Regresní model pro reakční dobu

Jaroslav Šafář

3. června 2021

1 Lineární regrese

Při lineární regresi předpokládáme, že střední hodnoty nekorelovaných náhodných veličin $Y_1, \dots Y_n$ lze popsat lineární funkcí $p+1$ neznámých parametrů $\beta = (\beta_0, \beta_1, \dots, \beta_p)^T$ následovně:

$$\mathbb{E}[Y_i] = \beta_0 + \beta_1 x_{i,1} + \dots \beta_p x_{i,p},$$

kde $x_{i,j}$ jsou známé konstanty. Tyto konstanty můžeme uspořádat do matice $\mathcal{X}$ řádu $n \times (p+1)$ (tzv. matice vysvětlujících proměnných) tvaru:

$$\begin{pmatrix} 1 & x_{1,1} & \dots & x_{1,p} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n,1} & \dots & x_{n,p} \end{pmatrix}.$$

Maticově potom můžeme psát:

$$\mathbb{E}[\mathbf{Y}] = \mathcal{X}\beta.$$

Dále předpokládáme, že $\sigma^2 > 0$ je rozptyl nekorelovaných náhodných chyb kolem středních hodnot $\mathcal{X}\beta$, tj. $\operatorname{Var} Y_i = \sigma^2$ pro všechna i. Jinými slovy:

$$\mathbf{Y} \sim \left(\mathcal{X}\beta, \sigma^2 \mathcal{I}_n\right),$$

Lineární model můžeme ekvivalentně zapsat také ve tvaru:

$$Y_i = \beta_0 + \beta_1 x_{i,1} + \dots \beta_p x_{i,p} + \varepsilon_i,$$

kde $\varepsilon_i \sim \left(0, \sigma^2\right)$ jsou již zmiňované nekorelované náhodné chyby. Maticově můžeme psát:

$$\mathbf{Y} = \mathcal{X}\beta + \mathcal{E}, \quad \mathcal{E} \sim \left(\mathbf{0}, \sigma^2 \mathcal{I}_n\right).$$

Velice často platí, že řádky $\mathbf{x}_i$ matice $\mathcal{X}$ jsou nějakou transformací $\mathbf{f}$ vektorů konstant $\mathbf{z}_i$ (např. pozorovaných či naměřených hodnot), tj.

$$\mathbf{x}_i = (1, x_{i,1}, \dots, x_{i,p})^T = \left(f_0(\mathbf{z}_i), f_1(\mathbf{z}_i), \dots, f_p(\mathbf{z}_i)\right)^T = \mathbf{f}(\mathbf{z}_i).$$

Parametry modelu odhadneme pomocí metody nejmenších čtverců a pozorovaných dvojic dat $(\mathbf{z_i}, y_i)$:

$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^{p+1}} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \arg\min_{\beta \in \mathbb{R}^{p+1}} \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^T \beta\right)^2 = \arg\min_{\beta \in \mathbb{R}^{p+1}} (\mathbf{y} - \mathcal{X}\beta)^T (\mathbf{y} - \mathcal{X}\beta).$$

Položíme derivaci této funkce rovnou nulovému vektoru a dostáváme tzv. normální rovnici:

$$\mathcal{X}^T \mathcal{X} \beta = \mathcal{X}^T \mathbf{y},$$

ze které, za předpokladu, že $\mathcal{X}$ má plnou hodnost, plyne:

$$\hat{\beta} = \left(\mathcal{X}^T \mathcal{X}\right)^{-1} \mathcal{X}^T \mathbf{y}.$$

Za dodatečného předpokladu normality náhodných chyb ε_i, tj. $\mathbf{Y} \sim N_n\left(\mathcal{X}\beta, \sigma^2 \mathcal{I}_n\right)$, lze ukázat, že při t-testu o j-tém regresním koeficientu platí:

$$T_j = \frac{\hat{\beta}_j - \beta_j}{S\sqrt{v_{j,j}}} \sim t_{n-(p+1)},$$

kde $S^2 = \hat{\sigma}^2$ je nevychýlený odhad rozptylu, pro který platí:

$$S^2 = \frac{\sum_{i=1}^{n} \left(y_i - \hat{y}_i\right)^2}{n - (p+1)}$$

a $v_{k,l}$ jsou prvky matice $\mathcal{V}$:

$$\mathcal{V} = \left(\mathcal{X}^T \mathcal{X}\right)^{-1}.$$

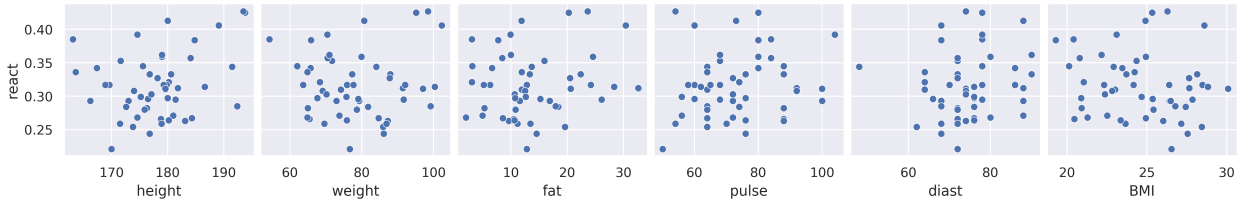
2 Regresní model pro reakční dobu

Za výše zmíněných předpokladů se budeme nyní snažit sestavit regresní model pro **reakční dobu** (react) pomocí následujících dat o policii pomocí lineární regrese. Pro naše dvojice pozorovaných dat $(\mathbf{z_i}, y_i)$ tedy platí, že $(\mathbf{z_i}, y_i) = ((height, weight, fat, pulse, diast)_i, react_i)$.

Jako další zajímavá vysvětlující proměnná by mohlo být BMI, které je definované následovně:

$$BMI_i = \frac{weight_i[kg]}{height_i^2[m^2]}.$$

V obrázku [1] jsou uvedené grafy závislosti reakční doby (react) na každé z těchto proměnných.



Obrázek 1: Závislost reakční doby (react) na jednotlivých proměnných.

Asi bychom nečekali, že samotná výška (height) bude mít nějaký vliv na reakční dobu. Jelikož je navíc součástí BMI, tak jí pro zjednodušení nadále vynecháme.

2.1 Konstrukce regresního modelu a jeho výstup

Pokusíme se o lineární regresi s lineární transformací $\mathbf{f}$, tj. matice konstant $\mathcal{X}$ bude rovna:

$$\mathcal{X} = \begin{pmatrix} 1 & x_{1,1} & \cdots & x_{1,5} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n,1} & \cdots & x_{n,5} \end{pmatrix} = \begin{pmatrix} 1 & weight_1 & fat_1 & pulse_1 & diast_1 & BMI_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & weight_n & fat_n & pulse_n & diast_n & BMI_n \end{pmatrix}.$$

K výpočtu použijeme následující kód v jazyku Python:

```
import statsmodels.api as sm
import numpy as np
import pandas as pd
df = pd.read_csv('police.csv', sep='\t', header=0)
df['BMI'] = 10000 * df['weight'] / (df['height'] * df['height'])
y, X = dmatrices('react ~ weight + fat + pulse + diast + BMI', data=df)
model = sm.OLS(y, X).fit()
model.summary()
```

Výsledkem je následující výstup:

Dep. Variable:	react	**R-squared:**	0.290
Model:	OLS	**Adj. R-squared:**	0.209
Method:	Least Squares	**F-statistic:**	3.596
Date:	Wed, 02 Jun 2021	**Prob (F-statistic):**	0.00819
Time:	18:38:37	**Log-Likelihood:**	89.729
No. Observations:	50	**AIC:**	-167.5
Df Residuals:	44	**BIC:**	-156.0
Df Model:	5		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Intercept	0.4375	0.109	4.004	0.000	0.217	0.658
weight	0.0017	0.001	1.500	0.141	-0.001	0.004
fat	0.0037	0.002	2.376	0.022	0.001	0.007
pulse	0.0003	0.001	0.514	0.610	-0.001	0.001
diast	0.0010	0.001	1.261	0.214	-0.001	0.003
BMI	-0.0162	0.004	-3.657	0.001	-0.025	-0.007

Omnibus:	1.162	**Durbin-Watson:**	1.650
Prob(Omnibus):	0.559	**Jarque-Bera (JB):**	1.139
Skew:	0.239	**Prob(JB):**	0.566
Kurtosis:	2.435	**Cond. No.**	2.42e+03

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.42e+03. This might indicate that there are
strong multicollinearity or other numerical problems.

