884236  
## final value 50.884236   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 160.686507  
## iter 20 value 138.366754  
## iter 30 value 116.300712  
## iter 40 value 62.442250  
## iter 50 value 53.370068  
## iter 60 value 50.812675  
## iter 70 value 49.115056  
## iter 80 value 48.028661  
## iter 90 value 47.498206  
## iter 100 value 47.368473  
## final value 47.368473   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 155.891793  
## iter 20 value 142.177802  
## iter 30 value 125.828484  
## iter 40 value 63.543873  
## iter 50 value 54.853272  
## iter 60 value 51.338545  
## iter 70 value 49.627869  
## iter 80 value 48.896194  
## iter 90 value 48.556437  
## iter 100 value 48.498693  
## final value 48.498693   
## stopped after 100 iterations  
## trying - freedom   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 155.512357  
## iter 20 value 139.414901  
## iter 30 value 132.712944  
## iter 40 value 76.223281  
## iter 50 value 63.879437  
## iter 60 value 62.481040  
## iter 70 value 61.685093  
## iter 80 value 61.208159  
## iter 90 value 60.951962  
## iter 100 value 60.902729  
## final value 60.902729   
## stopped after 100 iterations  
## trying - generosity   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 155.512567  
## iter 20 value 139.420113  
## iter 30 value 132.431118  
## iter 40 value 76.822461  
## iter 50 value 64.328368  
## iter 60 value 62.222617  
## iter 70 value 60.928692  
## iter 80 value 60.132813  
## iter 90 value 59.902681  
## iter 100 value 59.885578  
## final value 59.885578   
## stopped after 100 iterations  
## trying - trust\_government\_corruption   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 155.512889  
## iter 20 value 139.428103  
## iter 30 value 132.788196  
## iter 40 value 83.114261  
## iter 50 value 73.049811  
## iter 60 value 71.211220  
## iter 70 value 69.622972  
## iter 80 value 69.234382  
## iter 90 value 68.894906  
## iter 100 value 68.872064  
## final value 68.872064   
## stopped after 100 iterations  
## Df AIC  
## - agricultural\_production\_index\_2004\_2006\_100 42 178.5965  
## - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 42 178.7369  
## - unemployment\_percent\_of\_labour\_force 42 179.0292  
## - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent 42 180.9974  
## <none> 45 181.5753  
## - urban\_population\_percent\_of\_total\_population 42 183.2176  
## - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 42 183.9133  
## - individuals\_using\_the\_internet\_per\_100\_inhabitants 42 185.7685  
## - employment\_industry\_percent\_of\_employed 42 188.2830  
## - health\_total\_expenditure\_percent\_of\_gdp 42 188.3391  
## - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 42 191.3135  
## - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 42 191.3288  
## - generosity 42 203.7712  
## - freedom 42 205.8055  
## - trust\_government\_corruption 42 221.7441  
##   
## Step: AIC=178.6  
## category ~ employment\_industry\_percent\_of\_employed + unemployment\_percent\_of\_labour\_force +   
## urban\_population\_percent\_of\_total\_population + health\_total\_expenditure\_percent\_of\_gdp +   
## education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop +   
## mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 + individuals\_using\_the\_internet\_per\_100\_inhabitants +   
## pop\_using\_improved\_drinking\_water\_urban\_rural\_percent + pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +   
## freedom + generosity + trust\_government\_corruption  
##   
## trying - employment\_industry\_percent\_of\_employed   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 160.871481  
## iter 20 value 149.119007  
## iter 30 value 107.931667  
## iter 40 value 64.001749  
## iter 50 value 58.639864  
## iter 60 value 55.887152  
## iter 70 value 54.763152  
## iter 80 value 53.820477  
## iter 90 value 53.694042  
## iter 100 value 53.605133  
## final value 53.605133   
## stopped after 100 iterations  
## trying - unemployment\_percent\_of\_labour\_force   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 162.622202  
## iter 20 value 