setwd("D:/Skola/I-STGST/Cviceni/cvIV")

**NAHRANI DAT**

# Import Dataset

**# kod**

pop <- read.delim("D:/Skola/I-STGST/Cviceni/cvIV/population.txt")

nezam <- read.delim("D:/Skola/I-STGST/Cviceni/cvIV/unemployment.txt", dec=",")

crime <- read.csv("D:/Skola/I-STGST/Cviceni/cvIV/Crimes.csv", sep=";")

**#nahrani dat primo z excelu**

library('xlsx')

sesit <- paste(getwd(),"Eurostat\_data.xlsx",sep='/')

# sesit <- "D:/Skola/I-STGST/Cviceni/cvIV/Eurostat\_data.xlsx"

pop <- read.xlsx(sesit,'population')

nezam <- read.xlsx(sesit,2)

crime <- read.xlsx(sesit,3)

**Uprava datove sady**

nezam <- nezam[,-c(3,5,7)]

nezam.total <- nezam[,1:2]

colnames(nezam.total) <- c('GEO', 'nezamestnanost')

pop.2012 <- pop[,c('GEO','X2012')]

colnames(pop.2012) <- c('GEO','populace')

crime.2012 <- crime[,c('GEO', 'X2012')]

colnames(crime.2012) <- c('GEO','kriminalita')

UK <- crime.2012[crime.2012$GEO == 'Scotland',2] + crime.2012[crime.2012$GEO == 'Northern Ireland (UK)',2] + crime.2012[crime.2012$GEO == 'England and Wales',2]

crime.2012[,1] <- as.character(crime.2012[,1]) # zmena datoveho typu z faktoru na text

crime.2012[49,] <- c('United Kingdom', UK)

crime.2012[,2] <- as.numeric(crime.2012[,2]) # zmena datoveho typu z textu na cislo

**# spojeni datovych sad na zaklade nazvu jednoho z atributu**

nez.crime <- merge(nezam.total, crime.2012, by = 'GEO')

data.2012 <- merge(nez.crime, pop.2012, by = 'GEO')

**# vytvoreni noveho sloupce s poctem kriminalnich cinu na**

data.2012[,5] <- 10000\*data.2012$kriminalita/data.2012$populace

colnames(data.2012)[5] <- 'kriminalita\_pop'

write.table(data.2012, 'data.2012.txt') #vytvori rextovy dokument ceho napisi

**stredni hodnoty (miry polohy)**

**# aritmeticky prumer**

mean(data.2012)

mean(data.2012[,2], na.rm = T)

# mean(na.omit(data.2012[,2]))

mean(data.2012[,2], trim = 0.1, na.rm = T)

**# vypocet prumeru pro kazdy sloupec**

sapply(data.2012[,2:5],mean, na.rm = T)

sapply(data.2012[,2:5],mean, na.rm = T, trim = 0.1)

# for (i in 2:5) print(mean(data.2012[,i], na.rm = T))

sapply(data.2012[,2:5],function(x) mean(x, na.rm = T, trim = 0.1))

**# pripojeni datove sady, pouzivaji se jen nazvy sloupcu**

attach(data.2012)

**# median**

median(nezamestnanost, na.rm = T)

sapply(data.2012[,2:5],median, na.rm = T)

**#kvantily**

quantile(data.2012[,2], na.rm = T)

quantile(data.2012[,2], na.rm = T, prob = c(0.3,0.6))

quantile(data.2012[,2], na.rm = T, prob = seq(0, 1, 0.1))

quantile(data.2012[,2], na.rm = T, prob = c(1:4/4))

**#geometricky prumer**

gmean <- function(x) exp(mean(log(x))) #definice funkce #nebude po nas ted chtit:)

gmean(na.omit(data.2012[,2]))

**#harmonicky prumer - je definován jako podíl rozsahu souboru (počtu členů) a součtu převrácených**

# hodnot znaků. Jinými slovy je to převrácená hodnota aritmetického průměru převrácených hodnot zadaných členů

hmean <- function(x) 1/mean(1/x)

hmean(na.omit(data.2012[,2]))

