2. zápočtová úloha z 01RAD

Jazmína Kreanová

Popis úlohy

Datový soubor `Boston` je obsažen v balíku `MASS` a lze použít rovnou po načtení příslušné knihovny.

```
knitr::opts_chunk$set(echo = FALSE)
list_of_packages <- c("tidyverse", "MASS","GGally","moderndive", "dplyr","car","lmtest","nortest")
missing_packages <- list_of_packages[!(list_of_packages %in% installed.packages()[,"Package"])]
missing_packages
if(length(missing_packages)) install.packages(missing_packages)
lapply(list_of_packages, library, character.only = TRUE)
#library(tidyverse)
#library(GGally)
#library(knitr)
#library(MASS)
```

1. 'corrplot' · 'splines' · 'nortest' · 'lmtest' · 'zoo' · 'car' · 'carData' · 'moderndive' · 'GGally' · 'MASS' · 'forcats' · 'stringr' · 'dplyr' · 'purrr' · 'readr' · 'tidyr' · 'tibble' · 'ggplot2' · 'tidyverse' · 'stats' · 'graphics' · 'grDevices' · 'utils' · 'datasets' · 'methods' · 'base'
2. 'corrplot' · 'splines' · 'nortest' · 'lmtest' · 'zoo' · 'car' · 'carData' · 'moderndive' · 'GGally' · 'MASS' · 'forcats' · 'stringr' · 'dplyr' · 'purrr' · 'readr' · 'tidyr' · 'tibble' · 'ggplot2' · 'tidyverse' · 'stats' · 'graphics' · 'grDevices' · 'utils' · 'datasets' · 'methods' · 'base'
3. 'corrplot' · 'splines' · 'nortest' · 'lmtest' · 'zoo' · 'car' · 'carData' · 'moderndive' · 'GGally' · 'MASS' · 'forcats' · 'stringr' · 'dplyr' · 'purrr' · 'readr' · 'tidyr' · 'tibble' · 'ggplot2' · 'tidyverse' · 'stats' · 'graphics' · 'grDevices' · 'utils' · 'datasets' · 'methods' · 'base'
4. 'corrplot' · 'splines' · 'nortest' · 'lmtest' · 'zoo' · 'car' · 'carData' · 'moderndive' · 'GGally' · 'MASS' · 'forcats' · 'stringr' · 'dplyr' · 'purrr' · 'readr' · 'tidyr' · 'tibble' · 'ggplot2' · 'tidyverse' · 'stats' · 'graphics' · 'grDevices' · 'utils' · 'datasets' · 'methods' · 'base'
5. 'corrplot' · 'splines' · 'nortest' · 'lmtest' · 'zoo' · 'car' · 'carData' · 'moderndive' · 'GGally' · 'MASS' · 'forcats' · 'stringr' · 'dplyr' · 'purrr' · 'readr' · 'tidyr' · 'tibble' · 'ggplot2' · 'tidyverse' · 'stats' · 'graphics' · 'grDevices' · 'utils' · 'datasets' · 'methods' · 'base'
6. 'corrplot' · 'splines' · 'nortest' · 'lmtest' · 'zoo' · 'car' · 'carData' · 'moderndive' · 'GGally' · 'MASS' · 'forcats' · 'stringr' · 'dplyr' · 'purrr' · 'readr' · 'tidyr' · 'tibble' · 'ggplot2' · 'tidyverse' · 'stats' · 'graphics' · 'grDevices' · 'utils' · 'datasets' · 'methods' · 'base'

```
#install.packages("car")
#library(car)
fig <- function(width, heigth){
 options(repr.plot.width = width, repr.plot.height = heigth)
 }
 options(jupyter.plot_scale=2)
```

```
head(Boston)
```

A data.frame: 6 × 14

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	black	lstat	medv
	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	0.00632	18	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
2	0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
3	0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
4	0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
5	0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33	36.2
6	0.02985	0	2.18	0	0.458	6.430	58.7	6.0622	3	222	18.7	394.12	5.21	28.7

Obsahuje celkem 506 záznamů z obcí v předměstí města Boston, MA, USA a data pocházejí ze studie v roce 1978.Viz Harrison, D. and Rubinfeld, D.L. (1978) Hedonic prices and the demand for clean air. J. Environ. Economics and Management 5, 81–102.

Základní charakteristiky ohledně jednotlivých proměnných získáte pomocí funkcí `str(Boston)}` a summary(Boston)}.

Data celkem obsahují 14 proměnných, přičemž naším cílem je prozkoumat vliv 13 z nich na cenu nemovitostí `medv`. Přičemž anglický popis jednotlivých proměnných (sloupců) je následující:

Feature & Description

`crim` per capita crime rate by town

`zn` : proportion of residential land zoned for lots over 25,000 sq.ft `indus` : proportion of non-retail business acres per town

`chas` : Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)

`nox` : nitrogen oxides concentration (parts per 10 million)

`rm` : average number of rooms per dwelling

`age` : proportion of owner-occupied units built prior to 1940

`dis` : weighted mean of distances to five Boston employment centres

`rad` : index of accessibility to radial highways

`tax` : full-value property-tax rate per $10,000

`ptratio` : pupil-teacher ratio by town

`black` : $1000(B_k - 0.63)^2$ where B_k is the proportion of blacks by town

`lstat` : lower status of the population (percent)

`medv` : median value of owner-occupied homes in $1000s

Podmínky a body

Úkol i protokol vypracujte samostatně. Pokud na řešení nějaké úlohy budete přesto s někým spolupracovat, radit se, nezapomeňte to u odpovědi na danou otázku uvést. Tato zápočtová úloha obsahuje 10 otázek po 1 bodu. Celkem za 3 zápočtové úlohy bude možné získat 30 bodů, přičemž pro získání zápočtu je potřeba více jak 20. Další dodatečné body mohu případně individuálně udělit za extra práci na mini domácích úkolech nebo za aktivitu v hodině.

Odevzdání

Protokol ve formátu pdf (včetně příslušného Rmd souboru), nebo jak jupyter NB (ideálně s odkazem na Colab) odevzdejte prostřednictvím MS Teams, nejpozději do půlnoci 14. 12. 2022 (tj. za 3 týdny).

Příprava dat:

- Otázka 01

  Z dat vyfilitrujte jen pozorování, kde proměnná `chas` je rovna 0, proměnná rad je menší než 20 a odezva `medv` neobsahuje opakující se maximální hodnoty vzniklé nejspíše zaokrouhlením. Zkontrolujte, že výsledný datset neobsahuje chybějící hodnoty a vykreslete scatterplot pro proměnné `indus` a `medv`.

```
summary(Boston)
```

```
      crim                zn             indus            chas        
 Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000  
 1st Qu.: 0.08205   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000  
 Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000  
 Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917  
 3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000  
 Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000  
      nox               rm             age              dis        
 Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130  
 1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100  
 Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207  
 Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795  
 3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188  
 Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127  
      rad              tax           ptratio          black       
 Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32  
 1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38  
 Median : 5.000   Median :330.0   Median :19.05   Median :391.44  
 Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67  
 3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23  
 Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90  
     lstat            medv      
 Min.   : 1.73   Min.   : 5.00  
 1st Qu.: 6.95   1st Qu.:17.02  
 Median :11.36   Median :21.20  
 Mean   :12.65   Mean   :22.53  
 3rd Qu.:16.95   3rd Qu.:25.00  
 Max.   :37.97   Max.   :50.00  
```

```
#table(factor(Boston$medv))
```

Našou úlohou je vyfiltrovať dataset tak, aby zostali pozorovania s $ `chas=0`, `rad < 20` $ a aby dataset neobsahoval opakujúce sa maximálne hodnoty v odozve `medv`.

Vypíšeme si najprv pozorovania s opakujúcou sa hodnotou maxima v kýženej odozve `medv`. To činí 8 rôznych pozorovaní, pričom nepovažujeme za vhodné vybrať jedno pozorovanie, ktoré by sme do datasetu začlenili.

V ďalšom kroku teda vyfiltrujeme dáta tak, že nebydú obsahovať žiadne pozorovanie, ktoré v premennej `medv` nadobúda maximum. a overíme, že v datasete nemáme chýbajúce hodnoty. Nový dataset obsahuje 339 pozorovaní.

```
subset(Boston, Boston$chas==0 & Boston$rad<20 & Boston$medv==max(Boston$medv))
#dim(Boston_mx)
```

A data.frame: 8 × 14

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	black	lstat	me
	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<db
162	1.46336	0	19.58	0	0.6050	7.489	90.8	1.9709	5	403	14.7	374.43	1.73	
167	2.01019	0	19.58	0	0.6050	7.929	96.2	2.0459	5	403	14.7	369.30	3.70	
187	0.05602	0	2.46	0	0.4880	7.831	53.6	3.1992	3	193	17.8	392.63	4.45	
196	0.01381	80	0.46	0	0.4220	7.875	32.0	5.6484	4	255	14.4	394.23	2.97	
205	0.02009	95	2.68	0	0.4161	8.034	31.9	5.1180	4	224	14.7	390.55	2.88	
226	0.52693	0	6.20	0	0.5040	8.725	83.0	2.8944	8	307	17.4	382.00	4.63	
258	0.61154	20	3.97	0	0.6470	8.704	86.9	1.8010	5	264	13.0	389.70	5.12	
268	0.57834	20	3.97	0	0.5750	8.297	67.0	2.4216	5	264	13.0	384.54	7.44	

```
Boston_data<-subset(Boston, Boston$chas==0 & Boston$rad<20 & Boston$medv != max(Boston$medv))
```

Keďže premenná chas obsahuje jedinú hodnotu rovnakú pre všetky pozorovania, pre potreby tohto projektu ju odstránime z nového datasetu.

```
Boston_data<- Boston_data[,!names(Boston_data) %in% c("chas")]
```

```
head(Boston_data)
dim(Boston_data)
```

A data.frame: 6 × 13

	crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	black	lstat	medv
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	0.00632	18	2.31	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
2	0.02731	0	7.07	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
3	0.02729	0	7.07	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
4	0.03237	0	2.18	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
5	0.06905	0	2.18	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33	36.2
6	0.02985	0	2.18	0.458	6.430	58.7	6.0622	3	222	18.7	394.12	5.21	28.7

