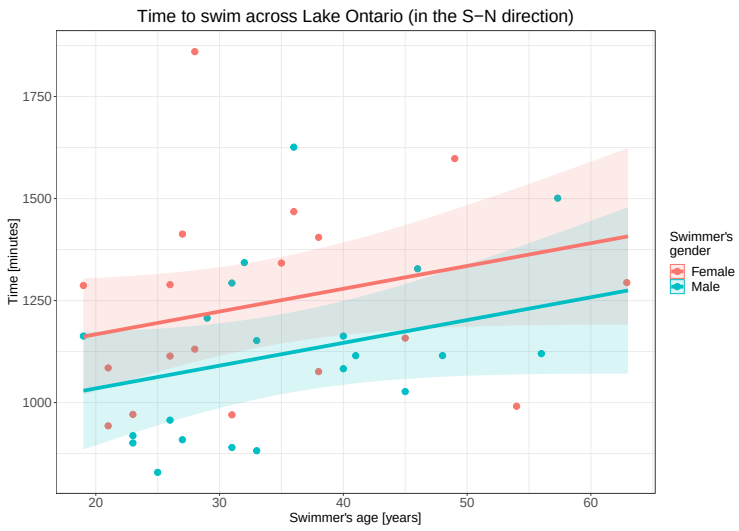
LINEÁRNÍ REGRESE V R

Mgr. ANDREA KRAUS M.Sc., Ph.D. & Mgr. VOJTĚCH ŠINDLÁŘ



Tento text vznikl jako doprovodný ke cvičením kurzu *M5120 Lineární statistické modely I* v rámci projektu *FRMU 1211/2019*.

Obsah

Kapitola 1

Seznámení s R

V této kapitole si připomeneme základní postupy při práci s R, jelikož se v následujících kapitolách základní znalost R předpokládá. Čtenář, který je na práci s R zvyklý, může tuto kapitolu bez újmy vynechat. Čtenář, pro kterého je následující přehled naopak příliš stručný, si jistě vybere z velkého množství mnohdy i volně dostupných knih, výukových materiálů, návodů a diskusí týkajících se R.

Při práci s R je rozumné ukládat si příkazy do skriptů, např. `skript.R`, které pak můžeme upravovat a opakovaně používat. Také objekty, se kterými v R pracujeme, je rozumné si ukládat. V R ukládáme pomocí příkazu `<-`.

```
> a <- 1  # store number 1 in a
> b <- 2  # store number 2 in b
```

Znak `#` označuje v R komentář, tj. to, co následuje po `#`, slouží jako poznámka pro autora nebo čtenáře kódu a R ji nevnímá jako příkaz.

Do `a` a `b` jsme si uložili čísla. Teď můžeme R požádat, aby nám řeklo, o jaká čísla se jedná, nebo aby s nimi pracovalo.

```
> a  # What is in a?

[1] 1

> a + b  # a + b = ?

[1] 3
```

Očekáváme-li, že výsledek operace `a + b` budeme později potřebovat, raději si jej také uložíme.

```
> d <- a + b  # save the result of a + b in d
```

Jelikož se ve statistice hodně využívá lineární algebra a R je statistický software, je speciálně připraveno pro práci s vektory a maticemi. Vektor lze v R vytvořit několika způsoby:

```
> e <- c(1, 2, 3, 4)  # a vector with elements 1, 2, 3, 4
> e

[1] 1 2 3 4

> f <- c(1:4)  # another way of defining the same vector
> f

[1] 1 2 3 4

> g <- rep(1, 4)  # a vector with element 1 repeated 4 times
> g

[1] 1 1 1 1
```

V kódu výše jsme poprvé použili *funkce*, `c()` a `rep()`. Uvnitř závorek jsme zadali požadované hodnoty jejich argumentů. Detaily o dané funkci najdeme v nápovědě, o kterou můžeme v R požádat příkazem `?`. Například `?c` otevře nápovědu k funkci `c()`.

K jednotlivým složkám vektoru přistupujeme pomocí hranatých závorek:

```
> f[3]  # show the third element of f

[1] 3
```

Operace s vektory vykonává R po složkách (nepožádáme-li explicitně o jiný přístup):

```
> # componentwise operations
> f + g

[1] 2 3 4 5

> f + 1

[1] 2 3 4 5

> f * g

[1] 1 2 3 4

> 1/f

[1] 1.0000000 0.5000000 0.3333333 0.2500000
```

Všimněme si, že v kódu výše jsme pro sčítání

$$\begin{pmatrix}1\\1\\1\\1\end{pmatrix}+\begin{pmatrix}1\\2\\3\\4\end{pmatrix}$$

vlastně nepotřebovali definovat vektor $\mathbf{g}$. Stačilo k vektoru $\mathbf{f}$ přičíst obyčejnou 1 a R si z ní samo vytvořilo vektor vhodné délky. Všimněme si také, že jsme v kódu výše zadávali násobení a dělení vektorem, $\mathbf{f}\,\mathbf{g}$ a $1/\mathbf{f}$, což matematicky nedává smysl. Tohoto značení se v R využívá pro násobení a dělení po složkách, jak vidíme i z výstupů výše. Toto značení má za úlohu zjednodušit uživateli zadávání operací, které statistici často potřebují.

Kdybychom, naopak, chtěli „opravdové“ násobení mezi vektory, $\mathbf{f}^\top\mathbf{g}$ nebo $\mathbf{f}\,\mathbf{g}^\top$, použili bychom v R maticové násobení `%*%` (a transpozici `t()`):

```
> t(f) %*% g  # transpose and matrix multiplication

     [,1]
[1,]   10

> f %*% t(g)

     [,1] [,2] [,3] [,4]
[1,]    1    1    1    1
[2,]    2    2    2    2
[3,]    3    3    3    3
[4,]    4    4    4    4
```

Všimněme si, že R ve skutečnosti vnímá vektory jako sloupcové a do řádku je vypisuje jenom pro úsporu místa.

Matici v R vytvoříme pomocí funkce `matrix()`.

```
> A <- matrix(c(1:4), nrow = 2, ncol = 2)
> # 2x2 matrix with elements 1:4 filled in by columns
> A

     [,1] [,2]
[1,]    1    3
[2,]    2    4

> A <- matrix(c(1:4), nrow = 2, ncol = 2, byrow = T)
> # 2x2 matrix with elements 1:4 filled in by rows
> A

     [,1] [,2]
[1,]    1    2
[2,]    3    4
```

K jejím složkám přistoupíme opět pomocí hranatých závorek:

```
> A[1, 1]  # element of A at position (1, 1)

[1] 1
```

Regulární matici můžeme invertovat pomocí příkazu `solve()`:

```
> solve(A)  # inversion of A

     [,1] [,2]
[1,] -2.0  1.0
[2,]  1.5 -0.5
```

Při psaní skriptů se mnohdy hodí také možnosti jakožto programovacího jazyka. Můžeme využít třeba *for cyklů*,

```
> # using a for cycle
> for (i in 1:10) {
+     print(i)
+ }

[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
[1] 6
[1] 7
[1] 8
[1] 9
[1] 10
```

zkonstruovat větvení programu pomocí *if/else*

```
> # using if/else
> if (i > 5) {
+     print("i is greater than 5")
+ } else {
+     print("i is less or equal to 5")
+ }

[1] "i is greater than 5"
```

anebo si vytvořit vlastní funkci pomocí příkazu `function()`.

```
> # defining a function
> mysum <- function(a, b) {
+     return(a + b)
+ }
> mysum(1, 2)

[1] 3
```

Do naší funkce `mysum()` vstupují dva argumenty, `a` a `b`, a funkce vrací jejich součet, `a + b`. Sumační funkce, pochopitelně, v R již existuje (jmenuje se `sum()`) a nebylo ji potřeba vyrábět. Funkci `mysum` proto smažeme:

```
> rm(mysum)   # remove unneeded objects
```

Samotné R obsahuje mnoho funkcí a další jsou dostupné ve specializovaných knihovnách (balíčcích). V následujících kapitolách se seznámíme s postupy, funkcemi a balíčky, které se hodí pro analýzu dat pomocí lineárních modelů.

Kapitola 2

Seznámení s daty

V této kapitole si představíme data, na kterých budeme v následujících kapitolách ilustrovat použití jednotlivých metod.

2.1 Data `swim`

Data `swim` obsahují údaje o plavcích, kteří překonali jezero Ontario mezi kanadským Torontem a ústím řeky Niagara na hranici mezi Kanadou a Spojenými státy, od prvního úspěšného pokusu v roce 1954 do konce roku 2018. Původní data jsou dostupná pod odkazem [2].

Máme-li data uložena v souboru `swim.txt` v podadresáři `Data` adresáře, ve kterém R momentálně pracuje, načteme je do R příkazem

```
> # read data from a .txt file into R object swim
> swim <- read.table("Data/swim.txt", header = TRUE)
```

Na adresář, ve kterém R momentálně pracuje, se můžeme R zeptat příkazem `getwd()`, případně jej můžeme nastavit příkazem `setwd()`, ve kterém do závorky zadáme cestu k požadovanému adresáři.

Příkaz `read.table()` načítá data z textových souborů (`.txt`). Pro data ve fromátu `.csv` lze použít příkaz `read.csv`. Pro načítání dat z jiných formátů existují v R jiné příkazy nebo balíčky. Výše jsme volbou parametru `header = TRUE` řekli, že první řádek souboru obsahuje názvy jednotlivých proměnných, hodnoty pak začínají od druhého řádku. R počítá s formátem, kde jsou hodnoty jednotlivých proměnných ve sloupcích a hodnoty pro jednotlivé jednotky/jedince/pozorování v řádcích. Všechny volby (`options`) příkazu `read.txt` si můžeme prohlédnout pomocí `?read.txt`.

Prvních několik řádků načtených dat prohlédneme příkazem

```
> head(swim)   # view the top of the data

            Name Sex      Age Start.Day Month Year Time.min.
```

```
1  Marilyn Bell   F 16.00000          8   Sep 1954      1255
2  John Jaremey   M 36.00000         23  July 1956      1273
3 Brenda Fisher   F 28.00000         12   Aug 1956      1131
4    Bill Sadlo   M 57.31781         23   Aug 1957      1501
5     Jim Woods   M 41.00000         26   Aug 1957      1115
6     Jim Woods   M 45.00000          2   Sep 1961      1027
  Direction
1        SN
2        SN
3        SN
4        SN
5        SN
6        SN
```

Všimněme si, že jde opravdu o formát, kde každý sloupec odpovídá jedné proměnné a každý řádek jednomu jedinci (v tomto případě plavci).