**# modus- je hodnota, která se v daném statistickém souboru vyskytuje nejčastěji**

library(lsr)

modeOf(data.2012[,2], na.rm = TRUE)

maxFreq(data.2012[,2], na.rm = TRUE)

**# funce round zaokrouhli hodnoty**

modeOf(round(data.2012[,2]), na.rm = T)

maxFreq(round(data.2012[,2]), na.rm = TRUE)

**charakteristiky (miry) variability**

**# minimum a maximum**

min(data.2012[,2], na.rm = T)

max(data.2012[,2], na.rm = T)

**# rozsah**

min <- min(data.2012[,2], na.rm = T)

max <- max(data.2012[,2], na.rm = T)

max - min

# max(nezamestnanost, na.rm = T) - min(nezamestnanost, na.rm = T)

range(data.2012[,2], na.rm = T)

**# mezikvartilove rozpeti**

q <- quantile(data.2012[,2], na.rm = T, prob = c(0.25,0.75))

q[2] - q[1]

IQR(data.2012[,2], na.rm = T)

**# rozptyl neboli variance**

var(data.2012[,2], na.rm = T)

**# smerodatna odchylka**

sqrt(var(data.2012[,2], na.rm = T))

sd(data.2012[,2], na.rm = T)

**# prumerna absolutni odchylka od prumeru**

aad(data.2012[,2], na.rm = T)

**# median absolutni odchylky od medianu**

mad(data.2012[,2], na.rm = T)

**miry tvaru**

**# sikmost**

library(e1071)

skewness(data.2012[,2], na.rm=T)

**#spicatost**

kurtosis(data.2012[,2], na.rm=T)

**#### automaticky vypis zakladni statistiky ####**

**summary(data.2012)**

**fivenum(data.2012[,2], na.rm=T)**

**# automaticky vzpis vcetne charakteristik variability**

**library(psych) #vypise prumer, median, min, max, atd...**

**describe(data.2012[,2], na.rm=T)**

**describeBy(data.2012, na.rm=T)**

**GRAFY**

**# XY bodovy graf**

attach(data.2012)

plot(nezamestnanost, kriminalita\_pop)

plot(nezamestnanost, kriminalita\_pop, xlab = 'Nezamestanost (%)', ylab = 'Trestne ciny na 10 000 obyvatel', main = 'Vztah mezi kriminalitou a nezamestnanosti \n v roce 2012')

plot(nezamestnanost, kriminalita\_pop, xlab = 'Nezamestanost (%)', ylab = 'Trestne ciny na 10 000 obyvatel', main = 'Vztah mezi kriminalitou a nezamestnanosti \n v roce 2012', pch = '\*')

plot(nezamestnanost, kriminalita\_pop, xlab = 'Nezamestanost (%)', ylab = 'Trestne ciny na 10 000 obyvatel', main = 'Vztah mezi kriminalitou a nezamestnanosti \n v roce 2012', type = 'h')

plot(nezamestnanost, kriminalita\_pop, xlab = 'Nezamestanost (%)', ylab = 'Trestne ciny na 10 000 obyvatel', main = 'Vztah mezi kriminalitou a nezamestnanosti \n v roce 2012', pch = '')

text(nezamestnanost, kriminalita\_pop, data.2012[,1], cex = 0.6, col = 'red')

lm <- lm(kriminalita\_pop ~ nezamestnanost, data = data.2012)

summary(lm)

abline(lm, col = 'red')

**# sloupcovy graf**

barplot(kriminalita\_pop)

barplot(kriminalita\_pop, main = 'Pocet trestnych cinu na 10 000 obyvatel', col = rainbow(20), names.arg = GEO)

x <- barplot(kriminalita\_pop, main = 'Pocet trestnych cinu na 10 000 obyvatel', col = rainbow(20))

text(cex=0.75, x=x-0.25, y=2.25, as.character(GEO), xpd=T, srt=90, adj = 1.04)

krim <- as.matrix(crime[1:5,2:5]); rownames(krim) <- as.character(crime[1:5,1])

barplot(krim, col = rainbow(5))

**# histogram**

hist(populace)

hist(log(populace), nclass = 12)

hist(kriminalita\_pop, breaks = 10)

hist(nezamestnanost, breaks = 10)

