Machine Learning completo

Matheus Assis e Oliveira

2023-10-11

instalar pacotes para o estudo

# instalação dos Pacotes e dependencias

install.packages(“tidyr”, dependencies = TRUE) install.packages(“readxl”, dependencies = TRUE) install.packages(“plyr”, dependencies = TRUE) install.packages(“corrplot”, dependencies = TRUE) install.packages(“RColorBrewer”, dependencies = TRUE) install.packages(“ggplot2”, dependencies = TRUE) install.packages(“ggpubr”, dependencies = TRUE) install.packages(“plyr”, dependencies = TRUE) install.packages(“caret”, dependencies = TRUE) install.packages(“randomForest”, dependencies = TRUE) install.packages(“caTools”, dependencies = TRUE)

carregar pacotes para a Engenharia de dados.

seguindo começamos a importar os dados em arquivo excel: xlsx, e converter anos a coluna, limpar o X a frente dos anos e converter anos para numérico.

renda\_per\_capita <- read\_excel("renda.per.capita.xlsx")  
#usar gather para converter anos à coluna  
renda\_per\_capita <- gather(renda\_per\_capita, ano, renda\_per\_capita, X2012:X2021, convert = TRUE)  
#Limpar o X a frente do ano  
renda\_per\_capita$ano <- gsub('X', '', renda\_per\_capita$ano)  
#converter ano de character para numeric  
renda\_per\_capita <- transform(renda\_per\_capita, ano = as.numeric(ano))  
renda\_per\_capita <- transform(renda\_per\_capita, renda\_per\_capita = as.numeric(renda\_per\_capita))

em seguida repete para demais arquivos com as seguintes variáveis.

sub\_esco\_pop <- read\_excel("sub.esco.pop.xlsx")  
sub\_esco\_pop <- gather(sub\_esco\_pop, ano, sub\_esco\_pop, X2012:X2021, convert = TRUE)  
sub\_esco\_pop$ano <- gsub('X', '', sub\_esco\_pop$ano)  
sub\_esco\_pop <- transform(sub\_esco\_pop, ano = as.numeric(ano))  
sub\_esco\_pop <- transform(sub\_esco\_pop, sub\_esco\_pop = as.numeric(sub\_esco\_pop))  
  
  
  
sub\_freq\_esco <- read\_excel("sub.freq.esco.xlsx")  
sub\_freq\_esco <- gather(sub\_freq\_esco, ano, sub\_freq\_esco, X2012:X2021, convert = TRUE)  
sub\_freq\_esco$ano <- gsub('X', '', sub\_freq\_esco$ano)  
sub\_freq\_esco <- transform(sub\_freq\_esco, ano = as.numeric(ano))  
sub\_freq\_esco <- transform(sub\_freq\_esco, sub\_freq\_esco = as.numeric(sub\_freq\_esco))  
  
  
  
  
esperança\_de\_vida <- read\_excel("esperança.de.vida.xlsx")  
esperança\_de\_vida <- gather(esperança\_de\_vida, ano, esperança\_de\_vida, X2012:X2021, convert = TRUE)  
esperança\_de\_vida$ano <- gsub('X', '', esperança\_de\_vida$ano)  
esperança\_de\_vida <- transform(esperança\_de\_vida, ano = as.numeric(ano))  
esperança\_de\_vida <- transform(esperança\_de\_vida, esperança\_de\_vida = as.numeric(esperança\_de\_vida))  
  
  
  
porcent\_pobres <- read\_excel("porcent\_pobres.xlsx")  
porcent\_pobres <- gather(porcent\_pobres, ano, porcent\_pobres, X2012:X2021, convert = TRUE)  
porcent\_pobres$ano <- gsub('X', '', porcent\_pobres$ano)  
porcent\_pobres <- transform(porcent\_pobres, ano = as.numeric(ano))  
porcent\_pobres <- transform(porcent\_pobres, porcent\_pobres = as.numeric(porcent\_pobres))  
  
  
  
população\_total <- read\_excel("população\_total.xlsx")  
população\_total <- gather(população\_total, ano, população\_total, X2012:X2021, convert = TRUE)  
população\_total$ano <- gsub('X', '', população\_total$ano)  
população\_total <- transform(população\_total, ano = as.numeric(ano))  
população\_total <- transform(população\_total, população\_total = as.numeric(população\_total))  
  
  
  
mortalidade\_infantil <- read\_excel("mortalidade\_infantil.xlsx")  
mortalidade\_infantil <- gather(mortalidade\_infantil, ano, mortalidade\_infantil, X2012:X2021, convert = TRUE)  
mortalidade\_infantil$ano <- gsub('X', '', mortalidade\_infantil$ano)  
mortalidade\_infantil <- transform(mortalidade\_infantil, ano = as.numeric(ano))  
mortalidade\_infantil <- transform(mortalidade\_infantil, mortalidade\_infantil = as.numeric(mortalidade\_infantil))  
  
  
  
media\_anos\_de\_estudo <- read\_excel("media\_anos\_de\_estudo.xlsx")  
media\_anos\_de\_estudo <- gather(media\_anos\_de\_estudo, ano, media\_anos\_de\_estudo, X2012:X2021, convert = TRUE)  
media\_anos\_de\_estudo$ano <- gsub('X', '', media\_anos\_de\_estudo$ano)  
media\_anos\_de\_estudo <- transform(media\_anos\_de\_estudo, ano = as.numeric(ano))  
media\_anos\_de\_estudo <- transform(media\_anos\_de\_estudo, media\_anos\_de\_estudo = as.numeric(media\_anos\_de\_estudo))  
  
  
  
indice\_gini <- read\_excel("indice\_gini.xlsx")  
indice\_gini <- gather(indice\_gini, ano, indice\_gini, X2012:X2021, convert = TRUE)  
indice\_gini$ano <- gsub('X', '', indice\_gini$ano)  
indice\_gini <- transform(indice\_gini, ano = as.numeric(ano))  
indice\_gini <- transform(indice\_gini, indice\_gini = as.numeric(indice\_gini))  
  
  
  
ind\_theil\_L <- read\_excel("ind\_theil\_L.xlsx")  
ind\_theil\_L <- gather(ind\_theil\_L, ano, ind\_theil\_L, X2012:X2021, convert = TRUE)  
ind\_theil\_L$ano <- gsub('X', '', ind\_theil\_L$ano)  
ind\_theil\_L <- transform(ind\_theil\_L, ano = as.numeric(ano))  
ind\_theil\_L <- transform(ind\_theil\_L, ind\_theil\_L = as.numeric(ind\_theil\_L))  
  
  
  
analfabetismo\_25\_anos <- read\_excel("analfabetismo\_25\_anos.xlsx")  
analfabetismo\_25\_anos <- gather(analfabetismo\_25\_anos, ano, analfabetismo\_25\_anos, X2012:X2021, convert = TRUE)  
analfabetismo\_25\_anos$ano <- gsub('X', '', analfabetismo\_25\_anos$ano)  
analfabetismo\_25\_anos <- transform(analfabetismo\_25\_anos, ano = as.numeric(ano))  
analfabetismo\_25\_anos <- transform(analfabetismo\_25\_anos, analfabetismo\_25\_anos = as.numeric(analfabetismo\_25\_anos))  
  
  
  
analfabetismo\_18\_anos <- read\_excel("analfabetismo\_18\_anos.xlsx")  
analfabetismo\_18\_anos <- gather(analfabetismo\_18\_anos, ano, analfabetismo\_18\_anos, X2012:X2021, convert = TRUE)  
analfabetismo\_18\_anos$ano <- gsub('X', '', analfabetismo\_18\_anos$ano)  
analfabetismo\_18\_anos <- transform(analfabetismo\_18\_anos, ano = as.numeric(ano))  
analfabetismo\_18\_anos <- transform(analfabetismo\_18\_anos, analfabetismo\_18\_anos = as.numeric(analfabetismo\_18\_anos))  
  
  
  
analfabetismo\_15\_anos <- read\_excel("analfabetismo\_15\_anos.xlsx")  
analfabetismo\_15\_anos <- gather(analfabetismo\_15\_anos, ano, analfabetismo\_15\_anos, X2012:X2021, convert = TRUE)  
analfabetismo\_15\_anos$ano <- gsub('X', '', analfabetismo\_15\_anos$ano)  
analfabetismo\_15\_anos <- transform(analfabetismo\_15\_anos, ano = as.numeric(ano))  
analfabetismo\_15\_anos <- transform(analfabetismo\_15\_anos, analfabetismo\_15\_anos = as.numeric(analfabetismo\_15\_anos))  
  
  
  
IDHM <- read\_excel("IDHM.xlsx")  
IDHM <- gather(IDHM, ano, IDHM, X2012:X2021, convert = TRUE)  
IDHM$ano <- gsub('X', '', IDHM$ano)  
IDHM <- transform(IDHM, ano = as.numeric(ano))  
IDHM <- transform(IDHM, IDHM = as.numeric(IDHM))

para cada indicator, conta número de linhas e o número total de NULLS e divide NULLS pela linha para obter a % de NULLS para indicador

print(paste0("renda\_per\_capita"))

## [1] "renda\_per\_capita"

renda\_per\_capita.na <- as.data.frame(sum(is.na(renda\_per\_capita$renda\_per\_capita)))  
renda\_per\_capita.n <- as.data.frame(nrow(renda\_per\_capita))  
renda\_per\_capita.na$`sum(is.na(renda\_per\_capita$renda\_per\_capita))`/renda\_per\_capita.n$`nrow(renda\_per\_capita)`\*100

## [1] 0

print(paste0("sub\_esco\_pop"))

## [1] "sub\_esco\_pop"

sub\_esco\_pop.na <- as.data.frame(sum(is.na(sub\_esco\_pop$sub\_esco\_pop)))  
sub\_esco\_pop.n <- as.data.frame(nrow(sub\_esco\_pop))  
sub\_esco\_pop.na$`sum(is.na(sub\_esco\_pop$sub\_esco\_pop))`/sub\_esco\_pop.n$`nrow(sub\_esco\_pop)`\*100

## [1] 0

print(paste0("sub\_freq\_esco"))

## [1] "sub\_freq\_esco"

sub\_freq\_esco.na <- as.data.frame(sum(is.na(sub\_freq\_esco$sub\_freq\_esco)))  
sub\_freq\_esco.n <- as.data.frame(nrow(sub\_freq\_esco))  
sub\_freq\_esco.na$`sum(is.na(sub\_freq\_esco$sub\_freq\_esco))`/sub\_freq\_esco.n$`nrow(sub\_freq\_esco)`\*100

## [1] 0

print(paste0("esperança\_de\_vida"))

## [1] "esperança\_de\_vida"

esperança\_de\_vida.na <- as.data.frame(sum(is.na(esperança\_de\_vida$esperança\_de\_vida)))  
esperança\_de\_vida.n <- as.data.frame(nrow(esperança\_de\_vida))  
esperança\_de\_vida.na$`sum(is.na(esperança\_de\_vida$esperança\_de\_vida))`/esperança\_de\_vida.n$`nrow(esperança\_de\_vida)`\*100

## [1] 0

print(paste0("porcent\_pobres"))

## [1] "porcent\_pobres"

porcent\_pobres.na <- as.data.frame(sum(is.na(porcent\_pobres$porcent\_pobres)))  
porcent\_pobres.n <- as.data.frame(nrow(porcent\_pobres))  
porcent\_pobres.na$`sum(is.na(rporcent\_pobres$porcent\_pobres))`/porcent\_pobres.n$`nrow(porcent\_pobres)`\*100

## numeric(0)

print(paste0("população\_total"))

## [1] "população\_total"

população\_total.na <- as.data.frame(sum(is.na(população\_total$população\_total)))  
população\_total.n <- as.data.frame(nrow(população\_total))  
população\_total.na$`sum(is.na(população\_total$população\_total))`/população\_total.n$`nrow(população\_total)`\*100

## [1] 0

print(paste0("mortalidade\_infantil"))

## [1] "mortalidade\_infantil"

mortalidade\_infantil.na <- as.data.frame(sum(is.na(mortalidade\_infantil$mortalidade\_infantil)))  
mortalidade\_infantil.n <- as.data.frame(nrow(mortalidade\_infantil))  
mortalidade\_infantil.na$`sum(is.na(mortalidade\_infantil$mortalidade\_infantil))`/mortalidade\_infantil.n$`nrow(mortalidade\_infantil)`\*100

## [1] 0

print(paste0("media\_anos\_de\_estudo"))

