Kazalo

1 IZBIRA MODELA **1**
1.1 Kakovost napovedi 1
1.2 Metode za ocenjevanje kakovosti napovedi osnovane na podatkih 2
1.2.1 PRESS statistika 2
1.2.2 Matrika **H** in računanje PRESS ostankov 5
1.2.3 Navzkrižno preverjanje 7
1.3 Asimptotske metode ocenjevanja kakovosti napovedi modela 11
1.3.1 Mallow-a C_p-statistika 11
1.3.2 Akaike informacijski kriterij AIC 12
1.4 Sekvenčne metode za izbiro najboljšega modela za napovedovanje 13
1.4.1 Izbira naprej 13
1.4.2 Izbira nazaj 14
1.4.3 Izbira po korakih 15
1.4.4 Problemi pri sekvenčnih metodah 16

2 VAJE **17**
2.1 Napovedovanje porabe goriva 17

1 IZBIRA MODELA

1.1 Kakovost napovedi

Pri izbiri regresijskega modela se pogosto soočamo z dilemo med kompleksnostjo modela in med njegovim prileganjem podatkom. Pogosto želimo imeti model, ki kar se da dobro obrazloži, kako napovedne spremenljivke vplivajo na odzivno spremenljivko. Včasih pa so nam pomembne predvsem modelske napovedi. O kakovosti oziroma sprejemljivosti modela se lahko odločamo tudi na podlagi analize njegovih napovedi. Pravimo, da analiziramo **kakovost napovedi modela** (*model predictive performance*).

Za predstavitev analize kakovosti modelskih napovedi poglejmo model enostavne linearne regresije:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \tag{1}$$

za katerega velja $\varepsilon \sim iid\, N(0, \sigma^2)$. Na podlagi podatkov $x_1, ..., x_n$ in $y_1, ..., y_n$ po metodi najmanjših kvadratov izračunamo b_0 in b_1, oceni za β_0 in β_1. Pri vrednosti napovedne spremenljivke x_* je tako napoved $y_* = \beta_0 + \beta_1 x_*$, ocena napovedi je $\hat{y}_* = b_0 + b_1 x_*$. Zanima nas **pričakovana vrednost kvadrata napake napovedi** (*expected squared prediction errors*):

$$E[(y_* - \hat{y}_*)^2]. \tag{2}$$

Pričakovano vrednost za (2) zapišimo še drugače, pri tem uporabimo zvezo, ki velja za varianco slučajne spremenljivke Z, $Var(Z) = E(Z^2) - E(Z)^2$, kar pomeni, da je $E(Z^2) = Var(Z) + E(Z)^2$:

$$\begin{aligned} E[(y_* - \hat{y}_*)^2] &= Var[y_* - \hat{y}_*] + E[y_* - \hat{y}_*]^2 \\ &= Var(y_*) + Var[\hat{y}_*] + E[y_* - \hat{y}_*]^2 \\ &= \sigma^2 + Var[\hat{y}_*] + E[y_* - \hat{y}_*]^2. \end{aligned} \tag{3}$$

V (3) je prvi člen varianca napak σ^2 in je s strani primerjave napovedi različnih modelov med seboj konstanta. Drugi člen $Var[\hat{y}_*]$ predstavlja varianco napovedi modela, tretji člen pa kvadrat pristranskosti napovedi $E[y_* - \hat{y}_*]^2$ (*square of prediction bias*). Za boljše razumevanje teh dveh členov primerjajmo model enostavne regresije z ničelnim modelom, ki vsebuje samo presečišče:

$$y_i = \beta_0 + \varepsilon_i. \tag{4}$$

V tem primeru za oceno β_0 minimiramo izraz $\sum_{i=1}^{n}(y_i - \beta_0)^2$ in po metodi najmanjših kvadratov dobimo $b_0 = \bar{y}$. Posledično je napoved $\hat{y}_* = \bar{y}$, njena varianca pa

$$Var(\hat{y}_*) = \frac{\sigma^2}{n}. \tag{5}$$

V primeru modela enostavne regresije (1) je varianca napovedi praviloma večja:

$$Var(\hat{y}_*) = \sigma^2 \left(\frac{1}{n} + \frac{(x_* - \bar{x})^2}{S_{xx}} \right). \tag{6}$$

V splošnem velja, da se z večanjem števila ocenjenih parametrov v modelu varianca napovedi veča.

Tretji člen v (3) predstavlja kvadrat pristranskosti napovedi, glede tega imamo dve situaciji:

- če je pravi model enostavne regresije, je $E[(y_* - \hat{y}_*)^2] = 0$. Če v tem primeru privzamemo napačen ničelni model, je člen pristranskosti napovedi lahko različen od nič;
- če je pravi ničelni model, je člen pristranskosti enak 0 za oba modela.

Iz tega sledi, da je člen pristranskosti napovedi za kompleksnejši model vedno manjši ali kvečjemu enak kot za model z manj ocenjenimi parametri. Kot smo videli, je z varianco napovedi ravno obratno, za kompleksnejše modele je večja.

Pri dodajanju nove napovedne spremenljivke v model mora biti povečanje variance napovedi na nek način uravnoteženo z zmanjšanjem pristranskosti napovedi (*trade off between contributions of bias and variance to prediction error*).

1.2 Metode za ocenjevanje kakovosti napovedi osnovane na podatkih

1.2.1 PRESS statistika

Kakovost modela po navadi najprej analiziramo na podlagi ostankov $e_i = y_i - \hat{y}_i$ in vsote njihovih kvadratov $SS_{residuals}$. Vemo, da se $SS_{residuals}$ zmanjša ob vsaki dodani napovedni spremenljivki, tudi če ta nima vpliva na odzivno spremenljivko. Pri ocenjevanju kakovosti napovedovanja z izbranim modelom za točko, ki ni v vzorcu, se lahko zgodi, da je z ostanki ocenjena napaka napovedi podcenjena. Boljšo mero za oceno napake napovedi za posamezno točko dobimo s t. i. **PRESS ostanki**:

$$e_{i,-i} = y_i - \hat{y}_{i,-i}. \tag{7}$$

V (7) je $\hat{y}_{i,-i}$ napoved za y_i na podlagi modela, ki je narejen na vseh podatkih brez i-te točke. Mera za prileganje modela je vsota kvadratov PRESS ostankov, t. i. $PRESS$**-statistika** (*Predictive Residual Error Sum of Squares*):

$$PRESS = \sum_{i=1}^{n} e_{i,-i}^2. \tag{8}$$

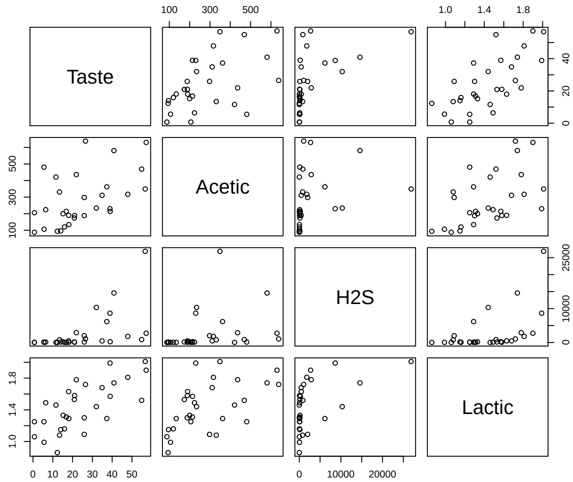
$PRESS$-statistika v nasprotju z $SS_{residuals}$ ne pada nujno z dodajanjem parametrov v model. Na podlagi $PRESS$-statistike lahko primerjamo različne modele narejene za isto odzivno spremenljivko. Model z najmanjšo vrednostjo $PRESS$-statistike je najboljši v smislu kakovosti napovedi.