339 · 13

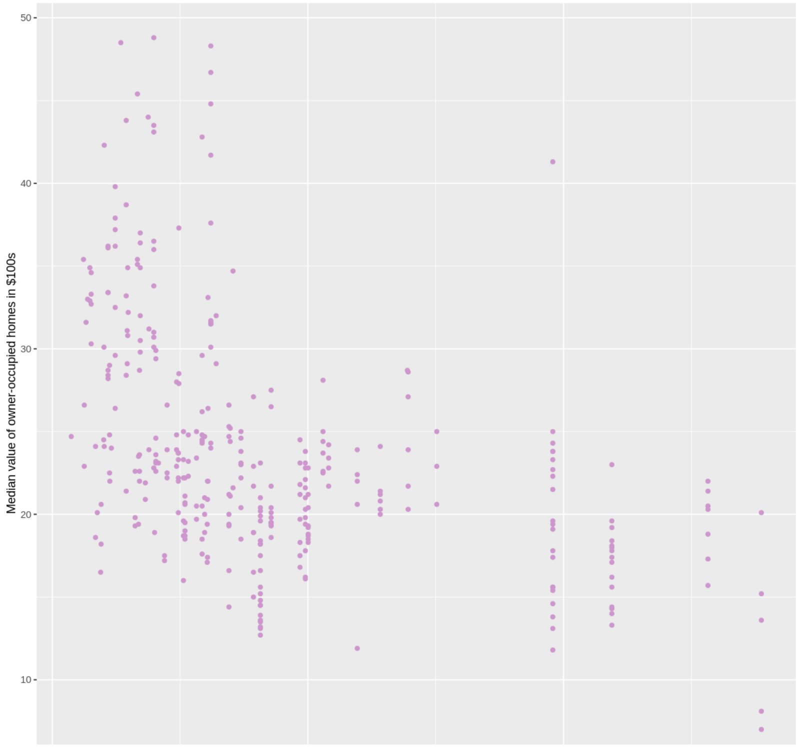
```
Boston_data %>%
  filter(if_any(everything(), is.na))
```

A data.frame: 0 × 13

crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	black	lstat	medv
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>

Vykreslíme scatterplot pre odozvu `medv` a vysvetľujúcu premennú `indus`. Podľa vykresleného scatterplotu nepozorujeme viditeľnú lineárnu závislosť mediánovej ceny nehnuteľností na podieli nemaloobchodného podnikania. Predpokladáme, že cena teda závisí aj na iných premenných, respektíve premenná `indus` môže byť závislá.

```
ggplot(Boston_data,aes(x=indus,y=medv))+
  geom_point(col="plum3")+
  labs(x= "Proportion of non-retail business acres per town", y="Median value of owner-occupied homes in $100s")
```



Regresní model závislosti mediánu ceny nemovitosti na zastoupení ne-maloobchodního podnikání v daném místě:

- Otázka 2

  Sestavte jednoduchý regresní model a na jeho základech zjistěte zdali proměnná `indus` ovlivňuje median ceny nemovitostí určených k bydlení `medv`. Pokud ano, o kolik je průměr mediánů cen nemovitostí nižší/vyšší při vzrůstu zastoupení nemaloobchodního podnikání o 5 jednotek?

Zostavíme jednoduchý regresný model závislosti ceny nehnuteľností od podielu podnikania v danej oblasti. Volíme model s interceptom, pretože predpokladáme, že nehnuteľnosť bude mať nenulovú cenu aj v oblasti bez dostatočnej občianskej vybavenosti, navyše z predošlej úvahy v otázke jedna predpokladáme aj iné závislosti.

```
model = lm(medv ~ indus, data = Boston_data)
summary(model)
```

```
Call:
lm(formula = medv ~ indus, data = Boston_data)

Residuals:
    Min      1Q  Median      3Q     Max
-11.371  -4.313  -1.028   3.385  23.361

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 28.43376    0.58668  48.466   <2e-16 ***
indus       -0.53598    0.05528  -9.696   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.447 on 337 degrees of freedom
Multiple R-squared:  0.2181,    Adjusted R-squared:  0.2158
F-statistic: 94.01 on 1 and 337 DF,  p-value: < 2.2e-16
```

```
model$coefficients[2]*5
```

indus: -2.6799182744655

Odhadli sme model: $Y = 28.434 - 0.536 \cdot X,$

kde nám intercept aj premenná `indus` vyšli ako štatisticky významné, *p-hodnota* oboch parametrov vyšla veľmi malá ($2 \cdot 10^{-16}$). Hodnota charakteristiky $R^2 = 0.2181$ je pomerne nízka (čo sme v predošlých úvahách očakávali).

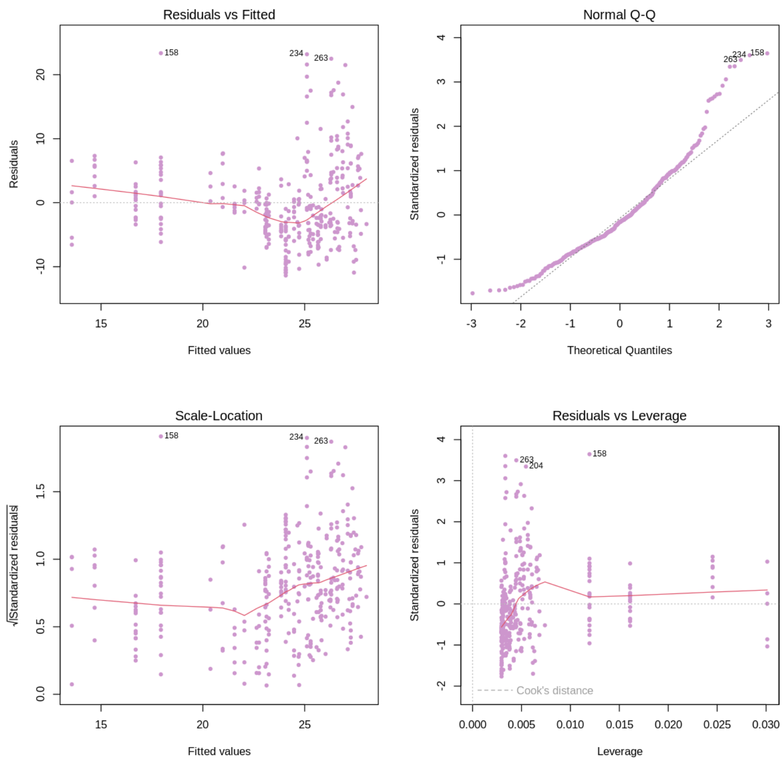
Na základe nášho modelu dôjde pri náraste nemaloobchodného podnikania o 5 jednotiek k poklesu (koeficient má zápornú hodnotu) cien v priemere o 2679.92 $.

- Otázka 3

  Vyzkoušejte model s mocninou a logaritmickou transformací odezvy. Pro výběr mocniné transformace vykreslete optimální log-věrohodnostní profil u Box-Coxovy transformace a porovnejte navrženou transformaci s provedenou logaritmickou.

Vykreslíme ploty reziduí našeho modelu. Z grafov Resuduals vs. Fitted a Scale-Location pozorujeme, že reziduá sa v rozptyle rozbiehajú, teda dochádza k porušeniu podmienku homoskedasticity, naviac Q-Q plot indikuje ťažké chvosty, čo nás navádza na transformáciu veličín.

```
#cv8,9
par(mfrow = c(2, 2))
plot(model, col="plum3",pch=20)
```



Mocninná transformácia Box-Coxova transformácia:

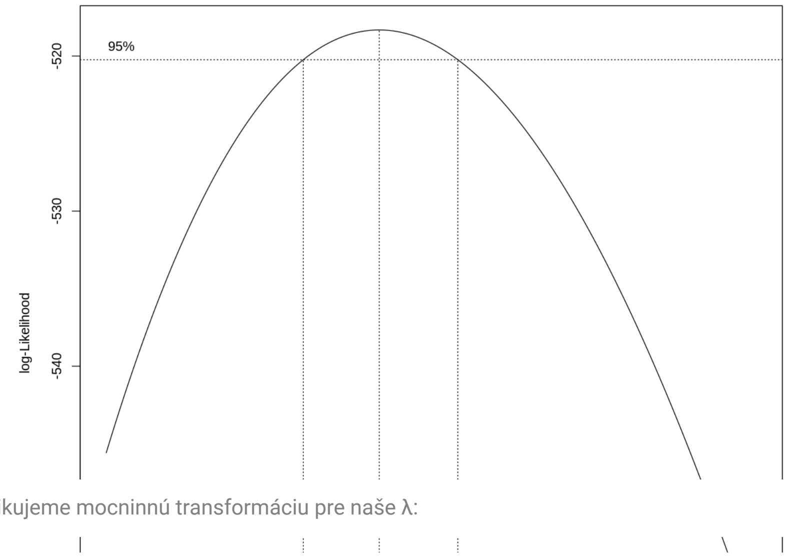
$$g(Y+a) = ln(Y+a) \text{ if } \lambda = 0,$$
$$g(Y+a) = \frac{(Y+a)^\lambda - 1}{\lambda} \text{ if } \lambda \neq 0,$$

Chceme transformovať dáta pomocou Box.Coxovej transformácie. Ako najvhodnejší exponent nám vyšlo $\lambda = -0.16$, čo spadá do intervalu spoľahlivosti 0 a teda vhodné by bolo použiť logaritmickú transformáciu.

Našou úlohou v tejto časti je porovnať mocninnú a logaritmickú transformáciu, preto nebudeme uvažovať $\lambda = 0$.

```
bc     <- boxcox(model, lambda = seq(-1,1 , 1/100))
lambda <- bc$x[which.max(bc$y)]
print("lambda:")
lambda
print("CI:")
c(min(bc$x[bc$y > max(bc$y) - 1/2 * qchisq(.95,1)]),max(bc$x[bc$y > max(bc$y) - 1/2 * qchisq(.95,1)]))
```

```
[1] "lambda:"
-0.16
[1] "CI:"
-0.39 · 0.0800000000000001
```

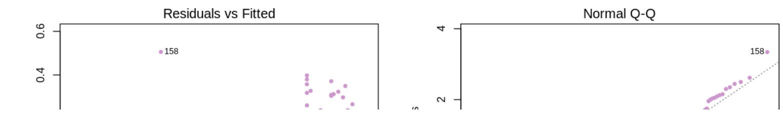


Aplikujeme mocninnú transformáciu pre naše λ:

```
# Power transform
Boston_bcT = Boston_data
Boston_bcT$medv_pwr = (Boston_bcT$medv^lambda-1)/lambda
model_pwr = lm(medv_pwr~indus, data = Boston_bcT)
summary(model_pwr)
```

```
Call:
lm(formula = medv_pwr ~ indus, data = Boston_bcT)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.50836 -0.09525 -0.00418  0.09975  0.50572 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.580596   0.013861  186.18   <2e-16 ***
indus       -0.014424   0.001306  -11.04   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1523 on 337 degrees of freedom
Multiple R-squared:  0.2657,	Adjusted R-squared:  0.2636 
F-statistic:   122 on 1 and 337 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2, 2))
plot(model_pwr, col="plum3",pch=20)
```