Z výstupu vidíme (sloupec **coef**), že koeficienty u vysvětlujících proměnných jsou velmi blízko nule. Ve výstupu vidíme také p-hodnotu testů důležitosti jednotlivých parametrů $\hat{\beta}$ pomocí již zmíněného t-testu (sloupec $\mathbf{P} > |\mathbf{t}|$), kde pro každý parametr uvažujeme následující nulovou a alternativní hypotézu:

- H_0: $\beta_i = 0$
- H_1: $\beta_i \neq 0$

V tomto případě tímto testujeme, nakolik i-tá vysvětlující proměnná vypovídá o chování střední hodnoty reakční doby nad to, co o jejím chování víme z ostatních nezávisle proměnných. Pokud zvolíme hladinu

významnosti $\alpha = 0.05$, pak hypotézu o nulovosti parametrů zamítáme v případech $\hat{\beta}_{Intercept}$, $\hat{\beta}_{fat}$ a $\hat{\beta}_{BMI}$. Nulovou hypotézu naopak nezamítáme v případě $\hat{\beta}_{weight}$, $\hat{\beta}_{pulse}$ a $\hat{\beta}_{diast}$. Můžeme tedy říci, že proměnné *weight*, *pulse* a *diast* nemají v tomto modelu statisticky signifikantní vliv na reakční dobu.

Regresní koeficienty příslušné vysvětlujícím proměnným udávají, o kolik se změní reakční doba při jednotkové změně dané proměnné za předpokladu, že hodnoty všech ostatních proměnných zůstanou stejné. Zajímavé např. je, že jediné $\hat{\beta}_{BMI}$ je záporné, což bychom čekali. Koeficient $\hat{\beta}_{Intercept}$ udává průměrnou reakční dobu při nulové hodnotě všech vysvětlujících proměnných. To ovšem v reálu nemůžeme čekat, jelikož např. člověk s nulovou hmotnostní neexistuje.

2.2 Grafické znázornění modelu

Pro grafické znázornění našeho modelu použijeme pro každou nezávisle proměnnou x funkci:

```
sm.graphics.plot_regress_exog(model, 'x')
```

která nám pro každé x vrátí následující čtveřici grafů:

1. graf vlevo nahoře znázorňuje pozorované hodnoty reakční doby (react) a jejich odhady (fitted) pomocí našeho regresního modelu proti hodnotám zvolené nezávisle proměnné. Dále jsou zde vidět konfidenční intervaly.
2. graf vpravo nahoře znázorňuje rezidua vzhledem k hodnotám zvolené nezávisle proměnné.
3. graf vlevo dole je částečný graf našeho modelu a udává závislost reakční doby na hodnotě zvolené nezávisle proměnné podmíněnou ostatními nezávisle proměnnými.
4. graf vpravo dole je CCPR graf.

Tyto grafy jsou znázorněny v následujících obrázcích [2, 3, 4, 5, 6]. Můžeme si všimnout, že v grafech reziduí nejsou nějaké pravidelnosti, které by nasvědčovaly tomu, že zvolený model není vhodný. Data jsou víceméně rovnoměrně rozprostřena kolem horizontální přímky procházející nulou. Pouze v případě vysvětlující proměnné *diast* [5] jsou data soustředěna na pravou stranu, to je ovšem způsobeno tím, že zde máme jedno odlehlé pozorování s nízkou hodnotou *diast* oproti ostatním pozorováním.

