01RAD - přednáška 12, 26.11.2024

5.2 Metody výběru modelu

1) Vyhodnocení všech možných modelů

- pro T dostupných regresorů tzn. naladit 2^T modelů, pak je porovnat pomocí nějakého kritéria
- náročné pro velká T (například $T = 10$ znamená 1024 modelů)

2) Zpětná eliminace (backward elimination)

- začneme s plným modelem a v každém kroku odstraníme jednu proměnnou
- tu, která nejméně přispívá modelu (měřeno F stat)
- nebo jejíž odstranění znamená největší zlepšení modelu (měřeno AIC)

Algoritmus:

1) naladíme model se všemi proměnnými
2) pro každou proměnnou spočteme částečnou F statistiku (nebo t-statistiku) jako by právě byla přidána do modelu, tzn. za předpokladu, že ostatní proměnné tam už jsou
3) pokud je nějaká F-statistika menší, než kritická hodnota F_{out}, vynecháme z modelu proměnnou s nejnižší hodnotou F
   ($F_{out} = F_{1-\alpha_{out}}(1, n-p)$, kde p je aktuální počet regresorů v modelu, $\alpha_{out} = 0.05, 0.1, \ldots$)
4) opakujeme kroky 2) a 3), dokud všechny částečné F statistiky nejsou větší, než příslušná kritická hodnota F_{out}, tzn. nelze už vyřadit žádnou proměnnou

Poznámka: Místo F lze používat AIC.

3) Dopředná regrese (forward regression)

- začneme pouze s interceptem (nebo nutným minimálním modelem)
- v každém kroku přidáme jednu proměnnou, která má za následek největší zlepšení modelu (největší nárůst F nebo největší pokles AIC)
- tato metoda neumožňuje odstranit proměnnou, která už do modelu byla jednou přidána

Algoritmus:

1) naladíme minimální model
2) pro každou dostupnou proměnnou spočteme F statistiku pro test významnosti jejího přidání do modelu
3) pokud některá z těchto F statistik překračuje kritickou hodnotu F_{in}, přidáme do modelu proměnnou s nejvyšší hodnotou F statistiky
4) opakujeme kroky 2) a 3), dokud všechny F-statistiky pro zbývající proměnné nebudou menší než F_{in} nebo dokud nezbude žádná proměnná na přidání do modelu

Poznámka: I když tento postup zjednodušuje výběr modelu, často bohužel vede na zařazení proměnných, které nemají významný příspěvek, jakmile jsou zařazeny další proměnné.

4) Postupná regrese (stepwise regression)

- kombinace dvou předchozích metod
- v každém kroku algoritmu přidáme jednu proměnnou a poté zkontrolujeme, zda není možné nějakou odebrat
- budeme potřebovat dvě kritické hodnoty F_{in}, F_{out} (pro použití F statistiky)

Algoritmus:

1) naladíme minimální model
2) zjistíme, zda přidání nějaké další proměnné může zlepšit model (F nebo AIC)
   pokud ano, přidáme do modelu proměnnou, která má za následek největší zlepšení modelu
   (největší pokles AIC)
3) v novém modelu zjistíme, zda nelze některou proměnnou vynechat (opět pomocí AIC nebo F)
   pokud ano, vynecháme proměnnou, jejíž vyřazení má za následek největší zlepšení modelu
   (největší pokles AIC)
4) opakujeme kroky 2) a 3) do té doby, až nebude možné přidat ani ubrat žádnou proměnnou

Poznámka 5.3 (Princip marginality)

- pokud jsou v modelu vyšší mocniny nějakého regressoru, měly by tam být obsaženy i všechny jeho nižší mocniny (i když jsou případně nevýznamné)
- pokud je v modelu obsažena interakce dvou regressorů, měly by tam být i oba individuální regresory
- s každou interakcí vyššího řádu by měl model obsahovat i všechny interakce řádu nižšího ($a : b : c \;\rightarrow\; a : b, a : c, b : c$)

Poznámka 5.4

Jakmile nalezneme optimální model, je třeba řádně ověřit adekvátnost.