## [1] "media\_anos\_de\_estudo"

media\_anos\_de\_estudo.na <- as.data.frame(sum(is.na(media\_anos\_de\_estudo$media\_anos\_de\_estudo)))  
media\_anos\_de\_estudo.n <- as.data.frame(nrow(media\_anos\_de\_estudo))  
media\_anos\_de\_estudo.na$`sum(is.na(media\_anos\_de\_estudo$media\_anos\_de\_estudo))`/media\_anos\_de\_estudo.n$`nrow(media\_anos\_de\_estudo)`\*100

## [1] 0

print(paste0("indice\_gini"))

## [1] "indice\_gini"

indice\_gini.na <- as.data.frame(sum(is.na(indice\_gini$indice\_gini)))  
indice\_gini.n <- as.data.frame(nrow(indice\_gini))  
indice\_gini.na$`sum(is.na(indice\_gini$indice\_gini))`/indice\_gini.n$`nrow(indice\_gini)`\*100

## [1] 0

print(paste0("ind\_theil\_L"))

## [1] "ind\_theil\_L"

ind\_theil\_L.na <- as.data.frame(sum(is.na(ind\_theil\_L$ind\_theil\_L)))  
ind\_theil\_L.n <- as.data.frame(nrow(ind\_theil\_L))  
ind\_theil\_L.na$`sum(is.na(ind\_theil\_L$ind\_theil\_L))`/ind\_theil\_L.n$`nrow(ind\_theil\_L)`\*100

## [1] 0

print(paste0("analfabetismo\_25\_anos"))

## [1] "analfabetismo\_25\_anos"

analfabetismo\_25\_anos.na <- as.data.frame(sum(is.na(analfabetismo\_25\_anos$analfabetismo\_25\_anos)))  
analfabetismo\_25\_anos.n <- as.data.frame(nrow(analfabetismo\_25\_anos))  
analfabetismo\_25\_anos.na$`sum(is.na(analfabetismo\_25\_anos$analfabetismo\_25\_anos))`/analfabetismo\_25\_anos.n$`nrow(analfabetismo\_25\_anos)`\*100

## [1] 0

print(paste0("analfabetismo\_18\_anos"))

## [1] "analfabetismo\_18\_anos"

analfabetismo\_18\_anos.na <- as.data.frame(sum(is.na(analfabetismo\_18\_anos$analfabetismo\_18\_anos)))  
analfabetismo\_18\_anos.n <- as.data.frame(nrow(analfabetismo\_18\_anos))  
analfabetismo\_18\_anos.na$`sum(is.na(analfabetismo\_18\_anos$analfabetismo\_18\_anos))`/analfabetismo\_18\_anos.n$`nrow(analfabetismo\_18\_anos)`\*100

## [1] 0

print(paste0("analfabetismo\_15\_anos"))

## [1] "analfabetismo\_15\_anos"

analfabetismo\_15\_anos.na <- as.data.frame(sum(is.na(analfabetismo\_15\_anos$analfabetismo\_15\_anos)))  
analfabetismo\_15\_anos.n <- as.data.frame(nrow(analfabetismo\_15\_anos))  
analfabetismo\_15\_anos.na$`sum(is.na(analfabetismo\_15\_anos$analfabetismo\_15\_anos))`/analfabetismo\_15\_anos.n$`nrow(analfabetismo\_15\_anos)`\*100

## [1] 0

print(paste0("IDHM"))

## [1] "IDHM"

IDHM.na <- as.data.frame(sum(is.na(IDHM$IDHM)))  
IDHM.n <- as.data.frame(nrow(IDHM))  
IDHM.na$`sum(is.na(IDHM$IDHM))`/IDHM.n$`nrow(IDHM)`\*100

## [1] 0

#criar data frame único com os indicadores IDHM.AED e IDHM.df

IDHM.AED = renda\_per\_capita  
IDHM.AED <- join(IDHM.AED, sub\_esco\_pop, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, sub\_freq\_esco, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, esperança\_de\_vida, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, porcent\_pobres, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, população\_total, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, mortalidade\_infantil, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, media\_anos\_de\_estudo, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, indice\_gini, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, ind\_theil\_L, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, analfabetismo\_25\_anos, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, analfabetismo\_18\_anos, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, analfabetismo\_15\_anos, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
IDHM.AED = join(IDHM.AED, IDHM, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))

AED.df = IDHM.AED  
  
  
  
# Verificar que o número de estados continua os mesmos e o data frame está correto  
  
  
sapply(AED.df, function(x) length(unique(x)))

## Territorialidades ano renda\_per\_capita   
## 28 10 280   
## sub\_esco\_pop sub\_freq\_esco esperança\_de\_vida   
## 179 156 238   
## porcent\_pobres população\_total mortalidade\_infantil   
## 261 280 255   
## media\_anos\_de\_estudo indice\_gini ind\_theil\_L   
## 202 134 182   
## analfabetismo\_25\_anos analfabetismo\_18\_anos analfabetismo\_15\_anos   
## 264 252 248   
## IDHM   
## 135

str(AED.df)

## 'data.frame': 280 obs. of 16 variables:  
## $ Territorialidades : chr "Brasil" "Acre" "Alagoas" "Amapá" ...  
## $ ano : num 2012 2012 2012 2012 2012 ...  
## $ renda\_per\_capita : num 759 517 395 528 559 ...  
## $ sub\_esco\_pop : num 0.606 0.59 0.487 0.67 0.613 0.51 0.54 0.765 0.613 0.619 ...  
## $ sub\_freq\_esco : num 0.731 0.681 0.645 0.653 0.642 0.639 0.742 0.77 0.735 0.741 ...  
## $ esperança\_de\_vida : num 74.5 72.5 70 72.8 70.8 ...  
## $ porcent\_pobres : num 11.4 23.8 23.4 18.4 22.2 ...  
## $ população\_total : num 1.98e+08 7.77e+05 3.22e+06 7.21e+05 3.54e+06 ...  
## $ mortalidade\_infantil : num 15.8 20.2 26.1 24.3 20.9 ...  
## $ media\_anos\_de\_estudo : num 8.56 7.72 6.8 9.09 8.63 ...  
## $ indice\_gini : num 0.54 0.566 0.503 0.528 0.589 0.563 0.545 0.601 0.489 0.474 ...  
## $ ind\_theil\_L : num 0.526 0.585 0.447 0.483 0.619 0.571 0.54 0.664 0.411 0.383 ...  
## $ analfabetismo\_25\_anos: num 10.22 18.22 24.22 7.93 9.46 ...  
## $ analfabetismo\_18\_anos: num 8.75 14.72 20.54 6.37 7.89 ...  
## $ analfabetismo\_15\_anos: num 8.21 13.48 18.97 5.76 7.22 ...  
## $ IDHM : num 0.746 0.701 0.651 0.707 0.691 0.678 0.701 0.825 0.758 0.744 ...

#AED.df é o data frame usado na etapa de análise exploratória de dados.

Engenharia de Dados

library(corrplot)

## Warning: package 'corrplot' was built under R version 4.3.1

## corrplot 0.92 loaded

library(RColorBrewer)

## Warning: package 'RColorBrewer' was built under R version 4.3.1

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.3.1

library(ggpubr)

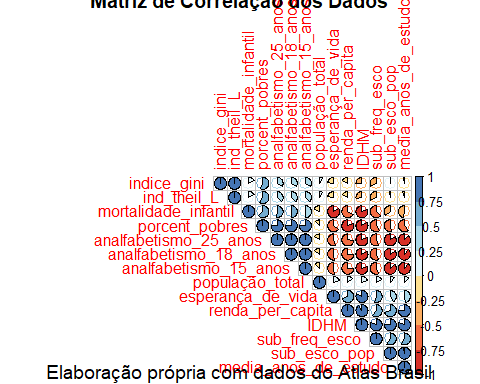
## Warning: package 'ggpubr' was built under R version 4.3.1

##   
## Attaching package: 'ggpubr'

## The following object is masked from 'package:plyr':  
##   
## mutate

matriz de correlação

# análise exploratória  
  
# Montar uma matriz de correlação básica de cada indicador  
#criar data frame que remova as linhas com nulos.  
  
cc = complete.cases(AED.df)  
AED.corr = AED.df[cc,]  
  
#remover não numérico  
AED.corr$Territorialidades <- NULL  
AED.corr$ano <- NULL  
  
### Matriz de corelação com Corrplot  
  
Matrix <-cor(AED.corr)  
corrplot(Matrix, type="upper", order="hclust", method="pie",  
 col=brewer.pal(n=8, name="RdYlBu"),  
 title="Matriz de Correlação dos Dados")  
  
# Adicionar uma nota de rodapé  
mtext("Elaboração própria com dados do Atlas Brasil", side=1, line=4, cex=1.2)

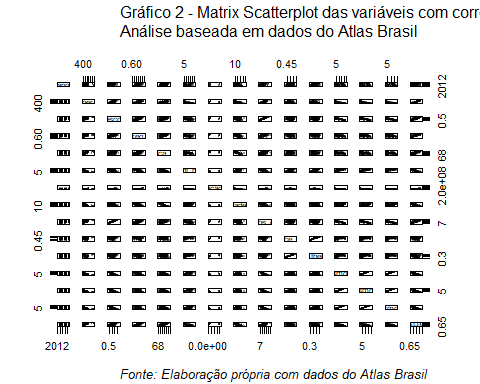


# gini, theil e população tem relação fraca com IDHM. mas irei manter para análise

# Montar um df para (IDHM ~ renda per capita, sub. ind. ecolaridade, sub. nd. frequencia escolar e esperança de vida.)  
predic.IDHM <- renda\_per\_capita  
predic.IDHM <- join(predic.IDHM, sub\_esco\_pop, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, sub\_freq\_esco, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, esperança\_de\_vida, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, porcent\_pobres, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, população\_total, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, mortalidade\_infantil, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, media\_anos\_de\_estudo, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, indice\_gini, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, ind\_theil\_L, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, analfabetismo\_25\_anos, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, analfabetismo\_18\_anos, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, analfabetismo\_15\_anos, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM <- join(predic.IDHM, IDHM, by = c("ano" = "ano", "Territorialidades" = "Territorialidades"))  
predic.IDHM$Territorialidades <- NULL   
cc = complete.cases(predic.IDHM)  
predic.IDHM = predic.IDHM[cc,]  
str(predic.IDHM)

## 'data.frame': 280 obs. of 15 variables:  
## $ ano : num 2012 2012 2012 2012 2012 ...  
## $ renda\_per\_capita : num 759 517 395 528 559 ...  
## $ sub\_esco\_pop : num 0.606 0.59 0.487 0.67 0.613 0.51 0.54 0.765 0.613 0.619 ...  
## $ sub\_freq\_esco : num 0.731 0.681 0.645 0.653 0.642 0.639 0.742 0.77 0.735 0.741 ...  
## $ esperança\_de\_vida : num 74.5 72.5 70 72.8 70.8 ...  
## $ porcent\_pobres : num 11.4 23.8 23.4 18.4 22.2 ...  
## $ população\_total : num 1.98e+08 7.77e+05 3.22e+06 7.21e+05 3.54e+06 ...  
## $ mortalidade\_infantil : num 15.8 20.2 26.1 24.3 20.9 ...  
## $ media\_anos\_de\_estudo : num 8.56 7.72 6.8 9.09 8.63 ...  
## $ indice\_gini : num 0.54 0.566 0.503 0.528 0.589 0.563 0.545 0.601 0.489 0.474 ...  
## $ ind\_theil\_L : num 0.526 0.585 0.447 0.483 0.619 0.571 0.54 0.664 0.411 0.383 ...  
## $ analfabetismo\_25\_anos: num 10.22 18.22 24.22 7.93 9.46 ...  
## $ analfabetismo\_18\_anos: num 8.75 14.72 20.54 6.37 7.89 ...  
## $ analfabetismo\_15\_anos: num 8.21 13.48 18.97 5.76 7.22 ...  
## $ IDHM : num 0.746 0.701 0.651 0.707 0.691 0.678 0.701 0.825 0.758 0.744 ...