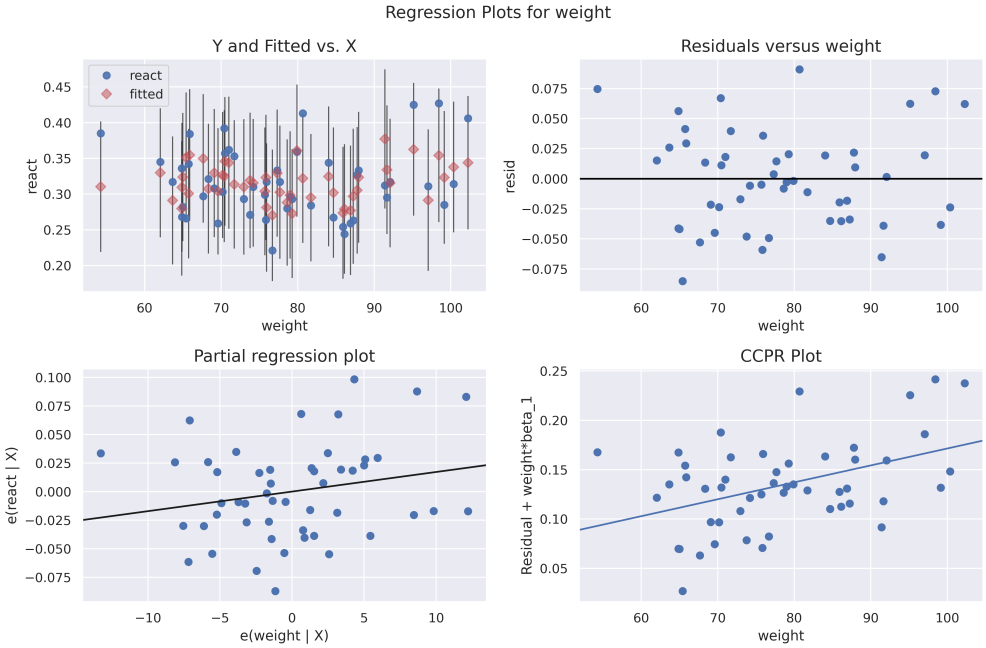
2.3 Vlivná pozorování

Pro detekci vlivných pozorování použijeme Cookovu vzdálenost, která měří, jak moc se predikované hodnoty $\hat{\mathbf{y}}$ změní při vynechání i-tého pozorování, kde

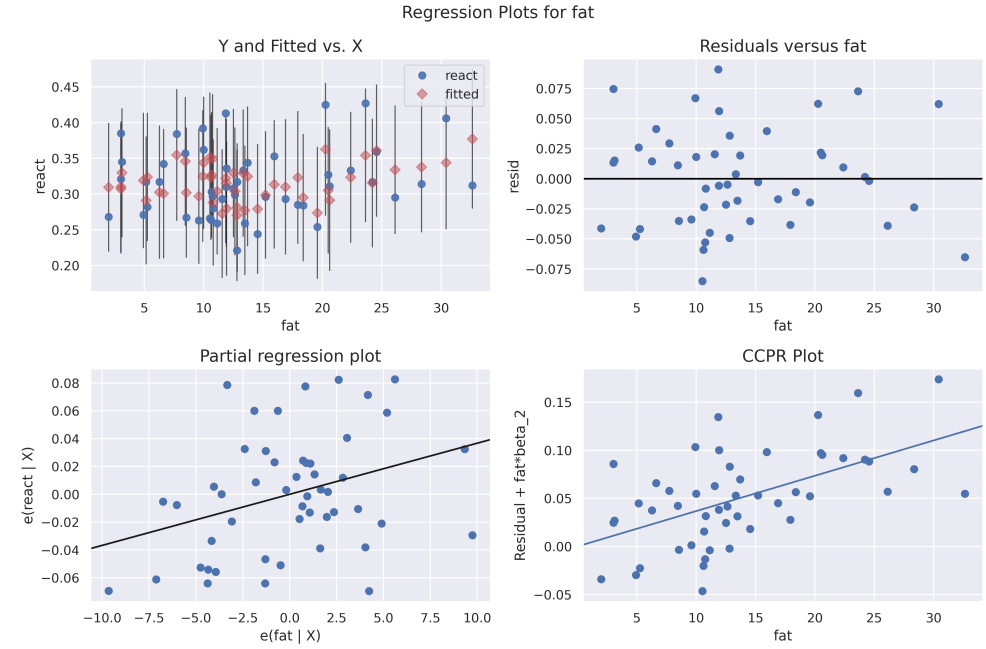
$$\hat{\mathbf{y}} = \mathcal{X}\hat{\beta} = \mathcal{X}\left(\mathcal{X}^T\mathcal{X}\right)^{-1}\mathcal{X}^T\mathbf{y} = \mathcal{H}\mathbf{y}.$$

Pro výpočet Cookovy vzdálenosti použijeme následující funkci:

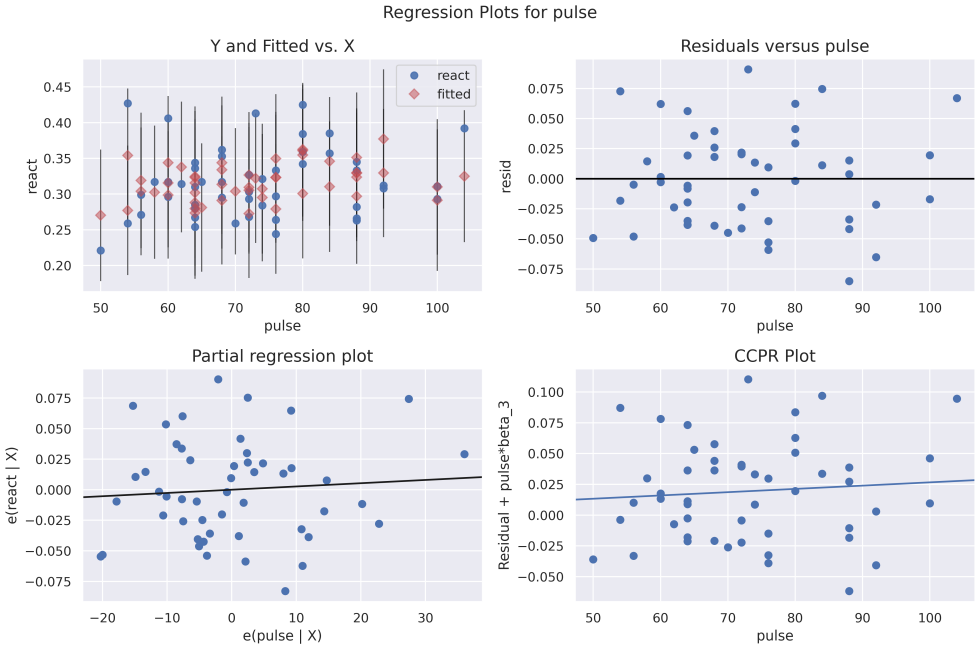
```
influence = model.get_influence()
D = influence.cooks_distance
print(D[0])
```



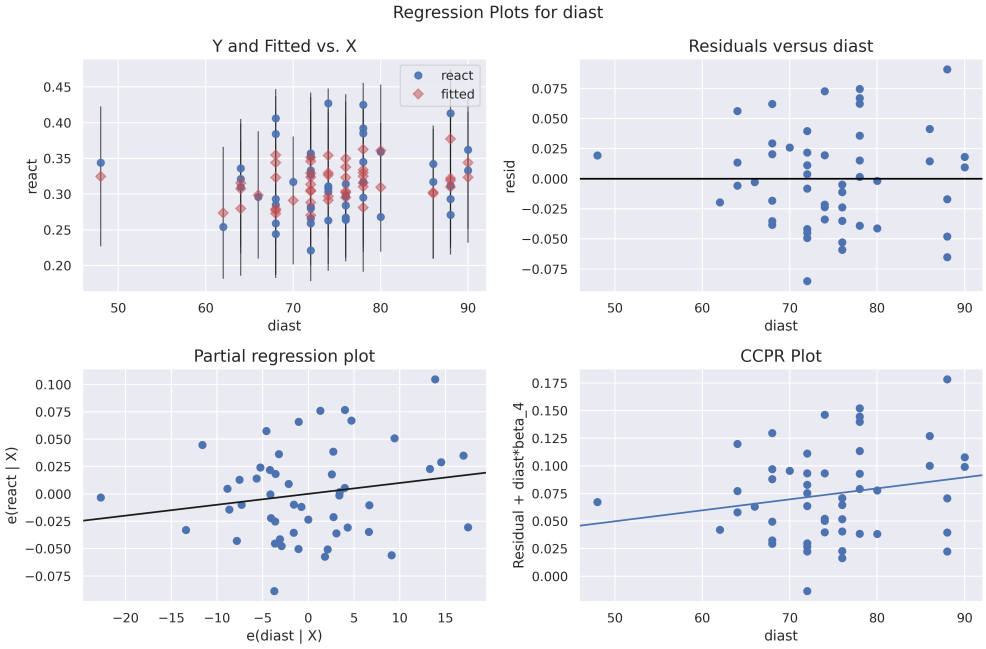
Obrázek 2: Grafy pro vysvětlující proměnnou *weight*