Primer: `cheese`

V podatkovnem okviru `cheese` v paketu `GLMsData` so podatki iz študije o siru čedar v dolini La Trobe v Victorii v Avstraliji. Subjektivno so ocenjevali okus sira (`Taste`), izmerili pa so koncentracijo ocetne kisline, koncentracijo žveplovodika in koncentracijo mlečne kisline (`Lactic`). V podatkovnem okviru `cheese` sta koncentraciji ocetne kisline in žveplovodika logaritmirani (`Acetic`, `H2S`). Zanimalo jih je, kako je okus odvisen od teh spremenljivk v smislu najboljše možne napovedi za `Taste`.

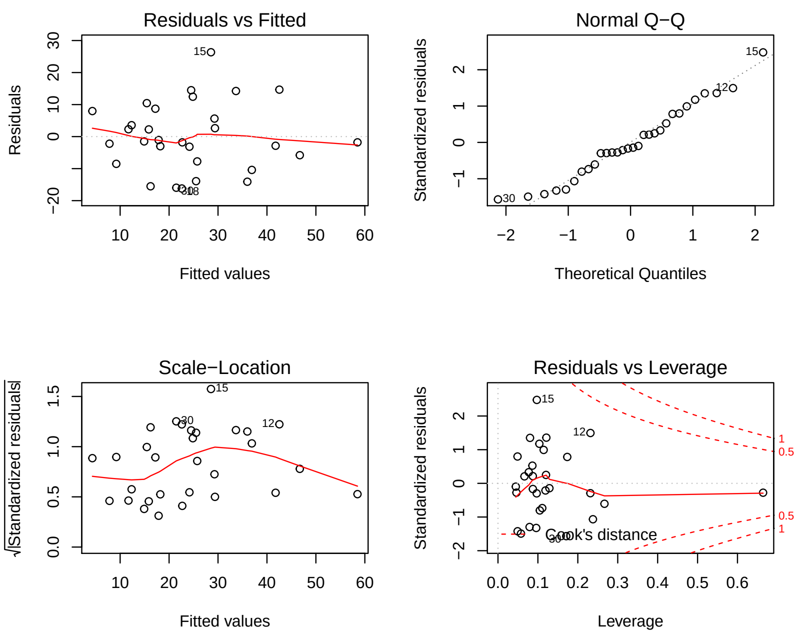
```
> library(GLMsData)
> data(cheese)
> str(cheese)

'data.frame':        30 obs. of  4 variables:
 $ Taste : num  12.3 20.9 39 47.9 5.6 25.9 37.3 21.9 18.1 21 ...
 $ Acetic: int  94 174 214 317 106 298 362 436 134 189 ...
 $ H2S   : int  23 155 230 1801 45 2000 6161 2881 47 65 ...
 $ Lactic: num  0.86 1.53 1.57 1.81 0.99 1.09 1.29 1.78 1.29 1.58 ...
```

```
> pairs(cheese)
```



Slika 1: Matrika razsevnih grafikonov za podatkovni okvir `cheese`

```
> mod.cheese<-lm(Taste ~ Acetic + H2S + Lactic, data=cheese)
> summary(mod.cheese)

Call:
lm(formula = Taste ~ Acetic + H2S + Lactic, data = cheese)

Residuals:
    Min      1Q  Median      3Q     Max
-16.208  -7.266  -1.652   7.385  26.338

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.898e+01  1.127e+01  -1.684   0.1042
Acetic       1.890e-02  1.563e-02   1.210   0.2373
H2S          7.668e-04  4.188e-04   1.831   0.0786 .
Lactic       2.501e+01  9.062e+00   2.760   0.0105 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.19 on 26 degrees of freedom
Multiple R-squared:  0.5754,       Adjusted R-squared:  0.5264
F-statistic: 11.74 on 3 and 26 DF,  p-value: 4.748e-05
```



Slika 2: Ostanki za `mod.cheese`

Primerjajmo ostanke in PRESS ostanke za nekaj točk za `mod.cheese`.

```
> e<-residuals(mod.cheese)[1:5]
> e.press<-numeric()
> for (i in 1:5){
+   mod<-lm(Taste ~ Acetic + H2S + Lactic, data=cheese[-i,])
+   novi<-cheese[i,]
+   e.press[i]<-cheese[i,"Taste"] - predict(mod, newdata=novi)
+ }
> round(data.frame(e,e.press),2)

      e e.press
1  7.97    9.65
2 -1.80   -1.97
3 14.49   15.75
4 14.23   16.20
5 -2.22   -2.52
```

Vidimo, da so v vseh petih primerih PRESS ostanki v absolutnem smislu večji kot navadni ostanki. Poglejmo, zakaj je tako.

1.2.2 Matrika H in računanje PRESS ostankov

Glede na definicijo PRESS ostankov jih izračunamo tako, da prilagodimo za vsako točko en model, torej n modelov. Pokaže se, da to ni potrebno. Izračunamo jih lahko na podlagi vzvodov h_{ii},

$i = 1, ..., n$. Vzvodi so diagonalni elementi matrike $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$. Pokazali smo že, da velja $\hat{y} = \mathbf{H}y$, kar pomeni, da če s h_{ii} pomnožimo y_i, dobimo prilegano vrednost $\hat{y}_i$. Torej vzvod predstavlja neko mero vpliva y_i na $\hat{y}_i$. Po drugi strani je vzvod h_{ii} odvisen samo od napovednih spremenljivk: vektor $\mathbf{x}_i = (1, x_{i1}, x_{i2}, ..., x_{ik})^T$ vsebuje komponente i-te vrstice modelske matrike $\mathbf{X}$ in velja:

$$h_{ii} = \mathbf{x}_i^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i. \tag{9}$$

Za varianco napovedi se pokaže, da je sorazmerna s h_{ii}:

$$\begin{aligned} Var(\hat{y}_i) &= Var(\mathbf{x}_i^T\mathbf{b}) \\ &= \mathbf{x}_i^T Var(\mathbf{b})\mathbf{x}_i = \\ &= \sigma^2\mathbf{x}_i^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_i \\ &= \sigma^2 h_{ii}. \end{aligned} \tag{10}$$

Vzvod ima vrednost med 0 in 1, kar pomeni, da je varianca napovedi vedno manjša od variance napak σ^2. Za enostavno linearno regresijo že vemo, da varianco napovedi pri x_i izrazimo

$$Var(\hat{y}(x_i)) = \sigma^2\left(\frac{1}{n} + \frac{(x_i - \bar{x})^2}{S_{xx}}\right), \tag{11}$$

kar pomeni, da je

$$h_{ii} = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{S_{xx}}. \tag{12}$$

Z uporabo algebre lahko pokažemo, da se PRESS ostanke izrazi z ostanki in vzvodi danega modela, torej ni potrebno oceniti n modelov:

$$e_{i,-i} = \frac{e_i}{1 - h_{ii}}. \tag{13}$$

PRESS ostanki predstavljajo povečane navadne ostanke modela, to povečanje je odvisno od tega, kako vplivna je posamezna točka v procesu ocenjevanja parametrov modela.

Zaradi opisanih lastnosti PRESS ostankov lahko $PRESS$-statistiko uporabimo za izbiro med kandidati za ustrezni model. Za vsak model izračunamo $PRESS$-statistiko in izberemo model z njeno najmanjšo vrednostjo.