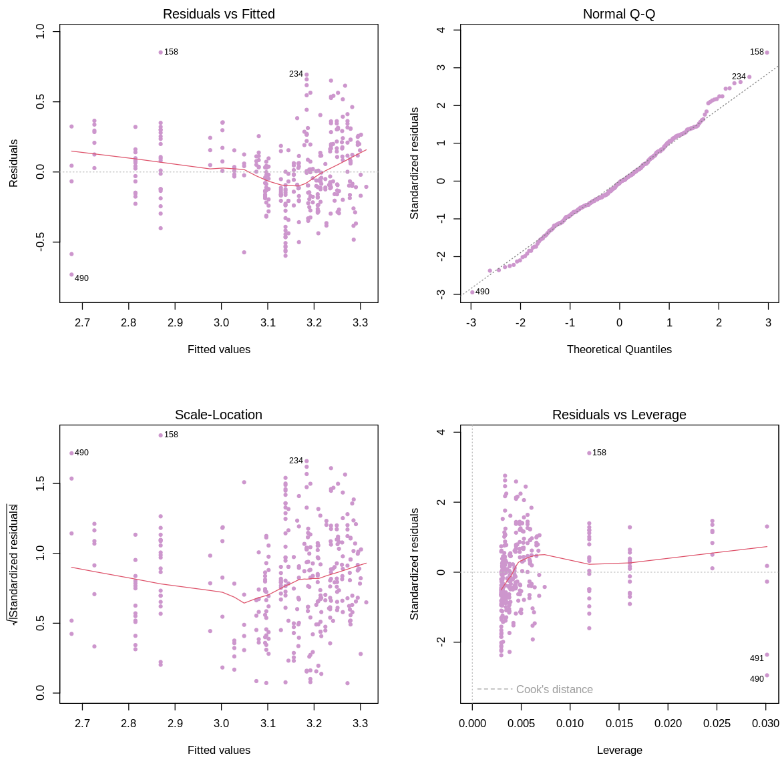
Logaritmická transformácia

$$\ln(y) = \mathbf{X} \cdot \beta + \epsilon$$
$$Y = e^{\mathbf{X} \cdot \beta + \epsilon} = e^{\mathbf{X} \cdot \beta} e^{\epsilon}$$

```
model_ln = lm(log(medv) ~ indus, Boston_data)
summary(model_ln)
```

```
Call:
lm(formula = log(medv) ~ indus, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.73082 -0.15957 -0.01244  0.16211  0.85196 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.33005    0.02293   145.2   <2e-16 ***
indus       -0.02355    0.00216   -10.9   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2519 on 337 degrees of freedom
Multiple R-squared:  0.2607,	Adjusted R-squared:  0.2585 
F-statistic: 118.9 on 1 and 337 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2, 2))
plot(model_ln, col="plum3",pch=20)
```



Porovnanie

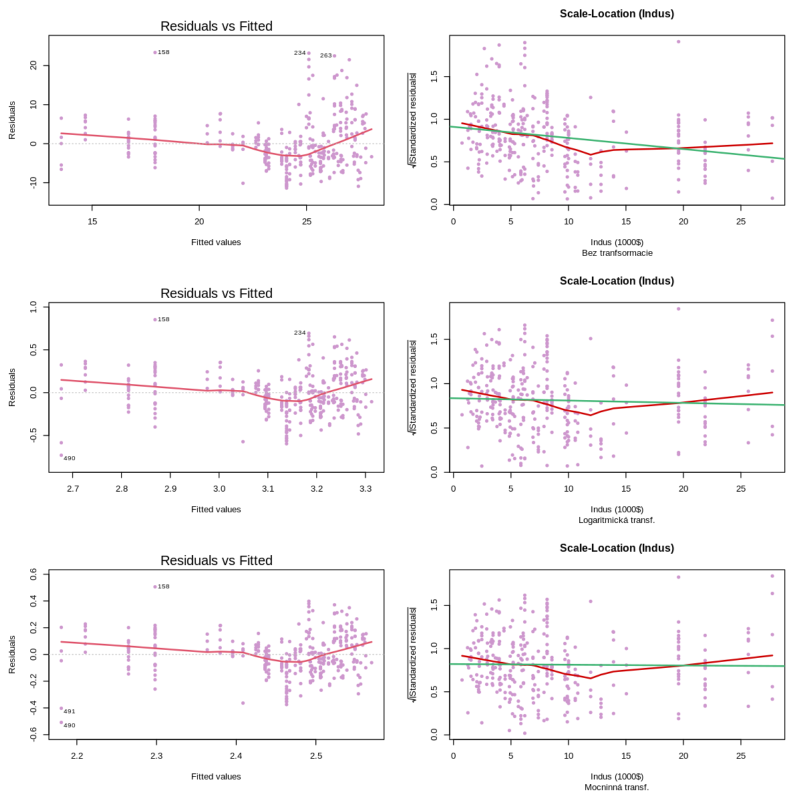
```
#summary(model)
#summary(model_ln)
#summary(model_pwr)
```

Chceme porovnať naše modely.

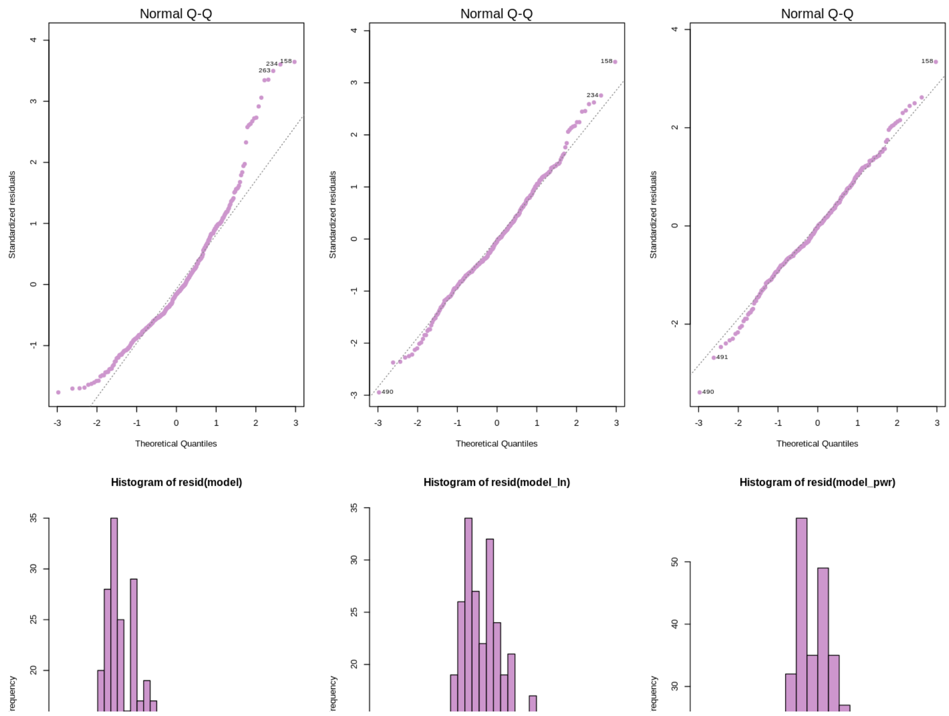
Pôvodný lineárny model mal hodnotu charakteristiky $R^2 = 0.2181$ a smerodajnú odchylku reziduí $sd_e = 6.447$. S použitím mocninnej transformácie sa hodnota zvýšila na $R^2 = 0.2657$, smerodajná odchylka poklesla na $sd_e = 0.1523$ a vrámci logaritmickej transformácie sme obdržali hodnotu $R^2 = 0.2607$, čo je nižsie ako pre mocninnú, ale vyššie než pre netransformovaný model a a $sd_e = 0.2519$ opäť trochu horšie než pre mocninný model a markantne lepšie než pre netransformovaný model.

Ďalej vyobrazíme Q-Q ploty pre jednotlivé modely. Môžeme pozorovať, že oba transformované modely grafy indikujú normalitu dát.

```
#fig(12,12)
op <- par(mfrow=c(3,2))
plot(model, sub = "Bez tranfsormacie", which = 1, pch = 20, col = "plum3", lwd = 2)
plot(Boston_data$indus, sqrt(abs(rstandard(model))), sub = "Bez tranfsormacie", pch = 20, col = "plum3", bg = "royalblue4",
     xlab = "Indus (1000$)", ylab = as.expression(substitute(sqrt(abs(yL)), list(yL = as.name("Standardized residuals")))),
     main = "Scale-Location (Indus)")
lines(lowess(Boston_data$indus, sqrt(abs(rstandard(model)))), col = "red3", lwd = 2)
abline(lm(sqrt(abs(rstandard(model)))~Boston_data$indus), col = "mediumseagreen", lwd = 2)
plot(model_ln, sub = "Logaritmická transf.", which = 1, pch = 20, col = "plum3", lwd = 2)
plot(Boston_data$indus, sqrt(abs(rstandard(model_ln))), pch = 20, col = "plum3", bg = "royalblue4",
     xlab = "Indus (1000$)", ylab = as.expression(substitute(sqrt(abs(yL)), list(yL = as.name("Standardized residuals")))),
     sub = "Logaritmická transf.",main = "Scale-Location (Indus)")
lines(lowess(Boston_data$indus, sqrt(abs(rstandard(model_ln)))), col = "red3", lwd = 2)
abline(lm(sqrt(abs(rstandard(model_ln)))~Boston_data$indus), col = "mediumseagreen", lwd = 2)

plot(model_pwr, sub = "Mocninná transf.", which = 1, pch = 20, col = "plum3", lwd = 2)
plot(Boston_data$indus, sqrt(abs(rstandard(model_pwr))), pch = 20, col = "plum3", bg = "royalblue4",
     xlab = "Indus (1000$)", ylab = as.expression(substitute(sqrt(abs(yL)), list(yL = as.name("Standardized residuals")))),
     sub = "Mocninná transf.",main = "Scale-Location (Indus)")
lines(lowess(Boston_data$indus, sqrt(abs(rstandard(model_pwr)))), col = "red3", lwd = 2)
abline(lm(sqrt(abs(rstandard(model_pwr)))~Boston_data$indus), col = "mediumseagreen", lwd = 2)
```