146.011451  
## iter 30 value 98.802152  
## iter 40 value 60.921714  
## iter 50 value 54.719910  
## iter 60 value 52.709053  
## iter 70 value 50.893086  
## iter 80 value 49.546310  
## iter 90 value 49.365589  
## iter 100 value 49.001001  
## final value 49.001001   
## stopped after 100 iterations  
## trying - urban\_population\_percent\_of\_total\_population   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 165.455644  
## iter 20 value 143.080826  
## iter 30 value 104.719191  
## iter 40 value 62.423142  
## iter 50 value 56.108413  
## iter 60 value 54.192761  
## iter 70 value 52.992349  
## iter 80 value 51.676033  
## iter 90 value 51.526440  
## iter 100 value 51.372265  
## final value 51.372265   
## stopped after 100 iterations  
## trying - health\_total\_expenditure\_percent\_of\_gdp   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 158.288753  
## iter 20 value 140.705274  
## iter 30 value 105.208351  
## iter 40 value 68.685776  
## iter 50 value 64.429485  
## iter 60 value 61.583647  
## iter 70 value 60.143362  
## iter 80 value 58.930071  
## iter 90 value 58.817894  
## iter 100 value 58.658725  
## final value 58.658725   
## stopped after 100 iterations  
## trying - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 168.465723  
## iter 20 value 147.550955  
## iter 30 value 99.162577  
## iter 40 value 62.159454  
## iter 50 value 57.006484  
## iter 60 value 55.656765  
## iter 70 value 54.122632  
## iter 80 value 52.075559  
## iter 90 value 51.902574  
## iter 100 value 51.452472  
## final value 51.452472   
## stopped after 100 iterations  
## trying - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 167.452022  
## iter 20 value 146.498538  
## iter 30 value 101.673073  
## iter 40 value 64.447821  
## iter 50 value 59.528341  
## iter 60 value 58.622792  
## iter 70 value 57.450273  
## iter 80 value 56.615956  
## iter 90 value 56.534242  
## iter 100 value 56.400931  
## final value 56.400931   
## stopped after 100 iterations  
## trying - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 170.885001  
## iter 20 value 154.805068  
## iter 30 value 109.912133  
## iter 40 value 63.952381  
## iter 50 value 58.277361  
## iter 60 value 56.046513  
## iter 70 value 55.333249  
## iter 80 value 54.807712  
## iter 90 value 54.773842  
## iter 100 value 54.737862  
## final value 54.737862   
## stopped after 100 iterations  
## trying - individuals\_using\_the\_internet\_per\_100\_inhabitants   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 164.175463  
## iter 20 value 156.655560  
## iter 30 value 97.741619  
## iter 40 value 62.742158  
## iter 50 value 56.453392  
## iter 60 value 54.099359  
## iter 70 value 52.719627  
## iter 80 value 52.388588  
## iter 90 value 52.328148  
## iter 100 value 52.259391  
## final value 52.259391   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 164.278712  
## iter 20 value 145.858073  
## iter 30 value 100.179771  
## iter 40 value 58.926798  
## iter 50 value 53.348909  
## iter 60 value 51.654118  
## iter 70 value 50.447693  
## iter 80 value 49.741564  
## iter 90 value 49.664176  
## iter 100 value 49.450582  
## final value 49.450582   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 159.325009  
## iter 20 value 143.480811  
## iter 30 value 99.394309  
## iter 40 value 61.239035  
## iter 50 value 54.844929  
## iter 60 value 52.256561  
## iter 70 value 50.839603  
## iter 80 value 50.174213  
## iter 90 value 50.116286  
## iter 100 value 50.084772  
## final value 50.084772   
## stopped after 100 iterations  
## trying - freedom   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 158.217773  
## iter 20 value 138.699414  
## iter 30 value 124.365921  
## iter 40 value 71.547039  
## iter 50 value 65.419019  
## iter 60 value 64.160632  
## iter 70 value 63.657890  
## iter 80 value 63.327244  
## iter 90 value 63.291906  
## iter 100 value 63.275826  
## final value 63.275826   
## stopped after 100 iterations  
## trying - generosity   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 158.218333  
## iter 20 value 138.699637  
## iter 30 value 124.591963  
## iter 40 