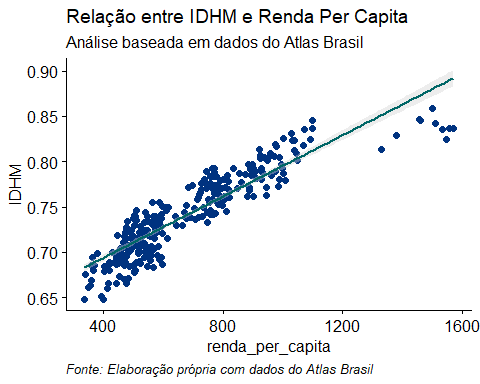
### Plotar a correlação como scatterplot matrix.  
# Criação do scatterplot sem título principal  
plot(predic.IDHM, pch=1, cex=.2, col=rgb(0,0,0,0.4), main="")  
  
# Adição do título, subtítulo e fonte  
mtext("Gráfico 2 - Matrix Scatterplot das variáveis com correlação com o IDHM", side=3, line=3, adj=0) # Título  
mtext("Análise baseada em dados do Atlas Brasil", side=3, line=2, adj=0) # Subtítulo  
mtext("Fonte: Elaboração própria com dados do Atlas Brasil", side=1, line=4, adj=0, cex=0.8, col="black", font=3) # Fonte



explorar relações lineares com potenciais relações diretas entre os indicadores

# Criar um scatterplot com linha de regressão para IDHM e renda per capita  
ggscatter(predic.IDHM, x = "renda\_per\_capita", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

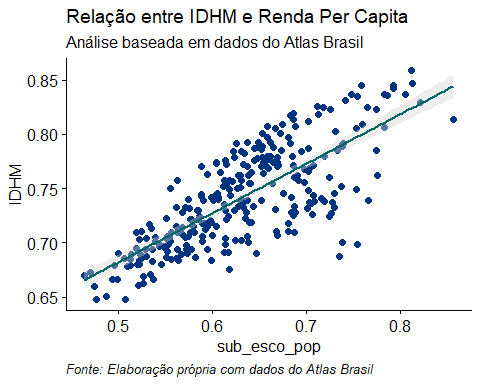


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ renda\_per\_capita, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ renda\_per\_capita, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.063202 -0.012445 0.001026 0.015348 0.036847   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.256e-01 3.516e-03 177.91 <2e-16 \*\*\*  
## renda\_per\_capita 1.699e-04 4.842e-06 35.08 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.01965 on 278 degrees of freedom  
## Multiple R-squared: 0.8157, Adjusted R-squared: 0.8151   
## F-statistic: 1231 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM e Subindice de Escolaridade  
ggscatter(predic.IDHM, x = "sub\_esco\_pop", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

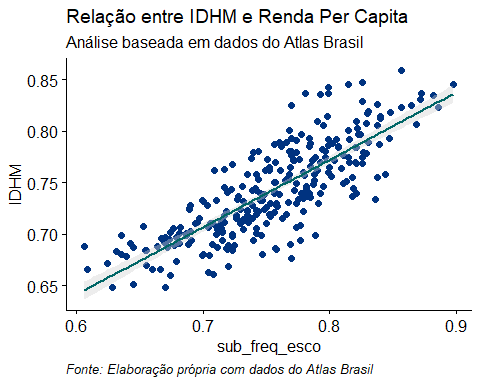


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ sub\_esco\_pop, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ sub\_esco\_pop, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.101216 -0.017439 -0.000147 0.020216 0.052755   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.45495 0.01448 31.41 <2e-16 \*\*\*  
## sub\_esco\_pop 0.45417 0.02276 19.96 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02935 on 278 degrees of freedom  
## Multiple R-squared: 0.5889, Adjusted R-squared: 0.5874   
## F-statistic: 398.2 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM e Subindice de frequência Escola  
ggscatter(predic.IDHM, x = "sub\_freq\_esco", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

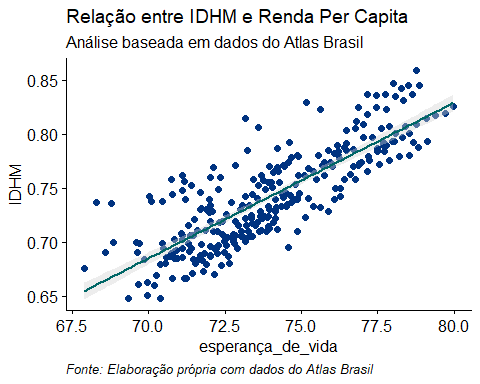


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ sub\_freq\_esco, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ sub\_freq\_esco, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.076437 -0.016621 -0.001193 0.016278 0.077382   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.24977 0.02139 11.68 <2e-16 \*\*\*  
## sub\_freq\_esco 0.65281 0.02830 23.07 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02682 on 278 degrees of freedom  
## Multiple R-squared: 0.6569, Adjusted R-squared: 0.6556   
## F-statistic: 532.2 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM e Esperança de Vida  
ggscatter(predic.IDHM, x = "esperança\_de\_vida", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

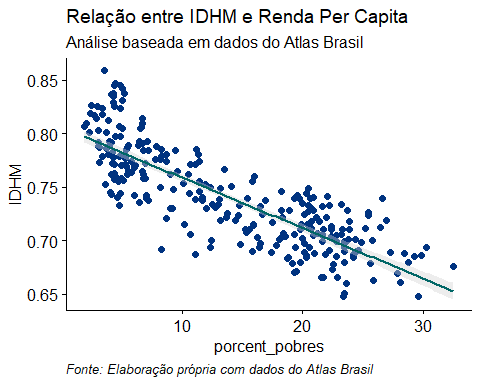


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ esperança\_de\_vida, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ esperança\_de\_vida, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.056084 -0.019263 -0.005752 0.015072 0.083536   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.324437 0.044098 -7.357 2.12e-12 \*\*\*  
## esperança\_de\_vida 0.014419 0.000596 24.194 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02598 on 278 degrees of freedom  
## Multiple R-squared: 0.678, Adjusted R-squared: 0.6768   
## F-statistic: 585.4 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM e Percentual de Pobres na População  
ggscatter(predic.IDHM, x = "porcent\_pobres", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

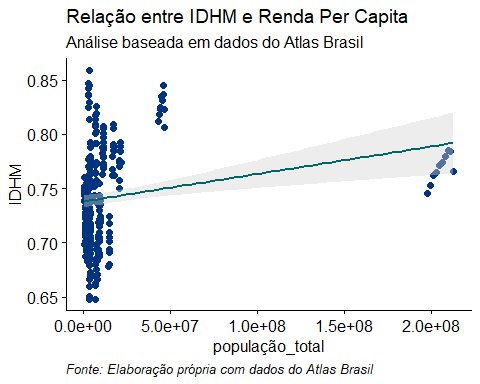


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ porcent\_pobres, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ porcent\_pobres, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.074741 -0.016308 -0.000152 0.016863 0.069873   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.8054598 0.0030056 267.99 <2e-16 \*\*\*  
## porcent\_pobres -0.0046933 0.0001906 -24.62 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02567 on 278 degrees of freedom  
## Multiple R-squared: 0.6856, Adjusted R-squared: 0.6844   
## F-statistic: 606.1 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM e População Total  
ggscatter(predic.IDHM, x = "população\_total", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

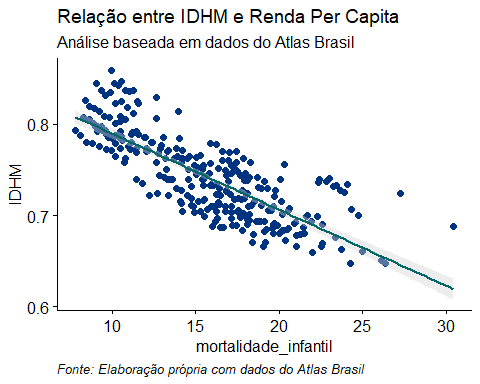


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ população\_total, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ população\_total, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.091832 -0.035392 -0.005422 0.031361 0.120117   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.381e-01 2.870e-03 257.141 < 2e-16 \*\*\*  
## população\_total 2.528e-10 7.092e-11 3.565 0.000428 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.04477 on 278 degrees of freedom  
## Multiple R-squared: 0.04372, Adjusted R-squared: 0.04028   
## F-statistic: 12.71 on 1 and 278 DF, p-value: 0.0004279

# Criar um scatterplot com linha de regressão para IDHM e Mortalidade Infantil  
ggscatter(predic.IDHM, x = "mortalidade\_infantil", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

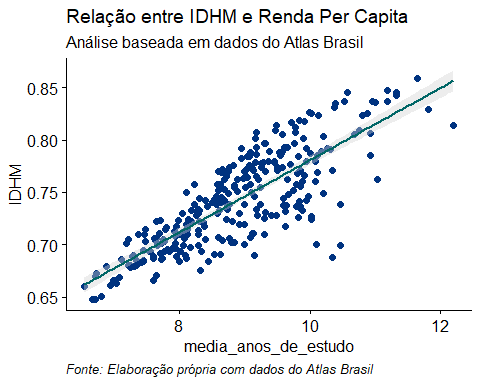


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ mortalidade\_infantil, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ mortalidade\_infantil, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.051939 -0.020627 -0.002814 0.015853 0.078314   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.8730255 0.0061886 141.1 <2e-16 \*\*\*  
## mortalidade\_infantil -0.0083336 0.0003788 -22.0 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02765 on 278 degrees of freedom  
## Multiple R-squared: 0.6351, Adjusted R-squared: 0.6338   
## F-statistic: 483.9 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM e Média de Anos de Estudo  
ggscatter(predic.IDHM, x = "media\_anos\_de\_estudo", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

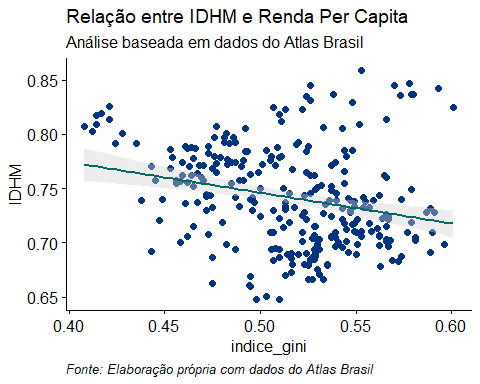


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ media\_anos\_de\_estudo, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ media\_anos\_de\_estudo, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.104787 -0.015780 0.001814 0.018576 0.054403   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.437262 0.013057 33.49 <2e-16 \*\*\*  
## media\_anos\_de\_estudo 0.034350 0.001462 23.50 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02649 on 278 degrees of freedom  
## Multiple R-squared: 0.6652, Adjusted R-squared: 0.664   
## F-statistic: 552.3 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM eÍndice de GINI  
ggscatter(predic.IDHM, x = "indice\_gini", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

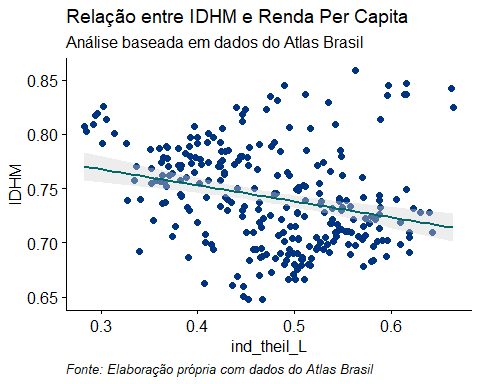


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ indice\_gini, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ indice\_gini, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.098729 -0.030321 -0.002438 0.026708 0.127824   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.88755 0.03385 26.217 < 2e-16 \*\*\*  
## indice\_gini -0.28277 0.06549 -4.318 2.2e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.04432 on 278 degrees of freedom  
## Multiple R-squared: 0.06284, Adjusted R-squared: 0.05947   
## F-statistic: 18.64 on 1 and 278 DF, p-value: 2.196e-05

# Criar um scatterplot com linha de regressão para IDHM e Índice de Theil  
ggscatter(predic.IDHM, x = "ind\_theil\_L", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

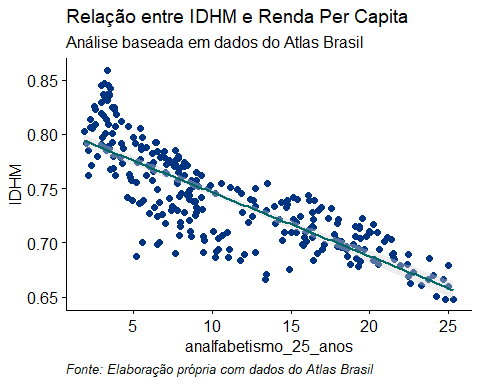


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ ind\_theil\_L, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ ind\_theil\_L, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.09734 -0.02999 -0.00193 0.02572 0.13008   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.81220 0.01553 52.306 < 2e-16 \*\*\*  
## ind\_theil\_L -0.14792 0.03217 -4.599 6.46e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.04413 on 278 degrees of freedom  
## Multiple R-squared: 0.0707, Adjusted R-squared: 0.06735   
## F-statistic: 21.15 on 1 and 278 DF, p-value: 6.462e-06