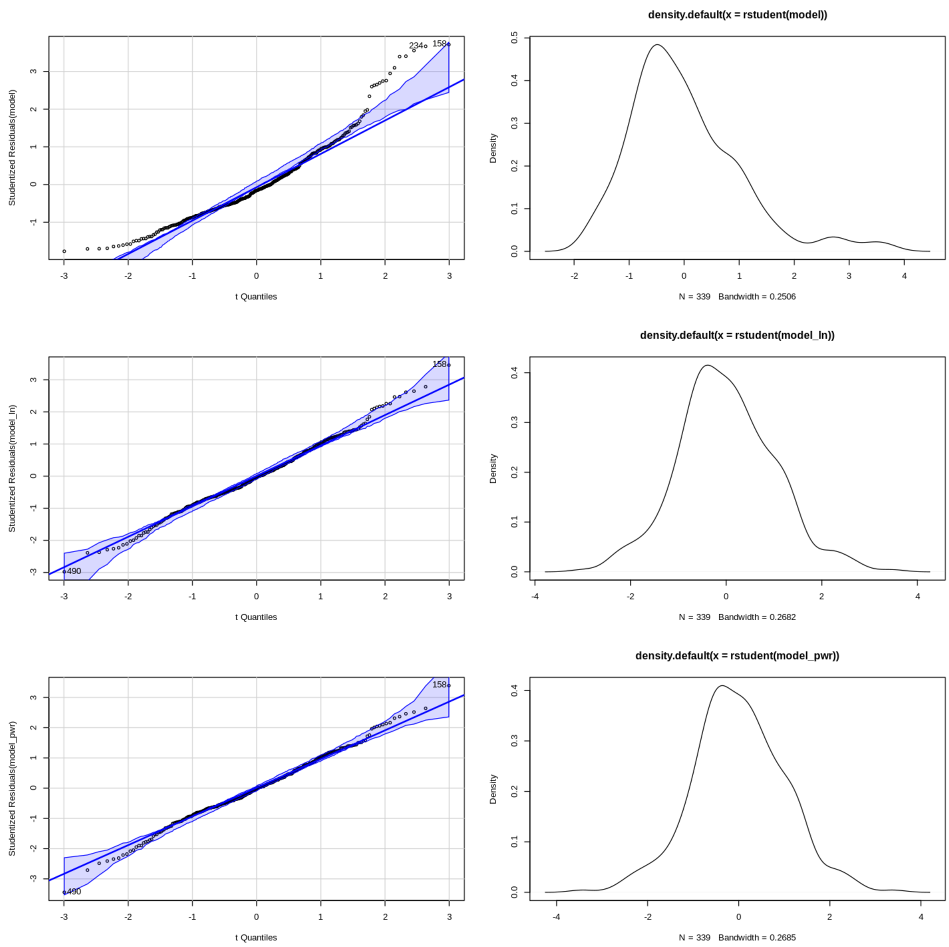



```
fig(12,12)
op <- par(mfrow=c(2,3))
plot(model,which=2, pch = 20, col = "plum3", lwd = 2,sub = "Bez transf.")
plot(model_ln,which=2, pch = 20, col = "plum3", lwd = 2,sub = "Logaritmická transf.")
plot(model_pwr,which=2,  pch = 20, col = "plum3", lwd = 2,sub = "Mocninná transf.")
hist(resid(model),breaks=30, col = "plum3",sub = "Bez transf.")
hist(resid(model_ln),breaks=30, col = "plum3",sub = "Logaritmická transf.")
hist(resid(model_pwr),breaks=30, col = "plum3",sub = "Mocninná transf.")
```




```
# compare models with and without log transform
op <- par(mfrow=c(3,2))
qqPlot(model, simulate=TRUE, line="quartiles")
plot(density(rstudent(model)))
qqPlot(model_ln, simulate=TRUE, line="quartiles")
plot(density(rstudent(model_ln)))
qqPlot(model_pwr, simulate=TRUE, line="quartiles")
plot(density(rstudent(model_pwr)))
par(op)
```

```
158:    154 234:    210
158:    154 490:    323
158:    154 490:    323
```



Aplikujeme príslušné testy, aby sme prípadne vyvrátili naše domnienky z grafov (podotknime, že v praxi často stačí záver na základe Q-Qplotov, či absencia kónusu reziduálneho scatterplotu).

Pre test normality využijeme Lillieforsov, Shapiro-Wilkov a Anderson-Darlingov test s levelom signifikantnosti α=5%. Pre netransformovaný model dostávame veľmi nízke p-hodnoty (pod 10^{-7}) pre všetky testy a teda zamietame hypotézu o normalite dát. Pre logaritmickú transformáciu máme hodnoty nad 0.077, čiže nemáme dostatok dókazov na zamietnutie hypotézy o normalite dát a pre mocninnú transformáciu sú p-hodnoty o niečo vyššie a opäť nemožno hypotézu normality zamietnuť.

```
#Normality
print("***********************Orig************************")
lillie.test(resid(model))   # Lilliefors test
shapiro.test(resid(model))      # Shapiro-Wilk test
ad.test(resid(model))
print("***********************Log************************")
lillie.test(resid(model_ln))   # Lilliefors test
shapiro.test(resid(model_ln))      # Shapiro-Wilk test
ad.test(resid(model_ln))
print("***********************Power************************")
lillie.test(resid(model_pwr))   # Lilliefors test
shapiro.test(resid(model_pwr))      # Shapiro-Wilk test
ad.test(resid(model_pwr))
```

```
[1] "***********************Orig************************"

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  resid(model)
D = 0.088726, p-value = 9.4e-07

        Shapiro-Wilk normality test

data:  resid(model)
W = 0.93378, p-value = 3.868e-11

        Anderson-Darling normality test

data:  resid(model)
A = 5.0044, p-value = 2.136e-12
[1] "***********************Log************************"

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  resid(model_ln)
D = 0.037909, p-value = 0.2774

        Shapiro-Wilk normality test

data:  resid(model_ln)
W = 0.99375, p-value = 0.1755

        Anderson-Darling normality test

data:  resid(model_ln)
A = 0.67475, p-value = 0.07742
[1] "***********************Power************************"

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  resid(model_pwr)
D = 0.035125, p-value = 0.3921

        Shapiro-Wilk normality test

data:  resid(model_pwr)
W = 0.99577, p-value = 0.4948

        Anderson-Darling normality test

data:  resid(model_pwr)
A = 0.49608, p-value = 0.2121
```

Rovnako potrebujeme otestovať homoskedasticitu dát. Využijeme Breusch-Pagan test, s nutnosťou normálne rozdelených reziduí. Testujeme len transformované modely. Dostávame hodnoty 0.79 pre logaritmickú a 0.24 pre mocninnú transformáciu v dôsledku čoho pri α=5% nezamietame homoskedasticitu.

Test homoskedasticity vyšiel významnejšie pre logaritmickú transformáciu, čo spolu s úzusom z praxe pre λ z konfidenčného intervalu 0 a ľahšou interpretovateľnosťou modelu s touto transformáciou považujeme za vhodné ďalej pracovať práve s logaritmicky transformovaným modelom

```
# for dependence on mean value (residual variance on the response expectation)
# Breusch-Pagan test statistic - library lmtest
```

```
print("***********************Log************************")
bptest(model_ln)
print("***********************Power************************")
bptest(model_pwr)
```

```
[1] "***********************Log************************"

	studentized Breusch-Pagan test

data:  model_ln
BP = 0.069204, df = 1, p-value = 0.7925
[1] "***********************Power************************"

	studentized Breusch-Pagan test

data:  model_pwr
BP = 1.3695, df = 1, p-value = 0.2419
```

cv_8 help

[ ] ↳ *1 skrytá bunka*

.

[ ] ↳ *34 skrytých buniek*

Vícerozměrný regresní model

- Otázka 7

  Zkonstruujte lineární model s logaritmicky transformovanou odezvou `medv` a zkuste najít vztah mezi cenou a dalšími nezávislými proměnnými, které máte k dispozici (stačí aditivní model bez interakcí). Na základě kritérií jako jsou `AIC`, `BIC`, `$R^2$`, `F`, atd. vyberte podle vás nejvhodnější model. Lze vztah mezi `indus` a `medv`, pokud existuje, vysvětlit pomocí jiných proměnných? Tj, že například v oblastech s větším zastoupením velkoobchodu a průmyslu bydlí chudší lidé, je tam větší znečištění, nebo větší kriminalita atd.?

```
head(Boston_data)
```

A data.frame: 6 × 13

	crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	black	lstat	medv
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	0.00632	18	2.31	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
2	0.02731	0	7.07	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
3	0.02729	0	7.07	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
4	0.03237	0	2.18	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
5	0.06905	0	2.18	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33	36.2
6	0.02985	0	2.18	0.458	6.430	58.7	6.0622	3	222	18.7	394.12	5.21	28.7

V prvom kroku vytvoríme model logaritmickej odozvy zo všetkých premenných. Tento model nadobúda $R^2 = 0.8553$. Štatisticky významnými premennými sú okrem interceptu `rm`, `age`, `dis`, `rad`, `tax`, `ptrario`, `black`, `lstat`. Podotkneme, že premenná `indus`, ktorú sme predtým používali pre modelovanie ceny je v modeli popri ostatných premenných najnevýznamnejšia.

```
#lm(Y ~ .,data=Z)
m_all=lm(log(medv)~.,data = Boston_data)
summary(m_all)
```

```
Call:
lm(formula = log(medv) ~ ., data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.50582 -0.06369 -0.00784  0.06203  0.37730 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.195e+00  1.811e-01  12.120  < 2e-16 ***
crim         1.069e-02  1.711e-02   0.625 0.532657    
zn           6.503e-04  3.579e-04   1.817 0.070098 .  
indus       -3.146e-04  1.591e-03  -0.198 0.843324    
```

Skontrolujeme multikolinearitu pomocou funkcie *vif()*. Hodnoty sú nízke (< 10) a teda môžeme prejsť k modelu s redukovanými premennými na významné.

```
vif(m_all)
```

crim: 2.63767178830982 zn: 2.27357678979342 indus: 2.68027851651607 nox: 4.83322630619225
rm: 2.40934398190735 age: 2.74633818593881 dis: 3.36440609708597 rad: 1.15345051016351
tax: 1.51645012888732 ptratio: 1.48377710681079 black: 1.3501237238269 lstat:

Vytvoríme model s významnými premennými. hodnota R^2 zostáva rovnaká.

```
m_more=lm(log(medv)~rm+ age+ dis+ rad+ tax+ ptratio+ black+ lstat,data=Boston_data)
summary(m_more)
```

```
Call:
lm(formula = log(medv) ~ rm + age + dis + rad + tax + ptratio + 
    black + lstat, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.50965 -0.06750 -0.01092  0.06310  0.38334 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.096e+00  1.451e-01  14.446  < 2e-16 ***
rm           2.787e-01  1.519e-02  18.345  < 2e-16 ***
age         -2.409e-03  3.398e-04  -7.088 8.28e-12 ***
dis         -2.715e-02  3.891e-03  -6.976 1.67e-11 ***
rad          1.346e-02  3.890e-03   3.460 0.000611 ***
tax         -5.883e-04  8.234e-05  -7.145 5.81e-12 ***
ptratio     -2.666e-02  3.005e-03  -8.872  < 2e-16 ***
black        6.257e-04  1.736e-04   3.604 0.000362 ***
lstat       -8.807e-03  1.822e-03  -4.833 2.06e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1134 on 330 degrees of freedom
Multiple R-squared:  0.8533,    Adjusted R-squared:  0.8498 
F-statistic:   240 on 8 and 330 DF,  p-value: < 2.2e-16
```

Vytvorili sme model s 8 premennými (regresormi) s R^2 o hodnote 0.8533 a smerodajnou odchylkou 0.1134. Tento model je pomerne zložitý, keďže pozostáva z 8 regresorov.