**#density**

d <- density(log(populace))

plot(d)

hist(log(populace), nclass = 12, freq = F, col = 'light blue')

lines(d, col = 'red', lty = 3)

d <- density(populace)

plot(d)

hist(populace, nclass = 8, freq = F, col = 'light blue')

lines(d, col = 'red', lty = 3)

**# boxplot**

boxplot(populace)

boxplot(data.2012[,2:5])

boxplot(scale(data.2012[,2:5]), notch = T)

**# violin plot**

library(vioplot)

# nezam <- read.delim("D:/Skola/I-STGST/Cviceni/cvV/unemployment.txt", dec=",")

vioplot(nezam[!is.na(nezam[,2]),2])

vioplot(na.omit(nezam[,2]))

par(mfrow = c(1,3))

vioplot(na.omit(nezam[,2]))

vioplot(na.omit(nezam[,3]))

vioplot(na.omit(nezam[,4]))

par(mfrow = c(1,1))

**# qqplot**

qqnorm(kriminalita\_pop)

qqline(kriminalita\_pop, col = 'red')

qqnorm(log(kriminalita\_pop))

qqline(log(kriminalita\_pop))

library(qualityTools)

qqPlot(kriminalita\_pop)

qqPlot(log(kriminalita\_pop))

**# dot plot**

dotchart(kriminalita\_pop, cex = 0.7, labels = GEO)

ord <- data.2012[order(data.2012$kriminalita\_pop),]

dotchart(ord[,5], cex = 0.7, labels = GEO)

abline(v = c(median(kriminalita\_pop, na.rm = T),mean(kriminalita\_pop, na.rm = T)), col = c("red", "blue"), lty = 3)

**# kruhovy diagram**

pie(na.omit(data.2012[,4]))

data.pop <- na.omit(data.2012[order(data.2012$populace, decreasing=F),])

pie(data.pop$populace, init.angle=90, radius = 1.6, labels = data.pop$GEO)

# <http://www.gisportal.cz/2013/05/jak-vybrat-spravny-graf-ci-diagram/>

**# scatterplot**

plot(data.eu[,-1])

**# korelace**

cor(nezamestnanost, kriminalita)

cor(kriminalita, populace)

cor(data.eu)

cor(data.eu[,-1])

**# vyznamnost korelace pomoci testu korelace**

cor.test(nezamestnanost, kriminalita)

cor.test(kriminalita, populace)

cor(kriminalita, populace, method = c("pearson"))

cor(kriminalita, populace, method = c("spearman"))

cor(kriminalita, populace, method = c("kendall"))

detach()

**REGRESE**

# podminky regrese

# Normální rozdělení závislé proměnné

# Vzájemná nezávislost závislé proměnné - bez autokorelace

# Smysluplna linearita vztahu

# Homoskedasticita – konstantní rozptyl, který je nezávislý na parametrech

**# normalni rozdeleni**

shapiro.test(kriminalita)

shapiro.test(log(kriminalita))

shapiro.test(populace)

shapiro.test(log(populace))

shapiro.test(kriminalita\_pop)

shapiro.test(log(kriminalita\_pop))

shapiro.test(nezamestnanost)

shapiro.test(log(nezamestnanost))

**# linearni regrese**

linm <- lm(kriminalita\_pop ~ nezamestnanost, data = data.eu)

linm

summary(linm)

par(mfrow = c(2,2))

plot(linm)

par(mfrow = c(1,1))

# normal Q-Q - normalita rezidui, mely by byt v liniich na uhlopricku, pokud je vse spravne

# residuals vs. fitted - linearita vztahu - nemel by byt videt zadny mozny vztah - nezavisle rozptylene body

# scale-location - homoskedacita (konstantni rozptyl) - data by mela byt nahodne rozptylena kolem horizontalni linie

# residuals vs. leverage - identifikace outlieru a vyznamnych bodu (mereni)

# - vysoka hodnota leverage - neobvykla kombinace prediktoru, velka cookova vzdalenost = velka vyznamnost

**# vykresleni**

plot(kriminalita\_pop ~ nezamestnanost, pch = '\*')

abline(linm, col = 'red')