Příklad: data cement - backward elimination

```
drop1(lm(y ~ x1 + x2 + x3 + x4), test="F")

## Model: y ~ x1 + x2 + x3 + x4
##        Df Sum of Sq    RSS    AIC F value  Pr(>F)
## <none>               47.864 26.944
## x1      1   25.9509 73.815 30.576  4.3375 0.07082 .
## x2      1    2.9725 50.836 25.728  0.4968 0.50090
## x3      1    0.1091 47.973 24.974  0.0182 0.89592
## x4      1    0.2470 48.111 25.011  0.0413 0.84407

drop1(lm(y ~ x1 + x2 + x4), test="F")

## Model: y ~ x1 + x2 + x4
##        Df Sum of Sq    RSS    AIC  F value    Pr(>F)
## <none>                47.97 24.974
## x1      1    820.91 868.88 60.629 154.0076 5.781e-07 ***
## x2      1     26.79  74.76 28.742   5.0259   0.05169 .
## x4      1      9.93  57.90 25.420   1.8633   0.20540

drop1(lm(y ~ x1 + x2), test="F")

## Model: y ~ x1 + x2
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## <none>                57.90 25.420
## x1      1    848.43  906.34 59.178  146.52 2.692e-07 ***
## x2      1   1207.78 1265.69 63.519  208.58 5.029e-08 ***
```

```
min.model <- lm(y ~ 1)
max.model <- lm(y ~ x1 + x2 + x3 + x4)

auto.backward <- step(max.model, direction = "backward",
        scope = list(lower=min.model, upper=max.model))

## Start:  AIC=26.94
## y ~ x1 + x2 + x3 + x4
##
##        Df Sum of Sq    RSS    AIC
## - x3    1    0.1091 47.973 24.974
## - x4    1    0.2470 48.111 25.011
## - x2    1    2.9725 50.836 25.728
## <none>              47.864 26.944
## - x1    1   25.9509 73.815 30.576
##
## Step:  AIC=24.97
## y ~ x1 + x2 + x4
##
##        Df Sum of Sq    RSS    AIC
## <none>               47.97 24.974
## - x4    1      9.93  57.90 25.420
## - x2    1     26.79  74.76 28.742
## - x1    1    820.91 868.88 60.629
```

Příklad: data cement - forward selection

```
add1(min.model, max.model, test = "F")

## Model: y ~ 1
##        Df Sum of Sq     RSS    AIC F value     Pr(>F)
## <none>              2715.76 71.444
## x1      1   1450.08 1265.69 63.519 12.6025 0.0045520 **
## x2      1   1809.43  906.34 59.178 21.9606 0.0006648 ***
## x3      1    776.36 1939.40 69.067  4.4034 0.0597623 .
## x4      1   1831.90  883.87 58.852 22.7985 0.0005762 ***

add1(lm(y ~ x4), max.model, test = "F")

## Model: y ~ x4
##        Df Sum of Sq    RSS    AIC  F value    Pr(>F)
## <none>              883.87 58.852
## x1      1    809.10  74.76 28.742 108.2239 1.105e-06 ***
## x2      1     14.99 868.88 60.629   0.1725    0.6867
## x3      1    708.13 175.74 39.853  40.2946 8.375e-05 ***

add1(lm(y ~ x1 + x4), max.model, test = "F")

## Model: y ~ x1 + x4
##        Df Sum of Sq    RSS    AIC F value  Pr(>F)
## <none>              74.762 28.742
## x2      1    26.789 47.973 24.974  5.0259 0.05169 .
## x3      1    23.926 50.836 25.728  4.2358 0.06969 .
```

```
add1(lm(y ~ x1 + x2 + x4), max.model, test = "F")

## Model: y ~ x1 + x2 + x4
##        Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>              47.973 24.974
## x3      1   0.10909 47.864 26.944  0.0182 0.8959

auto.forward <- step(min.model, direction = "forward",
         scope = list(lower=min.model, upper=max.model))
```