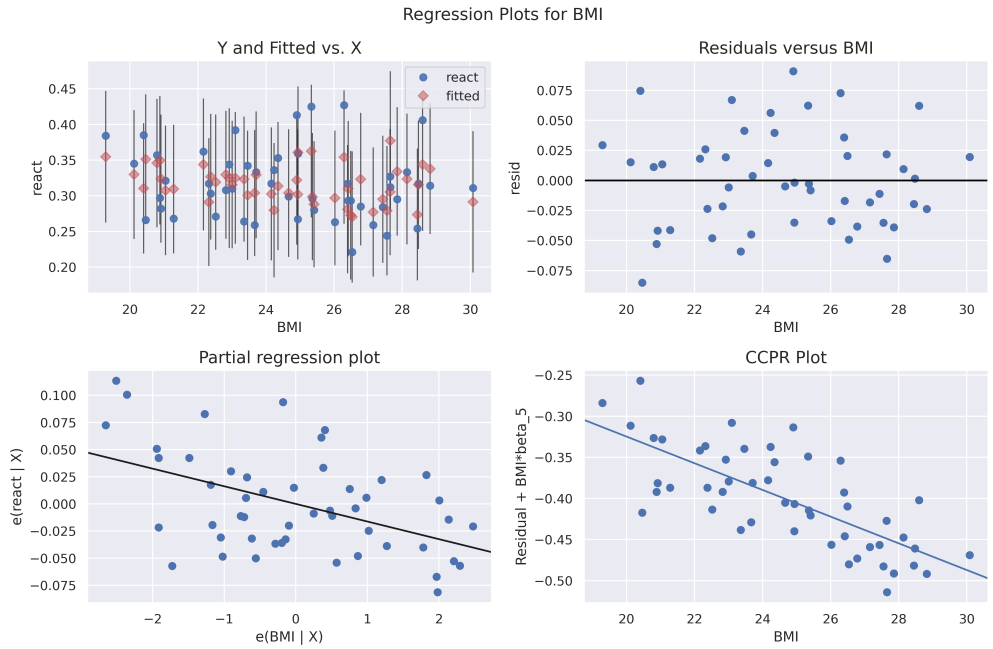
Obrázek 3: Grafy pro vysvětlující proměnnou fat



Obrázek 4: Grafy pro vysvětlující proměnnou *pulse*



Obrázek 5: Grafy pro vysvětlující proměnnou *diast*



Obrázek 6: Grafy pro vysvětlující proměnnou BMI

a dostáváme:

```
[2.48227318e-04, 2.43610269e-03, 8.44795442e-03, 6.88580840e-03,
1.48230147e-02, 8.40179388e-02, 1.88039687e-02, 2.16612368e-03,
7.69887445e-02, 7.56914620e-02, 1.12984603e-02, 1.36373046e-02,
9.02223564e-02, 7.87937053e-02, 2.00283251e-01, 1.91428460e-02,
4.00885847e-03, 2.15307680e-05, 1.35910191e-03, 4.49865928e-03,
1.22791935e-01, 9.26415376e-02, 2.32955071e-02, 8.29259559e-04,
9.61178067e-03, 5.43341625e-03, 3.19260172e-02, 7.95001957e-02,
1.58459640e-02, 6.79628435e-02, 1.02256918e-02, 1.19812792e-03,
3.79823178e-03, 4.10485917e-02, 8.50102540e-05, 5.64825866e-05,
7.90967946e-03, 1.42156410e-02, 5.08029514e-05, 3.89363916e-03,
2.39819299e-03, 2.21460577e-03, 2.55729931e-02, 1.43751399e-02,
1.50012585e-02, 2.89431679e-02, 1.73699534e-04, 3.23546157e-04,
1.66057841e-02, 5.96624178e-03]
```

Doporučené kritérium pro detekci vlivných pozorování uvedené např. zde je $D(i) > \frac{4}{n}$. Použitím tohoto kritéria můžeme získat indexy podezřelých pozorování jako:

```
np.argwhere(D[0] > 4 / np.size(D[0]))
```