Izračunajmo $PRESS$-statistiko za `mod.cheese`.

```
> h<-hatvalues(mod.cheese)
> press.ost<-residuals(mod.cheese)/(1-h)
> PRESS<-sum(press.ost^2)
> PRESS
```

```
[1] 4208.338
```

Izračunajmo $PRESS$-statistiko za vse možne modele na podlagi treh napovednih spremenljivk (brez interakcij):

```
> PRESS<-numeric()
> nap.sprem <- names(cheese)
> nap.sprem <- nap.sprem[! nap.sprem %in% "Taste"]
> n <- length(nap.sprem)
> # za vse možne kombinacije
> id <- unlist(lapply(1:n,function(i) combn(1:n,i,simplify=FALSE)), recursive=FALSE)
> formule <- sapply(id, function(i) paste("Taste~", paste(nap.sprem[i], collapse="+")))
> formule
```

```
[1] "Taste~ Acetic"            "Taste~ H2S"
[3] "Taste~ Lactic"            "Taste~ Acetic+H2S"
[5] "Taste~ Acetic+Lactic"     "Taste~ H2S+Lactic"
[7] "Taste~ Acetic+H2S+Lactic"
```

```
> for (i in (1:length(formule))){
+   mod<-lm(formule[i], data=cheese)
+   h<-lm.influence(mod)$hat
+   press.ost<-residuals(mod)/(1-h)
+   PRESS[i]<-sum(press.ost^2)
+  }
> data.frame(formule, PRESS)
```

```
                   formule    PRESS
1            Taste~ Acetic 6547.165
2               Taste~ H2S 5927.097
3            Taste~ Lactic 4375.643
4        Taste~ Acetic+H2S 5159.898
5     Taste~ Acetic+Lactic 4564.764
6        Taste~ H2S+Lactic 4101.883
7 Taste~ Acetic+H2S+Lactic 4208.338
```

Na podlagi $PRESS$-statistike izberemo model, ki kot napovedni spremenljivke vsebuje `Lactic` in `H2S`.

1.2.3 Navzkrižno preverjanje

Izbira ustreznega modela na podlagi $PRESS$-statistike predstavlja najbolj osnovni način **navzkrižnega preverjanja** (*cross validation*), ki ga pogosto imenujemo tudi **navzkrižno preverjanje brez ene enote** (*leave one out cross validation*). V splošnem pri osnovnem navzkrižnem preverjanju razdelimo enote iz vzorca na dva dela: **učni vzorec** (I_{ucni}), ki ga uporabimo za oceno parametrov modela (*training sample*) in **testni vzorec** (I_{test}), ki ga uporabimo za ugotavljanje kakovosti napovedi modela (*validation sample*). Izbira enot v posamezni vzorec mora biti slučajna.

Postopek osnovnega načina navzkrižnega preverjanja bomo predstavili na primeru `cheese`. Najprej enote razdelimo v učni in v testni podvzorec. To lahko naredimo na različne načine. Za primer izberimo enako velika podvzorca.

```
> n<-dim(cheese)[1]
> n.u<-n.t<-n/2
```

```
> # naredimo vektor z vrednostmi TRUE in FALSE, vsaka vrednost po 15 krat
> (ind<-rep(c(TRUE, FALSE), each=n.u))

 [1]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[13]  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[25] FALSE FALSE FALSE FALSE FALSE FALSE

> # slučajno razporedimo vrednosti
> set.seed(12345) # zaradi ponovljivosti
> (ind<-sample(ind))

 [1]  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE
[13]  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[25]  TRUE FALSE FALSE  TRUE FALSE FALSE

> cheese.ucni<-cheese[ind,]
> cheese.test<-cheese[!ind,]
```

Parametre modela $\mathbf{b}_{ucni}$ ocenimo na podlagi učnega vzorca I_{ucni}, napovedi za vse kandidate za model izračunamo na podlagi testnega vzorca I_{test}.

```
> # izračun napovedi samo za en izbrani model
> mod.ucni<-lm(Taste~H2S+Lactic, data=cheese.ucni)
> y.nap<-predict(mod.ucni, cheese.test)
```

Vsota kvadratov napak napovedi na testnem vzorcu predstavlja t. i. **kriterij navzkrižnega preverjanja** CVC:

$$CVC = \sum_{i \in I_{test}} (y_i - \hat{y}_i)^2 = \sum_{i \in I_{test}} e_i^2 \tag{14}$$

Na podlagi CVC izračunamo t. i. **celotno napako napovedi** oziroma **koren povprečja kvadratov napak napovedi** ($RMSE$, *root mean square prediction error*):

$$RMSE = \sqrt{\frac{1}{n_{test}} \sum_{i \in I_{test}} e_i^2} = \sqrt{\frac{CVC}{n_{test}}}. \tag{15}$$

```
> (CVC<-sum((cheese.test$Taste-y.nap)^2))

[1] 2232.068

> (RMSE<-sqrt(CVC/n.t))

[1] 12.19855
```

Postopek navzkrižnega preverjanja ponovimo za vse kandidate za model. Izberemo model z najmanjšo vrednostjo CVC oziroma $RMSE$, nato za izbrani model izračunamo ocene parametrov na podlagi vseh podatkov.

Osnovni način navzkrižnega preverjanja se izkaže kot neprimerna metoda, če osnovni vzorec ni dovolj velik. Druga težava je v tem, da razdelitev na vzorce vpliva na izid. Za ilustracijo tega vpliva naredimo pet različnih razporeditev enot v učni in testni vzorec na podlagi podatkovnega okvira `cheese` in izračunajmo CVC in $RMSE$:

```
> tabela<-data.frame(formule)
> for (j in 1:5) {
+   izbor<-rep(c(TRUE, FALSE), each=n.u)
+   set.seed(j*10)
+   izbor<-sample(izbor)
+   cheese.ucni<-cheese[izbor,]
+   cheese.test<-cheese[!izbor,]
+   CVC<-numeric()
+   for (i in (1:length(formule))){
+     mod<-lm(formule[i], data=cheese.ucni)
+     y.nap<-predict(mod, cheese.test)
+     CVC[i]<-sum((cheese.test$Taste-y.nap)^2)
+   }
+   # za primerjavo v nadaljevanju izračunamo tudi RMSE
+  tabela<-data.frame(tabela, round(CVC, 1), round(sqrt(CVC/15),1))
+ }
> names(tabela)<-c("formula", "CVC1", "RMSE1","CVC2", "RMSE2", "CVC3",
+                  "RMSE3", "CVC4", "RMSE4", "CVC5", "RMSE5")
> tabela[, c(1:2,4,6,8,10)]

                   formula   CVC1   CVC2   CVC3   CVC4   CVC5
1            Taste~ Acetic 4164.4 3263.5 2679.4 2832.3 3313.9
2               Taste~ H2S 3978.9 2350.9 4665.3 2453.3 2228.4
3            Taste~ Lactic 2618.3 1803.7 2394.7 2228.2 1290.4
4        Taste~ Acetic+H2S 3234.4 2186.0 3951.6 1795.4 2030.9
5     Taste~ Acetic+Lactic 2617.4 1972.1 2499.3 2105.6 1952.9
6        Taste~ H2S+Lactic 2751.7 1448.7 2725.2 1981.7 1023.7
7 Taste~ Acetic+H2S+Lactic 2877.2 1636.4 2916.5 1917.5 1580.0
```

V zgornji tabeli vidimo, da je izbira ustreznega modela na podlagi osnovnega navzkrižnega preverjanja odvisna od slučajne izbire enot v učni in testni vzorec, najmanjšo vrednost CVC imajo pri različnih delitvah v učni in testni podvzorec različni modeli.

Pomanjkljivosti osnovnega navzkrižnega preverjanja so v veliki meri odpravljene pri t. i. **K-kratnem navzkrižnem preverjanju** (*K-fold cross validation*). V tem primeru na začetku enote razdelimo v K enako velikih vzorcev. Parametre modela ocenjujemo K-krat. Vsakič izmed podatkov izločimo en vzorec (testni vzorec), na ostalih podatkih ($K-1$ vzorcev skupaj) ocenimo parametre modela in nato izračunamo CVC iz napovedi na testnem vzorcu. Na koncu K vrednosti CVC povprečimo za vsak model. Prednost tega načina navzkrižnega preverjanja je v tem, da vsak podatek nastopa $K-1$-krat v procesu ocenjevanja parametrov modela in natanko enkrat v testnem podvzorcu.