Příklad: data cement - stepwise regression

```
auto.both <- step(min.model, direction = "both",
    scope = list(lower=min.model, upper=max.model),
    k=log(nobs(max.model)))

signif(coef(auto.both), 3)

## (Intercept)          x1          x2
##      52.600       1.470       0.662

## Start:  AIC=72.01
## y ~ 1
##        Df Sum of Sq     RSS    AIC
## + x4    1   1831.90  883.87 59.982
## + x2    1   1809.43  906.34 60.308
## + x1    1   1450.08 1265.69 64.649
## + x3    1    776.36 1939.40 70.197
## <none>              2715.76 72.009

## Step:  AIC=59.98
## y ~ x4
##        Df Sum of Sq     RSS    AIC
## + x1    1    809.10   74.76 30.437
## + x3    1    708.13  175.74 41.547
## <none>               883.87 59.982
## + x2    1     14.99  868.88 62.324
## - x4    1   1831.90 2715.76 72.009
```

```
## Step:  AIC=30.44
## y ~ x4 + x1
##        Df Sum of Sq     RSS    AIC
## + x2    1     26.79   47.97 27.234
## + x3    1     23.93   50.84 27.987
## <none>                74.76 30.437
## - x1    1    809.10  883.87 59.982
## - x4    1   1190.92 1265.69 64.649

## Step:  AIC=27.23
## y ~ x4 + x1 + x2
##        Df Sum of Sq    RSS    AIC
## - x4    1      9.93  57.90 27.115
## <none>               47.97 27.234
## + x3    1      0.11  47.86 29.769
## - x2    1     26.79  74.76 30.437
## - x1    1    820.91 868.88 62.324

## Step:  AIC=27.11
## y ~ x1 + x2
##        Df Sum of Sq     RSS    AIC
## <none>                57.90 27.115
## + x4    1      9.93   47.97 27.234
## + x3    1      9.79   48.11 27.271
## - x1    1    848.43  906.34 60.308
## - x2    1   1207.78 1265.69 64.649
```

6. Kolinearita (multikolinearita)

- budeme předpokládat, že $\boldsymbol{X}$ má plnou hodnost a studovat situaci, kdy je $\boldsymbol{X}^T\boldsymbol{X}$ na pokraji singularity
- v tomto případě mluvíme o špatně podmíněné matici $\boldsymbol{X}$
- např. výpočet $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ může být problematický
- nejsou to však jediné potíže, které může špatná podmíněnost $\boldsymbol{X}$ způsobit
- kolinearita: alespoň jeden ze sloupců matice $\boldsymbol{X}$ je „skoro“ LK ostatních

Poznámka 6.1

Jak to vyjádřit? Jedno přiblížení:

$$\exists \boldsymbol{c} = (c_1, \ldots, c_{m+1})^T, \quad \|\boldsymbol{c}\| = 1, \quad \text{tak, že} \quad \sum_{i=1}^{m+1} c_i \boldsymbol{x}_i \approx \boldsymbol{\varepsilon}, \quad \text{kde } \|\boldsymbol{\varepsilon}\| \text{ je malá}$$

Dá se ukázat, že: kolinearita je přítomna $\iff$ $\boldsymbol{X}^T\boldsymbol{X}$ má alespoň jedno vlastní číslo malé

Zdroje kolinearity:

- způsob sběru dat
- omezení v populaci, ze které byla data vybírána
- špatná specifikace modelu (přeurčení)

Jak kolinearitu rozpoznat?