value 68.627984  
## iter 50 value 64.006528  
## iter 60 value 62.601463  
## iter 70 value 61.554174  
## iter 80 value 61.141251  
## iter 90 value 61.104706  
## iter 100 value 61.052608  
## final value 61.052608   
## stopped after 100 iterations  
## trying - trust\_government\_corruption   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 158.217581  
## iter 20 value 138.693357  
## iter 30 value 124.170623  
## iter 40 value 78.545340  
## iter 50 value 73.420060  
## iter 60 value 71.902325  
## iter 70 value 70.744065  
## iter 80 value 70.407107  
## iter 90 value 70.381891  
## iter 100 value 70.359130  
## final value 70.359130   
## stopped after 100 iterations  
## trying + agricultural\_production\_index\_2004\_2006\_100   
## # weights: 64 (45 variable)  
## initial value 216.261920   
## iter 10 value 155.511079  
## iter 20 value 139.388079  
## iter 30 value 132.243715  
## iter 40 value 71.086194  
## iter 50 value 55.677290  
## iter 60 value 51.749992  
## iter 70 value 49.855825  
## iter 80 value 48.324713  
## iter 90 value 46.060452  
## iter 100 value 45.787645  
## final value 45.787645   
## stopped after 100 iterations  
## Df AIC  
## - unemployment\_percent\_of\_labour\_force 39 176.0020  
## - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 39 176.9012  
## - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent 39 178.1695  
## <none> 42 178.5965  
## - urban\_population\_percent\_of\_total\_population 39 180.7445  
## - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 39 180.9049  
## + +agricultural\_production\_index\_2004\_2006\_100 45 181.5753  
## - individuals\_using\_the\_internet\_per\_100\_inhabitants 39 182.5188  
## - employment\_industry\_percent\_of\_employed 39 185.2103  
## - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 39 187.4757  
## - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 39 190.8019  
## - health\_total\_expenditure\_percent\_of\_gdp 39 195.3175  
## - generosity 39 200.1052  
## - freedom 39 204.5517  
## - trust\_government\_corruption 39 218.7183  
##   
## Step: AIC=176  
## 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\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent +   
## pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent +   
## freedom + generosity + trust\_government\_corruption  
##   
## trying - employment\_industry\_percent\_of\_employed   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 165.208045  
## iter 20 value 158.451907  
## iter 30 value 78.002652  
## iter 40 value 60.383640  
## iter 50 value 56.844136  
## iter 60 value 54.770120  
## iter 70 value 54.122860  
## iter 80 value 53.955308  
## iter 90 value 53.918623  
## iter 100 value 53.772499  
## final value 53.772499   
## stopped after 100 iterations  
## trying - urban\_population\_percent\_of\_total\_population   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 170.291244  
## iter 20 value 154.423259  
## iter 30 value 77.221775  
## iter 40 value 60.300367  
## iter 50 value 55.959480  
## iter 60 value 54.300053  
## iter 70 value 53.205019  
## iter 80 value 52.741618  
## iter 90 value 52.493326  
## iter 100 value 52.254039  
## final value 52.254039   
## stopped after 100 iterations  
## trying - health\_total\_expenditure\_percent\_of\_gdp   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 162.744175  
## iter 20 value 148.175084  
## iter 30 value 81.131543  
## iter 40 value 65.488032  
## iter 50 value 61.977847  
## iter 60 value 60.977700  
## iter 70 value 59.469664  
## iter 80 value 58.558707  
## iter 90 value 58.391235  
## iter 100 value 58.157021  
## final value 58.157021   
## stopped after 100 iterations  
## trying - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 172.034016  
## iter 20 value 153.267118  
## iter 30 value 86.123881  
## iter 40 value 63.752290  
## iter 50 value 58.904642  
## iter 60 value 55.786851  
## iter 70 value 54.752364  
## iter 80 value 54.164734  
## iter 90 value 54.010739  
## iter 100 value 53.725061  
## final value 53.725061   
## stopped after 100 iterations  
## trying - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 