Pomocí příkazu

```
> str(swim)  # overview of the data

'data.frame': 65 obs. of  8 variables:
 $ Name     : Factor w/ 54 levels "Angela Kondrak",..: 32 24 7 5 22 22 11 16 15 1 ...
 $ Sex      : Factor w/ 2 levels "F","M": 1 2 1 2 2 2 1 1 1 1 ...
 $ Age      : num  16 36 28 57.3 41 ...
 $ Start.Day: int  8 23 12 23 26 2 17 30 16 22 ...
 $ Month    : Factor w/ 3 levels "Aug","July","Sep": 3 2 1 1 1 3 1 1 1 1 ...
 $ Year     : int  1954 1956 1956 1957 1957 1961 1974 1974 1975 1976 ...
 $ Time.min.: num  1255 1273 1131 1501 1115 ...
 $ Direction: Factor w/ 3 levels "NS","NSN","SN": 3 3 3 3 3 3 3 1 3 3 ...
```

zjistíme, jak jednotlivé proměnné vnímá R. Z výstupu vidíme, že data vidí R jako `data.frame` se 65 pozorováními o 8 proměnných. Jedná se v podstatě o matici, u `data.frame` je ale povoleno, aby jednotlivé sloupce byly různých typů. Z výstupu `str(swim)` vidíme, že proměnné Name, Sex, Month a Direction vidí R jako diskrétní (načetlo je jako *faktory* s jistými *úrovněmi*), zatím co zbylé proměnné vidí jako spojité (načetlo je jako čísla).

Základní přehled o datech získáme příkazem

```
> summary(swim)  # descriptive statistics

               Name    Sex          Age          Start.Day
 Vicki Keith    : 4   F:39   Min.   :14.05   Min.   : 1
 Colleen Shields: 3   M:26   1st Qu.:21.00   1st Qu.: 8
 Kim Middleton  : 3          Median :28.00   Median :13
 Jim Woods      : 2          Mean   :31.03   Mean   :14
 John Scott     : 2          3rd Qu.:38.00   3rd Qu.:19
```

```
 Kim Lumsdon     : 2          Max.   :66.57   Max.   :31
 (Other)         :49
  Month          Year        Time.min.    Direction
 Aug :50   Min.   :1954   Min.   : 829   NS : 5
 July: 6   1st Qu.:1979   1st Qu.:1027   NSN: 1
 Sep : 9   Median :1993   Median :1199   SN :59
           Mean   :1992   Mean   :1279
           3rd Qu.:2007   3rd Qu.:1413
           Max.   :2018   Max.   :3370
```

Všimněme si, že R pro každou proměnnou vybere deskriptivní statistiky podle toho, zda ji vnímá jako diskrétní, nebo spojitou.

Na první pohled není vidět žádné problémy s načítáním datové sady. Ty by nastaly zejména v případě, načetlo-li by R numerickou proměnnou jako znak nebo faktor.

2.2 Data `fev`

Datový soubor `fev` obsahuje výběr z údajů o dětech a mladých lidech, kteří se účastnili studie *Childhood Respiratory Desease Study* v roce 1980 ve Spojených Státech, která zkoumala vývoj plicní funkce u dětí. Data i jejich popis lze najít pod odkazem [1]. Klíčová proměnná *Forced Expiratory Volume* (FEV) měří objem vzduchu vydechnutého za první sekundu intenzivního výdechu. Kromě této proměnné data obsahují údaje o věku, výšce, pohlaví a o tom, zda se jedná o kuřáka nebo ne.

Data načteme příkazem

```
> fev <- read.table("Data/fev.txt", header = TRUE)
```

Začátek databáze prohlédneme příkazem

```
> head(fev)

    ID Age   FEV Height    Sex Smoker
1  301   9 1.708   57.0 Female    Non
2  451   8 1.724   67.5 Female    Non
3  501   7 1.720   54.5 Female    Non
4  642   9 1.558   53.0   Male    Non
5  901   9 1.895   57.0   Male    Non
6 1701   8 2.336   61.0 Female    Non
```

Základní přehled o databázi získáme příkazem

```
> summary(fev)

       ID              Age              FEV             Height     
 Min.   :  201   Min.   : 3.000   Min.   :0.791   Min.   :46.00  
 1st Qu.:15811   1st Qu.: 8.000   1st Qu.:1.981   1st Qu.:57.00  
 Median :36071   Median :10.000   Median :2.547   Median :61.50  
 Mean   :37170   Mean   : 9.931   Mean   :2.637   Mean   :61.14  
 3rd Qu.:53639   3rd Qu.:12.000   3rd Qu.:3.119   3rd Qu.:65.50  
 Max.   :90001   Max.   :19.000   Max.   :5.793   Max.   :74.00  
    Sex           Smoker    
 Female:318   Current: 65  
 Male  :336   Non    :589  
```

Ani na těchto datech nejsou zřejmé žádné problémy s načítáním. Nyní si data prohlédneme podrobněji.

Začneme s rozsahem dat. Funkce `dim()` vrátí rozměr dat.

```
> dim(fev)  # dimensions of the data frame

[1] 654   6
```

Jedná se teda o rozsáhlejší data (6 proměnných o 654 jedincích).

Někdy je praktické přistoupit jen k části dat. Konkrétní řádky nebo sloupce vyžádáme následujícím způsobem.

```
> fev[1:10, ]  # first ten rows and all columns of the data

     ID Age   FEV Height    Sex Smoker
1   301   9 1.708   57.0 Female    Non
2   451   8 1.724   67.5 Female    Non
3   501   7 1.720   54.5 Female    Non
4   642   9 1.558   53.0   Male    Non
5   901   9 1.895   57.0   Male    Non
6  1701   8 2.336   61.0 Female    Non
7  1752   6 1.919   58.0 Female    Non
8  1753   6 1.415   56.0 Female    Non
9  1901   8 1.987   58.5 Female    Non
10 1951   9 1.942   60.0 Female    Non

> fev[c(1, 3, 5), ]  # rows 1, 3, 5 and all columns
```

```
   ID Age   FEV Height    Sex Smoker
1 301   9 1.708   57.0 Female    Non
3 501   7 1.720   54.5 Female    Non
5 901   9 1.895   57.0   Male    Non

> fev[1:10, 1:2]  # first ten rows and two columns

     ID Age
1   301   9
2   451   8
3   501   7
4   642   9
5   901   9
6  1701   8
7  1752   6
8  1753   6
9  1901   8
10 1951   9
```

O konkrétní sloupec lze požádat také jeho jménem a následně s ním pracovat jako s vektorem.

```
> range(fev$FEV)  # minimum and maximum

[1] 0.791 5.793

> mean(fev$FEV)  # mean

[1] 2.63678

> sd(fev$FEV)  # standard deviation

[1] 0.8670591
```

Rovněž je možné požádat jenom o část dat zadanou jejími vlastnostmi.

```
> # descriptive statistics for males
> summary(fev[fev$Sex == "Male", ])

       ID             Age              FEV             Height     
 Min.   :  201   Min.   : 3.00   Min.   :0.796   Min.   :47.00  
 1st Qu.:15317   1st Qu.: 8.00   1st Qu.:2.009   1st Qu.:57.00  
 Median :34171   Median :10.00   Median :2.606   Median :62.00  
 Mean   :36233   Mean   :10.01   Mean   :2.812   Mean   :62.03  
```

```
 3rd Qu.:52286   3rd Qu.:12.00   3rd Qu.:3.535   3rd Qu.:67.50
 Max.   :90001   Max.   :19.00   Max.   :5.793   Max.   :74.00
     Sex           Smoker
 Female:  0   Current: 26
 Male  :336   Non    :310
```

```
> # descriptive statistics for height in females
> summary(fev$Height[fev$Sex == "Female"])

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  46.00   57.50   61.00   60.21   63.50   71.00
```

Proměnná ID obsahuje identifikační čísla jedinců v rámci studie, ze které data pocházejí. V našich datech, která jsou podmnožinou původních dat ze studie, máme od každého jedince jen jedno pozorování, jak snadno ověříme třeba sečtením výstupů z funkce `duplicated()`. Tato funkce přiřadí hodnotu `TRUE`, tj. 1, těm prvkům vektoru, které se již vyskytly na předchozích pozicích, a `FALSE`, tj. 0, všem ostatním.

```
> sum(duplicated(fev$ID))

[1] 0
```

Identifikační čísla jedinců dále nebudeme potřebovat, proto se nám bude hodit tuto proměnnou z dat vymazat. Na výstupech výše jsme si možná také uvědomili, že výška je zadaná v palcích. Abychom se v ní lépe orientovali, převedeme ji na centimetry.

```
> # remove the first column of the data
> fev <- fev[, -1]
> # replace height in inches by height in centimeters
> fev$Height <- fev$Height * 2.54
```

Výsledná data uložíme, abychom je příště nemuseli opět upravovat.

```
> # save the modified data for later use
> save(fev, file = "fev.RData")
```

Upravená data posléze do R načteme příkazem `load("fev.RData")`. Pomocí funkce `save()` je možné do souboru typu `.RData` uložit více objektů současně. Ty se pak příkazem `load()` načtou přímo do prostředí R, kde budou figurovat s takovými názvy a vlastnostmi, jaké měly při ukládání. Do souboru typu `.txt` ukládáme data pomocí příkazu `write.table()`.

Kapitola 3

Deskriptivní statistiky

Když začínáme pracovat s nějakou datovou sadou, je rozumné si data nejdříve prohlédnout: zjistit, jaké obsahují proměnné, jakých hodnot tyto proměnné nabývají a jak jsou v datech vzájemně propojeny. Takový předběžný pohled nám umožní najít a odstranit případné chyby v datech, utvořit si představu o vhodném postupu samotné analýzy a předjímat případné problémy. Provést jej můžeme s využitím různých metod *deskriptivní statistiky*. V této fázi tedy nebudeme využívat žádných modelů, jenom vhodně zvolených charakteristik dat. Ty mohou být numerické nebo grafické a jejich výběr závisí od typu proměnné nebo proměnných, na které se chceme zaměřit.

V matematické statistice jsme se zabývali zejmnéna *diskrétními* a *spojitými* náhodnými veličinami, které se mezi sebou liší především počtem hodnot, kterých mohou nabývat (spočetně, nebo nespočetně mnoho). Jelikož i realizace spojitých veličin obvykle měříme na diskrétní stupnici (dané například rozlišením měřící technologie), v aplikované statistice se proměnné spíše dělí na *kvalitativní* (*kategoriální*) a *kvantitativní* (*číselné*). Kategoriální proměnné pak mohou být *nominální* (kategorie nemají uspořádání), anebo *ordinální* (kategorie lze uspořádat). V R kódujeme kategoriální proměnné třídou `factor`, případně `ordered factor` a kvantitativní proměnné třídou `numeric`, případně `integer`. Jsou-li data na vstupu dobře připravena, R často při jejich načítání samo přiřadí proměnným vhodné třídy. Je-li kategoriální proměnná v datech kódována číselně, lze jí v R změnit na `factor` pomocí funkce `factor()`.