Za K-kratno navzkrižno preverjanje lahko uporabimo funkcijo `cvFit` v paketu `cvTools`. Za vsak model funkcija `cvFit` izračuna celotno napako napovedi $RMSE$ kot povprečje K-tih $RMSE$ izračunanih na testnih vzorcih velikosti n/K.

```
> library(cvTools)
> cv<-numeric()
```

```
> for (i in (1:length(formule))){
+   mod<-lm(formule[i], data=cheese)
+   mod.cv<-cvFit(mod, data=cheese, y=cheese$Taste, K=5, seed=7)
+   cv[i]<-mod.cv$cv
+   }
> data.frame(formule, cv=round(cv,1))

                  formule   cv
1            Taste~ Acetic 14.6
2               Taste~ H2S 14.6
3            Taste~ Lactic 11.8
4        Taste~ Acetic+H2S 13.3
5     Taste~ Acetic+Lactic 11.7
6        Taste~ H2S+Lactic 11.7
7 Taste~ Acetic+H2S+Lactic 11.8
```

Za primerjavo so v spodnji tabeli $RMSE$ za osnovni način navzkrižnega preverjanja.

```
> tabela[, c(1,3,5,7,9,11)]

                  formula RMSE1 RMSE2 RMSE3 RMSE4 RMSE5
1            Taste~ Acetic  16.7  14.8  13.4  13.7  14.9
2               Taste~ H2S  16.3  12.5  17.6  12.8  12.2
3            Taste~ Lactic  13.2  11.0  12.6  12.2   9.3
4        Taste~ Acetic+H2S  14.7  12.1  16.2  10.9  11.6
5     Taste~ Acetic+Lactic  13.2  11.5  12.9  11.8  11.4
6        Taste~ H2S+Lactic  13.5   9.8  13.5  11.5   8.3
7 Taste~ Acetic+H2S+Lactic  13.8  10.4  13.9  11.3  10.3
```

1.3 Asimptotske metode ocenjevanja kakovosti napovedi modela

Asimptotske metode ocenjevanja kakovosti modela lahko uporabimo, kadar je n dovolj velik. Ideja teh metod je, da ocenimo **povprečje kvadratov napak napovedi** (MSE) na podlagi vsote kvadriranega odklona ostankov modela, velikosti vzorca n in števila regresorjev v modelu k. Podatkov pri tem ne delimo na učni in testni podvzorec.

1.3.1 Mallow-a C_p-statistika

Mallow-a C_p-statistika je osnovana na ideji, da želimo imeti model za katerega je ocena izraza (16) minimalna.

$$\sum_{i=1}^{n} \frac{MSE(\hat{y}(x_i))}{\sigma^2} = \sum_{i=1}^{n} \frac{E[(y(x_i) - \hat{y}(x_i))^2]}{\sigma^2}. \tag{16}$$

Če pričakovano vrednost kvadrata v števcu (16) razvijemo, kot smo to naredili v (3), se $MSE(\hat{y}(x_i))$ izrazi:

$$MSE(\hat{y}(x_i)) = Var[\hat{y}(x_i)] + E[y(x_i) - \hat{y}(x_i)]^2, \tag{17}$$

in (16) izrazimo kot vsoto dveh členov:

$$\frac{\sum_{i=1}^{n} Var[\hat{y}(x_i)]}{\sigma^2} + \frac{\sum_{i=1}^{n} E[y(x_i) - \hat{y}(x_i)]^2}{\sigma^2}. \tag{18}$$

Za model s p, $p < k$, regresorji lahko pokažemo, da je p nepristranska cenilka za prvi člen v (18). Nepristranska cenilka za drugi člen v (18) je $(n-p)(\hat{\sigma}_p^2 - \sigma^2)$. Torej je cenilka za izraz (16):

$$p + \frac{(n-p)(\hat{\sigma}_p^2 - \sigma^2)}{\sigma^2}. \tag{19}$$

V praksi je potrebno σ^2 oceniti. Nepristranska cenilka za σ^2 je $\hat{\sigma}_k^2$ ocenjena na podlagi polnega modela $p = k$ (model, ki vsebuje vse možne regresorje). Ko v (19) vstavimo oceno $\hat{\sigma}_k^2$, dobimo izraz za C_p-statistiko:

$$C_p = p + \frac{(n-p)(\hat{\sigma}_p^2 - \hat{\sigma}_k^2)}{\hat{\sigma}_k^2} = \frac{SS_{p,residual}}{MS_{k,residual}} - n + 2p, \tag{20}$$

$SS_{p,residual}$ je vsota kvadriranih ostankov modela s p regresorji in $MS_{k,residual}$ je srednji kvadrirani odklon ostankov oziroma ocena variance napak polnega modela s k-regresorji .

Z minimiranjem C_p uravnotežimo prileganje modela (če izpustimo pomemben regresor, bo imel izraz $(\hat{\sigma}^2 - \hat{\sigma}_k^2)$ pozitivno vrednost) in njegovo kompleksnost (število parametrov v modelu p).

Na primeru `cheese` bomo za izračun C_p-statistike uporabili funkcijo `leaps` (*all-subsets regression*) iz paketa `leaps`:

```
> library(leaps)
> cp<-leaps(x=cheese[, 2:4], y=cheese$Taste, names=names(cheese)[2:4], method="Cp")
> cbind(cp$which, Cp=cp$Cp)
```

```
  Acetic H2S Lactic        Cp
1      0   0      1  4.864561
1      0   1      0 15.997309
1      1   0      0 19.105885
2      0   1      1  3.463381
2      1   0      1  5.352588
2      1   1      0  9.616839
3      1   1      1  4.000000
```

Po Cp kriteriju je najboljši model, ki ima regresorja `H2S` in `Lactic`. Tak rezultat so dale tudi ostale metode za oceno kakovosti napovedi modela. V splošnem ni nujno, da različni kriteriji vrnejo kot najboljši isti model.

1.3.2 Akaike informacijski kriterij *AIC*

Akaike informacijski kriterij je v praksi najpogosteje uporabljen kriterij za izbiro najboljšega modela. Ni primeren samo za modele, kjer se parametre modela ocenjuje z metodo najmanjših kvaratov, temveč tudi za posplošene linearne modele, kjer se parametre ocenjuje po metodi največjega verjetja. V primeru normalnega linearnega modela obe metodi vrneta iste rezultate.

Akaike informacijski kriterij temelji na statistiki AIC:

$$AIC = -2ln\hat{L} + 2p. \tag{21}$$

V (21) je p število parametrov v modelu in $log\hat{L}$ je logaritem verjetja normalnega modela ovrednoten z ocenami parametrov modela $\mathbf{b}$:

$$log\hat{L} = -\frac{n}{2}ln(2\pi) - \frac{n}{2}ln(\hat{\sigma}^2) - \frac{1}{2\hat{\sigma}^2}(\mathbf{y} - \mathbf{X}\mathbf{b})^T(\mathbf{y} - \mathbf{X}\mathbf{b}). \tag{22}$$

Tudi AIC upošteva prileganje in kompleksnost modela. Absolutna vrednost AIC nima vsebinskega pomena, zanimiva je relativno glede na AIC vrednosti drugih modelov. Najboljši je model z najmanjšo vrednostjo AIC.

Izraz za AIC v okviru normalnih linearnih modelov poenostavimo, če v (22) $\hat{\sigma}^2$ zamenjamo za $SS_{residual}/n$ in $(\mathbf{y} - \mathbf{X}\mathbf{b})^T(\mathbf{y} - \mathbf{X}\mathbf{b})$ z $SS_{residual}$ ter v izrazu izpustimo konstantne člene, ki niso odvisni od prilagajanja modela:

$$AIC = 2p + n\ ln(SS_{residual}). \tag{23}$$

```
> AIC<-numeric()
> for (i in (1:length(formule))){
+   mod<-lm(formule[i], data=cheese)
+   AIC[i]<-AIC(mod)
+   }
> data.frame(formule, AIC=round(AIC,1))
```

```
                  formule   AIC
1            Taste~ Acetic 248.3
2               Taste~ H2S 246.1
3            Taste~ Lactic 236.9
4        Taste~ Acetic+H2S 241.4
5     Taste~ Acetic+Lactic 237.4
6        Taste~ H2S+Lactic 235.4
7 Taste~ Acetic+H2S+Lactic 235.7
```

Tudi na podlagi AIC kriterija izberemo model `Taste∼H2S+Lactic`.