170.154075  
## iter 20 value 153.373690  
## iter 30 value 83.531095  
## iter 40 value 62.568405  
## iter 50 value 59.453629  
## iter 60 value 57.744694  
## iter 70 value 55.484050  
## iter 80 value 55.252514  
## iter 90 value 55.174915  
## iter 100 value 55.028137  
## final value 55.028137   
## stopped after 100 iterations  
## trying - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 174.609729  
## iter 20 value 162.590566  
## iter 30 value 86.991911  
## iter 40 value 64.219236  
## iter 50 value 61.191251  
## iter 60 value 58.744290  
## iter 70 value 57.268372  
## iter 80 value 56.964197  
## iter 90 value 56.807119  
## iter 100 value 56.605139  
## final value 56.605139   
## stopped after 100 iterations  
## trying - individuals\_using\_the\_internet\_per\_100\_inhabitants   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 172.061205  
## iter 20 value 164.265415  
## iter 30 value 86.660086  
## iter 40 value 64.845006  
## iter 50 value 58.942212  
## iter 60 value 55.416609  
## iter 70 value 54.181712  
## iter 80 value 53.951861  
## iter 90 value 53.802091  
## iter 100 value 53.484842  
## final value 53.484842   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 168.998150  
## iter 20 value 151.253075  
## iter 30 value 79.319050  
## iter 40 value 62.002559  
## iter 50 value 58.830390  
## iter 60 value 56.370272  
## iter 70 value 54.909774  
## iter 80 value 54.341239  
## iter 90 value 54.220175  
## iter 100 value 52.752152  
## final value 52.752152   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 163.720838  
## iter 20 value 150.662005  
## iter 30 value 76.514075  
## iter 40 value 60.127542  
## iter 50 value 56.608963  
## iter 60 value 54.950181  
## iter 70 value 52.833489  
## iter 80 value 51.801384  
## iter 90 value 51.544747  
## iter 100 value 51.123919  
## final value 51.123919   
## stopped after 100 iterations  
## trying - freedom   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 162.624432  
## iter 20 value 146.057352  
## iter 30 value 118.505539  
## iter 40 value 73.663003  
## iter 50 value 70.331016  
## iter 60 value 69.394171  
## iter 70 value 68.842036  
## iter 80 value 68.696396  
## iter 90 value 68.679043  
## iter 100 value 68.661489  
## final value 68.661489   
## stopped after 100 iterations  
## trying - generosity   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 162.625017  
## iter 20 value 146.059780  
## iter 30 value 113.371904  
## iter 40 value 71.002520  
## iter 50 value 65.705909  
## iter 60 value 64.755190  
## iter 70 value 64.335051  
## iter 80 value 64.300711  
## iter 90 value 64.292989  
## iter 100 value 64.265352  
## final value 64.265352   
## stopped after 100 iterations  
## trying - trust\_government\_corruption   
## # weights: 52 (36 variable)  
## initial value 216.261920   
## iter 10 value 162.624302  
## iter 20 value 146.062173  
## iter 30 value 122.990002  
## iter 40 value 81.084090  
## iter 50 value 76.922711  
## iter 60 value 75.617391  
## iter 70 value 74.494424  
## iter 80 value 74.247306  
## iter 90 value 74.199767  
## iter 100 value 74.111892  
## final value 74.111892   
## stopped after 100 iterations  
## trying + unemployment\_percent\_of\_labour\_force   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 158.215518  
## iter 20 value 138.645668  
## iter 30 value 120.779362  
## iter 40 value 62.740196  
## iter 50 value 54.206825  
## iter 60 value 51.840731  
## iter 70 value 50.162992  
## iter 80 value 48.673380  
## iter 90 value 47.472260  
## iter 100 value 47.298269  
## final value 47.298269   
## stopped after 100 iterations  
## trying + agricultural\_production\_index\_2004\_2006\_100   
## # weights: 60 (42 variable)  
## initial value 216.261920   
## iter 10 value 158.776040  
## iter 20 value 141.544503  
## iter 30 value 134.119798  
## iter 40 value 66.558978  
## iter 50 value 56.639508  
## iter 60 value 53.680096  
## iter 70 value 49.987062  
## iter 80 value 48.219705  
## iter 90 value 47.643508  
## iter 100 value 47.514592  
## final value 47.514592   
## stopped after 100 iterations  
## Df AIC  
## - pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent 36 174.2478  
## <none> 39 176.0020  
## - urban\_population\_percent\_of\_total\_population 36 176.5081  
## - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 36 177.5043  
## + +unemployment\_percent\_of\_labour\_force 42 178.5965  
## - individuals\_using\_the\_internet\_per\_100\_inhabitants 36 178.9697  
## + +agricultural\_production\_index\_2004\_2006\_100 42 179.0292  
## - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 36 179.4501  
## - employment\_industry\_percent\_of\_employed 36 179.5450  
## - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 36 182.0563  
## - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 36 185.2103  
## - health\_total\_expenditure\_percent\_of\_gdp 36 188.3140  
## - generosity 36 200.5307  
## - freedom 36 209.3230  
## - trust\_government\_corruption 36 220.2238  
##   
## Step: AIC=174.25  
## 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\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop + mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 +   
## individuals\_using\_the\_internet\_per\_100\_inhabitants + pop\_using\_improved\_drinking\_water\_urban\_rural\_percent +   
## freedom + generosity + trust\_government\_corruption  
##   
## trying - employment\_industry\_percent\_of\_employed   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 172.075261  
## iter 20 value 166.477125  
## iter 30 value 75.661939  
## iter 40 value 67.955156  
## iter 50 value 64.671360  
## iter 60 value 61.689360  
## iter 70 value 60.700475  
## iter 80 value 60.667765  
## iter 90 value 60.572777  
## iter 100 value 60.473751  
## final value 60.473751   
## stopped after 100 iterations  
## trying - urban\_population\_percent\_of\_total\_population   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 171.350383  
## iter 20 value 159.219618  
## iter 30 value 70.297502  
## iter 40 value 60.567260  
## iter 50 value 58.609006  
## iter 60 value 56.689917  
## iter 70 value 55.202526  
## iter 80 value 55.130260  
## iter 90 value 54.980902  
## iter 100 value 54.461486  
## final value 54.461486   
## stopped after 100 iterations  
## trying - health\_total\_expenditure\_percent\_of\_gdp   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 163.834721  
## iter 20 value 151.910719  
## iter 30 value 72.588938  
## iter 40 value 66.630275  
## iter 50 value 63.885256  
## iter 60 value 62.649340  
## iter 70 value 61.962587  
## iter 80 value 61.899489  
## iter 90 value 61.845234  
## iter 100 value 61.432204  
## final value 61.432204   
## stopped after 100 iterations  
## trying - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 176.131631  
## iter 20 value 161.427610  
## iter 30 value 75.761969  
## iter 40 value 66.062823  
## iter 50 value 63.549697  
## iter 60 value 61.315387  
## iter 70 value 59.599981  
## iter 80 value 59.400133  
## iter 90 value 59.154029  
## iter 100 value 58.981088  
## final value 58.981088   
## stopped after 100 iterations  
## trying - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 171.856846  
## iter 20 value 160.997612  
## iter 30 value 77.450980  
## iter 40 value 66.420153  
## iter 50 value 64.891157  
## iter 60 value 64.345283  
## iter 70 value 64.035252  
## iter 80 value 64.019291  
## iter 90 value 63.988798  
## iter 100 value 63.832119  
## final value 63.832119   
## stopped after 100 iterations  
## trying - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 175.783876  
## iter 20 value 161.589249  
## iter 30 value 72.714135  
## iter 40 value 63.336102  
## iter 50 value 61.100891  
## iter 60 value 60.023866  
## iter 70 value 59.349591  
## iter 80 value 59.257196  
## iter 90 value 59.100739  
## iter 100 value 58.833800  
## final value 58.833800   
## stopped after 100 iterations  
## trying - individuals\_using\_the\_internet\_per\_100\_inhabitants   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 173.370458  
## iter 20 value 148.674589  
## iter 30 value 72.643481  
## iter 40 value 62.888408  
## iter 50 value 58.742897  
## iter 60 value 56.449297  
## iter 70 value 55.894804  
## iter 80 value 55.812925  