Pokúsime sa preto vybrať podmnožinu regresorov, ktorá bude dobre popisovať model. "Ideálny model" by mal mať najmenší možný počet regresorov, ktorý umožňuje adekvátnu interpretáciu (alebo predikciu) *--prednáška 01RAD*

Použijeme teda metódu postupnej (stepwise) regresie. V každom kroku pridámw premennú, a potom skontrolujeme či možno nejakú odobrať. Vyskúšame si kritérium na základe F štatistiky a BIC štatistiky.

```
intercept_only<-lm(log(medv)~1 ,data = Boston_data)
#summary(intercept_only)
AIC(intercept_only)
```

131.775971587473

Stepwise regression with F stat criterion

Vo výpise v bunke nižšie za základe F-value vidíme, že v každom kroku je pridanie ďalšej premennej štatisticky významné. V procese nie je navrhnuté akúkoľvek premennú odobrať.

Pripomíname, že nechceme príliš komplikovaný model. Po konzultácií úlohy s Bc. Eliškou Pečenkovou sa zameriame na modely s približne polovicou regresorov.

Ak vezmeme 3 regresory: `rm`, `lstat`, `ptratio`, dostávame $R^2 = 0.7928$ a $s_n = 0.1339$. V ďalšom kroku nám metóda radí pridať premennú `tax` (4 regresory: `rm`, `lstat`, `ptratio`,`tax`), hodnota R^2 stúpa na 0.8141 a s_n klesá na 0.1269.

Pozreli sme sa na prípad, kedy by sme miesto `tax` pridali premennú `age` a dostali by sme $R^2 = 0.8028$ a $s_n = 0.1307$.

Ak by sme chceli 5 regresorov, dostávame sa na $R^2 = 0.8216$ a $s_n = 0.1245$ pre `rm`, `lstat`, `ptratio`, `tax`, `age`. Ak by sme `age` nahradili premennou `black`, dostávame sa na $R^2 = 0.8202$ a $s_n = 0.125$

```
bothF <- step(intercept_only, direction='both', scope=list(lower=intercept_only, upper=m_more),test="F")
```

```
Start:  AIC=-832.26
log(medv) ~ 1

          Df Sum of Sq     RSS      AIC  F value    Pr(>F)    
+ rm       1   20.3201  8.6147 -1240.99 794.9043 < 2.2e-16 ***
+ lstat    1   15.9149 13.0199 -1100.98 411.9334 < 2.2e-16 ***
+ ptratio  1    5.4113 23.5235  -900.45  77.5229 < 2.2e-16 ***
+ tax      1    4.9755 23.9592  -894.23  69.9832 1.607e-15 ***
+ age      1    4.7150 24.2198  -890.56  65.6050 1.021e-14 ***
+ black    1    1.9622 26.9725  -854.07  24.5168 1.167e-06 ***
+ dis      1    0.9623 27.9725  -841.73  11.5932 0.0007418 ***
+ rad      1    0.1970 28.7377  -832.58   2.3107 0.1294268    
<none>                28.9348  -832.26                       
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=-1240.99
log(medv) ~ rm

          Df Sum of Sq     RSS      AIC  F value    Pr(>F)    
+ lstat    1    1.5648  7.0499 -1306.94  74.5763 2.388e-16 ***
+ age      1    1.5309  7.0838 -1305.32  72.6157 5.394e-16 ***
+ tax      1    1.0861  7.5286 -1284.67  48.4731 1.767e-11 ***
+ ptratio  1    1.0262  7.5885 -1281.99  45.4376 6.875e-11 ***
+ black    1    0.3421  8.2726 -1252.72  13.8934  0.000227 ***
+ dis      1    0.0672  8.5475 -1241.64   2.6416  0.105035    
<none>                 8.6147 -1240.99                       
+ rad      1    0.0308  8.5839 -1240.20   1.2056  0.272996    
- rm       1   20.3201 28.9348  -832.26 794.9043 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=-1306.94
log(medv) ~ rm + lstat

          Df Sum of Sq     RSS     AIC  F value    Pr(>F)    
+ ptratio  1    1.0443  6.0056 -1359.3  58.2545 2.420e-13 ***
+ tax      1    0.6337  6.4163 -1336.9  33.0840 1.990e-08 ***
+ age      1    0.3712  6.6788 -1323.3  18.6176 2.105e-05 ***
+ black    1    0.1300  6.9200 -1311.2   6.2928   0.01260 *  
+ dis      1    0.0634  6.9866 -1308.0   3.0384   0.08223 .  
+ rad      1    0.0536  6.9964 -1307.5   2.5648   0.11021    
<none>                 7.0499 -1306.9                      
- lstat    1    1.5648  8.6147 -1241.0  74.5763 2.388e-16 ***
- rm       1    5.9699 13.0199 -1101.0 284.5265 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=-1359.29
log(medv) ~ rm + lstat + ptratio

          Df Sum of Sq     RSS     AIC  F value    Pr(>F)    
+ tax      1    0.6274  5.3782 -1394.7  38.9625 1.310e-09 ***
+ age      1    0.3002  5.7054 -1374.7  17.5765 3.539e-05 ***
+ black    1    0.2779  5.7277 -1373.3  16.2039 7.042e-05 ***
+ dis      1    0.0688  5.9368 -1361.2   3.8690   0.05001 .  
+ rad      1    0.0409  5.9647 -1359.6   2.2892   0.13122    
<none>                 6.0056 -1359.3                      
```

```
m_3=lm(log(medv) ~ rm + lstat + ptratio, data = Boston_data)
summary(m_3)
m_4=lm(log(medv) ~ rm + lstat + ptratio+ tax, data = Boston_data)
summary(m_4)
m_5=lm(log(medv) ~ rm + lstat + ptratio+ tax + age, data = Boston_data)
summary(m_5)
```

```
Call:
lm(formula = log(medv) ~ rm + lstat + ptratio, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.66583 -0.07442  0.00392  0.07128  0.60381 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.063605   0.146692  14.068  < 2e-16 ***
rm           0.271148   0.016943  16.004  < 2e-16 ***
lstat       -0.016038   0.001707  -9.397  < 2e-16 ***
ptratio     -0.026638   0.003490  -7.632 2.42e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1339 on 335 degrees of freedom
Multiple R-squared:  0.7924,    Adjusted R-squared:  0.7906 
F-statistic: 426.3 on 3 and 335 DF,  p-value: < 2.2e-16

Call:
lm(formula = log(medv) ~ rm + lstat + ptratio + tax, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.55794 -0.07651  0.00066  0.06573  0.53350 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.255e+00  1.424e-01  15.839  < 2e-16 ***
rm           2.649e-01  1.609e-02  16.469  < 2e-16 ***
lstat       -1.386e-02  1.655e-03  -8.373 1.56e-15 ***
ptratio     -2.656e-02  3.308e-03  -8.029 1.69e-14 ***
tax         -5.542e-04  8.878e-05  -6.242 1.31e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1269 on 334 degrees of freedom
Multiple R-squared:  0.8141,    Adjusted R-squared:  0.8119 
F-statistic: 365.7 on 4 and 334 DF,  p-value: < 2.2e-16

Call:
lm(formula = log(medv) ~ rm + lstat + ptratio + tax + age, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.51713 -0.06778 -0.00273  0.06701  0.43257 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
```

```
summary(lm(log(medv) ~ rm + lstat + ptratio +age, data = Boston_data))
summary(lm(log(medv) ~ rm + lstat + ptratio +tax + black, data = Boston_data))
```

```
Call:
lm(formula = log(medv) ~ rm + lstat + ptratio + age, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.66620 -0.06827  0.00297  0.07446  0.49858 

Coefficients:
```

Stepwise regression with BIC

Využili sme výber na základe Bayesovského informačného kritéria, ktorého vlastnosťou je pokutovať so zvyšujúcim sa počtom regresorov.

Tento postup nám dal úplne rovnaké riešenie ako použitie F štatistiky, preto riešenie nebudeme popisovať a odkazujeme sa o bunku s popisom pre F value

```
bothBIC <- step(intercept_only, direction='both', scope=list(lower=intercept_only, upper=m_more),k=log(nobs(m_more)))
```

```
Start:  AIC=-828.44
log(medv) ~ 1

          Df Sum of Sq     RSS      AIC
+ rm       1   20.3201  8.6147 -1233.34
+ lstat    1   15.9149 13.0199 -1093.33
+ ptratio  1    5.4113 23.5235  -892.80
+ tax      1    4.9755 23.9592  -886.58
+ age      1    4.7150 24.2198  -882.91
+ black    1    1.9622 26.9725  -846.42
+ dis      1    0.9623 27.9725  -834.08
<none>                28.9348  -828.44
+ rad      1    0.1970 28.7377  -824.93

Step:  AIC=-1233.34
log(medv) ~ rm

          Df Sum of Sq     RSS      AIC
+ lstat    1    1.5648  7.0499 -1295.46
+ age      1    1.5309  7.0838 -1293.84
+ tax      1    1.0861  7.5286 -1273.19
+ ptratio  1    1.0262  7.5885 -1270.51
+ black    1    0.3421  8.2726 -1241.25
<none>                 8.6147 -1233.34
+ dis      1    0.0672  8.5475 -1230.16
+ rad      1    0.0308  8.5839 -1228.72
- rm       1   20.3201 28.9348  -828.44

Step:  AIC=-1295.46
log(medv) ~ rm + lstat

          Df Sum of Sq     RSS     AIC
+ ptratio  1    1.0443  6.0056 -1344.0
+ tax      1    0.6337  6.4163 -1321.6
+ age      1    0.3712  6.6788 -1308.0
+ black    1    0.1300  6.9200 -1296.0
<none>                 7.0499 -1295.5
+ dis      1    0.0634  6.9866 -1292.7
+ rad      1    0.0536  6.9964 -1292.2
- lstat    1    1.5648  8.6147 -1233.3
- rm       1    5.9699 13.0199 -1093.3

Step:  AIC=-1343.99
log(medv) ~ rm + lstat + ptratio

          Df Sum of Sq     RSS     AIC
+ tax      1    0.6274  5.3782 -1375.6
+ age      1    0.3002  5.7054 -1355.5
+ black    1    0.2779  5.7277 -1354.2
<none>                 6.0056 -1344.0
+ dis      1    0.0688  5.9368 -1342.1
+ rad      1    0.0409  5.9647 -1340.5
- ptratio  1    1.0443  7.0499 -1295.5
- lstat    1    1.5829  7.5885 -1270.5
- rm       1    4.5915 10.5971 -1157.3

Step:  AIC=-1375.57
log(medv) ~ rm + lstat + ptratio + tax
```

Model

Našou úlohou je vybrať čo najvhodnejší model. Rozhodli sme sa vybrať model so 4 regresormi (tj polovicu zo všetkých významných). Do modelu zakomponujeme premenné `rm`, `lstat`, `ptratio`, `tax` a teda model má $R^2 = 0.8141$ a $s_n = 0.1269$.