Třídy jednotlivých proměnných v konkrétním `data.frame` zjistíme například z prvního sloupce výstupu příkazu

```
> # types of variables in the first column of the output below
> str(fev)

'data.frame': 654 obs. of  5 variables:
 $ Age   : int  9 8 7 9 9 8 6 6 8 9 ...
 $ FEV   : num  1.71 1.72 1.72 1.56 1.9 ...
 $ Height: num  145 171 138 135 145 ...
 $ Sex   : Factor w/ 2 levels "Female","Male": 1 1 1 2 2 1 1 1 1 1 ...
 $ Smoker: Factor w/ 2 levels "Current","Non": 2 2 2 2 2 2 2 2 2 2 ...
```

Proměnná Age je tedy typu `integer`, proměnné FEV a Height jsou třídy `numeric` a proměnné Sex a Smoker jsou třídy `factor`, jak se nám hodí. Na třídu konkrétní proměnné se můžeme

zeptat také příkazem `class()`.

```
> # type of variable Sex in the fev data
> class(fev$Sex)

[1] "factor"
```

Jsou-li proměnné v R načteny se správnými třídami, vrátí nám příkaz `summary()` pro každou z nich vhodné číselné deskriptivní statistiky:

```
> # basic descriptive statistics chosen by the type of variable
> summary(fev)

      Age              FEV             Height          Sex
 Min.   : 3.000   Min.   :0.791   Min.   :116.8   Female:318
 1st Qu.: 8.000   1st Qu.:1.981   1st Qu.:144.8   Male  :336
 Median :10.000   Median :2.547   Median :156.2
 Mean   : 9.931   Mean   :2.637   Mean   :155.3
 3rd Qu.:12.000   3rd Qu.:3.119   3rd Qu.:166.4
 Max.   :19.000   Max.   :5.793   Max.   :188.0
    Smoker
 Current: 65
 Non    :589
```

Nyní se na vhodné číselné i grafické charakteristiky podle typu proměnné podíváme podrobněji.

3.1 Jednotlivé proměnné

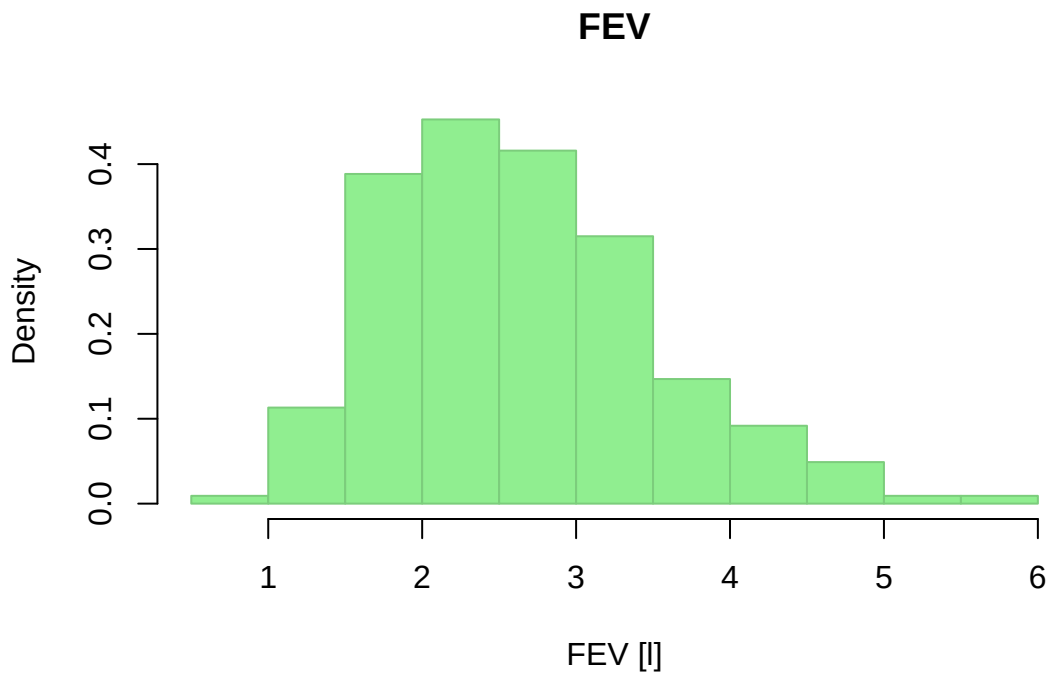
3.1.1 Kvantitativní proměnná

Rozložení kvantitativní proměnné můžeme číselně charakterizovat pomocí *měr polohy* nebo *měr variability*. Mezi oblíbené míry polohy patří průměr, a různé kvantily, zejména medián, maximum, minimum, dolní a horní kvartil. To jsou také charakteristiky na výstupu z funkce `summary()` aplikované na kvantitativní proměnnou. Mezi populární míry charakteristiky patří (mezi)kvartilové rozpětí, rozptyl a směrodatná odchylka. U všech uvedených charakteristik jde, pochopitelně, o výběrovou verzi.

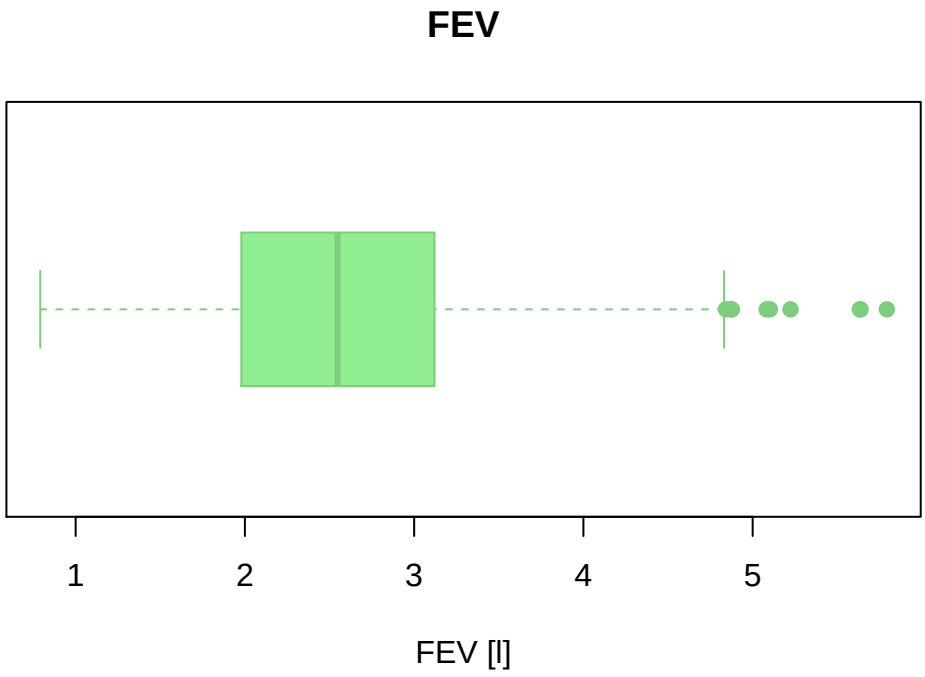
```
> IQR(fev$FEV)   # interquartile range

[1] 1.1375

> var(fev$FEV)   # variance

[1] 0.7517915

> sd(fev$FEV)   # standard deviation

[1] 0.8670591
```

Graficky lze rozložení kvantitativní proměnné znázornit pomocí histogramu nebo pomocí krabicového diagramu.

```
> # histogram
> hist(fev$FEV, freq = FALSE,
+      main = "FEV", xlab = "FEV [l]",
+      col = "lightgreen", border = "palegreen3")
```



```
> # boxplot
> boxplot(fev$FEV, horizontal = TRUE,
+         main = "FEV",  xlab = "FEV [l]",
+         col = "lightgreen", border = "palegreen3", pch = 19)
```



Rozdělení proměnné FEV je symetrické, až na několik vzdálenějších (možná odlehlých) větších hodnot.

3.1.2 Kategoriální proměnná

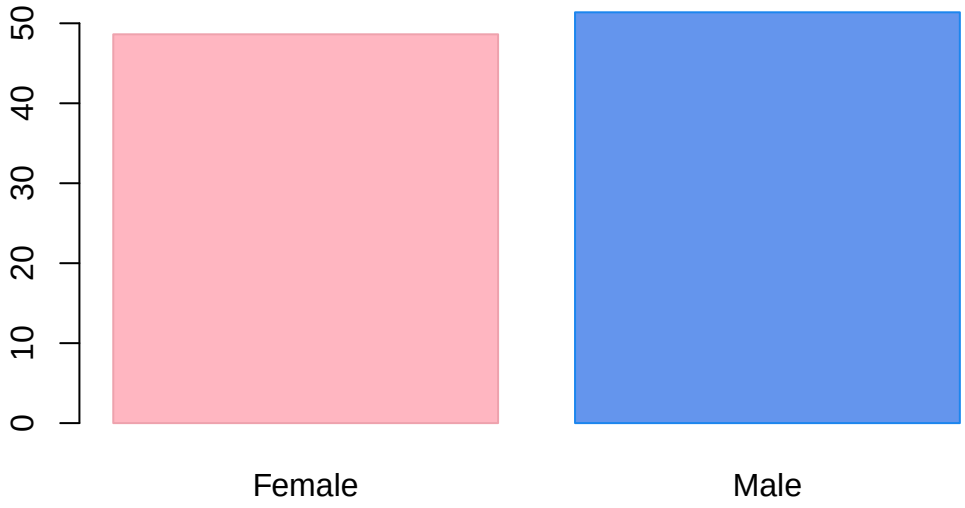
Rozložení kategoriální proměnné lze plně popsat i číselnými charakteristikami, konkrétně procentuálním rozložením dat mezi jednotlivými kategoriemi. Užitečný může být i počet pozorování v jednotlivých kategoriích. Ten také najdeme na výstupu z funkce `summary()` aplikované na kategoriální proměnnou. Pro ordinální kategoriální proměnné je informativní také kumulativní procentuální zastoupení v uspořádaných kategoriích.

```
> # nominal variable: percentages per category
> prop.table(table(fev$Sex))


   Female      Male
0.4862385 0.5137615
```

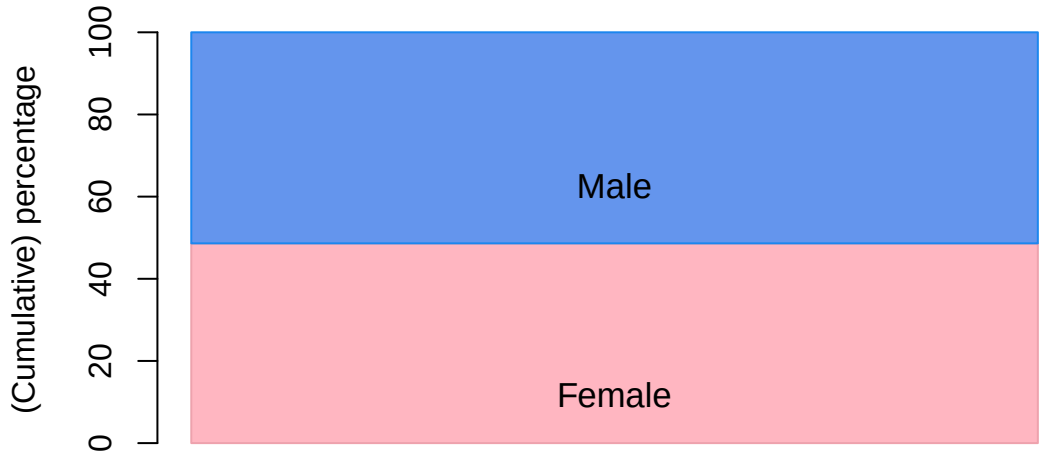
```
> # ordinal variable
> summary(swim$Month)

 Aug July  Sep
  50    6    9

> # first the right order
> swim$Month <- factor(swim$Month,
+                      levels = levels(swim$Month)[c(2, 1, 3)],
+                      ordered = TRUE)
> summary(swim$Month)

July  Aug  Sep
   6   50    9

> prop.table(table(swim$Month)) # percentages per category


      July        Aug        Sep
0.09230769 0.76923077 0.13846154

> cumsum(prop.table(table(swim$Month))) # cumulative percentages

      July        Aug        Sep
0.09230769 0.86153846 1.00000000
```

Pro grafické znázornění rozložení kategoriální proměnné lze využít sloupcový nebo koláčový graf.

