analysis.R

zuiuz

2021-10-27

#### Data processing ####  
library(foreign)  
data\_dir <- "../data/vaestotieto.sav"  
path <- file.path(data\_dir)  
dataset <- read.spss(path, to.data.frame=TRUE)

## re-encoding from CP1252

# Remove data of regions, or groups of country (Valtioiden muodostama alue)   
dataset.analysis <- subset(dataset, V3 != "Valtioiden muodostama alue")  
summary(dataset.analysis)

## V1 V2 V3   
## Length:216 Length:216 Itsenäinen valtio :194   
## Class :character Class :character Valtioiden muodostama alue: 0   
## Mode :character Mode :character Muu alue/territorio : 22   
##   
##   
##   
##   
## V4 V5 V6 V7   
## Min. : 10 Ei :148 Matala :34 Ei :150   
## 1st Qu.: 11610 Kyllä: 47 Alempi keskitaso:46 Kyllä: 44   
## Median : 96320 NA's : 21 Ylempi keskitaso:56 NA's : 22   
## Mean : 609099 Korkea :79   
## 3rd Qu.: 446300 NA's : 1   
## Max. :16376870   
## NA's :7   
## V9 V10 V11 V12   
## Min. :1.151e+04 Min. : 0.136 Min. : 13.03 Min. : 6.70   
## 1st Qu.:7.667e+05 1st Qu.: 37.463 1st Qu.: 42.06 1st Qu.:11.14   
## Median :6.679e+06 Median : 91.178 Median : 61.83 Median :17.56   
## Mean :3.516e+07 Mean : 359.367 Mean : 60.83 Mean :19.94   
## 3rd Qu.:2.503e+07 3rd Qu.: 219.079 3rd Qu.: 80.44 3rd Qu.:27.22   
## Max. :1.393e+09 Max. :20777.500 Max. :100.00 Max. :46.54   
## NA's :1 NA's :8 NA's :3 NA's :12   
## V13 V14 V15 V16   
## Min. : 1.169 Min. : 1.60 Min. : 17.52 Min. : 0.01413   
## 1st Qu.: 5.825 1st Qu.: 6.20 1st Qu.: 48.12 1st Qu.: 2.23888   
## Median : 7.236 Median :14.60 Median : 54.36 Median : 7.28794   
## Mean : 7.625 Mean :22.19 Mean : 58.61 Mean :17.83908   
## 3rd Qu.: 9.116 3rd Qu.:33.60 3rd Qu.: 67.62 3rd Qu.:28.41373   
## Max. :15.500 Max. :85.90 Max. :110.88 Max. :74.26484   
## NA's :12 NA's :24 NA's :24 NA's :139   
## V17 V18 V19 V20   
## Min. : 0.000 Min. : 0.00711 Min. : 0.110 Min. : 293   
## 1st Qu.: 1.533 1st Qu.: 2.13621 1st Qu.: 3.531 1st Qu.: 2211   
## Median : 4.798 Median : 5.61673 Median : 5.348 Median : 6331   
## Mean :12.329 Mean :15.10099 Mean : 7.021 Mean : 17219   
## 3rd Qu.:16.699 3rd Qu.:23.42948 3rd Qu.: 9.215 3rd Qu.: 20700   
## Max. :59.735 Max. :65.47724 Max. :26.920 Max. :173356   
## NA's :139 NA's :139 NA's :30 NA's :14   
## V21 V22 V23 V24   
## Min. :1.455 Min. : 0.000 Min. : 1.643 Min. : 27.41   
## 1st Qu.:3.355 1st Qu.: 0.957 1st Qu.: 4.528 1st Qu.: 89.76   
## Median :4.477 Median : 1.422 Median : 6.469 Median :112.70   
## Mean :4.448 Mean : 1.809 Mean : 6.643 Mean :111.03   
## 3rd Qu.:5.414 3rd Qu.: 2.116 3rd Qu.: 8.181 3rd Qu.:132.09   
## Max. :7.963 Max. :10.224 Max. :17.728 Max. :345.32   
## NA's :100 NA's :65 NA's :30 NA's :35   
## V25 V26 V27 V28   
## Min. :54.35 Min. :49.84 Min. :52.24 Min. : 0.100   
## 1st Qu.:69.31 1st Qu.:65.28 1st Qu.:67.30 1st Qu.: 0.200   
## Median :77.10 Median :70.91 Median :74.10 Median : 0.400   
## Mean :75.01 Mean :70.12 Mean :72.53 Mean : 2.093   
## 3rd Qu.:80.76 3rd Qu.:75.99 3rd Qu.:78.07 3rd Qu.: 1.375   
## Max. :87.60 Max. :81.90 Max. :84.68 Max. :28.400   
## NA's :18 NA's :18 NA's :18 NA's :94   
## V29 V30 V6\_1 V6\_2   
## Min. : 11 Min. : 5.861 Min. :0.0000 Min. :0.000   
## 1st Qu.: 1733 1st Qu.:13.404 1st Qu.:0.0000 1st Qu.:0.000   
## Median : 9787 Median :21.569 Median :0.0000 Median :0.000   
## Mean : 161976 Mean :27.051 Mean :0.1581 Mean :0.214   
## 3rd Qu.: 53989 3rd Qu.:34.779 3rd Qu.:0.0000 3rd Qu.:0.000   
## Max. :9893038 Max. :99.734 Max. :1.0000 Max. :1.000   
## NA's :13 NA's :23 NA's :1 NA's :1   
## V6\_3 V6\_4 filter\_.   
## Min. :0.0000 Min. :0.0000 Not Selected: 22   
## 1st Qu.:0.0000 1st Qu.:0.0000 Selected :194   
## Median :0.0000 Median :0.0000   
## Mean :0.2605 Mean :0.3674   
## 3rd Qu.:1.0000 3rd Qu.:1.0000   
## Max. :1.0000 Max. :1.0000   
## NA's :1 NA's :1