# Criar um scatterplot com linha de regressão para IDHM e Tx. Analfabetismo acima de 25 anos  
ggscatter(predic.IDHM, x = "analfabetismo\_25\_anos", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

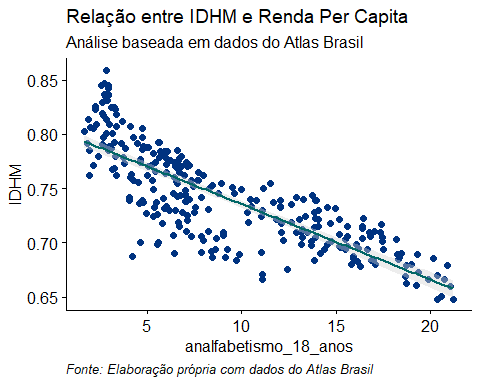


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ analfabetismo\_25\_anos, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ analfabetismo\_25\_anos, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.08688 -0.01411 0.00297 0.01686 0.07343   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.8053146 0.0031265 257.57 <2e-16 \*\*\*  
## analfabetismo\_25\_anos -0.0058758 0.0002496 -23.54 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02646 on 278 degrees of freedom  
## Multiple R-squared: 0.6659, Adjusted R-squared: 0.6647   
## F-statistic: 554.1 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM e Tx. Analfabetismo acima de 18 anos  
ggscatter(predic.IDHM, x = "analfabetismo\_18\_anos", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape

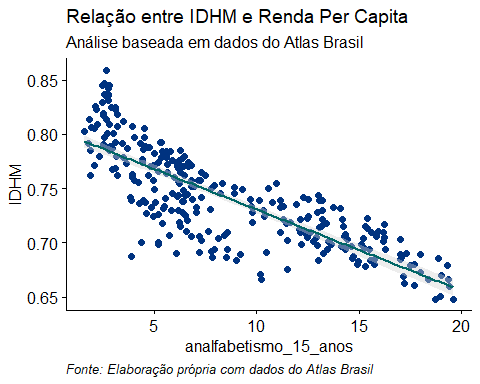


# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ analfabetismo\_18\_anos, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ analfabetismo\_18\_anos, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.087967 -0.014119 0.003096 0.017285 0.073505   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.8052461 0.0031911 252.3 <2e-16 \*\*\*  
## analfabetismo\_18\_anos -0.0069547 0.0003024 -23.0 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02687 on 278 degrees of freedom  
## Multiple R-squared: 0.6555, Adjusted R-squared: 0.6542   
## F-statistic: 528.9 on 1 and 278 DF, p-value: < 2.2e-16

# Criar um scatterplot com linha de regressão para IDHM e Tx. Analfabetismo acima de 15 anos  
ggscatter(predic.IDHM, x = "analfabetismo\_15\_anos", y = "IDHM",   
 color=rgb(0,.2,.5, 1), pch=1, add = "reg.line",   
 add.params = list(color=rgb(0,.4,.4, 1), fill = "light gray"),   
 conf.int = TRUE, main = "Relação entre IDHM e Renda Per Capita") +  
 labs(title = "Relação entre IDHM e Renda Per Capita",  
 subtitle = "Análise baseada em dados do Atlas Brasil",  
 caption = "Fonte: Elaboração própria com dados do Atlas Brasil") +  
 theme(plot.caption = element\_text(hjust = 0, face="italic"))

## Warning: Duplicated aesthetics after name standardisation: shape



# Printar estatística F para ver a significacia da regressão  
summary(lm(IDHM ~ analfabetismo\_15\_anos, data=predic.IDHM))

##   
## Call:  
## lm(formula = IDHM ~ analfabetismo\_15\_anos, data = predic.IDHM)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.088380 -0.014143 0.003496 0.017045 0.073506   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.8055144 0.0032362 248.91 <2e-16 \*\*\*  
## analfabetismo\_15\_anos -0.0074704 0.0003287 -22.73 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02708 on 278 degrees of freedom  
## Multiple R-squared: 0.6501, Adjusted R-squared: 0.6488   
## F-statistic: 516.4 on 1 and 278 DF, p-value: < 2.2e-16

remover indices de Theil e Gini junto com população total

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.3.1

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:plyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

predic.IDHM <- select(predic.IDHM, -ind\_theil\_L, -indice\_gini, -população\_total)

# Correlação positiva a todas as regressões( R2: renda per capita 0,8157, p-valor 2.2e-16

# escolaridade da população 0,5889 e p-valor 2.2e-16

# frequencia escolar 0.6569 e p-valor 2.2e-16

# esperança de vida 0.678 e p-valor 2.2e-16)

# população total R2 0.04372

# indice theil L R2 0,0707

# indice gini R2 0,06284

###Decils para avaliar a possibilidade de Outliers

print(paste0("renda\_per\_capita"))

## [1] "renda\_per\_capita"

quantile(predic.IDHM$renda\_per\_capita, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80%   
## 451.576 489.930 520.626 553.800 593.620 727.718 775.931 883.042   
## 90% 100%   
## 985.232 1568.870

sort(predic.IDHM$renda\_per\_capita)

## [1] 338.42 341.32 350.41 356.63 362.29 362.31 366.24 367.42 370.74  
## [10] 382.11 395.06 401.31 404.28 405.80 415.17 418.37 419.89 421.03  
## [19] 422.39 432.99 442.82 443.65 444.05 445.31 447.25 447.97 451.27  
## [28] 451.45 451.59 452.75 453.47 458.68 464.70 465.74 467.19 469.93  
## [37] 470.59 471.12 471.54 471.65 472.85 473.32 473.59 473.62 478.02  
## [46] 478.39 478.58 478.80 478.88 479.55 480.55 481.46 482.64 484.80  
## [55] 485.71 486.65 490.75 492.78 494.32 495.42 497.15 498.33 499.19  
## [64] 501.53 502.70 503.29 503.66 503.95 506.02 506.21 508.14 508.36  
## [73] 509.32 509.57 511.42 512.47 512.75 514.59 516.00 516.75 516.97  
## [82] 517.64 520.00 520.08 520.86 523.09 523.38 524.68 525.07 525.50  
## [91] 526.67 528.23 528.65 529.31 531.35 534.64 536.41 536.77 538.12  
## [100] 538.50 540.06 540.61 541.74 543.81 546.78 547.09 547.20 548.93  
## [109] 549.06 549.29 549.54 550.02 556.32 557.38 557.71 559.44 560.50  
## [118] 561.11 561.40 561.52 564.61 565.81 567.26 569.17 569.89 572.81  
## [127] 573.92 574.03 574.89 575.58 575.63 579.19 579.46 580.02 583.80  
## [136] 585.99 588.48 588.89 589.03 593.46 593.78 593.84 595.13 596.82  
## [145] 598.77 599.80 603.61 607.23 613.07 614.31 617.00 640.90 644.99  
## [154] 663.76 673.18 679.62 679.72 684.63 691.06 699.24 703.36 707.48  
## [163] 713.76 717.96 718.35 722.67 723.84 727.13 728.60 732.94 734.81  
## [172] 738.04 739.68 745.24 746.52 747.30 749.17 749.30 750.40 755.14  
## [181] 757.43 758.07 758.68 759.11 764.29 765.94 766.18 767.68 767.86  
## [190] 770.94 771.16 772.78 773.16 774.72 775.41 775.46 777.03 777.37  
## [199] 779.13 780.32 781.44 784.47 788.18 791.12 791.40 792.81 792.86  
## [208] 794.67 798.41 802.94 805.71 806.32 807.78 810.08 814.30 817.79  
## [217] 819.61 821.80 833.57 849.52 854.53 860.55 873.96 882.69 884.45  
## [226] 887.28 888.32 891.00 896.60 900.00 901.20 901.42 909.43 912.60  
## [235] 921.16 925.29 926.61 927.60 929.11 932.80 937.67 944.53 944.64  
## [244] 949.54 959.06 959.50 961.32 971.82 974.74 976.02 978.64 984.82  
## [253] 988.94 990.06 994.83 998.33 1000.82 1001.71 1007.29 1029.92 1032.89  
## [262] 1037.42 1038.98 1044.95 1047.74 1068.69 1072.88 1093.25 1096.85 1099.62  
## [271] 1326.87 1377.92 1456.83 1457.06 1498.74 1508.91 1533.05 1546.18 1553.68  
## [280] 1568.87

print(paste0("sub\_esco\_pop"))

## [1] "sub\_esco\_pop"

quantile(predic.IDHM$sub\_esco\_pop, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 0.5299 0.5620 0.5857 0.6126 0.6305 0.6490 0.6700 0.6986 0.7362 0.8570

sort(predic.IDHM$sub\_esco\_pop)

## [1] 0.464 0.470 0.475 0.477 0.487 0.494 0.496 0.499 0.500 0.507 0.508 0.510  
## [13] 0.513 0.513 0.514 0.519 0.519 0.520 0.521 0.523 0.523 0.523 0.524 0.525  
## [25] 0.526 0.527 0.527 0.529 0.530 0.530 0.533 0.534 0.535 0.536 0.536 0.537  
## [37] 0.540 0.540 0.544 0.544 0.546 0.549 0.549 0.551 0.551 0.552 0.552 0.552  
## [49] 0.553 0.554 0.555 0.556 0.557 0.557 0.562 0.562 0.562 0.563 0.563 0.563  
## [61] 0.564 0.564 0.567 0.568 0.568 0.568 0.572 0.572 0.573 0.573 0.574 0.575  
## [73] 0.575 0.576 0.577 0.577 0.578 0.578 0.580 0.581 0.582 0.584 0.585 0.585  
## [85] 0.586 0.588 0.588 0.589 0.590 0.590 0.590 0.591 0.593 0.593 0.595 0.595  
## [97] 0.595 0.598 0.599 0.599 0.599 0.600 0.601 0.603 0.604 0.605 0.605 0.606  
## [109] 0.607 0.607 0.609 0.612 0.613 0.613 0.613 0.614 0.614 0.614 0.614 0.615  
## [121] 0.615 0.615 0.617 0.617 0.618 0.618 0.618 0.619 0.622 0.623 0.624 0.624  
## [133] 0.625 0.626 0.627 0.628 0.628 0.630 0.630 0.630 0.631 0.631 0.631 0.631  
## [145] 0.633 0.633 0.633 0.634 0.635 0.636 0.637 0.637 0.637 0.638 0.639 0.639  
## [157] 0.640 0.640 0.641 0.642 0.642 0.642 0.644 0.645 0.645 0.647 0.648 0.649  
## [169] 0.649 0.652 0.652 0.652 0.652 0.653 0.654 0.654 0.655 0.655 0.657 0.659  
## [181] 0.659 0.659 0.659 0.660 0.660 0.662 0.662 0.664 0.665 0.665 0.667 0.667  
## [193] 0.668 0.669 0.670 0.670 0.670 0.671 0.673 0.674 0.675 0.676 0.681 0.681  
## [205] 0.681 0.683 0.683 0.684 0.684 0.685 0.686 0.686 0.687 0.687 0.688 0.688  
## [217] 0.688 0.689 0.689 0.691 0.692 0.695 0.698 0.698 0.701 0.702 0.702 0.703  
## [229] 0.703 0.704 0.704 0.706 0.707 0.708 0.708 0.709 0.712 0.713 0.716 0.718  
## [241] 0.719 0.720 0.725 0.725 0.726 0.727 0.728 0.730 0.730 0.731 0.733 0.736  
## [253] 0.738 0.739 0.739 0.739 0.740 0.742 0.748 0.754 0.754 0.755 0.755 0.759  
## [265] 0.760 0.765 0.766 0.775 0.775 0.776 0.783 0.783 0.787 0.793 0.793 0.806  
## [277] 0.812 0.813 0.822 0.857

print(paste0("sub\_freq\_esco"))

## [1] "sub\_freq\_esco"

quantile(predic.IDHM$sub\_freq\_esco, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 0.6758 0.7068 0.7230 0.7420 0.7560 0.7700 0.7843 0.8012 0.8260 0.8980

sort(predic.IDHM$sub\_freq\_esco)