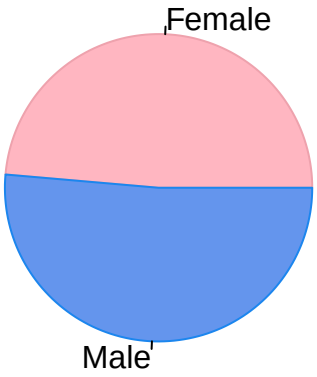
```
> # bar plot for Sex
> barplot(
+   100*prop.table(table(fev$Sex)),
+   col = c("lightpink", "cornflowerblue"),
+   border = c("lightpink2", "dodgerblue2")
+   )
```




```
> # bar plot for Sex
> barplot(
+   100*matrix(prop.table(table(fev$Sex)), nrow = 2, ncol = 1),
+   ylab = "(Cumulative) percentage",
+   col = c("lightpink", "cornflowerblue"),
+   border = c("lightpink2", "dodgerblue2")
+   )
> text(x = 0.7, y = 4, labels = "Female", cex = 1.1, pos = 3)
> text(x = 0.7, y = 55, labels = "Male", cex = 1.1, pos = 3)
```




```
> # pie chart for Sex
> pie(
+   summary(fev$Sex),
+   col = c("lightpink", "cornflowerblue"),
+   border = c("lightpink2", "dodgerblue2")
+   )
```

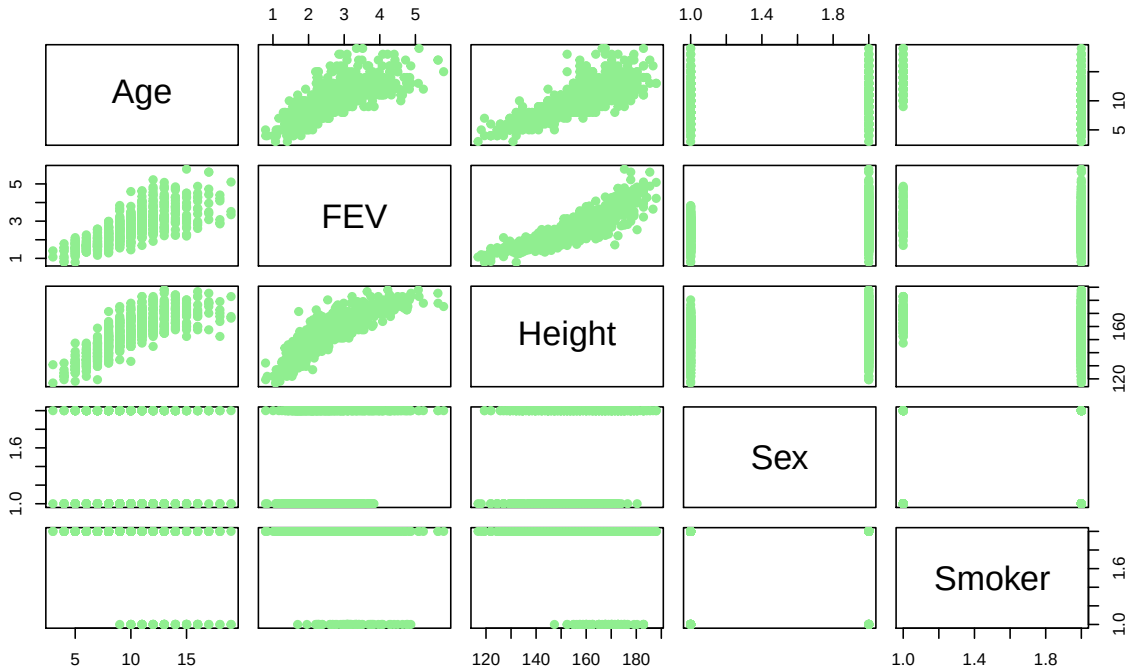


Data obsahují zhruba stejný počet děvčat a chlapců.

3.2 Vztahy mezi proměnnými

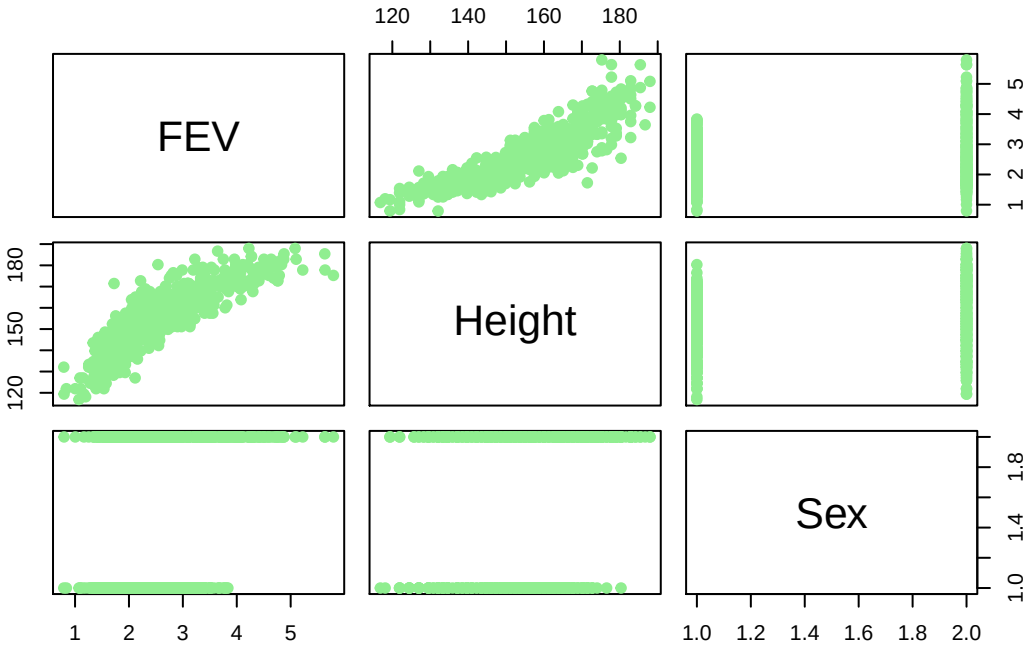
Číselnné charakteristiky pro všechny proměnné v datech získáme pomocí příkazu `summary()`. Příkaz `pairs()` vrátí grafické znázornění vztahů mezi proměnnými.

```
> # relationships between variables
> pairs(fev, col = "lightgreen", pch = 19)
```



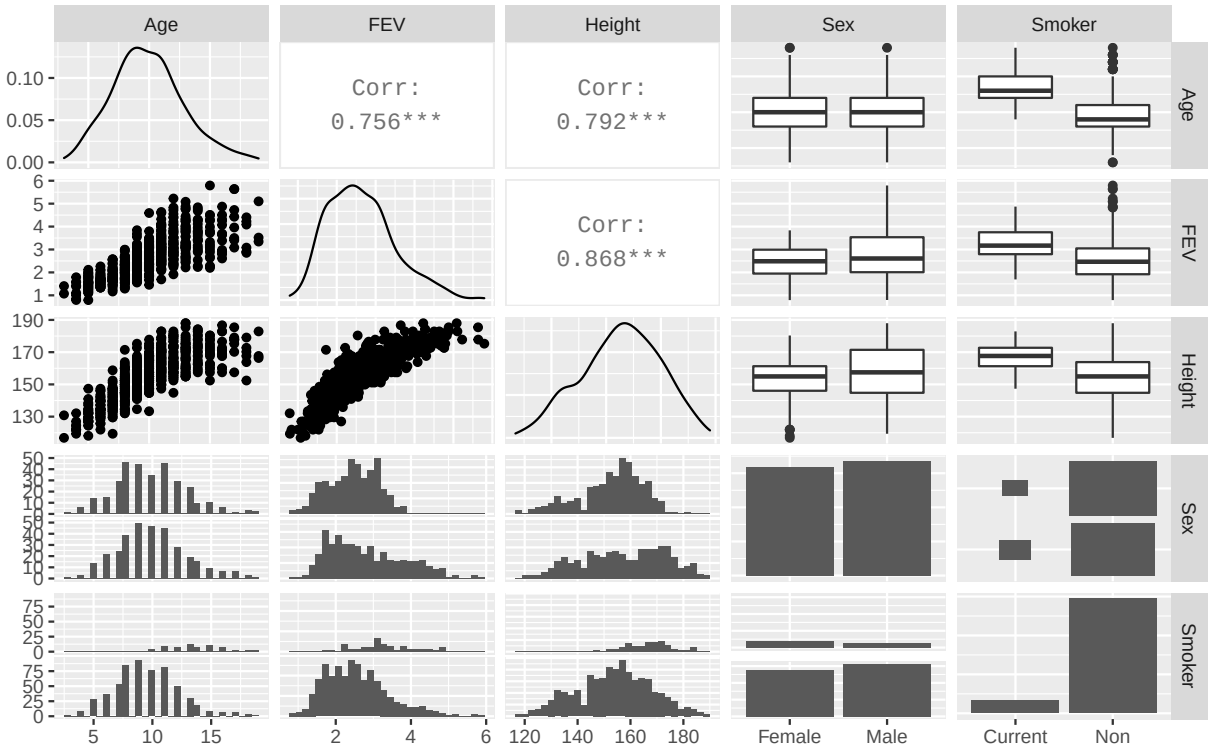
Tyto grafy jsou smysluplné především pro kvantitativní proměnné.

```
> pairs(fev[, c(2:4)], col = "lightgreen", pch = 19)
```



Funkce `ggpairs()` z knihovny `GGally` vybírá vhodné grafy pro kvantitativní i kategoriální proměnné.

```
> # relationships between variables (quantitative and categorical)
> library("GGally")
> ggpairs(fev)
```



Nyní se zaměříme na vhodné grafické i numerické popisy vztahů mezi proměnnými podle jejich typů.