**# vypocet predikce a konfidencnich intervalu z puvodnich hodnot**

linm.pred <- predict(linm, interval = 'confidence')

plot(kriminalita\_pop ~ nezamestnanost, pch = '\*')

points(data.eu$nezam, linm.pred[,'fit'], col = 'red', type='p')

points(data.eu$nezam, linm.pred[,'lwr'], col = 'blue', type='p')

points(data.eu$nezam, linm.pred[,'upr'], col = 'blue', type='p')

loglinm <- lm(log(kriminalita\_pop) ~ log(nezamestnanost), data = data.eu)

loglinm

summary(loglinm)

plot(loglinm)

plot(log(kriminalita\_pop) ~ log(nezamestnanost), pch = '\*')

abline(loglinm, col = 'red')

**# kvadraticka regrese**

kvm <- lm(kriminalita\_pop ~ nezamestnanost + I(nezamestnanost^2), data = data.eu)

kvm

summary(kvm)

plot(kvm)

plot(kriminalita\_pop ~ nezamestnanost, pch = '\*')

abline(kvm, col = 'red')

**# vypocet predikce a konfidencnich intervalu z puvodnich hodnot**

kvm.pred <- predict(kvm, interval = 'confidence')

plot(kriminalita\_pop ~ nezamestnanost, pch = '\*')

points(data.eu$nezam, kvm.pred[,'fit'], col = 'red', type='p')

points(data.eu$nezam, kvm.pred[,'lwr'], col = 'blue', type='p')

points(data.eu$nezam, kvm.pred[,'upr'], col = 'blue', type='p')

logkvm <- lm(log(kriminalita\_pop) ~ nezamestnanost + log(nezamestnanost2), data = data.eu)

logkvm

summary(logkvm)

plot(logkvm)

**# kubicka regrese**

kum <- lm(kriminalita\_pop ~ nezamestnanost + I(nezamestnanost^2) + I(nezamestnanost^3), data = data.eu)

kum

summary(kum)

plot(kum)

logkum <- lm(log(kriminalita\_pop) ~ nezamestnanost + I(nezamestnanost^2) + I(nezamestnanost^3), data = data.eu)

logkum

summary(logkum)

plot(logkum)

**# plotovani**

# vypocet predikce a konfidencnich intervalu z puvodnich hodnot

kvm.pred <- predict(kvm, interval = 'confidence')

plot(kriminalita\_pop ~ nezamestnanost, pch = '\*')

points(data.eu$nezam, kvm.pred[,'fit'], col = 'red', type='p')

points(data.eu$nezam, kvm.pred[,'lwr'], col = 'blue', type='p')

points(data.eu$nezam, kvm.pred[,'upr'], col = 'blue', type='p')

anova(linm, kum)

anova(loglinm, logkum)

**# vicenasobna regrese**

lm.vse <- lm(kriminalita\_pop ~ ., data = data.eu[,-1])

summary(lm.vse)

step(lm.vse)

**# model s interakci**

lm.vse.i <- lm(kriminalita\_pop ~ .^2, data = data.eu[,-1])

summary(lm.vse.i)

step(lm.vse.i)

#### test normálního rozdělení ####

shapiro.test(data[,2])

sapply(data[,-1], shapiro.test)

#### F-test shody rozptylů ####

var.test(data[,2], data[,3])

var.test(data[,4], data[,5])

#############################################

############ Parametricke metody ############

#############################################

#### t-test o shodě středních hodnot ####

#### jednostranné testy (<, menší nebo rovno, >, větší nebo rovno), oboustranné testy (=, nerovná se) ####

#### jednovýběrový t-test - srovnání výběru s předpokládaným průměrem ####

t.test(data$produktivita\_00, mu = 100)

t.test(data$produktivita\_00, mu = 100, alternative='less')

t.test(data$produktivita\_12, mu = 100)

t.test(data$produktivita\_12, mu = 100, , alternative='less')

#### dvouvýběrový t-test - srovnání středních hodnot dvou nezávislých výběrů ####

t.test(data$nezamestnanost\_00, data$nezamestnanost\_12)

t.test(data$produktivita\_00, data$produktivita\_12)