- první nápad: determinant $\boldsymbol{X}^T\boldsymbol{X}$ blízko 0 (nevhodné!!!)
- např. poměr největšího ku nejmenšímu vlastnímu číslu $\boldsymbol{X}^T\boldsymbol{X}$ lze použít jako charakteristiku
- pokud je matice singulární, alespoň jedno vl. číslo je nulové
- matice na pokraji singularity bude mít jedno vlastní číslo výrazně menší než to největší
- $h(\boldsymbol{X}) = m+1 \quad \Rightarrow \quad \boldsymbol{X}^T\boldsymbol{X}$ je PD a tedy vl. čísla $\lambda_i > 0$, $\forall i \in \widehat{m+1}$
- předpokládejme, že $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{m+1} > 0$ a položme $s_i = \sqrt{\lambda_i}$

Definice 6.1

j-tým indexem podmíněnosti matice $\boldsymbol{X}$ rozumíme veličinu $\eta_j = \frac{s_1}{s_j}$, $j = 1, \ldots, m+1$.

Index podmíněnosti matice $\boldsymbol{X}$ definujeme jako $\kappa(\boldsymbol{X}) = \eta_{m+1} = \frac{s_1}{s_{m+1}}$.

Věta 6.1

Nechť $\boldsymbol{PSQ}^T$ je singulární rozklad matice $\boldsymbol{X}$. Potom pro $j \in \widehat{m+1}$ platí

$$\mathrm{Var}(\widehat{\beta}_j) = \sigma^2 \sum_{i=1}^{m-1} \frac{q_{ji}^2}{s_i^2},$$

kde q_{ij} je i, j-tý prvek matice $\boldsymbol{Q}$ a $\widehat{\boldsymbol{\beta}} = (\widehat{\beta}_1, \ldots, \widehat{\beta}_{m+1})^T$ je OLS odhad parametru $\boldsymbol{\beta}$ v modelu $(**)$.

Důkaz.

Lemma (singulární rozklad matice):
Nechť matice $\boldsymbol{A} \in \mathbb{R}^{n,m}$, $n \geq m$, má hodnost $r \leq m$. Potom existují matice $\boldsymbol{P} \in \mathbb{R}^{n,m}$, $\boldsymbol{S} \in \mathbb{R}^{m,m}$ a $\boldsymbol{Q} \in \mathbb{R}^{m,m}$ takové, že platí

$$\boldsymbol{A} = \boldsymbol{PSQ}^T, \quad \boldsymbol{S} \text{ je diagonální}, \quad \boldsymbol{P}^T\boldsymbol{P} = \boldsymbol{I}_m, \quad \text{a} \quad \boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{QQ}^T = \boldsymbol{I}_m.$$

Důsledky:

- pokud je alespoň jedno s_i malé, rozptyl $\widehat{\beta}$ může být velký
- pokud je jedno s_i malé ve srovnání s ostatními s_j, bude mít i-tý člen sumy velkou váhu a může destabilizovat odhad

Věta 6.2

V modelu $\boldsymbol{Y} \sim (\boldsymbol{X}\beta, \sigma^2 \boldsymbol{I}_n)$ platí $\quad \mathsf{E}\|\widehat{\boldsymbol{Y}}\|^2 = \|\boldsymbol{X}\beta\|^2 + \sigma^2 h(\boldsymbol{X})$.

Má-li $\boldsymbol{X}$ lineárně nezávislé sloupce, platí $\quad \mathsf{E}\|\widehat{\beta}\|^2 = \|\beta\|^2 + \sigma^2 \mathrm{tr}(\boldsymbol{X}^T\boldsymbol{X})^{-1}$.

Důkaz.

Poznámka 6.2

- $\mathsf{E}\|\widehat{\boldsymbol{Y}}\|^2$ závisí pouze na $h(\boldsymbol{X})$, nikoli na tom, jak dobře jsou sloupce $\boldsymbol{X}$ LN
  kolinearita tu tedy nehraje významnou roli
- to neplatí pro $\mathsf{E}\|\widehat{\boldsymbol{\beta}}\|^2$, protože $\mathsf{E}\|\widehat{\boldsymbol{\beta}}\|^2 = \|\boldsymbol{\beta}\|^2 + \sum_{j=0}^{m} \mathsf{Var}\widehat{\beta}_j$ a $\mathsf{Var}\widehat{\beta}_j$ může být velké
- kolinearita ovlivňuje více interpretaci než predikci