3.2.1 Vztah mezi dvěmi kvantitativními proměnnými

Graficky můžema vztah mezi dvěmi kvantitativními proměnnými zobrazit pomocí klasického grafu (funkce `plot()`). Funkce `lowess()` neparametricky (bez parametrického modelu) odhadne závislost mezi proměnnými. Tu pak do grafu přidáme pomocí funkce `lines()`.

```
> plot(
+   fev$Height~fev$Age,
+   main = "Height by age",
+   ylab = "Height [cm]", xlab = "Age [y]",
+   pch = 19, col = "lightgreen"
+   )
>
> lines(lowess(fev$Height~fev$Age), lwd = 3, col = "palegreen3")
```

Číselným vyjádřením (lineární) závislosti mezi proměnnými je korelace.

```
> cor(fev$Height, fev$Age)

[1] 0.7919436
```

Těsnou závislost mezi věkem a výškou jsme očekávali. Také jsme očekávali postupné spomalování růstu výšky s věkem až po jeho pozvolné zastavení.

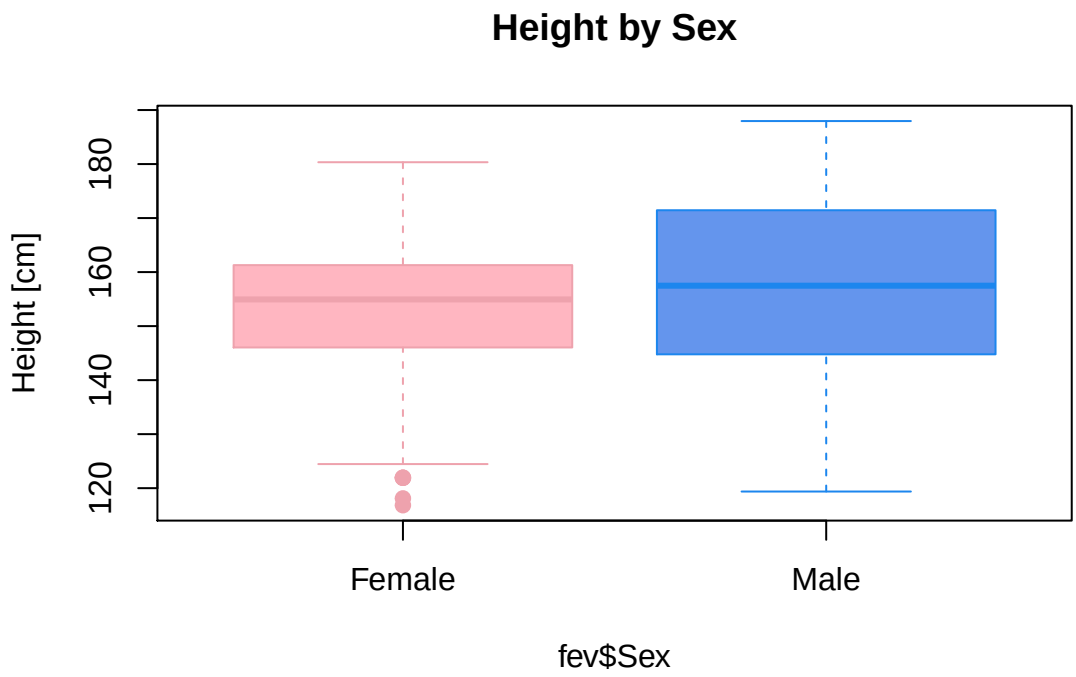
3.2.2 Vztah mezi kvantitativní a kategoriální proměnnou

Rozložení kvantitativní proměnné v závislosti na proměnné kategoriální můžeme studovat porovnáním podmíněného rozložení kvantitativní proměnné při daných úrovních kategoriální proměnné. To můžeme popsat pomocí charakteristik z části 3.1.1 jednotlivě vyhodnocených na datech rozdělených podle úrovní kategoriální proměnné.

```
> # descriptive statistics for Height
> # separately for girls and boys
> summary(fev$Height[fev$Sex == "Female"])

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  116.8   146.1   154.9   152.9   161.3   180.3
```

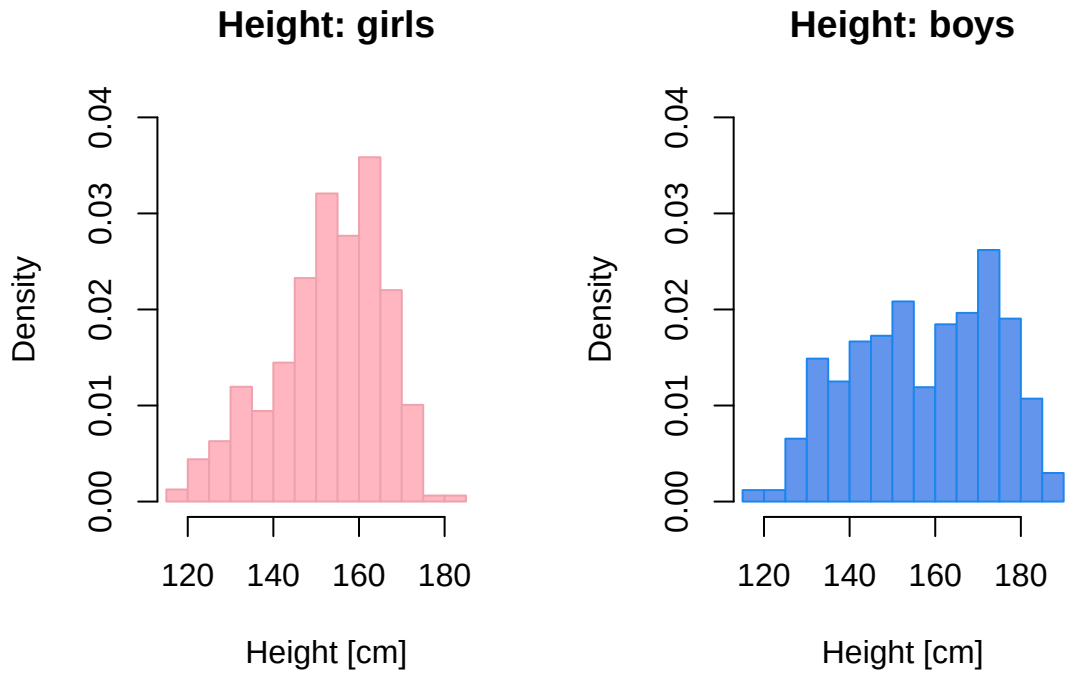
```
> summary(fev$Height[fev$Sex == "Male"])

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  119.4   144.8   157.5   157.5   171.4   188.0
```

```
> # boxplots for Height
> # separately for girls and boys
> boxplot(
+   fev$Height~fev$Sex,
+   main = "Height by Sex",  ylab = "Height [cm]",
+   col = c("lightpink", "cornflowerblue"),
+   border = c("lightpink2", "dodgerblue2"),
+   pch = 19
+   )
```




```
> par(mfrow = c(1, 2)) # divide the plot
> # into left and right panels
> # draw in the left panel
> hist(
+   fev$Height[fev$Sex == "Female"], freq = FALSE,
+   xlim = c(min(fev$Height) - 1, max(fev$Height) + 1),
```

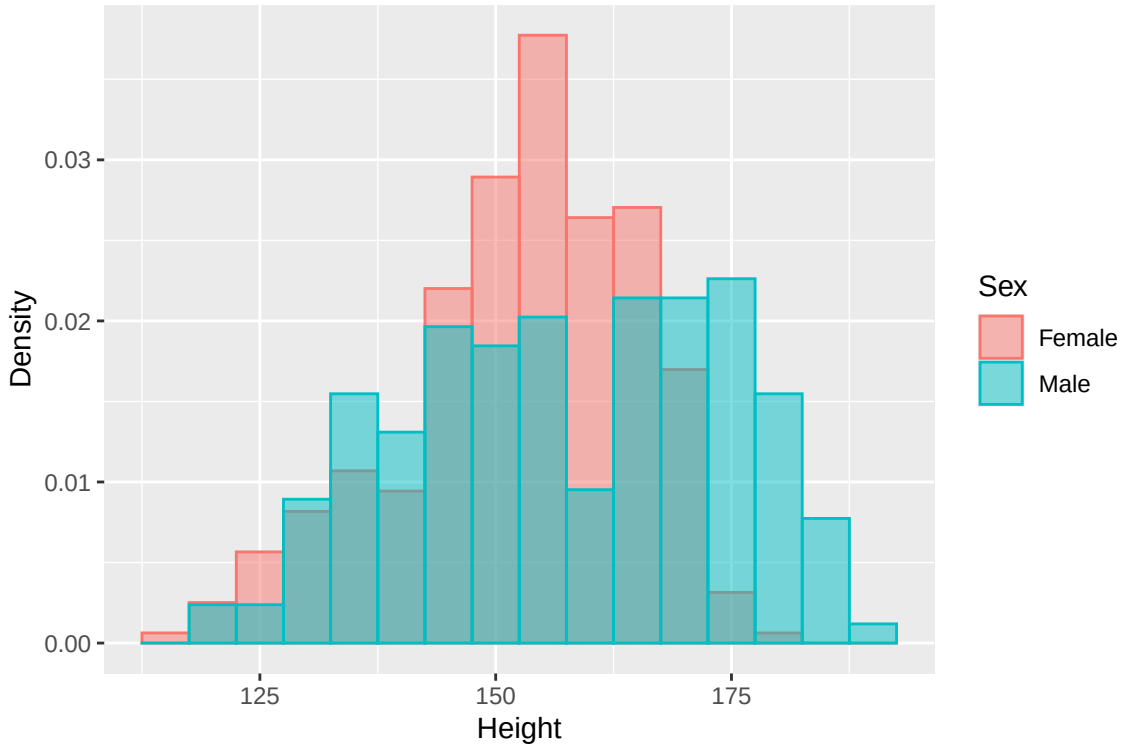
```
+   ylim = c(0, 0.04),
+   main = "Height: girls", xlab = "Height [cm]",
+   col = "lightpink", border = "lightpink2"
+   )
> # draw in the right panel
> hist(
+   fev$Height[fev$Sex == "Male"], freq = FALSE,
+   xlim = c(min(fev$Height) - 1, max(fev$Height) + 1),
+   ylim = c(0, 0.04),
+   main = "Height: boys", xlab = "Height [cm]",
+   col = "cornflowerblue", border = "dodgerblue2"
+   )
```



Zatímco krabicové diagramy vykreslilo do společného grafu, histogramy jsme museli nakreslit do dvou sousedících grafů. Aby byly porovnatelné, nastavili jsme jim stejné rozsahy os (možnosti `xlim()` a `ylim()`). Čtenáři seznámeni se syntaxí knihovny `ggplot2` mohou histogram podle skupin vykreslit s její pomocí.

```
> # boxplots for Height
> # separately for girls and boys using ggplot
> library(ggplot2)
> ggplot(data = fev, aes(x = Height, color = Sex,
```

```
+                          stat(density))) +
+   geom_histogram(position = "identity",
+                  aes(fill = Sex, color = Sex),
+                  binwidth=5, alpha = 0.5) +
+   ylab("Density")
```



Chlapci jsou podle očekávání o něco vyšší než dívky. Jejich výšky jsou v porovnání s děvčaty o něco více rozptýlené.