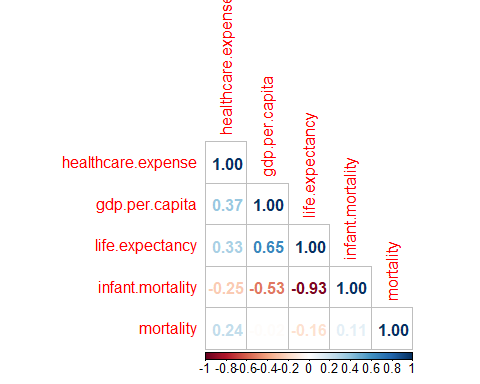
# Explanatory variables:   
# 1. V20-GDP per capita,   
# 2. V23 healthcare expense,   
gdp.per.capita <- dataset.analysis$V20  
healthcare.expense <- dataset.analysis$V23  
  
# Dependent variables:   
# 1. V27-life expectancy  
# 2. V14 infant mortality  
# 3. V13-mortality  
life.expectancy <- dataset.analysis$V27  
infant.mortality <- dataset.analysis$V14  
mortality <- dataset.analysis$V13  
  
# Create new dataframe to analyse and remove rows with NA value  
countries <- dataset.analysis$V1  
dataset.analysis <-na.omit(data.frame(countries,  
 gdp.per.capita,   
 healthcare.expense,  
 life.expectancy,  
 infant.mortality,  
 mortality,  
 stringsAsFactors=FALSE))  
row.names(dataset.analysis) <- NULL  
write.csv(dataset.analysis, file="../output/data/clean\_data.csv", row.names=FALSE)  
summary(dataset.analysis)

## countries gdp.per.capita healthcare.expense life.expectancy  
## Length:174 Min. : 293 Min. : 2.270 Min. :52.24   
## Class :character 1st Qu.: 1862 1st Qu.: 4.466 1st Qu.:67.02   
## Mode :character Median : 5413 Median : 6.421 Median :73.75   
## Mean : 13570 Mean : 6.510 Mean :72.15   
## 3rd Qu.: 15826 3rd Qu.: 8.135 3rd Qu.:77.57   
## Max. :107627 Max. :17.004 Max. :84.10   
## infant.mortality mortality   
## Min. : 1.70 Min. : 1.169   
## 1st Qu.: 6.20 1st Qu.: 5.849   
## Median :14.40 Median : 7.236   
## Mean :22.04 Mean : 7.700   
## 3rd Qu.:34.27 3rd Qu.: 9.200   
## Max. :85.90 Max. :15.500

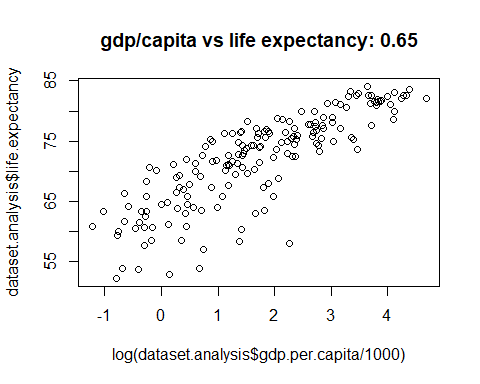
#### Exploratory data analysis ####  
# Calculate correlation coefficients between all variables  
# Get correlation matrix between 5 columns   
cor\_matrix <- cor(dataset.analysis[,2:6])  
  
library(corrplot)

## corrplot 0.90 loaded

corrplot(cor\_matrix, method="number", type="lower", order="hclust")