1.4 Sekvenčne metode za izbiro najboljšega modela za napovedovanje

V vseh do sedaj predstavljenih metodah izbire najboljšega modela smo računali različne statistike na podlagi katerih smo postavili kriterij izbora za vse kandidate za model. Če je število napovednih spremenljivk veliko, postane množica kandidatov za model zelo velika (2^k) in računanje postane računsko zahtevno. V takih primerih je smiselno uporabiti **sekvenčno metodo izbire najboljšega modela**. V splošnem se uporabljajo trije pristopi: **izbira naprej** (*forward selection*), **izbira nazaj** (*backward selection*) in **izbira po korakih** (*stepwise selection*).

1.4.1 Izbira naprej

Pri metodi izbire naprej začnemo z ničelnim modelom, ki vsebuje samo presečišče. V prvem koraku je to t. i. **veljaven model**. Postopamo po naslednjih korakih:

1. Izračunamo kriterijsko statistiko za veljaven model (AIC, Cp, CVC, prilagojen R^2, ...).
2. V veljaven model dodamo en regresor in ponovno izračunamo kriterijsko statistiko, to naredimo za vse regresorje, ki še niso v modelu.
3. Med modeli iz prejšnje točke poiščemo model z najmanjšo vrednostjo kriterijske statistike. Če je ta vrednost manjša od kriterijske statistike veljavnega modela, postane ta model veljaven in se vrnemo k točki 2.
4. Če nima noben model z dodanim enim regresorjem manjše vrednosti kriterijske statistike, postane veljaven model izbrani model.

Na tak način v procesu izbire naredimo $1+k(k+1)/2$ modelov, kar je pri velikem številu napovednih spremenljivk v modelu k precej manj kot 2^k (npr. $k = 20$, $1 + k(k + 1)/2 = 211$, $2^k = 1048576$). Metoda ne zagotavlja, da je izbrani model res najboljši.

Za sekvenčno metodo izbire naprej bomo uporabili funkcijo `stepAIC` iz paketa `MASS`.

```
> library(MASS)
> mod.nul<-lm(Taste~1, data=cheese)
> step<-stepAIC(mod.nul, scope=~H2S+Lactic+Acetic, direction="forward")

Start:  AIC=168.29
Taste ~ 1
```

```
         Df Sum of Sq    RSS    AIC
+ Lactic  1    3800.4 3862.5 149.74
+ H2S     1    2407.2 5255.7 158.98
+ Acetic  1    2018.2 5644.7 161.12
<none>                7662.9 168.29

Step:  AIC=149.74
Taste ~ Lactic

         Df Sum of Sq    RSS    AIC
+ H2S     1    425.63 3436.9 148.23
<none>                3862.5 149.74
+ Acetic  1    189.21 3673.3 150.23

Step:  AIC=148.23
Taste ~ Lactic + H2S

         Df Sum of Sq    RSS    AIC
<none>                3436.9 148.23
+ Acetic  1    183.13 3253.7 148.59
```

Ta metoda vrne za izbrani model `Taste~H2S+Lactic`.

1.4.2 Izbira nazaj

Pri metodi izbire nazaj je v prvem koraku veljaven polni model, ki vsebuje vse regresorje. Postopamo po naslednjih korakih:

1. Izračunamo kriterijsko statistiko za veljaven model (AIC,...).
2. Iz veljavnega modela odstranimo en regresor in ponovno izračunamo kriterijsko statistiko, to naredimo za vse regresorje, ki so v modelu.
3. Med modeli iz prejšnje točke poiščemo model z najmanjšo vrednostjo kriterijske statistike. Če je ta vrednost manjša od kriterijske statistike veljavnega modela, postane ta model veljaven in se vrnemo k točki 2.
4. Če nima noben model z odstranjenim enim regresorjem manjše vrednosti kriterijske statistike, postane veljaven model izbrani model.

```
> mod.polni<-lm(Taste~H2S+Lactic+Acetic, data=cheese)
> step<-stepAIC(mod.polni, direction="backward")

Start:  AIC=148.59
Taste ~ H2S + Lactic + Acetic

         Df Sum of Sq    RSS    AIC
- Acetic  1    183.13 3436.9 148.23
<none>                3253.7 148.59
- H2S     1    419.55 3673.3 150.23
```

```
- Lactic  1    953.20 4206.9 154.30

Step:  AIC=148.23
Taste ~ H2S + Lactic

         Df Sum of Sq    RSS    AIC
<none>                3436.9 148.23
- H2S     1    425.63 3862.5 149.74
- Lactic  1   1818.82 5255.7 158.98
```

Tudi ta metoda vrne za izbrani model `Taste∼H2S+Lactic`.

1.4.3 Izbira po korakih

Pri metodi izbire po korakih začnemo s poljubnim modelom. V prvem koraku je to veljaven model. Postopamo po naslednjih korakih:

1. Izračunamo kriterijsko statistiko za veljaven model (AIC,...).
2. Iz veljavnega modela odstranimo po en regresor in tudi dodamo po en regresor ter za vsak popravljeni model izračunamo kriterijsko statistiko.
3. Med modeli iz prejšnje točke poiščemo model z najmanjšo vrednostjo kriterijske statistike. Če je ta vrednost manjša od kriterijske statistike veljavnega modela, postane ta model veljaven in se vrnemo k točki 2.
4. Če nima noben model z odstranjenim enim regresorjem manjše vrednosti kriterijske statistike, postane veljaven model izbrani model.

Rezultati so odvisni od tega, kateri model izberemo v prvem koraku.

```
> mod.prvi<-lm(Taste~Acetic, data=cheese)
> step<-stepAIC(mod.prvi, scope=~H2S+Lactic+Acetic, direction="both")

Start:  AIC=161.12
Taste ~ Acetic

         Df Sum of Sq    RSS    AIC
+ Lactic  1    1971.4 3673.3 150.23
+ H2S     1    1437.8 4206.9 154.30
<none>                5644.7 161.12
- Acetic  1    2018.2 7662.9 168.29

Step:  AIC=150.23
Taste ~ Acetic + Lactic

         Df Sum of Sq    RSS    AIC
+ H2S     1    419.55 3253.7 148.59
- Acetic  1    189.21 3862.5 149.74
<none>                3673.3 150.23
```

```
- Lactic  1   1971.42 5644.7 161.12

Step:  AIC=148.59
Taste ~ Acetic + Lactic + H2S

         Df Sum of Sq    RSS    AIC
- Acetic  1    183.13 3436.9 148.23
<none>                3253.7 148.59
- H2S     1    419.55 3673.3 150.23
- Lactic  1    953.20 4206.9 154.30

Step:  AIC=148.23
Taste ~ Lactic + H2S

         Df Sum of Sq    RSS    AIC
<none>                3436.9 148.23
+ Acetic  1    183.13 3253.7 148.59
- H2S     1    425.63 3862.5 149.74
- Lactic  1   1818.82 5255.7 158.98
```

1.4.4 Problemi pri sekvenčnih metodah

V zgornjem primeru smo dobili enak rezultat izbire najboljšega modela v vseh treh primerih sekvenčnih metod. V praksi ni vedno tako. Kriterij izbire modela smo osnovali na podlagi minimalne vrednosti AIC. Namesto tega se v praksi še vedno pogosto uporablja F-statistika in F-test, kar je sporno zaradi zaradi večkratnega testiranja domnev. Prisoten je tudi problem pristranskosti ocen parametrov modela, ker se isti podatki uporabljajo za oceno parametrov in za proces izbire najboljšega modela.

Za ocene parametrov velja, da so nepristranske, če je vnaprej izbrani model pravi. V primeru, ko model izberemo na podlagi sekvenčne metode, inferenca na parametrih modela ni več upravičena. Ko uporabljamo sekvenčne metode, želimo odgovoriti na vprašanje, katera množica regresorjev vrne najboljšo napoved. Kakršnakoli nadaljnja inferenca ni veljavna, dobljeni model predstavlja inferenco.