## [1] 0.606 0.608 0.624 0.628 0.631 0.635 0.635 0.639 0.642 0.645 0.645 0.653  
## [13] 0.654 0.655 0.655 0.660 0.664 0.666 0.667 0.669 0.670 0.670 0.670 0.672  
## [25] 0.672 0.673 0.674 0.674 0.676 0.677 0.679 0.679 0.681 0.681 0.682 0.687  
## [37] 0.687 0.688 0.689 0.689 0.690 0.690 0.691 0.694 0.695 0.697 0.699 0.699  
## [49] 0.700 0.700 0.703 0.704 0.704 0.704 0.705 0.706 0.707 0.707 0.708 0.709  
## [61] 0.710 0.710 0.710 0.711 0.711 0.714 0.714 0.714 0.714 0.715 0.715 0.715  
## [73] 0.717 0.717 0.717 0.717 0.718 0.718 0.719 0.719 0.720 0.720 0.722 0.723  
## [85] 0.723 0.723 0.723 0.725 0.726 0.726 0.727 0.728 0.729 0.729 0.731 0.732  
## [97] 0.732 0.732 0.733 0.735 0.735 0.735 0.735 0.736 0.738 0.739 0.739 0.739  
## [109] 0.740 0.740 0.741 0.742 0.742 0.742 0.742 0.742 0.742 0.744 0.744 0.744  
## [121] 0.745 0.745 0.746 0.747 0.747 0.748 0.748 0.749 0.750 0.750 0.751 0.752  
## [133] 0.752 0.752 0.753 0.753 0.756 0.756 0.756 0.756 0.756 0.757 0.757 0.757  
## [145] 0.758 0.758 0.758 0.759 0.760 0.760 0.761 0.762 0.763 0.765 0.766 0.766  
## [157] 0.767 0.767 0.768 0.768 0.768 0.769 0.769 0.770 0.770 0.770 0.770 0.770  
## [169] 0.770 0.771 0.772 0.772 0.774 0.774 0.774 0.774 0.774 0.775 0.776 0.777  
## [181] 0.777 0.778 0.778 0.779 0.779 0.779 0.780 0.781 0.781 0.782 0.782 0.782  
## [193] 0.783 0.784 0.784 0.784 0.785 0.785 0.785 0.786 0.787 0.788 0.788 0.789  
## [205] 0.789 0.790 0.790 0.791 0.793 0.793 0.793 0.794 0.795 0.795 0.797 0.797  
## [217] 0.798 0.799 0.799 0.799 0.799 0.799 0.800 0.801 0.802 0.804 0.805 0.807  
## [229] 0.807 0.809 0.811 0.811 0.813 0.815 0.815 0.815 0.816 0.816 0.818 0.818  
## [241] 0.819 0.820 0.820 0.821 0.821 0.821 0.821 0.822 0.823 0.824 0.826 0.826  
## [253] 0.826 0.826 0.827 0.829 0.829 0.829 0.830 0.831 0.831 0.836 0.838 0.838  
## [265] 0.838 0.839 0.840 0.840 0.844 0.847 0.848 0.857 0.857 0.865 0.869 0.872  
## [277] 0.873 0.882 0.886 0.898

print(paste0("esperança\_de\_vida"))

## [1] "esperança\_de\_vida"

quantile(predic.IDHM$esperança\_de\_vida, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 70.770 71.596 72.257 73.042 73.705 74.358 75.275 76.500 77.752 79.980

sort(predic.IDHM$esperança\_de\_vida)

## [1] 67.90 68.28 68.57 68.77 68.86 69.34 69.61 69.62 69.65 69.71 69.87 69.96  
## [13] 69.96 70.03 70.10 70.19 70.25 70.37 70.38 70.44 70.45 70.50 70.54 70.57  
## [25] 70.71 70.76 70.76 70.77 70.77 70.83 70.85 70.85 70.91 70.94 71.06 71.10  
## [37] 71.11 71.11 71.11 71.12 71.12 71.16 71.16 71.20 71.23 71.30 71.30 71.34  
## [49] 71.36 71.39 71.41 71.44 71.47 71.47 71.50 71.54 71.61 71.64 71.69 71.70  
## [61] 71.71 71.72 71.74 71.76 71.80 71.84 71.85 71.88 71.90 71.99 72.00 72.01  
## [73] 72.02 72.02 72.03 72.03 72.03 72.08 72.11 72.11 72.15 72.15 72.22 72.25  
## [85] 72.26 72.29 72.30 72.31 72.34 72.39 72.41 72.42 72.47 72.48 72.50 72.53  
## [97] 72.55 72.59 72.60 72.66 72.71 72.77 72.78 72.79 72.80 72.83 72.84 72.84  
## [109] 72.91 72.94 73.01 73.03 73.05 73.07 73.09 73.16 73.16 73.16 73.18 73.21  
## [121] 73.24 73.26 73.29 73.29 73.37 73.37 73.39 73.40 73.41 73.43 73.49 73.56  
## [133] 73.56 73.58 73.61 73.64 73.65 73.66 73.69 73.69 73.72 73.75 73.78 73.81  
## [145] 73.87 73.89 73.91 73.92 73.92 73.93 73.94 73.95 73.96 74.02 74.05 74.09  
## [157] 74.09 74.15 74.16 74.16 74.16 74.18 74.20 74.20 74.20 74.26 74.29 74.35  
## [169] 74.37 74.42 74.42 74.44 74.47 74.48 74.51 74.53 74.59 74.59 74.60 74.65  
## [181] 74.66 74.68 74.78 74.80 74.86 74.88 74.88 74.88 74.90 74.91 74.94 75.11  
## [193] 75.13 75.17 75.22 75.23 75.38 75.40 75.49 75.56 75.62 75.63 75.68 75.74  
## [205] 75.75 75.76 75.87 75.88 75.96 76.00 76.08 76.09 76.12 76.18 76.21 76.22  
## [217] 76.24 76.28 76.29 76.40 76.40 76.47 76.47 76.47 76.62 76.69 76.72 76.76  
## [229] 76.78 76.80 76.86 76.87 76.91 76.96 77.02 77.04 77.15 77.17 77.17 77.21  
## [241] 77.23 77.32 77.47 77.47 77.49 77.49 77.54 77.61 77.70 77.70 77.73 77.75  
## [253] 77.77 77.78 77.89 77.89 77.97 78.02 78.04 78.07 78.07 78.22 78.27 78.30  
## [265] 78.35 78.42 78.52 78.53 78.55 78.63 78.68 78.76 78.76 78.83 78.87 79.09  
## [277] 79.12 79.40 79.70 79.98

print(paste0("porcent\_pobres"))

## [1] "porcent\_pobres"

quantile(predic.IDHM$porcent\_pobres, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 4.097 4.864 6.473 8.990 12.605 16.746 20.103 21.684 23.725 32.530

sort(predic.IDHM$porcent\_pobres)

## [1] 1.83 1.98 2.25 2.32 2.39 2.65 2.86 2.90 2.97 2.99 3.00 3.07  
## [13] 3.10 3.33 3.34 3.36 3.48 3.57 3.68 3.68 3.86 3.86 3.89 3.93  
## [25] 4.01 4.03 4.06 4.07 4.10 4.14 4.18 4.20 4.20 4.22 4.25 4.26  
## [37] 4.30 4.30 4.33 4.33 4.35 4.37 4.41 4.42 4.44 4.45 4.56 4.62  
## [49] 4.71 4.73 4.77 4.78 4.79 4.81 4.83 4.84 4.87 4.87 4.98 4.98  
## [61] 5.02 5.03 5.19 5.19 5.20 5.21 5.27 5.28 5.33 5.36 5.40 5.48  
## [73] 5.49 5.73 5.74 5.77 5.97 6.00 6.01 6.07 6.10 6.33 6.39 6.41  
## [85] 6.50 6.52 6.55 6.62 6.62 6.64 6.68 6.71 6.73 6.73 6.87 6.91  
## [97] 6.96 6.99 7.07 7.15 7.18 7.69 7.83 8.12 8.13 8.20 8.21 8.25  
## [109] 8.64 8.67 8.68 8.96 9.01 9.11 9.25 9.41 9.59 10.07 10.18 10.44  
## [121] 10.52 10.62 10.68 11.03 11.04 11.07 11.11 11.30 11.36 11.38 11.40 11.41  
## [133] 11.64 11.66 11.85 11.94 12.32 12.32 12.41 12.48 12.73 13.01 13.02 13.15  
## [145] 13.17 13.43 13.66 13.70 14.05 14.35 14.64 14.71 14.83 15.19 15.28 15.35  
## [157] 15.57 15.76 15.82 15.92 15.92 15.93 16.19 16.38 16.44 16.45 16.65 16.65  
## [169] 16.89 17.34 17.49 17.80 18.03 18.18 18.23 18.24 18.29 18.37 18.56 18.75  
## [181] 18.80 18.87 19.16 19.27 19.29 19.32 19.38 19.49 19.57 19.68 19.77 19.79  
## [193] 19.87 19.93 19.95 20.10 20.11 20.15 20.28 20.30 20.31 20.34 20.42 20.50  
## [205] 20.50 20.51 20.54 20.64 20.70 20.72 20.72 20.90 20.90 20.91 20.92 21.13  
## [217] 21.25 21.37 21.56 21.61 21.61 21.65 21.66 21.67 21.74 21.90 21.93 22.05  
## [229] 22.20 22.23 22.34 22.35 22.41 22.44 22.46 22.56 22.77 22.78 22.82 22.89  
## [241] 23.19 23.23 23.29 23.36 23.44 23.44 23.45 23.46 23.49 23.59 23.60 23.72  
## [253] 23.77 23.89 23.89 24.22 24.23 24.38 24.45 24.61 24.91 25.19 25.42 25.47  
## [265] 25.63 25.79 25.84 26.43 26.60 26.96 27.26 27.88 28.09 28.43 28.75 29.24  
## [277] 29.65 29.88 30.26 32.53

print(paste0("mortalidade\_infantil"))

## [1] "mortalidade\_infantil"

quantile(predic.IDHM$mortalidade\_infantil, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 9.920 11.130 13.021 14.684 15.910 17.028 18.073 19.362 21.453 30.410

sort(predic.IDHM$mortalidade\_infantil)

## [1] 7.82 8.12 8.28 8.39 8.45 8.61 8.65 8.81 8.88 8.93 8.97 9.05  
## [13] 9.10 9.13 9.23 9.23 9.32 9.35 9.39 9.52 9.60 9.60 9.64 9.67  
## [25] 9.74 9.84 9.89 9.92 9.92 9.95 10.02 10.15 10.17 10.18 10.18 10.21  
## [37] 10.21 10.37 10.46 10.46 10.48 10.49 10.51 10.53 10.56 10.57 10.62 10.74  
## [49] 10.80 10.80 10.81 10.88 10.95 11.04 11.09 11.13 11.13 11.18 11.25 11.28  
## [61] 11.42 11.42 11.42 11.46 11.53 11.55 11.57 11.78 11.95 12.00 12.03 12.03  
## [73] 12.18 12.21 12.35 12.42 12.63 12.63 12.66 12.72 12.78 12.82 12.88 12.93  
## [85] 13.06 13.14 13.21 13.27 13.30 13.31 13.34 13.37 13.42 13.61 13.70 13.84  
## [97] 13.86 13.90 13.93 14.02 14.07 14.11 14.12 14.13 14.22 14.22 14.22 14.48  
## [109] 14.48 14.56 14.57 14.60 14.74 14.75 14.76 14.84 14.94 14.94 15.00 15.02  
## [121] 15.04 15.11 15.16 15.27 15.28 15.37 15.39 15.44 15.44 15.44 15.46 15.52  
## [133] 15.52 15.53 15.71 15.79 15.85 15.88 15.89 15.91 15.91 15.97 16.03 16.11  
## [145] 16.18 16.23 16.24 16.25 16.26 16.30 16.34 16.35 16.42 16.44 16.47 16.49  
## [157] 16.59 16.60 16.67 16.76 16.76 16.78 16.81 16.83 16.84 16.96 16.97 17.00  
## [169] 17.07 17.12 17.13 17.13 17.14 17.16 17.21 17.29 17.32 17.41 17.42 17.45  
## [181] 17.47 17.51 17.55 17.58 17.68 17.68 17.69 17.75 17.78 17.80 17.83 17.85  
## [193] 17.89 17.94 18.00 18.07 18.08 18.09 18.09 18.10 18.17 18.19 18.21 18.24  
## [205] 18.27 18.34 18.39 18.48 18.51 18.66 18.71 18.72 18.77 18.90 18.91 18.93  
## [217] 19.04 19.09 19.10 19.13 19.16 19.25 19.31 19.32 19.53 19.57 19.65 19.67  
## [229] 19.67 19.69 19.70 19.76 19.87 20.02 20.04 20.12 20.18 20.20 20.21 20.27  
## [241] 20.31 20.34 20.49 20.54 20.55 20.63 20.88 20.99 21.03 21.04 21.24 21.45  
## [253] 21.48 21.65 21.66 21.93 22.00 22.30 22.35 22.41 22.55 22.56 22.57 22.81  
## [265] 23.01 23.23 23.38 23.44 23.66 23.73 23.88 24.10 24.25 24.33 24.73 24.98  
## [277] 26.14 26.33 27.28 30.41

print(paste0("media\_anos\_de\_estudo"))