## iter 90 value 55.491997  
## iter 100 value 54.933531  
## final value 54.933531   
## stopped after 100 iterations  
## trying - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 170.141891  
## iter 20 value 155.863375  
## iter 30 value 71.036961  
## iter 40 value 61.630984  
## iter 50 value 59.437622  
## iter 60 value 58.445488  
## iter 70 value 57.451307  
## iter 80 value 57.357867  
## iter 90 value 57.251932  
## iter 100 value 56.974585  
## final value 56.974585   
## stopped after 100 iterations  
## trying - freedom   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 163.723051  
## iter 20 value 150.645477  
## iter 30 value 85.499143  
## iter 40 value 71.873425  
## iter 50 value 70.577975  
## iter 60 value 69.838501  
## iter 70 value 69.452466  
## iter 80 value 69.424215  
## iter 90 value 69.376317  
## iter 100 value 69.217048  
## final value 69.217048   
## stopped after 100 iterations  
## trying - generosity   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 163.723755  
## iter 20 value 150.666917  
## iter 30 value 86.120089  
## iter 40 value 71.550045  
## iter 50 value 69.649893  
## iter 60 value 68.218202  
## iter 70 value 68.113181  
## iter 80 value 68.104007  
## iter 90 value 68.100827  
## iter 100 value 68.050144  
## final value 68.050144   
## stopped after 100 iterations  
## trying - trust\_government\_corruption   
## # weights: 48 (33 variable)  
## initial value 216.261920   
## iter 10 value 163.722918  
## iter 20 value 150.648046  
## iter 30 value 96.021338  
## iter 40 value 84.176071  
## iter 50 value 80.106167  
## iter 60 value 78.460431  
## iter 70 value 77.760242  
## iter 80 value 77.726764  
## iter 90 value 77.697285  
## iter 100 value 77.449845  
## final value 77.449845   
## stopped after 100 iterations  
## trying + unemployment\_percent\_of\_labour\_force   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 159.325009  
## iter 20 value 143.480811  
## iter 30 value 99.394309  
## iter 40 value 61.239035  
## iter 50 value 54.844929  
## iter 60 value 52.256561  
## iter 70 value 50.839603  
## iter 80 value 50.174213  
## iter 90 value 50.116286  
## iter 100 value 50.084772  
## final value 50.084772   
## stopped after 100 iterations  
## trying + agricultural\_production\_index\_2004\_2006\_100   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 159.143702  
## iter 20 value 144.004710  
## iter 30 value 105.861176  
## iter 40 value 62.390431  
## iter 50 value 55.826353  
## iter 60 value 53.676539  
## iter 70 value 51.034610  
## iter 80 value 49.459146  
## iter 90 value 49.076004  
## iter 100 value 48.947243  
## final value 48.947243   
## stopped after 100 iterations  
## trying + pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent   
## # weights: 56 (39 variable)  
## initial value 216.261920   
## iter 10 value 162.622202  
## iter 20 value 146.011451  
## iter 30 value 98.802152  
## iter 40 value 60.921714  
## iter 50 value 54.719910  
## iter 60 value 52.709053  
## iter 70 value 50.893086  
## iter 80 value 49.546310  
## iter 90 value 49.365589  
## iter 100 value 49.001001  
## final value 49.001001   
## stopped after 100 iterations  
## Df AIC  
## <none> 36 174.2478  
## - urban\_population\_percent\_of\_total\_population 33 174.9230  
## - individuals\_using\_the\_internet\_per\_100\_inhabitants 33 175.8671  
## + +agricultural\_production\_index\_2004\_2006\_100 39 175.8945  
## + +pop\_using\_improved\_sanitation\_facilities\_urban\_rural\_percent 39 176.0020  
## + +unemployment\_percent\_of\_labour\_force 39 178.1695  
## - pop\_using\_improved\_drinking\_water\_urban\_rural\_percent 33 179.9492  
## - mobile\_cellular\_subscriptions\_per\_100\_inhabitants\_40 33 183.6676  
## - education\_primary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 33 183.9622  
## - employment\_industry\_percent\_of\_employed 33 186.9475  
## - health\_total\_expenditure\_percent\_of\_gdp 33 188.8644  
## - education\_secondary\_gross\_enrol\_ratio\_f\_m\_per\_100\_pop 33 193.6642  
## - generosity 33 202.1003  
## - freedom 33 204.4341  
## - trust\_government\_corruption 33 220.8997