2 VAJE

2.1 Napovedovanje porabe goriva

Zanima nas, kako bi najbolje napovedali porabo goriva na avtocestah v odvisnosti od lastnosti avtomobila: `MPG.highway`, `Weight`, `EngineSize`, `Horsepower`, `Type` in `Origin`. Podatki so v podatkovnem okviru `Cars93` v paketu `MASS`. Uporabite različne pristope za izbiro najboljšega modela za napovedovanje porabe goriva: $PRESS$-statistika, navzkrižno preverjanje, C_p-statistika, AIC, sekvenčne metode).

```
> library(car)
> library(MASS)
```

Najprej spremenite podatke v nam razumljive merske enote.

```
> # spremenimo podatke v nam razumljive merske enote.
> Cars93$Poraba<-235.21/Cars93$MPG.highway  # v l/100 km
> Cars93$Masa<-Cars93$Weight*0.45359/100    # v 100 kg
> Cars93$Prostornina<-Cars93$EngineSize     # v litih
> Cars93$Moc<-Cars93$Horsepower             # v KM
> Cars93$Poreklo<-Cars93$Origin
> Cars93$Tip<-Cars93$Type
> avti <- subset(Cars93, select=c(Poraba, Masa, Prostornina, Moc, Poreklo, Tip))
> rownames(avti)<-Cars93$Make
> str(avti)

'data.frame':        93 obs. of  6 variables:
 $ Poraba     : num  7.59 9.41 9.05 9.05 7.84 ...
 $ Masa       : num  12.3 16.1 15.3 15.4 16.5 ...
 $ Prostornina: num  1.8 3.2 2.8 2.8 3.5 2.2 3.8 5.7 3.8 4.9 ...
 $ Moc        : int  140 200 172 172 208 110 170 180 170 200 ...
 $ Poreklo    : Factor w/ 2 levels "USA","non-USA": 2 2 2 2 2 1 1 1 1 1 ...
 $ Tip        : Factor w/ 6 levels "Compact","Large",..: 4 3 1 3 3 3 2 2 3 2 ...
```

Izbira modela glede na kakovost napovedi

```
> # pripravimo formule za vse možne modele
> nap.sprem<-names(avti)
> (nap.sprem<-nap.sprem[!nap.sprem %in% c("Poraba")])

[1] "Masa"        "Prostornina" "Moc"         "Poreklo"     "Tip"

> n<-length(nap.sprem)
> id<-unlist(lapply(1:n, function(i) combn(1:n, i, simplify=FALSE)), recursive=FALSE)
> formule<-sapply(id, function(i) paste("Poraba~", paste(nap.sprem[i], collapse="+")))
> formule[1:10]
```

```
 [1] "Poraba~ Masa"             "Poraba~ Prostornina"
 [3] "Poraba~ Moc"              "Poraba~ Poreklo"
 [5] "Poraba~ Tip"              "Poraba~ Masa+Prostornina"
 [7] "Poraba~ Masa+Moc"         "Poraba~ Masa+Poreklo"
 [9] "Poraba~ Masa+Tip"         "Poraba~ Prostornina+Moc"
```

```
> (length(formule))
```

```
[1] 31
```

PRESS-statistika

```
> PRESS<-numeric()
> for(i in (1:length(formule))){
+   mod<-lm(formule[i], data=avti)
+   h<-lm.influence(mod)$hat
+   press.ostanki<-residuals(mod)/(1-h)
+   PRESS[i]<-sum(press.ostanki^2)
+ }
> izpis<-data.frame(formule, PRESS)
> ### uredimo po vrednostih PRESS
> izpis1 <- izpis[order(PRESS),]
> izpis1[1:5,]
```

```
                           formule    PRESS
20             Poraba~ Masa+Moc+Tip 45.45614
9                  Poraba~ Masa+Tip 46.01660
27 Poraba~ Masa+Prostornina+Moc+Tip 46.10538
29     Poraba~ Masa+Moc+Poreklo+Tip 46.47284
18     Poraba~ Masa+Prostornina+Tip 46.86436
```

Najboljši model: `Poraba` $\sim$ `Masa+Moc+Tip`

K-kratno navzkrižno preverjanje: `cvTools`

```
> library(cvTools)
> cv<-numeric()
> for (i in (1:length(formule))){
+ mod<-lm(formule[i], data=avti)
+ mod.cv<-cvFit(mod, data=avti, y=avti$Poraba, K=5, seed=77)
+ cv[i]<-mod.cv$cv
+ }
> izpis<-data.frame(formule, cv)
> izpis1<-izpis[order(cv),]; izpis1[1:5,]
```

```
                                   formule        cv
20                       Poraba~ Masa+Moc+Tip 0.7338565
```

```
27          Poraba~ Masa+Prostornina+Moc+Tip 0.7345918
9                           Poraba~ Masa+Tip 0.7358218
18              Poraba~ Masa+Prostornina+Tip 0.7389315
31 Poraba~ Masa+Prostornina+Moc+Poreklo+Tip 0.7436436
```

Najboljši model je isti: `Poraba` $\sim$ `Masa+Moc+Tip`

C_p statistika

C_p statistike z ukazom `leaps` ne moremo izračunati, ker so med napovednimi spremenljivkami tudi opisne spremenljivke.

Akaike informacijski kriterij

```
> AIC<-numeric()
> for(i in (1:length(formule))){
+   mod<-lm(formule[i], data=avti)
+   AIC[i]<-AIC(mod)
+ }
> izpis<-data.frame(formule, AIC)
> izpis1<-izpis[order(AIC),]
> izpis1[1:5,]
```

```
                             formule      AIC
20             Poraba~ Masa+Moc+Tip 199.1887
9                  Poraba~ Masa+Tip 200.3304
29     Poraba~ Masa+Moc+Poreklo+Tip 200.8361
27 Poraba~ Masa+Prostornina+Moc+Tip 201.0815
21         Poraba~ Masa+Poreklo+Tip 202.2335
```

Najboljši model je isti: `Poraba` $\sim$ `Masa+Moc+Tip`

Sekvenčna metoda: izbira naprej

```
> poraba.0<-lm(Poraba~1, data=avti)
> step<-stepAIC(poraba.0, scope=~ Tip + Poreklo + Masa + Prostornina + Moc,
+               direction="forward")
```

```
Start:  AIC=62.26
Poraba ~ 1

              Df Sum of Sq     RSS     AIC
+ Masa         1   121.727  56.057 -43.080
+ Tip          5   116.980  60.804 -27.520
+ Prostornina  1    69.743 108.041  17.942
+ Moc          1    66.383 111.401  20.790
+ Poreklo      1     4.018 173.766  62.135
```

```
<none>                      177.784  62.261

Step:  AIC=-43.08
Poraba ~ Masa

              Df Sum of Sq    RSS     AIC
+ Tip          5   16.5384 39.519 -65.592
+ Prostornina  1    3.3086 52.748 -46.737
<none>                     56.057 -43.080
+ Poreklo      1    0.1458 55.911 -41.322
+ Moc          1    0.0000 56.057 -41.080

Step:  AIC=-65.59
Poraba ~ Masa + Tip

              Df Sum of Sq    RSS     AIC
+ Moc          1   1.31270 38.206 -66.734
<none>                     39.519 -65.592
+ Poreklo      1   0.04114 39.477 -63.689
+ Prostornina  1   0.00234 39.516 -63.598

Step:  AIC=-66.73
Poraba ~ Masa + Tip + Moc

              Df Sum of Sq    RSS     AIC
<none>                     38.206 -66.734
+ Poreklo      1  0.144549 38.061 -65.086
+ Prostornina  1  0.044015 38.162 -64.841
```