Poznámka 6.3

- pozor na změnu $\kappa(\boldsymbol{X})$ při různém škálování různých sloupců matice $\boldsymbol{X}$!
  (např. změna jednotek jedné proměnné)
- proto se mnohdy jednotlivé regresory centrují a škálují, aby měly stejnou délku
  (pokud je délka 1, bude $\boldsymbol{X}_{sc}^T\boldsymbol{X}_{sc}$ matice výběrová korelační matice původních regresorů)
- centrování není vhodné, pokud má intercept vlastní věcnou interpretaci

Příklad: data Trees

```
mod <- lm(Volume ~ Girth + Height)
summary(mod)
## Coeff:      Estimate Std. Error t value Pr(>|t|)
## (Intercept) -57.9877     8.6382  -6.713 2.75e-07 ***
## Girth         4.7082     0.2643  17.816  < 2e-16 ***
## Height        0.3393     0.1302   2.607   0.0145 *
## ---
## Residual standard error: 3.882 on 28 degrees of freedom
## Multiple R-squared:  0.948, Adjusted R-squared:  0.9442
## F-statistic:   255 on 2 and 28 DF,  p-value: < 2.2e-16

vif(mod)
##   Girth  Height
## 1.36921 1.36921

# jednotky XX  palce, stopy
C <- model.matrix(mod)
kappa(C,exact = TRUE)

## 959.377

# jednotky XX cm, metry
Cm <- cbind(rep(1,31), 2.54*Girth, 0.305*Height)
kappa(Cm,exact = TRUE)

## 514.5681
```

```
mod.scale <- lm(scale(Volume) ~ scale(Girth)
                            + scale(Height))
summary(mod.scale)

##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -3.813e-16  4.241e-02   0.000   1.0000
## scale(Girth)   8.988e-01  5.045e-02  17.816   <2e-16 ***
## scale(Height)  1.315e-01  5.045e-02   2.607   0.0145 *
## ---
## Residual standard error: 0.2362 on 28 degrees of freedom
## Multiple R-squared: 0.948, Adjusted R-squared: 0.9442
## F-statistic:   255 on 2 and 28 DF,  p-value: < 2.2e-16

D <- model.matrix(mod.scale)
kappa(D,exact = TRUE)

## 1.777759
```

Regrese standardizovaných veličin (regression in correlation form)

- uvažujme model s interceptem, $h(\boldsymbol{X}) = m+1$ a $\boldsymbol{e} \sim (\boldsymbol{0}, \sigma^2 \boldsymbol{I}_n)$, tzn. pro $i = 1, \dots, n$

$$Y_i = \beta_0 + \sum_{j=1}^{m} x_{ij}\beta_j + e_i = \left(\beta_0 + \sum_{j=1}^{m} \overline{x}_j \beta_j\right) + \sum_{j=1}^{m} (x_{ij} - \overline{x}_j)\beta_j + e_i$$

- podělením zatím neurčeným T_0 dostaneme

$$\frac{Y_i}{T_0} = \frac{\beta_0 + \sum_{j=1}^{m} \overline{x}_j \beta_j}{T_0} + \sum_{j=1}^{m} \frac{x_{ij} - \overline{x}_j}{T_j} \frac{T_j}{T_0} \beta_j + \frac{e_i}{T_0} = \frac{\beta_0^0}{T_0} + \sum_{j=1}^{m} x_{ij}^* \beta_j^* + e_i^* \,,$$

kde

$$\overline{x}_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij}, \quad T_j = \sqrt{\sum_{i=1}^{n} (x_{ij} - \overline{x}_j)^2}, \quad x_{ij}^* = \frac{x_{ij} - \overline{x}_j}{T_j}, \quad \beta_j^* = \frac{T_j}{T_0}\beta_j, \quad \beta_0^0 = \beta_0 + \sum_{j=1}^{m} \overline{x}_j \beta_j \ \text{ a } \ e_i^* = \frac{e_i}{T_0}$$