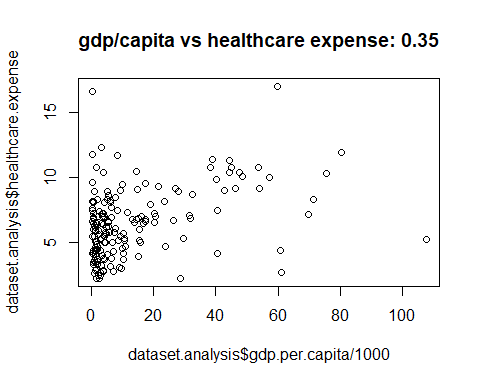
plot(log(dataset.analysis$gdp.per.capita/1000),  
 dataset.analysis$life.expectancy,  
 main="gdp/capita vs life expectancy: 0.65")



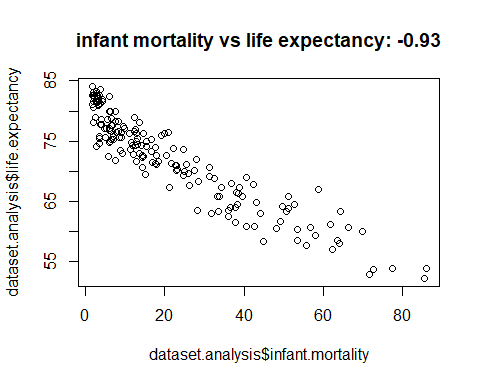
# Strong relationship between life expectancy and log of gdp/capita  
cor(dataset.analysis$life.expectancy,   
 dataset.analysis$gdp.per.capita,   
 method="spearman")

## [1] 0.8631761

plot(dataset.analysis$gdp.per.capita/1000,  
 dataset.analysis$healthcare.expense,  
 main="gdp/capita vs healthcare expense: 0.35")



# Weak relationship  
  
# Strongest negative linear relationship  
plot(dataset.analysis$infant.mortality,  
 dataset.analysis$life.expectancy,  
 main="infant mortality vs life expectancy: -0.93")



#### Sampling ####  
n <- 35  
set.seed(n)  
random\_id <- sample(1:174, n)  
dataset.sample <- dataset.analysis[random\_id,]  
write.csv(dataset.sample, file="../output/data/sample\_data.csv", row.names=FALSE)  
summary(dataset.sample)

## countries gdp.per.capita healthcare.expense life.expectancy  
## Length:35 Min. : 293 Min. : 2.271 Min. :58.06   
## Class :character 1st Qu.: 2212 1st Qu.: 5.248 1st Qu.:68.91   
## Mode :character Median : 4586 Median : 6.230 Median :72.64   
## Mean :11041 Mean : 6.407 Mean :72.33   
## 3rd Qu.: 9079 3rd Qu.: 7.388 3rd Qu.:75.95   
## Max. :75497 Max. :11.711 Max. :82.95   
## infant.mortality mortality   
## Min. : 2.00 Min. : 1.169   
## 1st Qu.: 7.00 1st Qu.: 6.085   
## Median :15.40 Median : 7.261   
## Mean :21.58 Mean : 7.516   
## 3rd Qu.:31.95 3rd Qu.: 8.995   
## Max. :64.30 Max. :14.200

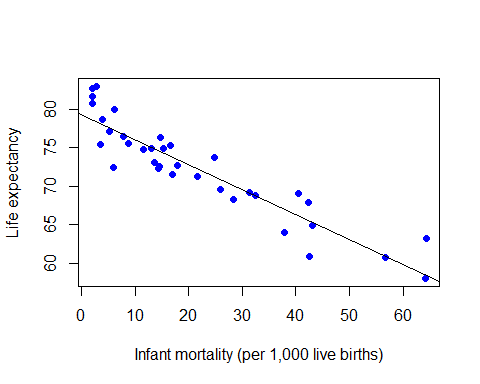
X <- dataset.sample$infant.mortality  
Y <- dataset.sample$life.expectancy  
plot(X, Y,  
 xlab='Infant mortality (per 1,000 live births)',  
 ylab='Life expectancy',   
 pch = 16, col = "blue")  
  
  
#### Fit linear line to the data ####  
# Without R  
X\_1 <- cbind(1,X)  
BM<-solve(t(X\_1)%\*%X\_1)%\*%t(X\_1)%\*%Y  
BM