Rozhodli sme sa uprednostniť premennú `tax` pred premnennou `black`, hoci by mal model o niečo vyšší R^2 a to z dôvodu, že nechceme do modelu zahŕňať prvky rasizmu.

Na záver zostáva preskúmať premennú `indus`.

Skúsime vytvoriť model prostredníctvom stepwise regresie s východzím modelom postavenom na premennej `indus`, ktorý má $R^2 = 0.2607$ a premenná je veľmi významná.

Keď k modelu pridáme premennú `rm`, obe premenné sú významné a kvalita modelu sa zvýši razantne, teda neuvažujeme vzťah týchto dvoch premenných.

Pridaním premennej `lstat` sa významnosť `indus` zníži, ak navyše volíme interakciu, `lstat:indus`, potom premenná indus je nevýznamná. Z tohto pozorovania možno usúdiť, že v oblastiach fabrík a skladov býva viac ľudí s nižším sociálnym postavením. Navyše vidíme aj vyšší korelačný koeficient medzi premennými.

Kombinácia `indus`, `lstat` a `tax` zaistí nevýznamnosť premennej indus v modeli. Povšimneme si, že všetky koeficienty sú záporné a teda ak sa zvýši nemaloobchodné podnikanie (továrne, sklady), daň z nehnuteľnosti sa zníži.

Ak vytvoríme model len z premennej `crim`, táto premenná je štatisticky významná, avšak ak pridáme premennú `indus` tak kriminalita už nemá štatistický význam pre náš model a teda platí, že v oblasti s vyšším počtom neobchodných území klesá kriminalita (oba koeficienty majú záporné znamienka, ak jednému uberieme, druhému sa pridá). Navyše vidíme aj vyšší korelačný koeficient medzi premennými.

Ak vytvoríme model v závislosti na `indus`,`nox` potom premenná `nox` už je automaticky štatisticky nevýznamná a medzi premennými vidíme spojitosť. Navyše vidíme aj vyšší korelačný koeficient medzi premennými.

```
m_indus<- step(model_ln, direction='both', scope=list(lower=model_ln, upper=m_all),test="F")
```

```
Start:  AIC=-932.67
log(medv) ~ indus

          Df Sum of Sq     RSS      AIC  F value    Pr(>F)    
+ rm       1   13.5059  7.8847 -1269.00 575.5417 < 2.2e-16 ***
+ lstat    1    8.6772 12.7134 -1107.05 229.3290 < 2.2e-16 ***
+ ptratio  1    3.3781 18.0125  -988.94  63.0140 3.102e-14 ***
+ tax      1    0.9878 20.4028  -946.70  16.2674 6.812e-05 ***
+ age      1    0.7223 20.6683  -942.32  11.7431 0.0006865 ***
+ dis      1    0.6664 20.7242  -941.40  10.8051 0.0011192 ** 
+ zn       1    0.4911 20.8995  -938.55   7.8959 0.0052450 ** 
+ black    1    0.4314 20.9592  -937.58   6.9156 0.0089377 ** 
+ crim     1    0.1697 21.2209  -933.37   2.6866 0.1021299    
+ rad      1    0.1532 21.2374  -933.11   2.4235 0.1204687    
<none>                21.3906  -932.67                       
+ nox      1    0.0491 21.3415  -931.45   0.7732 0.3798632    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=-1269
log(medv) ~ indus + rm

          Df Sum of Sq     RSS      AIC  F value    Pr(>F)    
+ lstat    1    0.9668  6.9179 -1311.35  46.8167 3.718e-11 ***
+ ptratio  1    0.9175  6.9672 -1308.94  44.1161 1.252e-10 ***
+ age      1    0.8789  7.0058 -1307.07  42.0256 3.226e-10 ***
+ tax      1    0.5519  7.3328 -1291.60  25.2135 8.357e-07 ***
+ black    1    0.1614  7.7233 -1274.02   7.0027  0.008523 ** 
+ dis      1    0.0938  7.7909 -1271.06   4.0348  0.045373 *  
<none>                 7.8847 -1269.00                       
+ rad      1    0.0332  7.8515 -1268.43   1.4163  0.234848    
+ zn       1    0.0241  7.8606 -1268.04   1.0257  0.311902    
+ nox      1    0.0044  7.8803 -1267.19   0.1856  0.666869    
+ crim     1    0.0006  7.8841 -1267.03   0.0254  0.873499    
- rm       1   13.5059 21.3906  -932.67 575.5417 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=-1311.35
log(medv) ~ indus + rm + lstat

          Df Sum of Sq     RSS     AIC  F value    Pr(>F)    
+ ptratio  1    0.9937  5.9242 -1361.9  56.0260 6.408e-13 ***
+ tax      1    0.5046  6.4134 -1335.0  26.2771 5.018e-07 ***
+ age      1    0.2678  6.6501 -1322.7  13.4507 0.0002849 ***
+ dis      1    0.2599  6.6580 -1322.3  13.0395 0.0003518 ***
+ black    1    0.0925  6.8255 -1313.9   4.5241 0.0341539 *  
+ nox      1    0.0535  6.8644 -1312.0   2.6044 0.1075102    
+ rad      1    0.0516  6.8663 -1311.9   2.5091 0.1141390    
+ crim     1    0.0515  6.8664 -1311.9   2.5067 0.1143091    
<none>                 6.9179 -1311.3                       
+ zn       1    0.0027  6.9153 -1309.5   0.1287 0.7200027    
- lstat    1    0.9668  7.8847 -1269.0  46.8167 3.718e-11 ***
- rm       1    5.7955 12.7134 -1107.0 280.6448 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=-1361.92
```

```
summary(model_ln)
print("***************************************************")
print("***************************************************")
summary(lm(log(medv) ~ indus + rm,data=Boston_data))
print("***************************************************")
print("***************************************************")
summary(lm(log(medv) ~ indus + lstat,data=Boston_data))
print("***************************************************")
print("***************************************************")
summary(lm(log(medv) ~ indus +lstat:indus,data=Boston_data))
print("***************************************************")
print("***************************************************")
summary(lm(log(medv) ~  indus + lstat + tax , data = Boston_data))
print("***************************************************")
print("***************************************************")
summary(lm(log(medv) ~ indus  + crim,data=Boston_data))
print("***************************************************")
print("***************************************************")
summary(lm(log(medv) ~ crim,data=Boston_data))
```

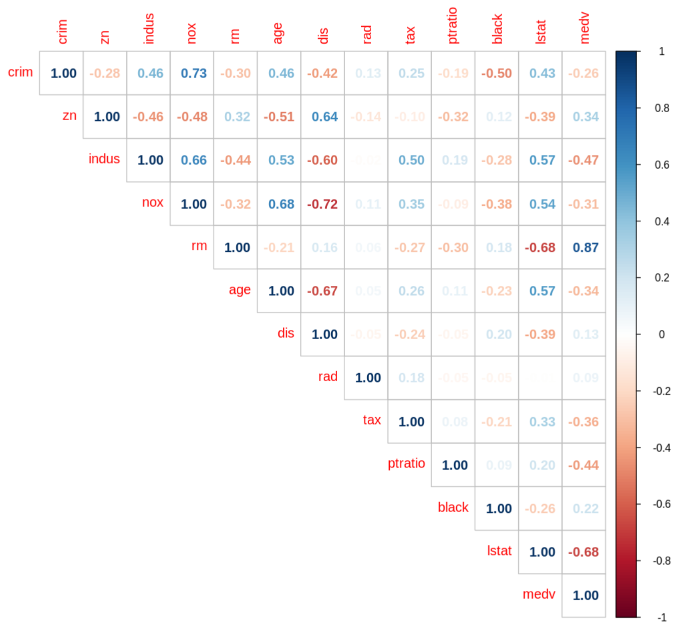
```
Call:
lm(formula = log(medv) ~ indus, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.73082 -0.15957 -0.01244  0.16211  0.85196 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.33005    0.02293   145.2   <2e-16 ***
indus       -0.02355    0.00216   -10.9   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2519 on 337 degrees of freedom
Multiple R-squared:  0.2607,	Adjusted R-squared:  0.2585 
F-statistic: 118.9 on 1 and 337 DF,  p-value: < 2.2e-16

[1] "****************************************************"
[1] "****************************************************"

Call:
lm(formula = log(medv) ~ indus + rm, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.69044 -0.07901  0.00234  0.09999  0.47767 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.868217   0.103560   8.384 1.43e-15 ***
indus       -0.008154   0.001462  -5.577 5.03e-08 ***
rm           0.368364   0.015355  23.990  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1532 on 336 degrees of freedom
Multiple R-squared:  0.7275,	Adjusted R-squared:  0.7259 
F-statistic: 448.5 on 2 and 336 DF,  p-value: < 2.2e-16

[1] "****************************************************"
[1] "****************************************************"

Call:
lm(formula = log(medv) ~ indus + lstat, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.72496 -0.13295 -0.02117  0.11262  0.68351 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.535184   0.022290 158.600   <2e-16 ***
indus       -0.005802   0.002039  -2.846   0.0047 ** 
lstat       -0.033563   0.002216 -15.144   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1945 on 336 degrees of freedom
Multiple R-squared:  0.5606,	Adjusted R-squared:  0.558 
F-statistic: 214.4 on 2 and 336 DF,  p-value: < 2.2e-16

[1] "****************************************************"
[1] "****************************************************"

Call:
lm(formula = log(medv) ~ indus + lstat:indus, data = Boston_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.7042 -0.1421 -0.0377  0.1446  0.6491 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.2489359  0.0230533 140.932  < 2e-16 ***
indus        0.0068005  0.0041387   1.643    0.101    
indus:lstat -0.0015890  0.0001906  -8.338 1.97e-15 ***
---
```

```
summary(lm(log(medv)~indus+nox,data=Boston_data))
```

```
Call:
lm(formula = log(medv) ~ indus + nox, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.74617 -0.15744 -0.01244  0.15809  0.84944

Coefficients:
```

```
# improved correlation matrix
install.packages("corrplot")
library(corrplot)

corrplot(cor(Boston_data),
  method = "number",
  type = "upper" # show only upper side
)
```

```
Installing package into '/usr/local/lib/R/site-library'
(as 'lib' is unspecified)
```