## [1] "media\_anos\_de\_estudo"

quantile(predic.IDHM$media\_anos\_de\_estudo, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 7.459 7.880 8.227 8.600 8.840 9.144 9.490 9.820 10.183 12.200

sort(predic.IDHM$media\_anos\_de\_estudo)

## [1] 6.54 6.66 6.70 6.71 6.72 6.80 6.87 6.94 6.99 7.00 7.03 7.08  
## [13] 7.11 7.19 7.21 7.22 7.25 7.25 7.28 7.29 7.31 7.34 7.34 7.37  
## [25] 7.41 7.44 7.45 7.45 7.46 7.48 7.48 7.53 7.55 7.60 7.60 7.60  
## [37] 7.60 7.61 7.63 7.64 7.64 7.65 7.66 7.69 7.70 7.71 7.72 7.72  
## [49] 7.74 7.75 7.76 7.76 7.78 7.80 7.81 7.84 7.89 7.92 7.93 7.93  
## [61] 7.94 7.95 7.95 7.98 7.98 7.99 8.01 8.01 8.04 8.04 8.06 8.08  
## [73] 8.09 8.10 8.10 8.11 8.11 8.12 8.12 8.13 8.16 8.18 8.18 8.22  
## [85] 8.23 8.25 8.26 8.28 8.28 8.29 8.29 8.29 8.31 8.31 8.31 8.32  
## [97] 8.33 8.36 8.38 8.39 8.40 8.42 8.44 8.50 8.53 8.54 8.56 8.57  
## [109] 8.58 8.58 8.60 8.60 8.60 8.61 8.61 8.63 8.63 8.64 8.64 8.64  
## [121] 8.65 8.66 8.68 8.68 8.69 8.70 8.72 8.72 8.72 8.73 8.74 8.74  
## [133] 8.76 8.76 8.77 8.79 8.81 8.82 8.84 8.84 8.84 8.88 8.89 8.92  
## [145] 8.92 8.95 8.95 8.97 8.99 8.99 8.99 9.00 9.01 9.02 9.03 9.04  
## [157] 9.04 9.07 9.07 9.09 9.09 9.09 9.10 9.11 9.12 9.13 9.13 9.14  
## [169] 9.15 9.16 9.16 9.17 9.17 9.18 9.18 9.18 9.19 9.21 9.24 9.24  
## [181] 9.25 9.26 9.28 9.30 9.30 9.30 9.32 9.32 9.32 9.36 9.37 9.41  
## [193] 9.45 9.45 9.47 9.49 9.49 9.50 9.50 9.53 9.55 9.56 9.57 9.57  
## [205] 9.58 9.59 9.59 9.60 9.62 9.62 9.63 9.65 9.66 9.67 9.70 9.72  
## [217] 9.75 9.76 9.77 9.78 9.79 9.79 9.80 9.82 9.82 9.83 9.84 9.84  
## [229] 9.85 9.85 9.85 9.89 9.89 9.90 9.91 9.91 9.94 9.94 9.95 9.97  
## [241] 9.99 10.00 10.02 10.02 10.02 10.08 10.08 10.10 10.11 10.12 10.13 10.18  
## [253] 10.21 10.21 10.27 10.31 10.35 10.35 10.37 10.39 10.47 10.47 10.52 10.58  
## [265] 10.68 10.76 10.80 10.88 10.92 10.93 10.95 11.01 11.04 11.19 11.19 11.33  
## [277] 11.33 11.65 11.82 12.20

print(paste0("analfabetismo\_25\_anos"))

## [1] "analfabetismo\_25\_anos"

quantile(predic.IDHM$analfabetismo\_25\_anos, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 3.407 4.898 6.431 7.756 8.760 11.058 14.960 17.434 20.251 25.310

sort(predic.IDHM$analfabetismo\_25\_anos)

## [1] 1.90 2.01 2.17 2.18 2.25 2.26 2.36 2.40 2.56 2.59 2.60 2.71  
## [13] 2.95 2.95 2.96 2.97 3.03 3.10 3.10 3.12 3.17 3.19 3.20 3.30  
## [25] 3.31 3.35 3.36 3.38 3.41 3.41 3.42 3.45 3.46 3.50 3.54 3.57  
## [37] 3.68 3.69 3.70 3.71 3.73 3.79 3.82 3.86 3.87 3.91 4.21 4.22  
## [49] 4.24 4.28 4.36 4.62 4.71 4.74 4.80 4.85 4.91 4.98 5.13 5.18  
## [61] 5.20 5.27 5.36 5.42 5.47 5.49 5.52 5.54 5.58 5.59 5.61 5.82  
## [73] 5.83 5.86 5.96 5.99 6.01 6.18 6.21 6.21 6.26 6.30 6.32 6.41  
## [85] 6.44 6.51 6.51 6.52 6.54 6.65 6.68 6.72 6.83 6.86 6.95 6.96  
## [97] 6.97 7.04 7.05 7.07 7.07 7.28 7.29 7.42 7.42 7.45 7.60 7.63  
## [109] 7.64 7.64 7.66 7.69 7.80 7.81 7.82 7.87 7.88 7.88 7.90 7.90  
## [121] 7.93 7.98 7.99 8.02 8.03 8.13 8.23 8.24 8.25 8.26 8.38 8.40  
## [133] 8.41 8.55 8.55 8.56 8.56 8.63 8.66 8.71 8.81 8.83 8.91 8.91  
## [145] 8.97 8.99 8.99 9.00 9.03 9.11 9.30 9.32 9.38 9.43 9.46 9.52  
## [157] 9.77 9.87 10.08 10.21 10.22 10.28 10.41 10.41 10.59 10.67 10.99 11.05  
## [169] 11.07 11.17 11.60 11.80 11.84 11.92 12.05 12.28 12.41 12.58 12.60 12.73  
## [181] 12.77 12.77 13.39 13.44 13.46 13.49 13.92 14.00 14.24 14.37 14.43 14.61  
## [193] 14.71 14.76 14.79 14.93 15.03 15.14 15.15 15.34 15.35 15.42 15.57 15.73  
## [205] 15.79 15.84 15.90 15.98 16.09 16.15 16.18 16.36 16.37 16.38 16.41 16.44  
## [217] 16.45 16.48 16.70 16.75 16.96 17.02 17.13 17.40 17.57 17.76 17.91 17.92  
## [229] 17.96 17.98 18.17 18.22 18.50 18.53 18.55 18.76 18.98 19.06 19.08 19.16  
## [241] 19.25 19.27 19.33 19.35 19.41 19.42 19.51 19.69 19.77 19.97 20.05 20.25  
## [253] 20.26 20.28 20.47 20.52 20.59 20.95 21.13 21.38 21.44 21.47 21.51 21.58  
## [265] 21.97 22.37 22.43 22.44 22.55 22.75 23.25 23.72 23.78 24.04 24.22 24.65  
## [277] 24.78 24.96 24.97 25.31

print(paste0("analfabetismo\_18\_anos"))

## [1] "analfabetismo\_18\_anos"

quantile(predic.IDHM$analfabetismo\_18\_anos, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 2.930 4.204 5.447 6.502 7.225 9.186 12.629 14.572 16.966 21.200

sort(predic.IDHM$analfabetismo\_18\_anos)

## [1] 1.68 1.83 1.90 1.95 1.95 1.97 2.13 2.14 2.25 2.27 2.31 2.41  
## [13] 2.59 2.61 2.61 2.63 2.67 2.69 2.71 2.71 2.75 2.79 2.84 2.84  
## [25] 2.85 2.86 2.90 2.93 2.93 2.94 2.94 2.94 3.01 3.02 3.08 3.10  
## [37] 3.14 3.18 3.26 3.27 3.28 3.30 3.31 3.35 3.36 3.37 3.65 3.69  
## [49] 3.72 3.79 3.84 4.05 4.05 4.12 4.13 4.18 4.21 4.25 4.29 4.50  
## [61] 4.53 4.56 4.63 4.64 4.64 4.72 4.72 4.74 4.76 4.79 4.80 4.95  
## [73] 4.97 5.02 5.05 5.07 5.24 5.24 5.26 5.31 5.32 5.35 5.36 5.44  
## [85] 5.45 5.48 5.53 5.56 5.58 5.60 5.62 5.66 5.67 5.81 5.85 5.91  
## [97] 5.92 5.93 5.95 5.98 5.99 6.01 6.04 6.08 6.19 6.24 6.37 6.40  
## [109] 6.43 6.44 6.46 6.49 6.51 6.56 6.57 6.61 6.61 6.62 6.62 6.63  
## [121] 6.63 6.70 6.72 6.72 6.78 6.81 6.82 6.87 6.94 7.03 7.03 7.03  
## [133] 7.04 7.07 7.09 7.09 7.11 7.11 7.12 7.22 7.23 7.32 7.33 7.42  
## [145] 7.45 7.47 7.57 7.65 7.73 7.73 7.87 7.88 7.89 7.96 7.97 8.21  
## [157] 8.26 8.29 8.41 8.44 8.47 8.59 8.67 8.75 8.93 9.04 9.04 9.15  
## [169] 9.24 9.50 9.60 9.74 9.90 9.98 10.29 10.36 10.44 10.52 10.75 11.04  
## [181] 11.05 11.07 11.08 11.09 11.12 11.53 11.70 11.84 12.03 12.06 12.07 12.08  
## [193] 12.12 12.18 12.40 12.62 12.65 12.89 12.90 13.03 13.13 13.15 13.19 13.23  
## [205] 13.27 13.36 13.46 13.68 13.68 13.73 13.83 13.83 13.84 13.90 13.91 13.92  
## [217] 13.95 14.03 14.07 14.07 14.14 14.39 14.51 14.54 14.70 14.72 14.98 15.16  
## [229] 15.17 15.18 15.22 15.46 15.46 15.57 15.58 15.61 15.81 15.89 15.89 16.07  
## [241] 16.12 16.19 16.29 16.36 16.40 16.40 16.45 16.64 16.77 16.80 16.85 16.96  
## [253] 17.02 17.05 17.32 17.37 17.45 17.46 17.47 18.16 18.24 18.28 18.32 18.42  
## [265] 18.68 18.72 18.73 18.99 19.25 19.28 19.37 19.86 20.16 20.35 20.35 20.54  
## [277] 20.73 20.88 21.06 21.20

print(paste0("analfabetismo\_15\_anos"))

## [1] "analfabetismo\_15\_anos"

quantile(predic.IDHM$analfabetismo\_15\_anos, probs = seq(.1, 1, by = .1))

## 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%   
## 2.769 3.958 5.148 6.134 6.810 8.562 11.726 13.514 15.741 19.600

sort(predic.IDHM$analfabetismo\_15\_anos)