- ozn. $\boldsymbol{X}^* = \boldsymbol{X}_{sc} = (x_{ij}^*)_{n \times m}$ (sloupce $\boldsymbol{X}^*$ jsou OG vůči $\boldsymbol{1}_n$ a mají délku 1)

- normální rovnice budou

$$\begin{pmatrix} n & \mathbf{0}^T \\ \mathbf{0} & (\boldsymbol{X}^*)^T\boldsymbol{X}^* \end{pmatrix} \begin{pmatrix} \frac{\beta_0^0}{T_0} \\ \beta^* \end{pmatrix} = \begin{pmatrix} \frac{n\overline{y}}{T_0} \\ \frac{(\boldsymbol{X}^*)^T\boldsymbol{y}}{T_0} \end{pmatrix} \quad \Longrightarrow \quad \begin{aligned} \widehat{\beta}_0^0 &= \overline{y} \\ \widehat{\beta}^* &= ((\boldsymbol{X}^*)^T\boldsymbol{X}^*)^{-1}(\boldsymbol{X}^*)^T\frac{\boldsymbol{y}}{T_0} \end{aligned}$$

- snadno se ukáže, že $(\boldsymbol{X}^*)^T\boldsymbol{X}^* = \boldsymbol{R}_{xx}$ je výběrová korelační matice původních regresorů
- pokud zvolíme $T_0 = \sqrt{\sum_{i=1}^n (y_i - \overline{y})^2}$, bude vektor $(\boldsymbol{X}^*)^T\frac{\boldsymbol{y}}{T_0} = r_{xy}$ vektorem výběrových korelačních koeficientů mezi původními $\boldsymbol{x}_j$ a $\boldsymbol{y}$
- tzn. $\widehat{\beta}^* = R_{xx}^{-1} r_{xy}$
- formálně lze postup chápat tak, že jsme od x_{ij}, y_i přešli k $x_{ij}^*, y_i^* = \frac{y_i - \overline{y}}{T_0}$
- model bude:

$$Y_i^* = \sum_{j=1}^m x_{ij}^* \beta_j^* + e_i^*, \qquad \text{kde} \quad \beta_j^* = \frac{T_j}{T_0}\beta_j, \quad e_i^* = \frac{e_i}{T_0}$$

- odhad původních parametrů: $\widehat{\beta}_j = \frac{T_0}{T_j}\widehat{\beta}_j^*, \quad \widehat{\beta}_0 = \overline{y} - \sum_{j=1}^m \widehat{\beta}_j \overline{x}_j$

Poznámka: pro volbu $T_0 = 1$ dostaneme pouze centrované proměnné Y

Poznámka 6.4

- odhady $\widehat{\beta}_j^*$ se někdy nazývají beta weights
- mají důležitou interpretaci: ukazují relativní vliv jednotlivých regresorů na y

  (vztahují se k regresorům vyjádřeným bezrozměrně a ve stejném měřítku)

- vztah mezi SSE^* a R^2 původního modelu (R^2 je stejná i ve stand. modelu)

$$SSE^* = \sum_{i=1}^{n} (\widehat{e}_i^*)^2 = \sum_{i=1}^{n} \left(\frac{\widehat{e}_i}{T_0} \right)^2 = \frac{SSE}{T_0^2} = 1 - \left(1 - \frac{SSE}{T_0^2} \right) = 1 - R^2$$