#### párový t-test - srovnání středních hodnot dvou závislých výběrů ####

t.test(data$nezamestnanost\_00, data$nezamestnanost\_12, paired = T)

t.test(data$produktivita\_00, data$produktivita\_12, paired = T)

#### ANOVA + Tukey HSD - analýza variance a vícenásobné porovnávání ####

#### zjišťování rozdílů mezi více než dvěma skupinami, srovnání vnitro- a mezi-skupinového rozptylu ####

#### v případě zamítnutí H0: µ1 = µ2 nerovná se µ3 nebo µ1 nerovná se µ2 = µ3 nebo µ1 nerovná se µ2 nerovná se µ3 ####

# rozdeleni do skupin

data$skupina <- 1

data[10:18,'skupina'] <- 2

data[19:27,'skupina'] <- 3

data$skupina <- as.factor(data$skupina)

anova <- aov(nezamestnanost\_00 ~ skupina, data = data)

summary(anova)

tukey <- TukeyHSD(anova)

plot(tukey)

a <- quantile(data$nezamestnanost\_00, probs=c(0.33, 0.66))

data$skupina <- ifelse(data$nezamestnanost\_00 < a[1], 1, ifelse(data$nezamestnanost\_00 < a[2], 2, 3))

data$skupina <- as.factor(data$skupina)

anova2 <- aov(produktivita\_00 ~ skupina, data = data)

summary(anova2)

tukey2 <- TukeyHSD(anova2); tukey2

plot(tukey2)

**Neparametricke metody**

EarlyLeavers <- read.delim("D:/Skola/I-STGST/Cviceni/cvIX/EarlyLeavers.txt", dec=",")

sapply(EarlyLeavers[,-1], shapiro.test)

#### Mann - Whitney test - srovnání shody rozdělení dvou nezávislých souborů ####

#### neparametrická obdoba dvouvýběrového t-testu ####

wilcox.test(EarlyLeavers$Men, EarlyLeavers$Women)

wilcox.test(EarlyLeavers$r2013, EarlyLeavers$Target)

#### Wilcoxonův test - pro párová pozorování, neparametrická obdoba párového t-testu ####

#### H0: medián rozdílů je 0 ####

wilcox.test(EarlyLeavers$Men, EarlyLeavers$Women, paired = T)

wilcox.test(EarlyLeavers$r2005, EarlyLeavers$r2013, alternative = 'less', paired = T)

wilcox.test(EarlyLeavers$r2013, EarlyLeavers$Target, paired = T)

**Kruskal - Wallis test - neparametrická obdoba ANOVY**

EarlyLeavers$skupina <- 0

b <- quantile(EarlyLeavers$r2005, probs=c(0.33, 0.66))

EarlyLeavers$skupina <- ifelse(EarlyLeavers$r2005 < b[1], 1, ifelse(EarlyLeavers$r2005 < b[2], 2, 3))

EarlyLeavers$skupina <- as.factor(EarlyLeavers$skupina)

kruskal <- kruskal.test(r2005 ~ skupina, data = EarlyLeavers)

kruskal

library(pgirmess)

kruskalmc(r2005 ~ skupina, data = EarlyLeavers)

kruskal2 <- kruskal.test(r2013 ~ skupina, data = EarlyLeavers)

kruskal2

kruskalmc(r2013 ~ skupina, data = EarlyLeavers)

**GEOSTATISTIKA**

# nacteni dat a jejich pocatecni pruzkum

data <- read.table("D:/Skola/I-STGST/Cviceni/cvXI/dataXI.txt", header=T, quote="\"")

shapiro.test(data[,3])

library(geoR)

geo.data <- as.geodata(data, coords.col = 1:2, data.col=3)

summary(geo.data)

# prostorova pruzkumna analyza

points(geo.data,cex.min=1,cex.max=4, col="gray")

# points(geo.data,cex.min=1,cex.max=4, col=heat.colors(5))

points(geo.data, trend="1st", abs=T, col="gray", cex.max=3) # rezidua po nafitovani linearniho trendu

points(geo.data, trend="2nd", abs=T, col="gray", cex.max=3) # rezidua po nafitovani kvadratickeho trendu

