

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky



Aplikovaná štatistika

11. cvičenie - riešené príklady v R

Jana Petrillová, Mária Švecová

```
# PRÍKLAD 7.3
```

```
x=c(160,250,320,500,750,1000,1500,2000)
y=c(789,800,851,990,1193,1335,1704,2073)
alpha = 0.05
n=length(x)
```

```
# a)-----
# lineárny model (balík stats, zaleží na poradí y~x alebo x ~ y)
```

```
model = lm(y~x)
summary(model)
b=model$coefficients # bodové odhady parametrov regresného modelu
fy=model$fitted.values # odhadované hodnoty pomocou regresného modelu (alebo fy=predict.lm(model))
e=resid(model) # vypočítaná reziduá (alebo e=model$residuals)
```

```
# -----
# scatter.smooth(x,y) zakreslí body x,y a preloží to zhruba ako by mohla vyzerat závislosť medzi nimi
plot(x,y, xlab = "", ylab = "", type="p", col="red", pch=19, cex.axis=1.3)
title(xlab = list("Počet vyrobených stoličiek", cex = 1.4), ylab = list("Množstvo nakúpeného dreva", cex = 1.4))
abline(model,col="blue",lty=2,lwd=2) # alebo abline(reg = model) alebo abline(coef = coef(model))
```

```
# -----
# grafická analýza reziduí
plot(fy,e, xlab = "", ylab = "", grid(nx=NULL,ny=NULL), cex.axis=1.3)
title(xlab = list("Teoretické hodnoty", cex = 1.4), ylab = list("Reziduá", cex = 1.4))
abline(0,0,col="red", lwd=2)
```

```
# alebo (balík olsrr)
ols_plot_resid_fit(model)
```

```
# -----
# overenie normality
shapiro.test(e)
p=shapiro.test(e)$p.value
```

```
# vyhodnotenie pomocou p-hodnoty
if(p > alpha)
  cat("p > alpha, H0 nezamietame na hladine alpha") else
  cat("p <= alpha, H0 zamietame prijimame H1")
```

```
# alebo (balík olsrr)
ols_test_normality(model) # vypíše hodnoty testovacích kriterií a p-hodnot pre rôzne testy dobrej zhody
ols_plot_resid_qq(model) # qq plot pre reziduá na grafické overenie normality
ols_plot_resid_hist(model) # histogram pre reziduá na grafické overenie normality
```

```
# -----
# test, či nahodné chyby majú strednú hodnotu nulovú
t.test(e,mu=0)
p=t.test(e,mu=0)$p.value
alpha<-0.05
```

```
# vyhodnotenie pomocou p-hodnoty
if(p > alpha)
  cat("p > alpha, H0 nezamietame na hladine alpha") else
  cat("p <= alpha, H0 zamietame prijimame H1")
```

```
# -----
# test, či náhodné chyby majú konštantný rozptyl - Goldfeldov-Quandtov test
```

```
vrx=sort(x)
xh=c(vrx[1:4])
xd=c(vrx[5:8])
vry=sort(y)
yh=c(vry[1:4])
yd=c(vry[5:8])
modeld = lm(yd~xd)
summary(modeld)
modelh = lm(yh~xh)
summary(modelh)
ed=resid(modeld)
SSED=sum(ed^2)
eh=resid(modelh)
SSEH=sum(eh^2)
```

```

TS=SSED/SSEH # testovacie kritérium
KH1=qf(1-alpha,2,2) # kritická hodnota

cat("Kritická oblasť (",KH1," , Inf)")

# vyhodnotenie pomocou kritickej oblasti
if(TS > KH1) cat("TS patrí do kritickej oblasti, H0 zamietame prijímame H1") else
  cat("TS nepatrí do kritickej oblasti, H0 nezamietame")

# alebo (balík lmtest)

gqtest(model)
p=gqtest(model)$p.value

# vyhodnotenie pomocou p-hodnoty
if(p > alpha)
  cat("p > alpha, H0 nezamietame na hladine alpha") else
  cat("p <= alpha, H0 zamietame prijímame H1")

# -----
# Durbin-Watsonova štatistika

sum1=0
for (i in 2:n) {sum1=sum1+((e[i]-e[i-1])^2)}
sum2=sum(e^2)
dw=sum1/sum2

# vyhodnotenie
if(dw < 1.4)
  cat("dw < 1.4, rezíduá sú kladne korelované a model je nevyhovujúci") else
  if(dw > 2.6)
    cat("dw > 2.6, rezíduá sú záporne korelované a model je nevyhovujúci") else
    cat("1.4 < dw < 2.6, rezíduá nevykazujú autokoreláciu a model je dobrý")

# alebo (balík lmtest)

dw<-dwtest(model)$statistic

# -----
# test pomocou bodov zvratu

z=3
TS=(z-((2*n-4)/3))/sqrt((16*n-29)/90)
KH1 <- qnorm(1-alpha/2,0,1) # kritická hodnota

cat("Kritická oblasť (-Inf, ",-KH1," ) zjednotenie (",KH1," , Inf)")

# vyhodnotenie pomocou kritickej oblasti
if(TS > KH1) cat("TS patrí do kritickej oblasti, H0 zamietame prijímame H1") else
  if (TS < -KH1) cat("TS patrí do kritickej oblasti, H0 zamietame prijímame H1") else
    cat("TS nepatrí do kritickej oblasti, H0 nezamietame")

```