## [1] 1.60 1.77 1.84 1.89 1.89 1.94 2.06 2.07 2.14 2.18 2.23 2.32  
## [13] 2.46 2.48 2.50 2.51 2.54 2.57 2.57 2.58 2.62 2.66 2.68 2.69  
## [25] 2.69 2.71 2.76 2.76 2.77 2.77 2.80 2.82 2.84 2.88 2.93 2.97  
## [37] 2.97 3.00 3.10 3.12 3.12 3.13 3.16 3.19 3.20 3.20 3.43 3.49  
## [49] 3.51 3.63 3.67 3.85 3.87 3.90 3.95 3.95 3.96 4.03 4.05 4.24  
## [61] 4.30 4.32 4.36 4.41 4.41 4.46 4.46 4.50 4.53 4.56 4.56 4.61  
## [73] 4.70 4.73 4.78 4.82 4.87 4.94 4.97 4.98 5.00 5.00 5.00 5.05  
## [85] 5.19 5.19 5.21 5.23 5.23 5.31 5.33 5.34 5.39 5.41 5.46 5.48  
## [97] 5.57 5.59 5.59 5.60 5.61 5.64 5.64 5.76 5.77 5.85 5.89 5.90  
## [109] 6.00 6.07 6.09 6.11 6.15 6.16 6.17 6.20 6.20 6.20 6.21 6.22  
## [121] 6.24 6.27 6.27 6.28 6.37 6.37 6.40 6.41 6.41 6.44 6.46 6.47  
## [133] 6.54 6.54 6.57 6.61 6.62 6.68 6.69 6.81 6.81 6.81 6.84 6.85  
## [145] 6.96 7.06 7.07 7.18 7.22 7.22 7.27 7.28 7.29 7.31 7.50 7.63  
## [157] 7.75 7.77 7.79 7.84 7.93 7.93 8.13 8.21 8.33 8.41 8.53 8.55  
## [169] 8.58 8.89 8.90 8.99 9.20 9.30 9.70 9.74 9.80 9.80 10.11 10.20  
## [181] 10.21 10.31 10.37 10.46 10.59 10.75 10.95 11.14 11.16 11.17 11.31 11.32  
## [193] 11.44 11.48 11.49 11.63 11.95 12.08 12.12 12.15 12.18 12.29 12.32 12.48  
## [205] 12.48 12.51 12.54 12.63 12.76 12.93 12.95 12.98 13.00 13.01 13.05 13.06  
## [217] 13.16 13.19 13.23 13.23 13.27 13.38 13.45 13.48 13.65 13.72 14.10 14.14  
## [229] 14.19 14.23 14.28 14.29 14.34 14.35 14.42 14.61 14.67 14.72 14.89 14.98  
## [241] 15.10 15.20 15.25 15.28 15.30 15.40 15.42 15.49 15.56 15.61 15.72 15.73  
## [253] 15.84 15.93 16.17 16.20 16.23 16.24 16.30 17.01 17.02 17.03 17.05 17.15  
## [265] 17.19 17.32 17.49 17.68 17.70 17.99 17.99 18.38 18.70 18.87 18.97 19.02  
## [277] 19.32 19.33 19.40 19.60

Machine Learning – parte Final

library(plyr)  
library(caret)

## Warning: package 'caret' was built under R version 4.3.1

## Carregando pacotes exigidos: lattice

library(randomForest)

## Warning: package 'randomForest' was built under R version 4.3.1

## randomForest 4.7-1.1

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':  
##   
## combine

## The following object is masked from 'package:ggplot2':  
##   
## margin

library(caTools)

## Warning: package 'caTools' was built under R version 4.3.1

# —————– SUPERVISED RANDOM FOREST REGRESSION —————–

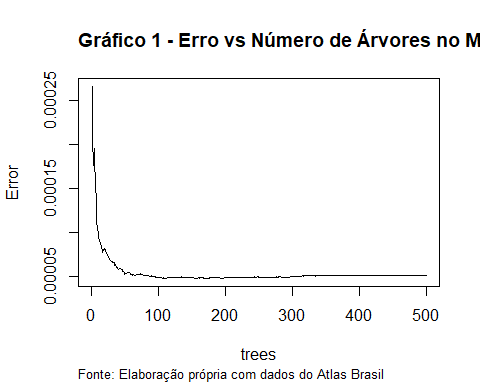
#usar o data frame fina da análise exploratória (predic.IDHM), e selecionar os dados para treino e teste

#usar o data frame fina da análise exploratória (predic.IDHM), e selecionar os dados para treino (80%) e teste (20%)  
  
set.seed(123)  
amostra.IDHM <- predic.IDHM$IDHM %>%  
 createDataPartition(p = 0.8, list = FALSE)  
treino.IDHM <- predic.IDHM[amostra.IDHM, ]  
teste.IDHM <- predic.IDHM[-amostra.IDHM, ]

# random forest para regressão, iniciando com 500 arvores e mtry of 3  
IDHM.model.1 <- randomForest(IDHM ~ ., data = treino.IDHM, ntree=500, mtry = 3,   
 importance = TRUE, na.action = na.omit)   
print(IDHM.model.1)

##   
## Call:  
## randomForest(formula = IDHM ~ ., data = treino.IDHM, ntree = 500, mtry = 3, importance = TRUE, na.action = na.omit)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 3  
##   
## Mean of squared residuals: 5.068181e-05  
## % Var explained: 97.6

# Plotar erro vs numero de arvores   
plot(IDHM.model.1, main = "")  
  
# Adicionar título e fonte no estilo ABNT  
title(main = "Gráfico 1 - Erro vs Número de Árvores no Modelo 1", adj = 0)  
mtext("Fonte: Elaboração própria com dados do Atlas Brasil", side=1, line=4, adj = 0, cex=0.8)



#Usar tuneRF para determinar se há melhor mtry na tentativa de encontrar o valor que produz o menor erro.  
mtry <- tuneRF(treino.IDHM[-6],treino.IDHM$IDHM, ntreeTry=500,  
 stepFactor=1,improve=0.01, trace=TRUE, plot=FALSE)

## mtry = 3 OOB error = 1.782552e-05   
## Searching left ...  
## Searching right ...

print(mtry)

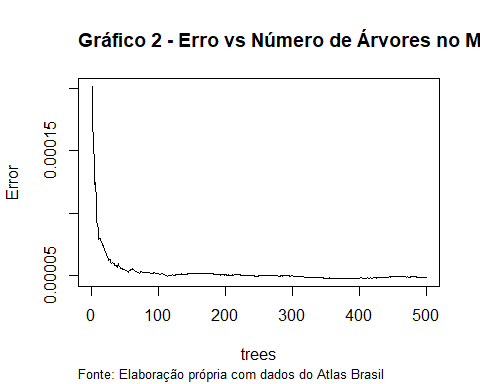
## mtry OOBError  
## 3 3 1.782552e-05

melhor mtry = 4

#o valor ótimo para mtry é 4, que produz o menor erro.  
  
set.seed(123)  
# random forest para regressão com mtry=4  
IDHM.model.2 <- randomForest(IDHM ~ ., data = treino.IDHM, ntree=500, mtry = 4,   
 importance = TRUE, na.action = na.omit)   
print(IDHM.model.2)

##   
## Call:  
## randomForest(formula = IDHM ~ ., data = treino.IDHM, ntree = 500, mtry = 4, importance = TRUE, na.action = na.omit)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 4  
##   
## Mean of squared residuals: 4.829864e-05  
## % Var explained: 97.71

# Plot the error vs the number of trees graph   
plot(IDHM.model.2, main = "")  
  
# Adicionar título e fonte no estilo ABNT  
title(main = "Gráfico 2 - Erro vs Número de Árvores no Modelo 2", adj = 0)  
mtext("Fonte: Elaboração própria com dados do Atlas Brasil", side=1, line=4, adj = 0, cex=0.8)



desenvolver e avaliar ambos os modelos

# Fazer predições com dados de teste usando modelo 1 (mtry = 3)  
IDHM.predições.1 <- IDHM.model.1 %>% predict(teste.IDHM)  
head(IDHM.predições.1)

## 4 5 6 21 25 26   
## 0.7146746 0.7019420 0.6878327 0.7019142 0.7868317 0.7983661

# Fazer predições com dados de teste usando modelo 2 (mtry = 4)  
IDHM.predições.2 <- IDHM.model.2 %>% predict(teste.IDHM)  
head(IDHM.predições.2)

## 4 5 6 21 25 26   
## 0.7141945 0.7021202 0.6884029 0.7004927 0.7869232 0.8016628

# Calcular o erro médio de previsão -- erro quadrático médio da raiz (RMSE) de ambos os modelos  
RMSE(IDHM.predições.1, teste.IDHM$IDHM)

## [1] 0.008417118

RMSE(IDHM.predições.2, teste.IDHM$IDHM)

## [1] 0.00817621

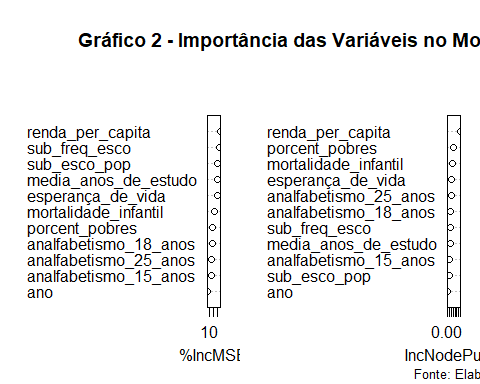
### O modelo original com mtry=4 (hdi.rf.1) na verdade tem um RMSE maior, portanto,o modelo 2 é o melhor modelo a ser usado daqui para frente. Um RMSE de 0.008546427 é consideravelmente baixo e indica um modelo de previsão altamente válido. analisar a significância de cada variável para ver possíveis mudanças na média.

avaliar importância das variáveis

#avaliar a importância das variáveis para o modelo   
importance(IDHM.model.1)

## %IncMSE IncNodePurity  
## ano 7.868941 0.002079949  
## renda\_per\_capita 23.030204 0.136166918  
## sub\_esco\_pop 20.115061 0.016921115  
## sub\_freq\_esco 21.777708 0.029760092  
## esperança\_de\_vida 19.615980 0.045243865  
## porcent\_pobres 13.947132 0.063730867  
## mortalidade\_infantil 16.330721 0.056961273  
## media\_anos\_de\_estudo 19.724511 0.025218818  
## analfabetismo\_25\_anos 12.473426 0.040472876  
## analfabetismo\_18\_anos 13.835095 0.030522313  
## analfabetismo\_15\_anos 11.938146 0.024789071

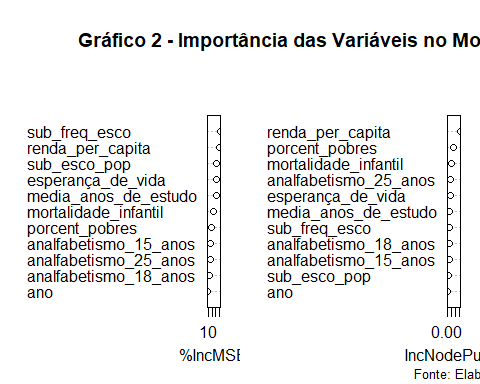
varImpPlot(IDHM.model.1, main = "")  
  
# Adicionar título e fonte no estilo ABNT  
title(main = "Gráfico 2 - Importância das Variáveis no Modelo Random Forest", adj = 0)  
mtext("Fonte: Elaboração própria com dados do Atlas Brasil", side=1, line=4, adj = 0, cex=0.8)



#avaliar a importância das variáveis para o modelo   
importance(IDHM.model.2)

## %IncMSE IncNodePurity  
## ano 8.530811 0.001727589  
## renda\_per\_capita 25.042790 0.160428577  
## sub\_esco\_pop 21.549152 0.016031995  
## sub\_freq\_esco 26.672853 0.023984730  
## esperança\_de\_vida 21.082599 0.034064658  
## porcent\_pobres 14.087930 0.074457295  
## mortalidade\_infantil 16.680057 0.052798519  
## media\_anos\_de\_estudo 21.035407 0.025611437  
## analfabetismo\_25\_anos 12.204311 0.041127026  
## analfabetismo\_18\_anos 11.118136 0.022789319  
## analfabetismo\_15\_anos 12.844512 0.019948849

varImpPlot(IDHM.model.2, main = "")  
  
# Adicionar título e fonte no estilo ABNT  
title(main = "Gráfico 2 - Importância das Variáveis no Modelo Random Forest", adj = 0)  
mtext("Fonte: Elaboração própria com dados do Atlas Brasil", side=1, line=4, adj = 0, cex=0.8)



###Join predicions to test table \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Converter predições para um data frame   
IDHM.predições.df <- as.data.frame(IDHM.predições.1)  
# Mesclar com base no índice  
IDHM.predições.df <- merge(teste.IDHM, IDHM.predições.df, by.x = 0, by.y = 0, all.x = TRUE, all.y = TRUE)  
# Criar uma nova coluna calculada com a diferença da predição do IDHM, e o valor Real  
IDHM.predições.df$diff <- with(IDHM.predições.df, IDHM.predições.df$IDHM - IDHM.predições.df$IDHM.predições.1)  
# Obter a média da diferença  
IDHM.predições.df