plot(predictorEffects(model\_logistic2))



eval\_classification(model\_logistic2, classification\_train2)

## Truth  
## Prediction 0 1 2 3  
## 0 26 0 9 0  
## 1 0 39 0 0  
## 2 12 0 29 0  
## 3 1 0 1 39  
## # A tibble: 3 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 accuracy multiclass 0.853  
## 2 mcc multiclass 0.804  
## 3 f\_meas macro 0.851



class\_predictions <- function() {  
 tibble(  
 migration\_growth = predict(model\_linear\_migration, classification\_test),  
 natural\_growth = predict(model\_linear\_natural, classification\_test)  
 ) %>%  
 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  
 ))) %>%  
 pull(category) %>%  
 factor(levels = c(0, 1, 2, 3))  
}  
  
  
df\_pred\_truth <- tibble(  
 predicted\_1 =  
 factor(predict(model\_logistic, classification\_test)),  
 predicted\_2 =  
 factor(predict(model\_logistic2, classification\_test)),  
 predicted\_3 = class\_predictions(),  
 truth = classification\_test$category  
)  
  
  
classification\_metrics <- metric\_set(accuracy, mcc, f\_meas)  
  
  
conf\_mat(df\_pred\_truth,  
 truth = truth,  
 estimate = predicted\_1  
)

## Truth  
## Prediction 0 1 2 3  
## 0 9 2 6 0  
## 1 5 2 0 0  
## 2 6 0 14 0  
## 3 1 1 0 2

conf\_mat(df\_pred\_truth,  
 truth = truth,  
 estimate = predicted\_2  
)

## Truth  
## Prediction 0 1 2 3  
## 0 12 2 6 0  
## 1 4 2 1 1  
## 2 4 0 12 0  
## 3 1 1 1 1

conf\_mat(df\_pred\_truth,  
 truth = truth,  
 estimate = predicted\_3  
)

## Truth  
## Prediction 0 1 2 3  
## 0 15 2 3 1  
## 1 1 0 0 0  
## 2 5 2 17 1  
## 3 0 1 0 0

classification\_metrics(df\_pred\_truth, truth, estimate = predicted\_1)

## # A tibble: 3 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 accuracy multiclass 0.562  
## 2 mcc multiclass 0.333  
## 3 f\_meas macro 0.543

classification\_metrics(df\_pred\_truth, truth, estimate = predicted\_2)

## # A tibble: 3 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 accuracy multiclass 0.562  
## 2 mcc multiclass 0.339  
## 3 f\_meas macro 0.473

classification\_metrics(df\_pred\_truth, truth, estimate = predicted\_3)

## # A tibble: 3 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 accuracy multiclass 0.667  
## 2 mcc multiclass 0.442  
## 3 f\_meas macro 0.367