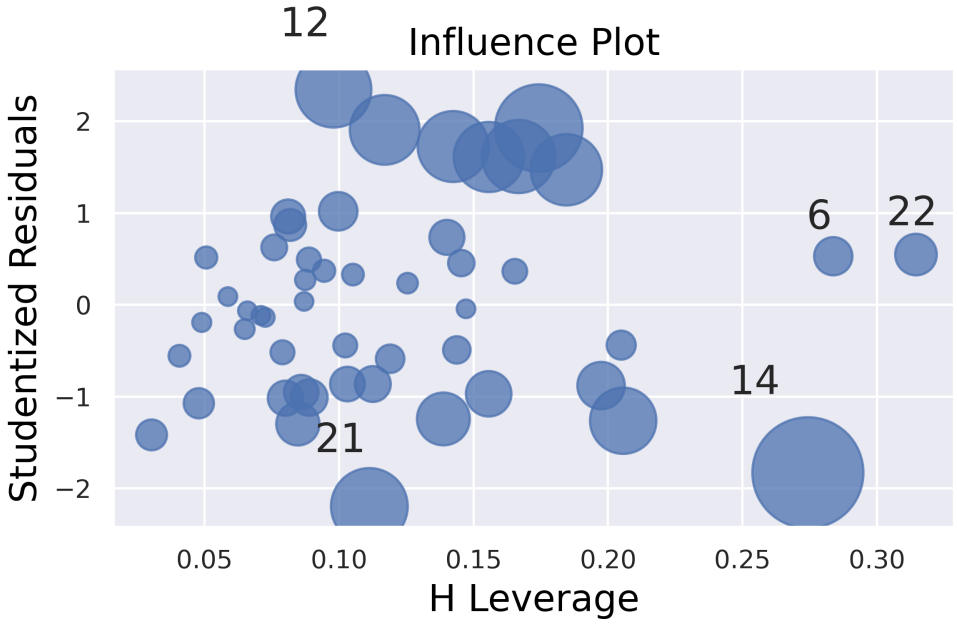
a dostáváme:

```
array([[ 5],
       [12],
       [14],
       [20],
       [21]])
```

Můžeme se také pokusit o grafické znázornění Cookovy vzdálenosti pomocí influenčního grafu, který znázorňuje hodnotu studentizovaných reziduí vzhledem k diagonálním prvkům matice $\mathcal{H}$, kterým se také říká *leverage*. Tento graf můžeme získat následujícím příkazem:

```
sm.graphics.influence_plot(model, criterion="cooks")
```

a výsledkem je graf znázorněný v následujícím obrázku [7], kde velká kolečka odpovídají vlivnějším pozorováním. Můžeme vidět, že v pozorování 12 a 21 mají menší *leverage*, ale velká studentizovaná rezidua. Pozorování 6 a 22 mají velký *leverage*, ale malá studentizovaná rezidua. Pozorování 14 má velký *leverage* a také velké studentizované reziduum a má tedy velký vliv.



Obrázek 7: Influenční graf

2.4 Splnění předpokladů modelu

Předpoklady kladené na model lineární regrese jsou uvedené v sekci 1. Předpoklad o tvaru regresní funkce vypadá v pořádku vzhledem ke grafům reziduí popsaných výše. Konstantnost reziduálního rozptylu také, jelikož reziduální grafy nemají tvar trychtýře. Odlehlá a vlivná pozorování jsme popsali v předcházejících sekcích, kde jsme zjistili, že pozorování 14 má velký *leverage* a také velké studentizované reziduum a má

tedy velký vliv. Z toho důvodu bychom mohli uvažovat o jeho vyřazení. Samozřejmě také předpokládáme nezávislost a normalitu jednotlivých náhodných chyb.

Mohli bychom se nyní pokusit celý model vybudovat znovu např. s menším počtem vysvětlujících proměnných. Jako kandidáti na vyřazení se jeví některé z proměnných, u kterých jsme nezamítli hypotézu o nulovosti regresních parametrů, konkrétně: proměnné *weight*, *pulse* a *diast*. Také bychom mohli např. vyřadit vlivné pozorování 14.

3 Strojové učení

Metody strojového učení jsou zaměřené spíše na vybudování prediktivních modelů, které můžeme použít na nová, dosud neviděná, data. Interpretace modelů strojového učení je proto většinou velice komplikovaná. V případě neuronových sítí se v dnešní době začíná výzkum soustředit také na to, abychom byli schopni interpretovat, co se uvnitř takového modelu děje. Lineární regrese je koneckonců součástí standardních tzv. fully connected neuronových sítí - tvoří jádro jednoho neuronu, jehož výsledek je pak ještě použit jako vstup do nějaké nelineární aktivační funkce (např. tanh, sigmoid či ReLu).

Modely neuronových sítí tedy neposkytují dodatečnou analýzu a testování jednotlivých odhadnutých koeficientů. Při dostatečném množství trénovacích dat jsou ovšem tyto modely schopné se naučit i velice složité nelineární závislosti.