## Row.names ano renda\_per\_capita sub\_esco\_pop sub\_freq\_esco esperança\_de\_vida  
## 1 117 2016 467.19 0.681 0.717 71.85  
## 2 119 2016 473.32 0.585 0.779 73.72  
## 3 120 2016 1456.83 0.813 0.826 78.02  
## 4 121 2016 738.04 0.654 0.774 78.22  
## 5 122 2016 722.67 0.655 0.809 74.09  
## 6 131 2016 469.93 0.513 0.719 70.94  
## 7 137 2016 921.16 0.686 0.840 79.09  
## 8 147 2017 503.66 0.585 0.791 73.94  
## 9 152 2017 791.40 0.628 0.752 75.74  
## 10 153 2017 766.18 0.652 0.816 74.44  
## 11 154 2017 757.43 0.631 0.821 77.49  
## 12 157 2017 447.25 0.595 0.691 72.22  
## 13 163 2017 596.82 0.598 0.756 71.50  
## 14 168 2017 575.58 0.631 0.783 73.56  
## 15 174 2018 499.19 0.564 0.714 73.87  
## 16 182 2018 767.86 0.623 0.820 77.73  
## 17 196 2018 617.00 0.618 0.811 73.81  
## 18 207 2019 366.24 0.578 0.777 71.39  
## 19 21 2012 478.88 0.519 0.673 74.59  
## 20 216 2019 1044.95 0.760 0.790 77.04  
## 21 217 2019 595.13 0.599 0.739 76.29  
## 22 218 2019 1047.74 0.673 0.768 78.55  
## 23 224 2019 598.77 0.630 0.813 74.05  
## 24 225 2020 779.13 0.695 0.807 76.21  
## 25 229 2020 478.39 0.727 0.759 71.99  
## 26 230 2020 547.09 0.617 0.729 73.66  
## 27 234 2020 713.76 0.692 0.844 70.76  
## 28 239 2020 497.15 0.568 0.732 73.93  
## 29 244 2020 961.32 0.775 0.784 73.24  
## 30 245 2020 603.61 0.627 0.756 76.28  
## 31 247 2020 640.90 0.652 0.793 71.20  
## 32 25 2012 927.60 0.626 0.805 77.70  
## 33 252 2020 588.48 0.664 0.818 74.02  
## 34 257 2021 432.99 0.739 0.710 69.61  
## 35 26 2012 1037.42 0.703 0.840 76.80  
## 36 263 2021 341.32 0.618 0.770 67.90  
## 37 265 2021 707.48 0.706 0.785 68.77  
## 38 274 2021 944.53 0.708 0.772 72.84  
## 39 34 2013 508.14 0.520 0.635 72.66  
## 40 36 2013 1568.87 0.783 0.799 77.17  
## 41 4 2012 528.23 0.670 0.653 72.77  
## 42 45 2013 484.80 0.534 0.624 71.44  
## 43 47 2013 503.95 0.496 0.699 70.38  
## 44 48 2013 888.32 0.698 0.723 75.23  
## 45 5 2012 559.44 0.613 0.642 70.85  
## 46 55 2013 567.26 0.536 0.631 71.76  
## 47 56 2013 550.02 0.599 0.742 72.42  
## 48 6 2012 503.29 0.510 0.639 72.39  
## 49 60 2014 575.63 0.687 0.689 73.37  
## 50 65 2014 807.78 0.633 0.751 77.54  
## 51 75 2014 502.70 0.507 0.718 70.57  
## 52 90 2015 520.00 0.556 0.687 73.16  
## 53 91 2015 478.58 0.562 0.763 73.49  
## 54 93 2015 758.07 0.645 0.770 77.89  
## porcent\_pobres mortalidade\_infantil media\_anos\_de\_estudo  
## 1 25.19 18.34 9.11  
## 2 21.93 14.48 7.65  
## 3 4.41 10.57 11.19  
## 4 8.21 8.81 9.16  
## 5 5.77 15.02 9.04  
## 6 22.82 19.25 7.34  
## 7 2.86 9.23 9.49  
## 8 20.72 13.86 8.04  
## 9 5.40 13.70 9.21  
## 10 6.07 16.60 8.99  
## 11 7.18 10.46 9.04  
## 12 20.54 16.18 8.33  
## 13 8.64 19.70 8.36  
## 14 11.66 15.44 8.70  
## 15 21.67 16.11 8.12  
## 16 6.71 10.02 9.07  
## 17 13.15 15.00 8.74  
## 18 30.26 18.71 7.84  
## 19 17.80 18.21 7.44  
## 20 6.64 10.46 10.76  
## 21 19.57 13.14 8.68  
## 22 3.68 8.88 9.90  
## 23 14.05 14.57 8.89  
## 24 9.41 14.02 9.82  
## 25 20.50 19.76 10.02  
## 26 16.65 18.09 8.72  
## 27 4.87 17.45 9.79  
## 28 15.92 16.34 8.29  
## 29 8.20 13.06 10.93  
## 30 12.32 15.39 9.10  
## 31 6.99 22.41 9.24  
## 32 3.00 10.51 8.95  
## 33 11.40 17.13 9.24  
## 34 25.79 21.66 10.10  
## 35 3.86 11.25 9.77  
## 36 32.53 23.38 8.32  
## 37 6.39 22.35 9.83  
## 38 4.35 12.18 10.35  
## 39 19.79 20.02 7.34  
## 40 4.20 11.28 10.95  
## 41 18.37 24.33 9.09  
## 42 19.49 18.39 7.64  
## 43 19.16 21.24 6.87  
## 44 6.01 12.78 9.80  
## 45 22.23 20.88 8.63  
## 46 18.75 19.09 7.41  
## 47 13.17 17.55 8.11  
## 48 22.56 21.04 7.25  
## 49 13.70 23.88 9.30  
## 50 6.91 9.67 9.02  
## 51 20.31 20.54 7.11  
## 52 18.03 18.24 7.93  
## 53 20.34 15.16 7.48  
## 54 8.13 9.23 9.07  
## analfabetismo\_25\_anos analfabetismo\_18\_anos analfabetismo\_15\_anos IDHM  
## 1 8.63 7.09 6.57 0.711  
## 2 17.98 15.16 14.19 0.722  
## 3 3.17 2.71 2.57 0.847  
## 4 6.97 5.95 5.59 0.779  
## 5 7.07 5.98 5.61 0.764  
## 6 21.58 18.28 17.03 0.685  
## 7 3.31 2.85 2.69 0.814  
## 8 16.75 14.07 13.19 0.730  
## 9 5.86 5.02 4.73 0.762  
## 10 7.63 6.49 6.11 0.770  
## 11 6.95 5.92 5.59 0.784  
## 12 11.07 9.24 8.58 0.694  
## 13 8.56 7.12 6.62 0.721  
## 14 12.41 10.44 9.70 0.740  
## 15 15.34 13.15 12.32 0.710  
## 16 6.96 6.08 5.77 0.784  
## 17 11.60 9.98 9.30 0.749  
## 18 18.50 15.46 14.34 0.694  
## 19 19.25 16.19 15.25 0.695  
## 20 2.59 2.31 2.23 0.809  
## 21 15.14 12.89 12.29 0.742  
## 22 3.20 2.84 2.71 0.801  
## 23 11.17 9.50 8.90 0.751  
## 24 6.52 5.67 5.39 0.784  
## 25 5.99 4.97 4.61 0.727  
## 26 12.58 11.04 10.37 0.724  
## 27 4.85 4.25 4.05 0.758  
## 28 16.45 13.83 13.05 0.714  
## 29 2.17 1.95 1.89 0.785  
## 30 12.77 11.12 10.59 0.750  
## 31 5.61 4.74 4.46 0.739  
## 32 4.24 3.65 3.43 0.792  
## 33 10.59 8.93 8.33 0.755  
## 34 5.59 4.64 4.36 0.700  
## 35 4.22 3.69 3.49 0.812  
## 36 14.93 12.40 11.49 0.676  
## 37 5.36 4.56 4.30 0.736  
## 38 2.36 2.14 2.07 0.771  
## 39 18.76 15.81 14.89 0.680  
## 40 3.45 2.86 2.69 0.837  
## 41 7.93 6.37 5.76 0.707  
## 42 13.44 11.07 10.20 0.671  
## 43 24.96 20.88 19.32 0.679  
## 44 3.57 3.14 2.97 0.768  
## 45 9.46 7.89 7.22 0.691  
## 46 19.41 16.12 14.98 0.683  
## 47 15.35 12.62 11.63 0.719  
## 48 19.27 16.29 15.28 0.678  
## 49 8.91 7.11 6.54 0.725  
## 50 7.69 6.57 6.20 0.772  
## 51 23.78 20.16 18.87 0.686  
## 52 15.98 13.68 12.76 0.701  
## 53 19.35 16.40 15.30 0.715  
## 54 7.66 6.63 6.21 0.776  
## IDHM.predições.1 diff  
## 1 0.7095903 0.0014097333  
## 2 0.7150993 0.0069006667  
## 3 0.8385089 0.0084910667  
## 4 0.7800819 -0.0010818667  
## 5 0.7665971 -0.0025971333  
## 6 0.6886255 -0.0036255333  
## 7 0.8100509 0.0039491333  
## 8 0.7231550 0.0068450000  
## 9 0.7685123 -0.0065123333  
## 10 0.7674915 0.0025085333  
## 11 0.7804120 0.0035880000  
## 12 0.7030948 -0.0090948333  
## 13 0.7264488 -0.0054487667  
## 14 0.7371906 0.0028094000  
## 15 0.7119679 -0.0019679000  
## 16 0.7811909 0.0028091000  
## 17 0.7407610 0.0082389667  
## 18 0.6964571 -0.0024571000  
## 19 0.7019142 -0.0069142333  
## 20 0.8222143 -0.0132143333  
## 21 0.7351314 0.0068686000  
## 22 0.8130208 -0.0120208000  
## 23 0.7404551 0.0105449000  
## 24 0.7732565 0.0107435000  
## 25 0.7297161 -0.0027161333  
## 26 0.7258185 -0.0018185000  
## 27 0.7605134 -0.0025134000  
## 28 0.7206246 -0.0066246000  
## 29 0.7920688 -0.0070688333  
## 30 0.7399933 0.0100067333  
## 31 0.7413534 -0.0023533667  
## 32 0.7868317 0.0051683000  
## 33 0.7424384 0.0125615667  
## 34 0.7022753 -0.0022753000  
## 35 0.7983661 0.0136339333  
## 36 0.7008335 -0.0248335333  
## 37 0.7493369 -0.0133369000  
## 38 0.7891303 -0.0181302667  
## 39 0.6903452 -0.0103452000  
## 40 0.8314648 0.0055351667  
## 41 0.7146746 -0.0076745667  
## 42 0.6847285 -0.0137285333  
## 43 0.6693097 0.0096902667  
## 44 0.7758335 -0.0078334667  
## 45 0.7019420 -0.0109420000  
## 46 0.6890759 -0.0060759333  
## 47 0.7146250 0.0043750000  
## 48 0.6878327 -0.0098327000  
## 49 0.7261022 -0.0011021667  
## 50 0.7740489 -0.0020489333  
## 51 0.6744937 0.0115063000  
## 52 0.7059610 -0.0049609667  
## 53 0.7103681 0.0046319333  
## 54 0.7765281 -0.0005280667

mean(IDHM.predições.df[,"diff"])

## [1] -0.00127523

### visualizando as predições do modelo em comparação com os valores reais em plots  
# Redefinir o índice de linha(row.names)  
rownames(IDHM.predições.df ) <- NULL  
# Ordenar os dados (sort)  
IDHM.predições.df <- IDHM.predições.df[order(IDHM.predições.df$IDHM),]  
# Plotar as predições versus o IDHM real  
plot(IDHM.predições.df$IDHM.predições.1, type = "l", col="red",   
 xlab="Dados Testados", ylab="Real vs. Predição", main="")  
  
# Adicionar linhas do IDHM Real  
lines(IDHM.predições.df$IDHM, lwd=2)  
  
# Adicionar a legenda  
legend("topright",   
 legend=c("Predição IDHM", "IDHM Real"),  
 col=c("red", "black"),  
 lty=1,  
 cex=0.8)  
  
# Adicionar título e fonte no estilo ABNT  
title(main = "Gráfico 3 - Variação da Predição do IDHM", adj = 0)  
mtext("Fonte: Elaboração própria com dados do Atlas Brasil", side=1, line=4, adj = 0, cex=0.8)