3.2.3 Vztah mezi dvěmi kategoriálními proměnnými

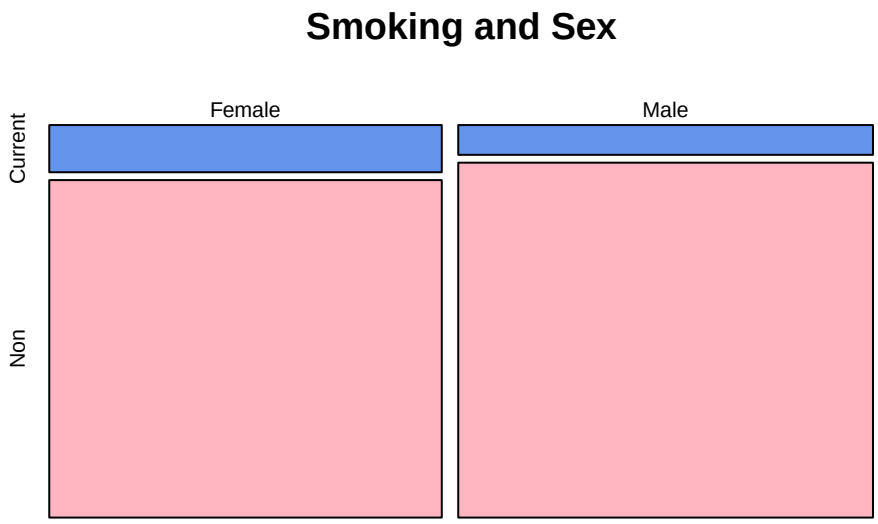
Vzájemný vztah mezi dvěma kategoriálními proměnnými také můžeme popsat pomocí podmíněných rozdělení. Za tímto účelem využijeme funkce `table()`, která vrátí počty jedinců pro jednotlivé kombinace kategorií. Na výsledek můžeme dále aplikovat funkci `prop.table()`, která počty přepočítá na proporce. V základní formě se procenta v celé tabulce sčítají na 1 (popisují sdružené rozdělení). Pomocí voleb `margin = 1` a `margin = 2` se na 1 sčítají procenta v řadcích nebo sloupcích (popisují marginální rozdělení).

```
> # joint distribution of Smoking and Sex
> table(fev$Smoker, fev$Sex)
```

```
         Female Male
  Current     39   26
  Non        279  310

> # conditional distribution of Sex for given Smoking status
> prop.table(table(fev$Smoker, fev$Sex), margin = 1)


            Female      Male
  Current 0.6000000 0.4000000
  Non     0.4736842 0.5263158

> # conditional distribution of Smoking for given Sex
> prop.table(table(fev$Smoker, fev$Sex), margin = 2)


             Female       Male
  Current 0.12264151 0.07738095
  Non     0.87735849 0.92261905
```

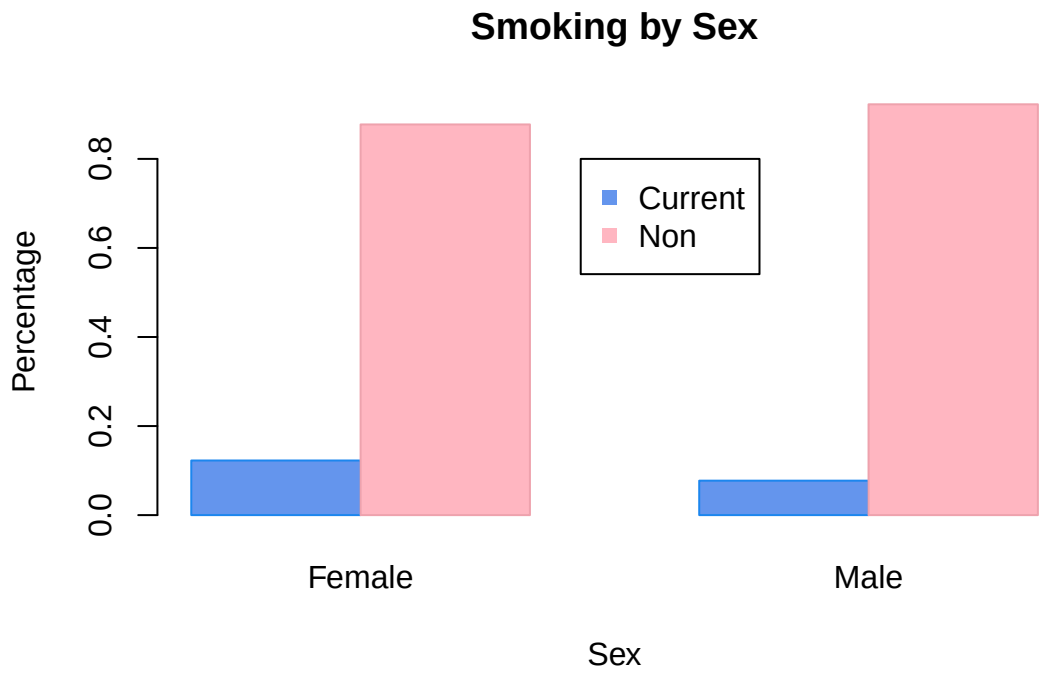
Sdružené rozdělení můžeme vizualizovat pomocí funkce `mosaicplot()`.

```
> # mosaic plot for Sex and Smoking
> mosaicplot(
+   table(fev$Sex, fev$Smoker),
+   main = "Smoking and Sex",
+   col = c("cornflowerblue", "lightpink")
+   )
```

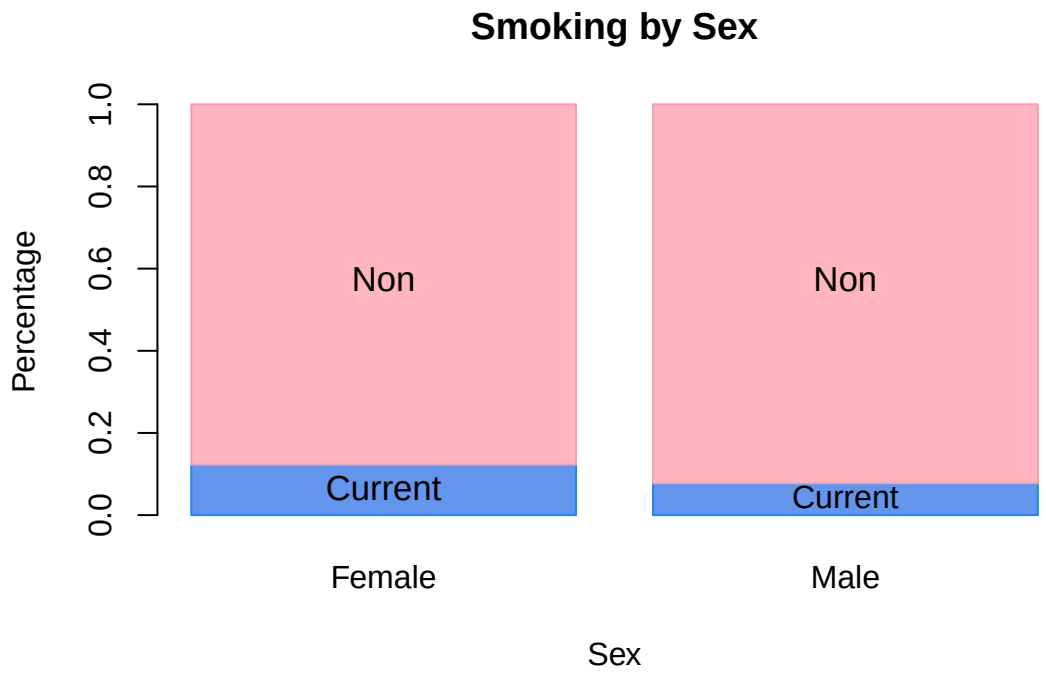


Podmíněné rozdělení můžeme také vizualizovat pomocí sloupcového grafu rozděleného podle jednotlivých úrovní kategoriální proměnné.

```
> # bar plot for Smoking by Sex
> barplot(
+   height = prop.table(table(fev$Smoker, fev$Sex), margin = 2),
+   beside = TRUE,
+   main = "Smoking by Sex",
+   ylab = "Percentage", xlab = "Sex",
+   col = c("cornflowerblue", "lightpink"),
+   border = c("dodgerblue2", "lightpink2")
+   )
> legend(
+   x = 3.3, y = 0.8,
+   legend = c("Current", "Non"),
+   col = c("cornflowerblue", "lightpink"),
+   pch = 15
+   )
```



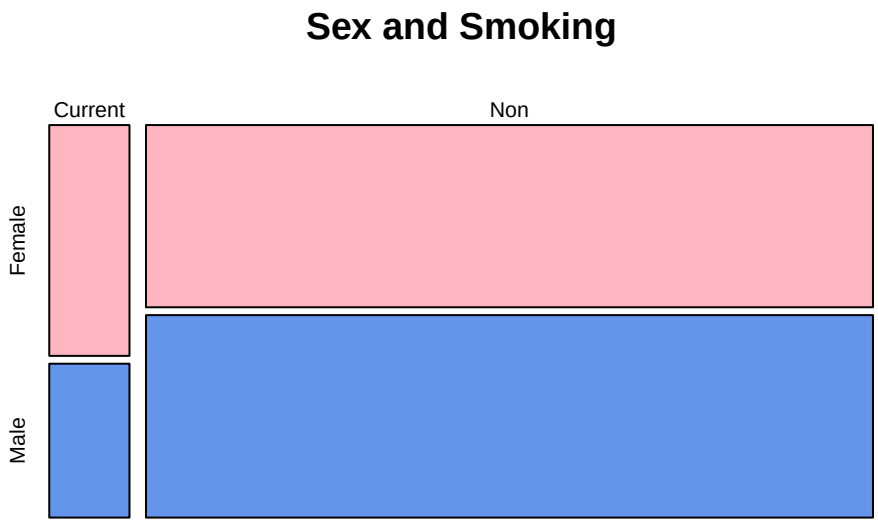

```
> # bar plot for Smoking by Sex
> barplot(
+   height = prop.table(table(fev$Smoker, fev$Sex), margin = 2),
+   beside = FALSE,
+   main = "Smoking by Sex",
+   ylab = "Percentage", xlab = "Sex",
+   col = c("cornflowerblue", "lightpink"),
+   border = c("dodgerblue2", "lightpink2")
+   )
> text(
+   x = 1.2 * c(0:2) + 0.7, y = c(-.01, -.03),
+   labels = "Current",
+   cex = c(1.1, 1), pos = 3
+   )
> text(
+   x = 1.2 * c(0:2) + 0.7, y = 0.5,
+   labels = "Non",
+   cex = 1.1, pos = 3
+   )
```