Sekvenčna metoda: izbira nazaj

```
> poraba.polni<-lm(Poraba~Tip + Poreklo + Masa + Prostornina + Moc, data=avti)
> step<-stepAIC(poraba.polni, direction="backward")

Start:  AIC=-63.48
Poraba ~ Tip + Poreklo + Masa + Prostornina + Moc

              Df Sum of Sq    RSS     AIC
- Prostornina  1    0.1626 38.061 -65.086
- Poreklo      1    0.2631 38.162 -64.841
<none>                     37.899 -63.485
- Moc          1    1.5779 39.477 -61.691
- Masa         1    4.8261 42.725 -54.337
- Tip          5   14.1646 52.063 -43.953

Step:  AIC=-65.09
Poraba ~ Tip + Poreklo + Masa + Moc
```

```
          Df Sum of Sq    RSS     AIC
- Poreklo  1    0.1445 38.206 -66.734
<none>                 38.061 -65.086
- Moc      1    1.4161 39.477 -63.689
- Masa     1    5.2231 43.284 -55.127
- Tip      5   17.8468 55.908 -39.327

Step:  AIC=-66.73
Poraba ~ Tip + Masa + Moc

       Df Sum of Sq    RSS     AIC
<none>              38.206 -66.734
- Moc   1    1.3127 39.519 -65.592
- Masa  1    5.4433 43.649 -56.347
- Tip   5   17.8510 56.057 -41.080
```

Sekvenčna metoda: izbira po korakih

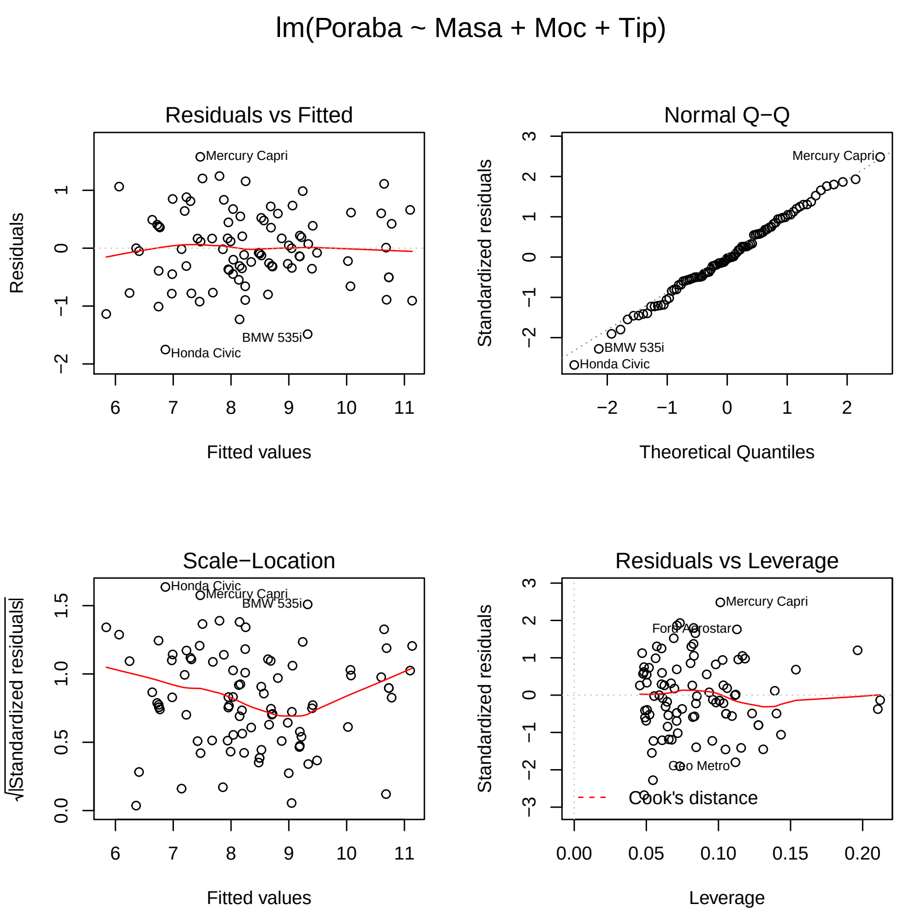
```
> poraba.1<-lm(Poraba~Tip + Masa, data=avti)
> step<-stepAIC(poraba.polni, scope=~ Tip + Poreklo + Masa + Prostornina + Moc,
+               direction="both")

Start:  AIC=-63.48
Poraba ~ Tip + Poreklo + Masa + Prostornina + Moc

              Df Sum of Sq    RSS     AIC
- Prostornina  1    0.1626 38.061 -65.086
- Poreklo      1    0.2631 38.162 -64.841
<none>                     37.899 -63.485
- Moc          1    1.5779 39.477 -61.691
- Masa         1    4.8261 42.725 -54.337
- Tip          5   14.1646 52.063 -43.953

Step:  AIC=-65.09
Poraba ~ Tip + Poreklo + Masa + Moc

              Df Sum of Sq    RSS     AIC
- Poreklo      1    0.1445 38.206 -66.734
<none>                     38.061 -65.086
- Moc          1    1.4161 39.477 -63.689
+ Prostornina  1    0.1626 37.899 -63.485
- Masa         1    5.2231 43.284 -55.127
- Tip          5   17.8468 55.908 -39.327

Step:  AIC=-66.73
Poraba ~ Tip + Masa + Moc

              Df Sum of Sq    RSS     AIC
```

```
<none>                    38.206 -66.734
- Moc           1   1.3127 39.519 -65.592
+ Poreklo       1   0.1445 38.061 -65.086
+ Prostornina   1   0.0440 38.162 -64.841
- Masa          1   5.4433 43.649 -56.347
- Tip           5  17.8510 56.057 -41.080
```

Najboljši model je po vseh kriterijih isti. Analizirajmo ga.

```
> mod.opt<-lm(Poraba~ Masa + Moc + Tip, data=avti)
> vif(mod.opt)
```

```
         GVIF Df GVIF^(1/(2*Df))
Masa 9.315755  1        3.052172
Moc  3.472018  1        1.863335
Tip  6.919416  5        1.213408
```



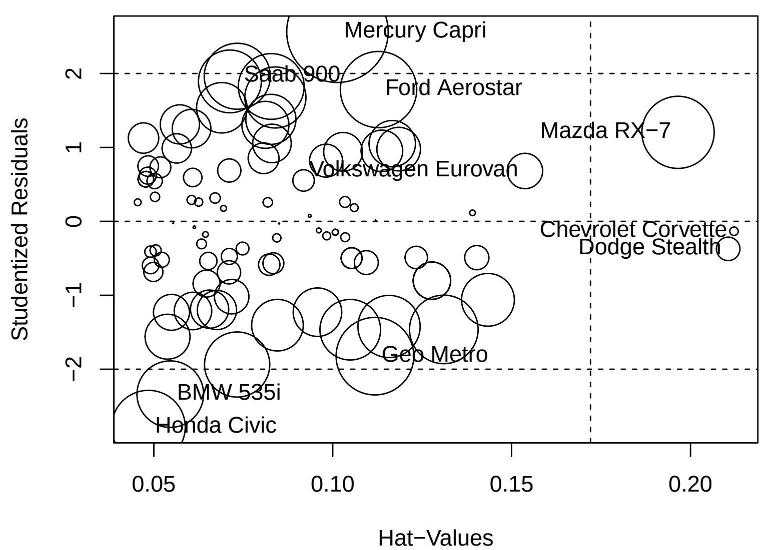
Slika 3: Ostanki za `mod.opt`

Slika ostankov kaže, da je `mod.opt` sprejemljiv, morda nas malo bega leva sličica spodaj, ki ne odraža povsem konstantne varinace. Varianco bi bilo morda smiselno modelirati z `varPower(form= fitted(.))` (teoretična podlaga za to sledi v naslednjem poglavju), kar pa se izkaže kot nepotrebno, saj dobimo enakovreden model ($p = 0.1863$):

```
> library(nlme)
> mod.opt.gls<-gls(Poraba~ Masa + Moc + Tip, weight=varPower(form=~fitted(.)),
+                  method="ML", data=avti)
> anova(mod.opt.gls,mod.opt)

            Model df      AIC      BIC    logLik   Test  L.Ratio p-value
mod.opt.gls     1 10 199.4423 224.7683 -89.72115
mod.opt         2  9 199.1887 221.9821 -90.59433 1 vs 2 1.746356  0.1863
```