- víme $\mathrm{Cov}(\widehat{\beta}^*) = \sigma^{*2}((\boldsymbol{X}^*)^T \boldsymbol{X}^*)^{-1} = \sigma^{2*} R_{xx}^{-1}$
- při použití odhadu σ^{2*} ve tvaru $s_n^{*2} = \frac{SSE^*}{n-m-1}$ dostaneme odhad

$$\widehat{\mathrm{Cov}}(\widehat{\beta}^*) = \frac{SSE^*}{n-m-1} R_{xx}^{-1} = \frac{1-R^2}{n-m-1} R_{xx}^{-1} \quad \text{a} \quad \widehat{\mathrm{Var}}(\widehat{\beta}_j^*) = \frac{1-R^2}{n-m-1} r_{jj},$$

kde r_{jj} je j-tý diagonální prvek R_{xx}^{-1}

- dá se ukázat, že $r_{jj} = \frac{1}{1-R_j^2}$, kde R_j^2 je koeficient determinace v modelu pro j-tý sloupec matice $\boldsymbol{X}$ v závislosti na ostatních
- celkem tedy

$$\widehat{\text{Var}}(\widehat{\beta}_j^*) = \frac{1-R^2}{n-m-1}\frac{1}{1-R_j^2}$$

- charakteristika $VIF_j = \dfrac{1}{1-R_j^2}$ se nazývá inflační faktor *(variance inflation factor)*

  - udává, kolikrát se zhorší rozptyl odhadu $\widehat{\beta}_j^*$ v důsledku korelovanosti j-tého regresoru s ostatními regresory
- stejný význam má i pro odhad $\widehat{\beta}_j$, neboť

$$\widehat{\beta}_j = \frac{T_0}{T_j}\widehat{\beta}_j^* \quad \Longrightarrow \quad \widehat{\text{Var}}(\widehat{\beta}_j) = \frac{1-R^2}{n-m-1}\left(\frac{T_0}{T_j}\right)^2\frac{1}{1-R_j^2}$$

Poznámka 6.5

- testovací statistika pro test $H_0: \beta_j = 0$ je

$$T_j = \frac{\widehat{\beta}_j}{\sqrt{\widehat{\text{Var}}\widehat{\beta}_j}} = \frac{\frac{T_0}{T_j}\widehat{\beta}_j^*}{\sqrt{\widehat{\text{Var}}\left(\frac{T_0}{T_j}\widehat{\beta}_j^*\right)}} = \frac{\widehat{\beta}_j^*}{\sqrt{\widehat{\text{Var}}\widehat{\beta}_j^*}} = T_j^*$$

- t-testy jsou tedy stejné v původním i standardizovaném modelu
- velký VIF_j vyžaduje pro zamítnutí H_0 větší hodnotu $|\widehat{\beta}_j^*|$

Poznámka 6.6

- při výpočtu $\widehat{\beta}_j^*$ si v R můžeme pomoci funkcí `scale()`, která provádí centrování a škálování vektoru $\boldsymbol{x}$ pomocí $\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2}$
- při použití funkce `scale()` se v modelu ponechává intercept, aby byly zachovány potřebné stupně volnosti

Zjišťování (detekce) kolinearity

A) pomocí indexu podmíněnosti $\kappa(\boldsymbol{X}^*) = \kappa(\boldsymbol{X}_{sc})$:

$\kappa(\boldsymbol{X}^*) > 100$ silná kolinearita (vypuštění nějakého sloupce)

$\kappa(\boldsymbol{X}^*) > \kappa$, kde $\kappa \in (10, 30)$ $\rightarrow$ použití metody na potlačení kolinearity

B) pomocí hodnoty *VIF*: $VIF_j > 10$ může indikovat problémy

- $VIF_j \geq 10 \quad \leftrightarrow \quad R_j^2 \geq 0.9$
- R_j^2 nemusí být jednoznačně svázán s mírou LZ, *VIF* nemusí být vždy spolehlivý
- má ale jednoduchou stat. interpretaci a snadno se počítá
- většinou představuje spolehlivou náhradu $\kappa(\boldsymbol{X}^*)$

C) další možné příznaky:

- velké hodnoty odhadnutých parametrů $\widehat{\beta}$
- velké směrodatné odchylky $\widehat{\beta}$
- jeden nebo více odhadnutých regresních koeficientů se špatným znaménkem