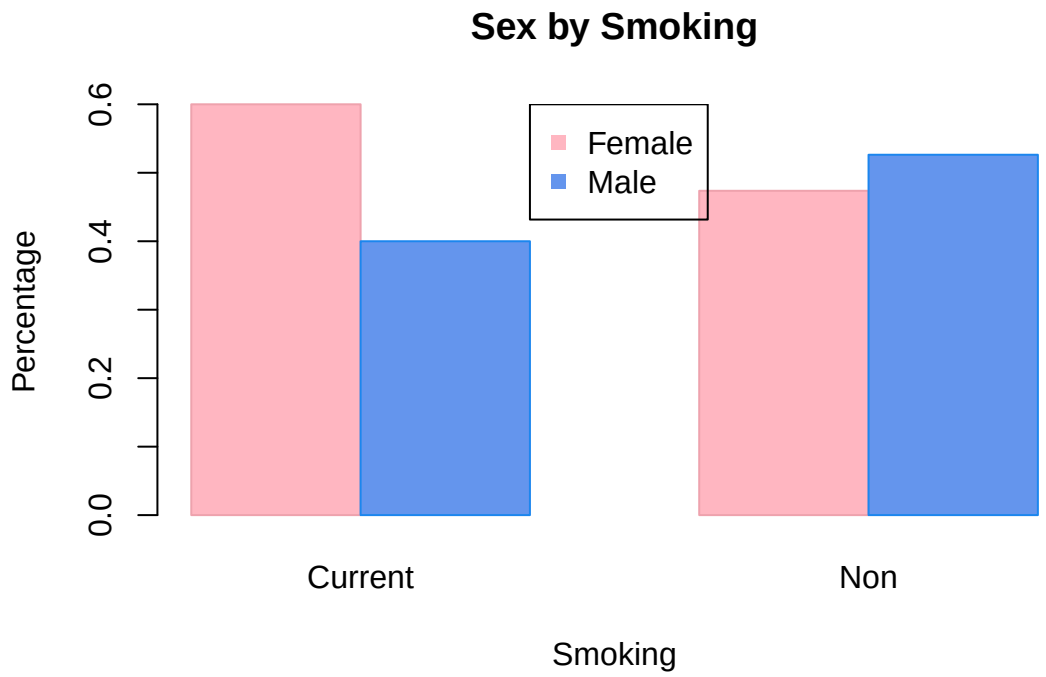
Všimněme si, že představu o kouření mezi chlapci a děvčaty jsme si mohli udělat i na základě mozaikového grafu výše. Mezi děvčaty najdeme kuřačky o něco častěji než kuřáky mezi chlapci.

Rozdělení pohlaví mezi kuřáky a nekuřáky získáme výměnou pořadí proměnných v kódech uvedených výše.

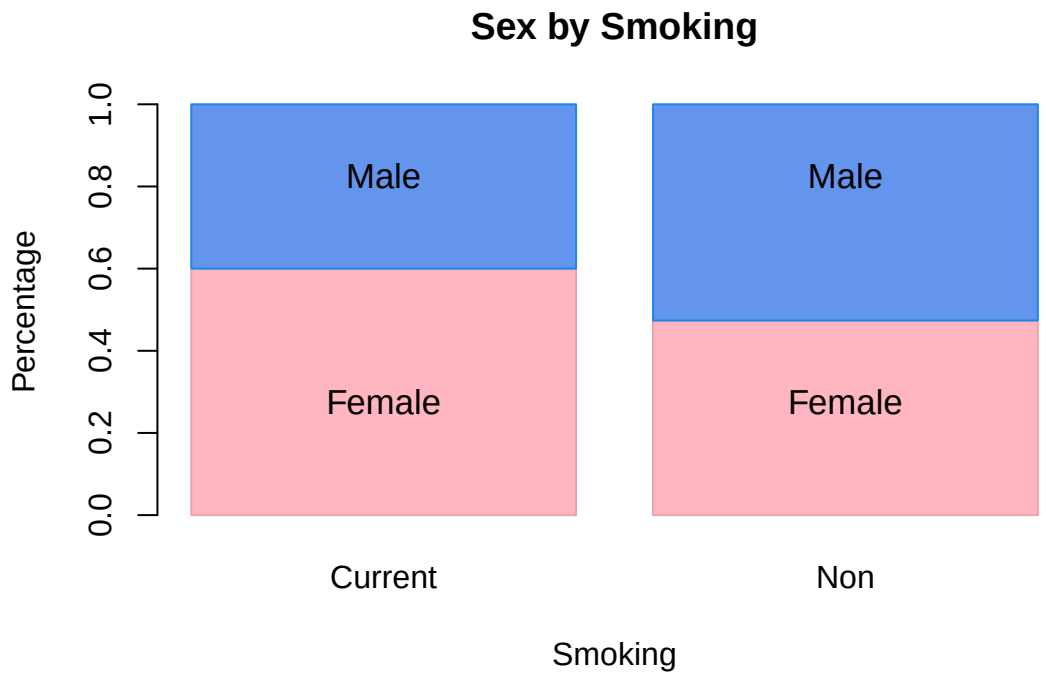
```
> # mosaic plot for Sex and Smoking
> mosaicplot(
+   table(fev$Smoker, fev$Sex),
+   main = "Sex and Smoking",
+   col = c("lightpink", "cornflowerblue")
+   )
```




```
> # bar plot for Sex by Smoking
> barplot(
+   height = prop.table(table(fev$Sex, fev$Smoker), margin = 2),
+   beside = TRUE,
+   main = "Sex by Smoking",
+   ylab = "Percentage", xlab = "Smoking",
+   col = c("lightpink", "cornflowerblue"),
+   border = c("lightpink2", "dodgerblue2")
+   )
> legend(
+   x = 3, y = 0.6,
+   legend = c("Female", "Male"),
+   col = c("lightpink", "cornflowerblue"),
+   pch = 15
+   )
```




```
> # bar plot for Sex by Smoking
> barplot(
+   height = prop.table(table(fev$Sex, fev$Smoker), margin = 2),
+   beside = FALSE,
+   main = "Sex by Smoking",
+   ylab = "Percentage", xlab = "Smoking",
+   col = c("lightpink", "cornflowerblue"),
+   border = c("lightpink2", "dodgerblue2")
+   )
> text(
+   x = 1.2 * c(0:2) + 0.7, y = 0.2,
+   labels="Female",
+   cex = 1.1, pos = 3
+   )
> text(
+   x = 1.2 * c(0:2) + 0.7, y = 0.75,
+   labels = "Male",
+   cex = 1.1, pos = 3
+   )
```



Mezi kuřáky jsou o něco více zastoupena děvčata než chlapci, mezi nekuřáky je tomu obráceně.

Kapitola 4

Lineární model

Lineární model popisuje závislost mezi *vysvětlovanou proměnnou* Y (říkáme jí také *závisle proměnná* nebo *odezva*, anglicky *outcome*, *response* anebo také *dependent variable*) a *vysvětlujícími proměnnými* $x_1, \ldots, x_k$ (říkáme jim také *nezávisle proměnné* nebo *prediktory*, anglicky *covariates*, *predictors* anebo také *independent variables* nebo *explanatory variables*). V lineárním modelu je jejich závislost ve tvaru

$$Y_i = \beta_0 + \beta_1 x_{i,1} + \ldots + \beta_k x_{i,k} + \varepsilon_i, \quad i = 1, \ldots, n, \tag{4.1}$$

kde $\beta_0, \ldots, \beta_k$ jsou *regresní koeficienty* (anglicky *regression coefficients*) a ε_i jsou *náhodné chyby* (anglicky *random errors*). Jak již napovídá terminologie, náhodné chyby považujeme za náhodné veličiny. Naproti tomu prediktory považujeme za nenáhodné, respektive jsou-li prediktory náhodné, studujeme závislost odezvy Y na prediktorech $X_1, \ldots, X_k$ podmíněně při jejich pozorovaných hodnotách $x_1, \ldots, x_k$, a tuto závislost pak popisujeme rovnicí (4.1). Odezva Y je v obou přístupech náhodná veličina, zatímco koeficienty $\beta_0, \ldots, \beta_k$ jsou nenáhodné.

Má-li se jednat o lineární model, pak vše, co je v závislosti odezvy na prediktorech systematické, musí být popsáno *lineárním prediktorem* $\eta_i = \beta_0 + \beta_1 x_{i,1} + \ldots + \beta_k x_{i,k}$, tj. $\mathrm{E}Y_i = \eta_i$, resp. $\mathrm{E}\varepsilon_i = 0$, $i = 1, \ldots, n$. Navíc předpokládáme, že odezvy pro jednotlivá pozorování jsou nekorelované, $\mathrm{Cor}(Y_i, Y_j) = 0$ pro $i \neq j$, $i, j = 1, \ldots, n$, a mají stejný rozptyl $\mathrm{Var}\, Y_i = \sigma^2$, $i = 1, \ldots, n$. Jelikož $\mathrm{Var}\, Y_i = \mathrm{Var}\, \varepsilon_i$, lze rovnost rozptylů chápat také tak, že odezvy pro jednotlivá pozorování kolem svých středních hodnot kolísají se stejnou přesností. Všechny předpoklady lineáního modelu můžeme formulovat v řeči náhodných chyb ε_i, od kterých požadujeme stejné rozdělení, nulovou střední hodnotu, nekorelovanost a stejný rozptyl. Někdy navíc požadujeme normální rozdělení a jelikož v normálním rozdělení nekorelovanost implikuje nezávislost, předpoklady v takovém případě můžeme zapsat ve tvaru $\varepsilon_i \overset{\mathrm{iid}}{\sim} \mathrm{N}(0, \sigma^2)$, $i \in \{1, \ldots, n\}$, kde skratka *iid* pochází z anglického *independent, identically distributed*: nezávislé, stejně rozdělené.

Lineární prediktor $\eta_i = \beta_0 + \beta_1 x_{i,1} + \ldots + \beta_k x_{i,k}$ je funkce lineární jak v prediktorech tak v koeficientech. Linearita v prediktorech a linearita v koeficientech ale v lineárním modelu nehrají stejnou roli. Uvažujme například model

$$Y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \varepsilon_i, \quad i = 1, \ldots, n.$$

Jde o speciální případ modelu (4.1), kde $k = 2$, a prediktory jsou x_1 a x_2. Se stejnými daty můžeme uvažovat i model

$$Y_i = \beta_0 + \beta_1 x_{i,1}^2 + \beta_2 x_{i,2} + \varepsilon_i, \quad i = 1, \ldots, n, \tag{4.2}$$

nebo

$$Y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,1}^2 + \beta_3 x_{i,2} + \varepsilon_i, \quad i = 1, \ldots, n, \tag{4.3}$$

anebo

$$Y_i = \beta_0 + \beta_1 \frac{x_{i,1}}{x_{i,2}} + \varepsilon_i, \quad i = 1, \ldots, n. \tag{4.4}$$

Ve všech případech se jedná o lineární model. Model (4.2) je speciálním případem modelu (4.1) s $k = 2$ a prediktory x_1^2 a x_2, Model (4.3) je speciálním případem modelu (4.1) s $k = 3$ a prediktory x_1, x_1^2 a x_2, a model (4.4) je speciálním případem modelu (4.1) s $k = 1$ a prediktorem x_1/x_2. V dalším textu nebudeme modely vždy nutně přepisovat do tvaru (4.1). Dojde-li k *transformaci* prediktoru nebo prediktorů, například tím, že prediktor x_1 nahradíme nebo doplníme prediktorem x_1^2, anebo prediktory x_1 a x_2 nahradíme prediktorem x_1/x_2, ponecháme modely ve tvarech (4.2), (4.3), anebo (4.4). Také nutně nepřestaneme původní prediktory x_1 a x_2 nazývat prediktory, ani když se v modelu vyskytovat nebudou (vždy si můžeme představit, že se v modelu vyskytují, ale příslušný koeficient β je nulový). Linearita prediktoru v koeficientech naopak zachována být musí. Například model

$$Y_i = \beta_0 + \beta_2 x_{i,1}^{\beta_1} + \varepsilon_i, \quad i = 1, \ldots, n,$$

už není lineárním modelem, protože jej nelze přepsat do tvaru (4.1).