## [,1]  
## 79.3272588  
## X -0.3243522

# With R  
lmTemp = lm(Y~X)  
summary(lmTemp)

##   
## Call:  
## lm(formula = Y ~ X)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.9816 -1.5386 -0.2949 1.8721 4.8236   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 79.32726 0.64797 122.42 < 2e-16 \*\*\*  
## X -0.32435 0.02328 -13.94 2.22e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.422 on 33 degrees of freedom  
## Multiple R-squared: 0.8548, Adjusted R-squared: 0.8504   
## F-statistic: 194.2 on 1 and 33 DF, p-value: 2.219e-15

abline(lmTemp) # Draw the fit line



#### Cross-validation OLS ####  
library(bootstrap)  
# The fit-function for crossval(), n is the degree of the model  
theta.fitn <- function (x,y,n){  
 a <- cbind(x)  
 if (n>1) {  
 for (i in 2:n) {  
 a <- cbind(a, I(x^i))  
 }  
 }  
 lsfit(a, y)  
}  
# The predict-function for crossval()  
theta.predictn<- function (fit,x){  
 a <- cbind(1, x)  
 n <- length(fit$coef) - 1  
 if (n>1) {  
 for (i in 2:n) {  
 a <- cbind(a, I(x^i))  
 }  
 }  
 a%\*%fit$coef  
}  
  
# Go through models up to the 3rd degree  
deg <- 3  
qs <- vector(length=deg)  
for (i in 1:deg) {  
 results<-crossval(dataset.sample$infant.mortality,  
 dataset.sample$life.expectancy,   
 theta.fitn, theta.predictn, n=i)  
 # Mean of the loss-function  
 Q <- sum((dataset.sample$life.expectancy-results$cv.fit)^2)/52  
 qs[i] <- Q  
}  
# Choose the degree with the smallest mean of the loss-function  
# Degree  
which.min(qs)

## [1] 2

# Smallest Q  
qs

## [1] 4.358701 4.141500 4.544503

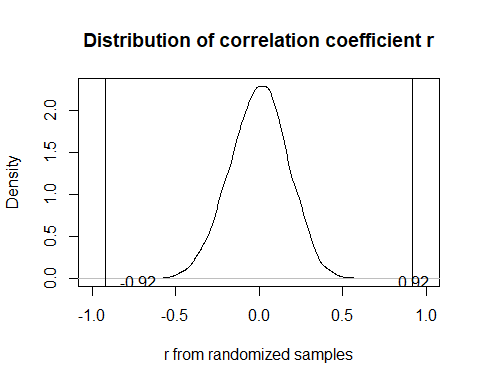
qs[which.min(qs)]

## [1] 4.1415

#### Hypothesis testing ####  
set.seed(n)  
r.obt <- cor(X, Y)  
cat("The obtained correlation is ", r.obt,'\n')

## The obtained correlation is -0.9245287

nreps <- 1e4  
r.random <- numeric(nreps)  
for (i in 1:nreps) {  
 X.permute <- sample(X, length(X), replace=FALSE)  
 r.random[i] <- cor(X.permute, Y)  
}  
plot(density(r.random),  
 main = "Distribution of correlation coefficient r",   
 xlab = "r from randomized samples",  
 xlim = c(-1,1))  
r.obt <- round(r.obt, digits = 2)  
abline(v=r.obt)  
abline(v=-r.obt)  
legend(-1, .2, r.obt, bty = "n")  
legend(0.66, .2, -r.obt, bty = "n")



r.obt

## [1] -0.92

prob <- sum(r.random[-r.obt <= r.random] + r.random[r.random >= r.obt])/nreps  
prob

## [1] 0

#### Estimate correlation coefficients using Bootstrap and Jackknife method ####  
set.seed(n)  
library(bootstrap)  
library(boot)  
  
data <- cbind(X,Y)  
# Jackknife  
theta1 <- function(x,m){  
 cor(m[x,1], m[x,2])  
}  
l1 <- jackknife(1:n, theta1, data)  
# Pseudo-values and the Jackknife-estimator  
bjack <- n\*cor(X,Y) - (n-1)\*l1$jack.values   
theta.est <- sum(bjack) / n  
theta.est

## [1] -0.9237229

# Confidence interval  
t.value <- qt(0.975, n-1)  
dev <- t.value\*sqrt(var(bjack) / n)  
lower <- theta.est - dev  
upper <- theta.est + dev  
CI <- c(lower, upper)  
CI

## [1] -0.9700611 -0.8773847

# Bootstrap  
theta2 <- function(m,x) {  
 cor(m[x,1], m[x,2])  
}  
l2 <- boot(data, theta2, R=1000)  
l2$t0

## [1] -0.9245287

boot.ci(l2, type = "perc")

## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS  
## Based on 1000 bootstrap replicates  
##   
## CALL :   
## boot.ci(boot.out = l2, type = "perc")  
##   
## Intervals :   
## Level Percentile   
## 95% (-0.9603, -0.8762 )   
## Calculations and Intervals on Original Scale