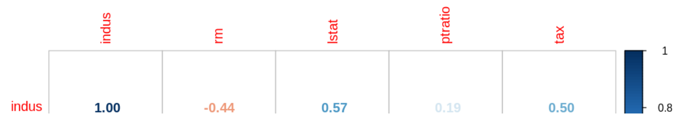



```
Residual standard error: 0.2791 on 337 degrees of freedom
```

- Otázka 8

  Použijte ve výsledném modelu proměnnou `indus` a porovnejte jak se změnil její vliv na medián ceny nemovitostí oproti jednoduchému regresnímu modelu s log transformovanou odezvou (viz otázka 4). Jaké je snížení průměrné ceny nemovitostí při vzrůstu proměnné indus o jednu jednotku? Pokud proměnnou `indus` v modelu nemáte tak ji pro tuto otázku do modelu přiřaďte na úkor jiné proměnné s kterou je nejvíce korelovaná.

```
corrplot(cor(Boston_data[,c("indus","rm","lstat","ptratio","tax")]),
  method = "number",
  type = "upper" # show only upper side
)
```



Model, ktorý sme zvolili v otázke 7) je založený na premenných `rm`, `lstat`, `ptratio` a `tax`. Premenná `indus` najviac koreluje s premennou `lstat` (ρ=0.57), preto vložíme do modelu na jej úkor.

```
summary(model_ln)
```

```
Call:
lm(formula = log(medv) ~ indus, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.73082 -0.15957 -0.01244  0.16211  0.85196 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.33005    0.02293   145.2   <2e-16 ***
indus       -0.02355    0.00216   -10.9   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2519 on 337 degrees of freedom
Multiple R-squared:  0.2607,	Adjusted R-squared:  0.2585 
F-statistic: 118.9 on 1 and 337 DF,  p-value: < 2.2e-16
```

```
m_i_add <- lm(log(medv) ~  rm + indus + ptratio + tax, data = Boston_data)
summary(m_i_add)
```

```
Call:
lm(formula = log(medv) ~ rm + indus + ptratio + tax, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.51729 -0.07809  0.01042  0.08197  0.41848 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.6739569  0.1325167  12.632  < 2e-16 ***
rm           0.3378917  0.0143277  23.583  < 2e-16 ***
indus       -0.0039302  0.0014703  -2.673  0.00789 ** 
ptratio     -0.0256338  0.0036097  -7.101 7.48e-12 ***
tax         -0.0005877  0.0001051  -5.591 4.69e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1381 on 334 degrees of freedom
Multiple R-squared:  0.7798,	Adjusted R-squared:  0.7772 
F-statistic: 295.7 on 4 and 334 DF,  p-value: < 2.2e-16
```

```
m_i_add$coefficients
```

(Intercept): 1.6739569053096 rm: 0.337891747423145 indus: -0.00393017913857012 ptratio: -0.0256338310602392 tax: -0.000587747849002026

V pôvodnom jednoduchom modeli pri náraste podielu nemaloobchodných zón o jednotku dôjde k poklesu cien nehnuteľností o 2.33%. V našom novom modeli, ktorý má 4 vysvetľujúce premenné dôjde pri rovnakej jednotkovej zmene premennej `indus` k poklesu o 0.39%.

Teda premenná `indus` má v rozsiahlejšom modeli nižší vplyv na vývoj ceny, než keby bola jedinou vysvetľujúcou premennou.

```
unname(100*(exp(model_ln$coefficients[2])-1))
unname(100*(exp(m_i_add$coefficients[3])-1))
```

-2.3276413163198
-0.39224660923991

- Otázka 9

  Prezentujte váš výsledný model pro predikci `medv`, diskutujte výsledné parametry R^2, σ, F a porovnejte je s jednoduchým lin. modelem z otázky 6. Jak se změnily a dala se tato změna očekávat? Validujte model (jak graficky, tak pomocí příslušných testů hypotéz). Pomocí Partial regression plots a Partial residual plots diskutujte linearitu použitých proměnných.

V predošlej otázke sme vystavali model na základe premenných `indus`, `rm`, `ptratio`, `tax`. Pozrieme sa tera na základné štatistiky tohto modelu, do zátvorky vždy uvedieme hodnoty pre jednduchý lineárny model založenom len na premennej `indus`.

- $R^2 = 0.7798$ (0.2519), nárast štatistiky sme očakávali, pretože s pridaním premenných sme boli schopný vysvetliť väčšiu časť variability modelu než tomu bolo pri jednoduchom modeli.
- $\sigma = s_n = 0.1381$ (0.2607), pokles smerodajnej odchylky reziduí sa dal opäť očakávať, pretože s pridaním významných premenných sa dáta dajú lepšie modelovať a teda sú bližšie rozptýlené a linear regression fit sa zlepšuje.
- $F - stat = 295.7$ s $p - val = 2.2 \cdot 10^{-16}$ (118.9;$2.2 \cdot 10^{-16}$), zvýšenie sme opäť predpokladali a p-value < 0.05 nám naznačuje že aspoň 1 nezávislá premenná vysvetľuje Y. (*If the p-value associated with the F-statistic < 0.05: Then, AT LEAST 1 independent variable is related to Y.*)

```
summary(model_ln)
```

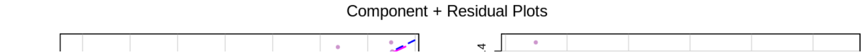
```
Call:
lm(formula = log(medv) ~ indus, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.73082 -0.15957 -0.01244  0.16211  0.85196 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.33005    0.02293   145.2   <2e-16 ***
indus       -0.02355    0.00216   -10.9   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2519 on 337 degrees of freedom
Multiple R-squared:  0.2607,	Adjusted R-squared:  0.2585 
F-statistic: 118.9 on 1 and 337 DF,  p-value: < 2.2e-16
```

```
summary(m_i_add)
```

```
Call:
lm(formula = log(medv) ~ rm + indus + ptratio + tax, data = Boston_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.51729 -0.07809  0.01042  0.08197  0.41848 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.6739569  0.1325167  12.632  < 2e-16 ***
rm           0.3378917  0.0143277  23.583  < 2e-16 ***
indus       -0.0039302  0.0014703  -2.673  0.00789 ** 
ptratio     -0.0256338  0.0036097  -7.101 7.48e-12 ***
tax         -0.0005877  0.0001051  -5.591 4.69e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1381 on 334 degrees of freedom
Multiple R-squared:  0.7798,	Adjusted R-squared:  0.7772 
F-statistic: 295.7 on 4 and 334 DF,  p-value: < 2.2e-16
```

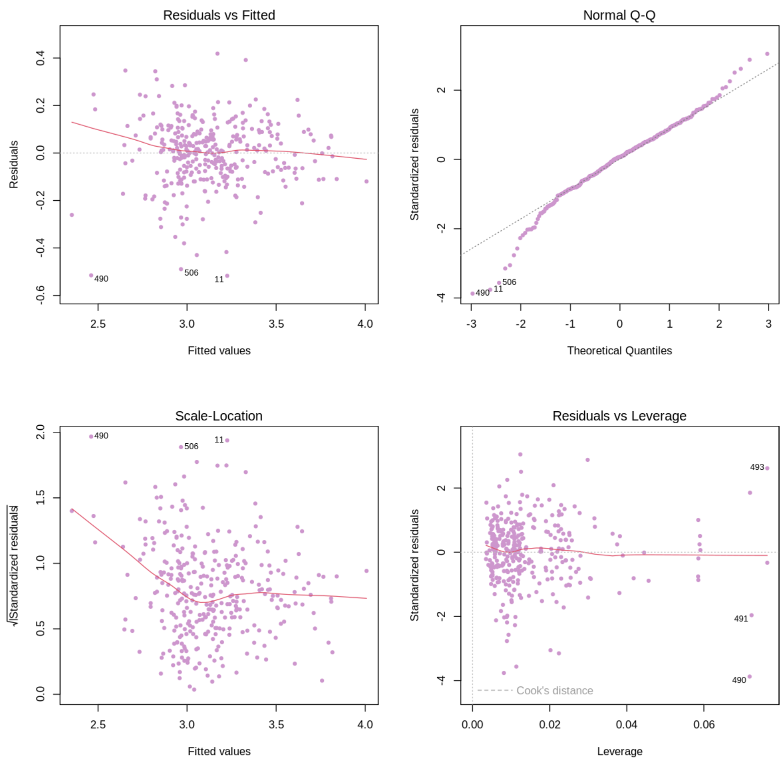
Najprv vybraný model znázorníme graficky.

```
crPlots(m_i_add, col="plum3",pch=20)
```



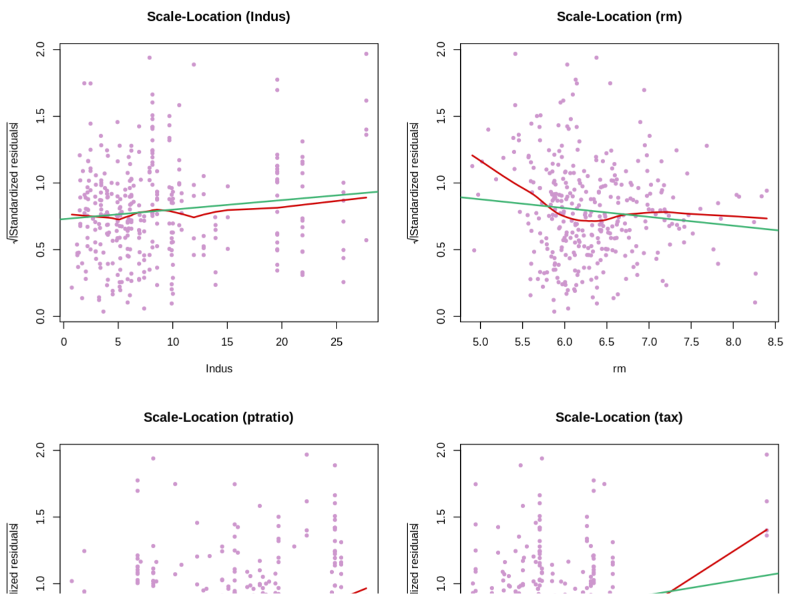
Na obrázku *Residuals vs Fitted* si všimneme, že dáta majú v oblasti 3.0-3.25 x-ovej osi väčší rozptul reziduú ako v oblasti 3.8-4.0. Ďalej na Q-Q plote pozorujeme problém s chvostami rozdelenia, ťažké chvosty nie sú znakom normálneho rozdelenia. Podozrenie na porušenie homoskedasticity a normality otestujeme príslušnými testami.

```
fig(10,10)
par(mfrow=c(2,2))
plot(m_i_add, col="plum3",pch=20)
```