Nadaljujemo z analizo posebnih točk za `mod.opt`. Vplivnih točk ni.

```
> influencePlot(mod.opt, id=list(n=4))

                       StudRes        Hat        CookD
BMW 535i            -2.3374414 0.05459716 0.0374728773
Chevrolet Corvette  -0.1337284 0.21214934 0.0006089793
Dodge Stealth       -0.3722991 0.21052481 0.0046674790
Ford Aerostar        1.7825381 0.11280381 0.0492386968
Geo Metro           -1.8234408 0.11179679 0.0509203061
Honda Civic         -2.7828078 0.04849608 0.0457102524
Mazda RX-7           1.2038986 0.19650959 0.0440760801
Mercury Capri        2.5637325 0.10131797 0.0869275353
Saab 900             1.9626050 0.07332433 0.0368607649
Volkswagen Eurovan   0.6809334 0.15372890 0.0105953250
```



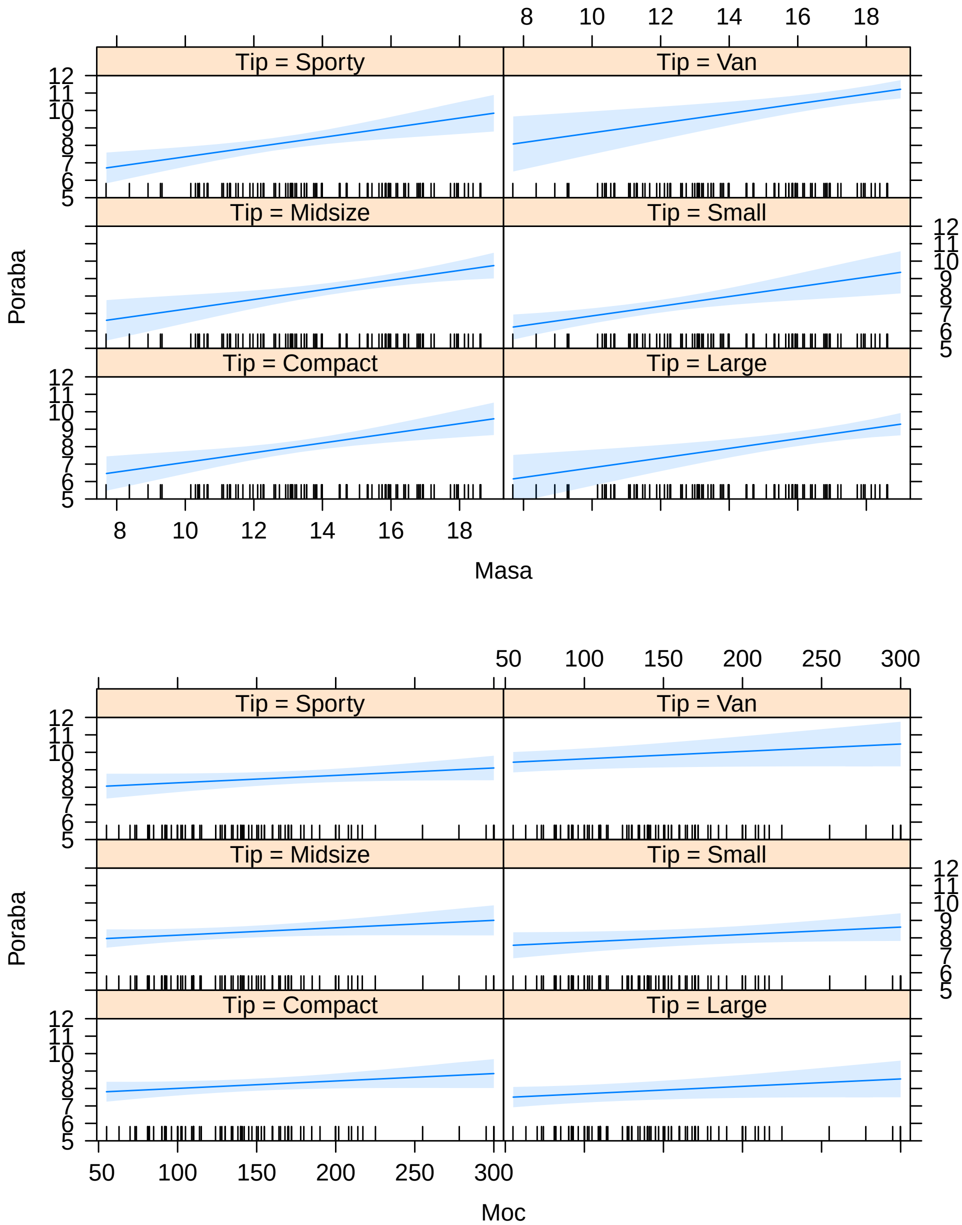
Slika 4: Grafični prikaz posebnih točk `mod.opt`

Vzvodne točke so: Dodge Stealth, Mazda RX-7 in Chevrolet Corvette, to so športni avtomobili z veliko močjo in maso:

```
> avti[rownames(avti)=="Dodge Stealth",c("Masa","Moc","Tip")]

                 Masa Moc    Tip
Dodge Stealth 17.2591 300 Sporty

> avti[rownames(avti)=="Mazda RX-7", c("Masa","Moc","Tip")]

               Masa Moc    Tip
Mazda RX-7 13.13143 255 Sporty
```

```
> avti[rownames(avti)=="Chevrolet Corvette", c("Masa","Moc","Tip")]

                       Masa Moc    Tip
Chevrolet Corvette 15.33134 300 Sporty

> summary(mod.opt)

Call:
lm(formula = Poraba ~ Masa + Moc + Tip, data = avti)

Residuals:
     Min       1Q   Median       3Q      Max
-1.75172 -0.36516 -0.01868  0.44592  1.57849

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.713199   0.864810   4.294 4.64e-05 ***
Masa         0.277464   0.079732   3.480 0.000794 ***
Moc          0.004250   0.002487   1.709 0.091110 .
TipLarge    -0.308124   0.336498  -0.916 0.362426
TipMidsize   0.147298   0.251627   0.585 0.559843
TipSmall    -0.239278   0.275017  -0.870 0.386724
TipSporty    0.246436   0.257293   0.958 0.340879
TipVan       1.618911   0.408672   3.961 0.000154 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6704 on 85 degrees of freedom
Multiple R-squared:  0.7851,       Adjusted R-squared:  0.7674
F-statistic: 44.36 on 7 and 85 DF,  p-value: < 2.2e-16
```

```
> library(effects)
> p1<-plot(Effect(c("Tip","Masa"), mod.opt), x.var="Masa", ci.style="band",
+       main="", ylim=c(5, 12))
> p2<-plot(Effect(c("Tip","Moc"), mod.opt), x.var="Moc", ci.style="band",
+       main="", ylim=c(5, 12))
> library(gridExtra)
> grid.arrange(p1, p2, ncol=1)
```



Slika 5: Napovedi in pripadajoči 95 % intervali zaupanja za povprečne napovedi za `Poraba` na podlagi `mod.opt` pri povprečni vrednosti za `Moc` (zgoraj) in pri povprečni vrednosti za `Masa` (spodaj)

Sklepi:

- v modelu, ki napoveduje porabo goriva, so naslednje napovedne spremenljivke: `Masa`, `Moc` in `Tip`;
- z modelom je pojasnjene 78.5 % variabilnosti porabe;
- napovedi za `Poraba` v odvisnosti od `Masa` in `Moc` pri različni vrednostih `Tip` prikazuje Slika 5.