# kladna (cerne) a zaporna (bile) rezidua se vyskytuji u sebe - indikuje prostorove autokorelace

plot(geo.data,lowess=TRUE)

plot(geo.data,lowess=TRUE,trend="1st")

plot(geo.data,lowess=TRUE,trend="2nd")

# variogram

plot(variog(geo.data, option = 'cloud'))

variogram <- variog(geo.data)

plot(variogram)

varfit <- variofit(variogram)

lines(varfit)

varfit <- variofit(variogram, nugget = 50, max.dist = 500)

lines(varfit, col = 'red')

varfit <- variofit(variogram, cov.model = 'gau', nugget = 50, max.dist = 500)

lines(varfit, col = 'blue')

# eyefit <- eyefit(variogram)

# lines(eyefit, col = "green")

var4 <- variog4(geo.data)

plot(var4)

# tvorba rastru

# summary(geo.data)

# b rozestupy, l pocet intervalu

loci <- expand.grid(seq(min(geo.data$coords[,1]), max(geo.data$coords[,1]), b = 5),seq(min(geo.data$coords[,2]), max(geo.data$coords[,2]), b = 5))

plot(loci)

kriging <- krige.conv(geo.data, loc = loci, krige = krige.control(obj.model = varfit))

# kc <- krige.control(obj.model = varfit)

# kriging <- krige.conv(geo.data, loc = loci, krige = kc)

str(kriging)

image(kriging)

contour(kriging,add=T)

contour(kriging, filled = T)

par(mfrow = c(1,2))

image(kriging)

contour(kriging,add=T)

image(kriging, val = sqrt(kriging$krige.var))

contour(kriging, val = sqrt(kriging$krige.var), add=T)

par(mfrow = c(1,1))

# image(kriging, col=rev(heat.colors(10, alpha= 0.2)))

# contour(kriging,add=T)

##### kriging s linearnim trendem #####

plot(geo.data,lowess=TRUE)

plot(geo.data,lowess=TRUE,trend="1st")

variogram.t <- variog(geo.data, trend = "1st")

plot(variogram.t)

varfit.t <- variofit(variogram.t, nugget = 55, max.dist = 320)

lines(varfit.t, col = 'red')

kriging.t <- krige.conv(geo.data, loc = loci, krige = krige.control(obj.model = varfit.t))

image(kriging.t, col=rev(heat.colors(10, alpha= 0.5))); contour(kriging.t, add = T)

par(mfrow = c(1,2))

image(kriging, col=rev(heat.colors(10, alpha= 0.5))); contour(kriging, add = T)

image(kriging.t, col=rev(heat.colors(10, alpha= 0.5))); contour(kriging.t, add = T)

par(mfrow = c(1,1))

#### kriging s daty, ktera nemaji normalni rozdeleni a navic se v nich vyskytuje trend ####

geo.data.n <- as.geodata(data, coords.col = 1:2, data.col=4)

plot(geo.data.n)

shapiro.test(geo.data.n$data)

shapiro.test(log(geo.data.n$data))

geo.data.n$data <- log(geo.data.n$data)

plot(geo.data.n, lowess = T, trend = "1st")

variogram.vt <- variog(geo.data.n, trend = "1st")

plot(variogram.vt)

varfit.vt <- variofit(variogram.vt, nugget = 55, max.dist = 320)

lines(varfit.vt, col = 'red')

kriging.vt <- krige.conv(geo.data.n, loc = loci, krige = krige.control(obj.model = varfit.vt))

summary(data$volume2)

summary(kriging.vt$predict)

# puvodne logarimovane hodnoty je potreba odlogaritmovat nebo pozdeji alespon zobrazit spravne

# kriging.vt.exp <- kriging.vt

# kriging.vt.exp$predict <- exp(kriging.vt.exp$predict)

par(mfrow = c(1,2))

image(kriging.vt, col=rev(heat.colors(10, alpha= 0.5))); contour(kriging.vt, add = T)

# parametr ... val = exp(kriging.vt$predict) ... predstavuje prevedeni cisel zpet na puvodni rozsah (odlogaritmovani)

image(kriging.vt, val = exp(kriging.vt$predict), col=rev(heat.colors(10, alpha= 0.5))); contour(kriging.t, val = exp(kriging.vt$predict), add = T)

par(mfrow = c(1,1))