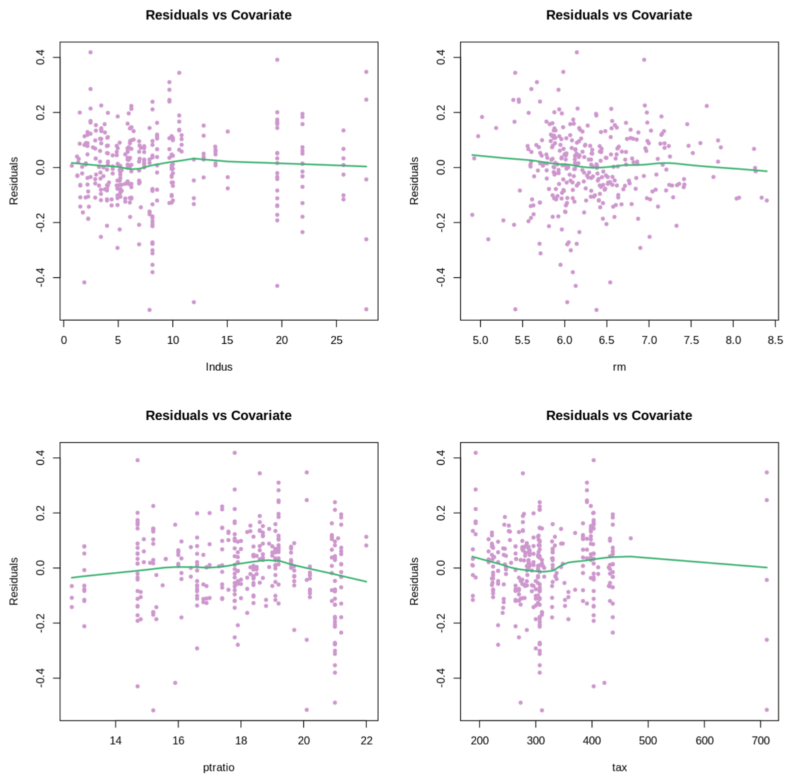


Vykreslíme reziduá v závislosti na jednotlivých premenných. Nepozorujeme však šiadne podozrivé rozloženie reziduí, naopak všímame si rovnomernosť.

```
fig(10,10)
par(mfrow=c(2,2))
plot(Boston_data$indus, sqrt(abs(rstandard(m_i_add))), pch = 20, col = "plum3", bg = "royalblue4",
     xlab = "Indus", ylab = as.expression(substitute(sqrt(abs(yL)), list(yL = as.name("Standardized residuals")))),
     main = "Scale-Location (Indus)")
lines(lowess(Boston_data$indus, sqrt(abs(rstandard(m_i_add)))), col = "red3", lwd = 2)
abline(lm(sqrt(abs(rstandard(m_i_add)))~Boston_data$indus), col = "mediumseagreen", lwd = 2)
plot(Boston_data$rm, sqrt(abs(rstandard(m_i_add))), pch = 20, col = "plum3", bg = "royalblue4",
     xlab = "rm", ylab = as.expression(substitute(sqrt(abs(yL)), list(yL = as.name("Standardized residuals")))),
     main = "Scale-Location (rm)")
lines(lowess(Boston_data$rm, sqrt(abs(rstandard(m_i_add)))), col = "red3", lwd = 2)
abline(lm(sqrt(abs(rstandard(m_i_add)))~Boston_data$rm), col = "mediumseagreen", lwd = 2)
plot(Boston_data$ptratio, sqrt(abs(rstandard(m_i_add))), pch = 20, col = "plum3", bg = "royalblue4",
     xlab = "ptratio", ylab = as.expression(substitute(sqrt(abs(yL)), list(yL = as.name("Standardized residuals")))),
     main = "Scale-Location (ptratio)")
lines(lowess(Boston_data$ptratio, sqrt(abs(rstandard(m_i_add)))), col = "red3", lwd = 2)
abline(lm(sqrt(abs(rstandard(m_i_add)))~Boston_data$ptratio), col = "mediumseagreen", lwd = 2)
plot(Boston_data$tax, sqrt(abs(rstandard(m_i_add))), pch = 20, col = "plum3", bg = "royalblue4",
     xlab = "tax", ylab = as.expression(substitute(sqrt(abs(yL)), list(yL = as.name("Standardized residuals")))),
     main = "Scale-Location (tax)")
lines(lowess(Boston_data$tax, sqrt(abs(rstandard(m_i_add)))), col = "red3", lwd = 2)
abline(lm(sqrt(abs(rstandard(m_i_add)))~Boston_data$tax), col = "mediumseagreen", lwd = 2)
```




```
fig(10,10)
par(mfrow=c(2,2))
plot(Boston_data$indus, residuals(m_i_add), pch = 20, col = "plum3",  xlab = "Indus", ylab = "Residuals", main = "Residuals vs Covariate'
lines(lowess(Boston_data$indus, residuals(m_i_add)), col = "mediumseagreen", lwd = 2)
plot(Boston_data$rm, residuals(m_i_add), pch = 20, col = "plum3",  xlab = "rm", ylab = "Residuals", main = "Residuals vs Covariate")
lines(lowess(Boston_data$rm, residuals(m_i_add)), col = "mediumseagreen", lwd = 2)
plot(Boston_data$ptratio, residuals(m_i_add), pch = 20, col = "plum3",  xlab = "ptratio", ylab = "Residuals", main = "Residuals vs Covari
lines(lowess(Boston_data$ptratio, residuals(m_i_add)), col = "mediumseagreen", lwd = 2)
plot(Boston_data$tax, residuals(m_i_add), pch = 20, col = "plum3",  xlab = "tax", ylab = "Residuals", main = "Residuals vs Covariate")
lines(lowess(Boston_data$tax, residuals(m_i_add)), col = "mediumseagreen", lwd = 2)
```



Prejdeme k jednotlivým testom.

Normalita

Overíme si, či naše podozrenie porušenia normality reziduí je oprávnené a teda využijeme Lilieforsov, Shapiro-Wilkov a Anderson-Darlingov test normality na hladine α=0.05.

Lilieforsov test tesne nezamieta hypotézu o normalite, ale Shapiro-Wilkov a Anderson Darlingov test normalitu rozdelenia zamietajú. Naše obavy sa potvrdili.

```
#Normality
lillie.test(resid(m_i_add))   # Lilliefors test
shapiro.test(resid(m_i_add))      # Shapiro-Wilk test
ad.test(resid(m_i_add))
```

```
	Lilliefors (Kolmogorov-Smirnov) normality test

data:  resid(m_i_add)
D = 0.047342, p-value = 0.06478

	Shapiro-Wilk normality test

data:  resid(m_i_add)
W = 0.97415, p-value = 9.161e-06

	Anderson-Darling normality test

data:  resid(m_i_add)
A = 1.7081, p-value = 0.0002193
```

Homoskedasticita

Chceme otestovať homoskedasticitu. Bohužiaľ nemáme normálne rozdelené reziduá, čo je podmienka Breusch-Paganovho test. Avšak, na hodinách sme si takýto prípad neukazovali a na internete som našla len Goldfeld-Quandt Tests, ktorého neparametrická varianta by teoreticky šla použiť, ale daná knižnica/balíček nejde nainštalovať.

Napriek okolnostiam použijeme `bptest()` na úrovni α=0.05.

Testom dostávame p-val=0.001838<α a teda zamietame nulovú hypotézu o tom, že by šlo o homoskedasticitu.

```
#NOT WORKING
#install.packages("skedastic")
#library(skedastic)
#goldfeld_quandt(mtcars_lm, deflator = "qsec", method = "nonparametric", restype = "blus", alternative = "two.sided")
```

```
bptest(m_i_add)
```

```
	studentized Breusch-Pagan test

data:  m_i_add
BP = 17.113, df = 4, p-value = 0.001838
```

Nezávislosť

Nezávislosť reziduí otestujeme Durbin-Watsnovým testom, pričom nezávislosť zamietame s p-hodnotou $2.2 \cdot 10^{-16}$.

```
dwtest(m_i_add, alternative = "two.sided")  # Two-sided
```

```
	Durbin-Watson test

data:  m_i_add
DW = 1.1017, p-value < 2.2e-16
alternative hypothesis: true autocorrelation is not 0
```

- Otázka 10

  Na základě vašeho modelu odpovězte, zdali si myslíte, že pokud bychom dokázali snížit/zvýšit podíl maloobchodu v dané lokalitě, vedlo by to ke zvýšení cen nemovitostí určených k bydlení v dané lokalitě?

Výsledný model má tvar:

$$ln(Y) = 1.674 + 0.338 \cdot rm - 0.004 \cdot indus - 0.026 \cdot ptratio - 0.001 \cdot tax,$$

čo možno prepísať:

$$Y = e^{1.674} e^{0.338 \cdot rm} e^{-0.004 \cdot indus} e^{-0.026 \cdot ptratio} e^{-0.001 \cdot tax}$$

a Y predstavuje `medv`.

V našom modeli sa premenná `indus` vyskytuje so záporným znamienkom. A teda ak znížime podiel nemaloobchodných zón v danej lokalite, viedlo by to k zvýšeniu cien nehnuteľností. Poznamenajme, že koeficient je pomerne nízky (oproti hodnote interceptu či koeficientu premennej `rm`) a táto zmena v cene bude nízka pre malé zmeny - 0.39% pri jednotkovej zmene podielu neobchodných zón.

Medziročný nárast cien domov v ČR je 17,6% (k dňu 16.11.22 viď). K takejto zmene by došlo pri poklese spomínaného podielu o 45 jednotiek (takmer ako dvojnásobok maximálneho non-retail business podielu v našej databáze).

```
17.6/0.39
```

```
summary(Boston_data$indus)
```

45.1282051282051

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  0.740   3.985   6.410   8.515  10.010  27.740 
```

✓ 0 s dokončené o 19:18