Zápis (4.1) definuje linéarní model po složkách. Mnohdy se hodí využít maticového zápisu

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{4.5}$$

kde

$$\underbrace{\begin{pmatrix} Y_1 \\ Y_2 \\ \ldots \\ \ldots \\ \ldots \\ Y_n \end{pmatrix}}_{\mathbf{Y}} = \underbrace{\begin{pmatrix} 1 & x_{1,1} & x_{1,2} & \ldots & x_{1,k} \\ 1 & x_{2,1} & x_{2,2} & \ldots & x_{2,k} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & x_{n,1} & x_{n,2} & \ldots & x_{n,k} \end{pmatrix}}_{\mathbf{X}} \times \underbrace{\begin{pmatrix} \beta_0 \\ \beta_1 \\ \ldots \\ \beta_k \end{pmatrix}}_{\boldsymbol{\beta}} + \underbrace{\begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \ldots \\ \ldots \\ \ldots \\ \varepsilon_n \end{pmatrix}}_{\boldsymbol{\varepsilon}}.$$

Matici $\mathbf{X}$ v zapisu (4.5) říkáme *regresní matice*, *matice modelu* nebo také *matice plánu*, anglicky *design matrix* nebo *model matrix*.

Nyní se podívámé na několik lineárních modelů pro data `fev` ze 2. kapitoly. Nejprve si ale připomeneme, jak data vypadají.

```
> head(fev)   # first rows of the data
```

```
  Age   FEV Height    Sex Smoker
1   9 1.708 144.78 Female    Non
2   8 1.724 171.45 Female    Non
3   7 1.720 138.43 Female    Non
4   9 1.558 134.62   Male    Non
5   9 1.895 144.78   Male    Non
6   8 2.336 154.94 Female    Non

> tail(fev)  # last rows of the data

    Age   FEV Height    Sex  Smoker
649  16 4.872 182.88   Male Current
650  16 4.270 170.18   Male Current
651  15 3.727 172.72   Male Current
652  18 2.853 152.40 Female     Non
653  16 2.795 160.02 Female Current
654  15 3.211 168.91 Female     Non
```

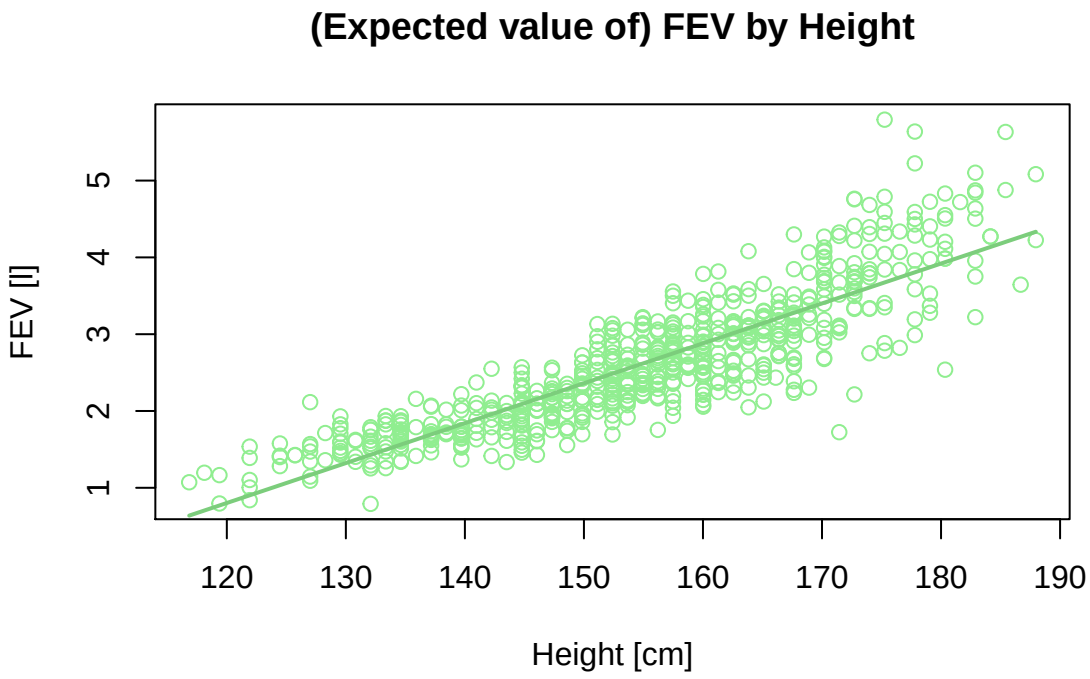
(1.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \varepsilon_i, \quad i = 1, \ldots, n$

Odezvou je v tomto modelu indikátor objemu plic FEV a jediným prediktorem je výška Height. Máme tedy $k = 1$. Pomocí vektorů a matic bychom model zapsali následovně.

$$\begin{pmatrix} 1.708 \\ 1.724 \\ \ldots \\ 2.795 \\ 3.211 \end{pmatrix} = \begin{pmatrix} 1 & 144.78 \\ 1 & 171.45 \\ \ldots & \ldots \\ 1 & 160.02 \\ 1 & 168.91 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \ldots \\ \varepsilon_{653} \\ \varepsilon_{654} \end{pmatrix}.$$

Pro lineární prediktor platí $\eta_i = \mathrm{E}(\mathsf{FEV}_i) = \beta_0 + \beta_1 \times \mathsf{Height}_i$, střední hodnota FEV podle modelu tedy lineárně roste s výškou. Koeficient β_1 vyjádřuje, o kolik litrů je střední hodnota FEV vyšší u dítěte, které je o 1 cm vyšší. Obecně netvrdíme, že nárůst střední hodnoty FEV je *způsobený* rozdílem ve výšce: to jenom na základě modelu tvrdit nemůžeme. Proto raději mluvíme o *asociaci*: u vyšších dětí pozorujeme větší plíce. Koeficient β_0 v modelu reprezentuje střední hodnotu FEV pro dítě s nulovou výškou: v tomto modelu tedy rozumnou interpretaci nemá. Vztah mezi střední hodnotou FEV a výškou popsaný modelem by mohl vypadat jako na obrázku níže.

```
# Coefficient beta_1 represents the increase in expected value
# of FEV associated with a 1 cm increase in Height.
# We talk about associations rather than causal dependence:
# the increase in expected value of FEV is *associated with*
# an increase in Height, not *caused by* it.
# beta_0 represents the expected value of FEV for a child
# with Height of 0 cm, which makes little sense.
```



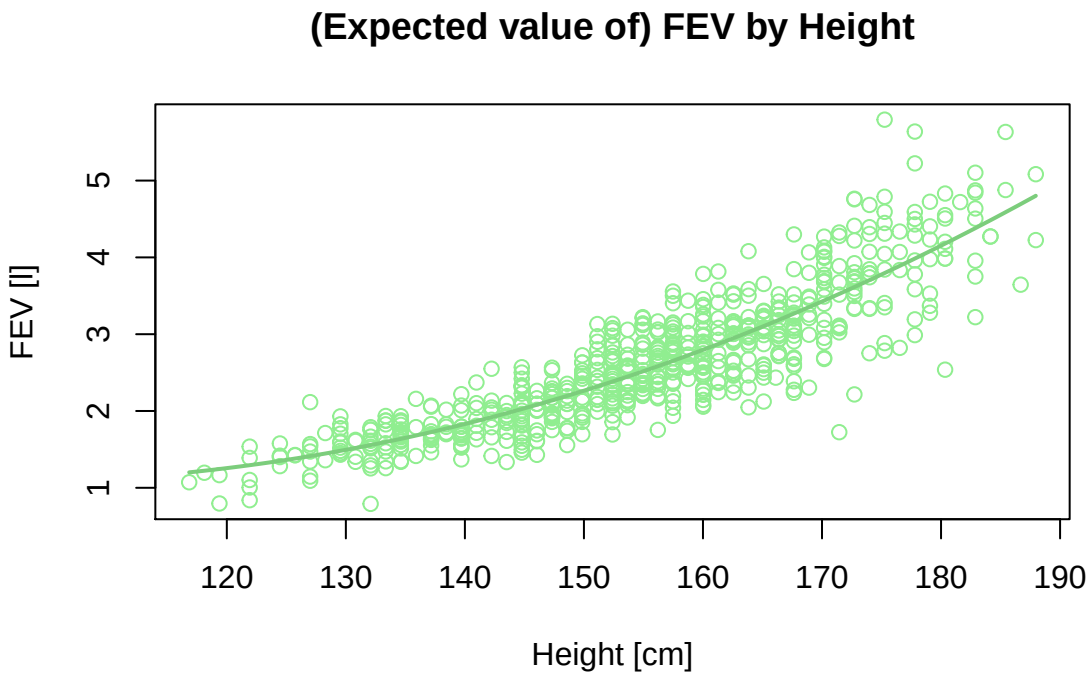
(2.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Height}_i^2 + \varepsilon_i, \quad i = 1, \ldots, n$

I v tomto modelu máme odezvu FEV, ale nyní je $k = 2$. Technicky jsou prediktory výška Height a kvadrát výšky Height^2, ve skutečnosti ale jde o funkce jediného prediktoru: výšky. Maticový zápis je

$$\begin{pmatrix} 1.708 \\ 1.724 \\ \ldots \\ 2.795 \\ 3.211 \end{pmatrix} = \begin{pmatrix} 1 & 144.78 & 144.78^2 \\ 1 & 171.45 & 171.45^2 \\ \ldots & \ldots & \ldots \\ 1 & 160.02 & 160.02^2 \\ 1 & 168.91 & 168.91^2 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \ldots \\ \varepsilon_{653} \\ \varepsilon_{654} \end{pmatrix}.$$

Pro lineární prediktor platí $\eta_i = \mathrm{E}(\mathsf{FEV}_i) = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Height}_i^2$. Střední hodnota FEV podle modelu tedy kvadraticky roste s výškou. Interpretaci jednotlivých koeficientů je v takovém modelu vhodnější nahradit obrázkem anebo výpočtem či porovnáním střední hodnoty FEV pro několik zajímavých hodnot výšky Height.

```
# In a model like this, it is more practical to replace
# the interpretation of individual coefficients by a picture
# or by computing and comparing expected values of FEV
# for several interesting values of Age.
```



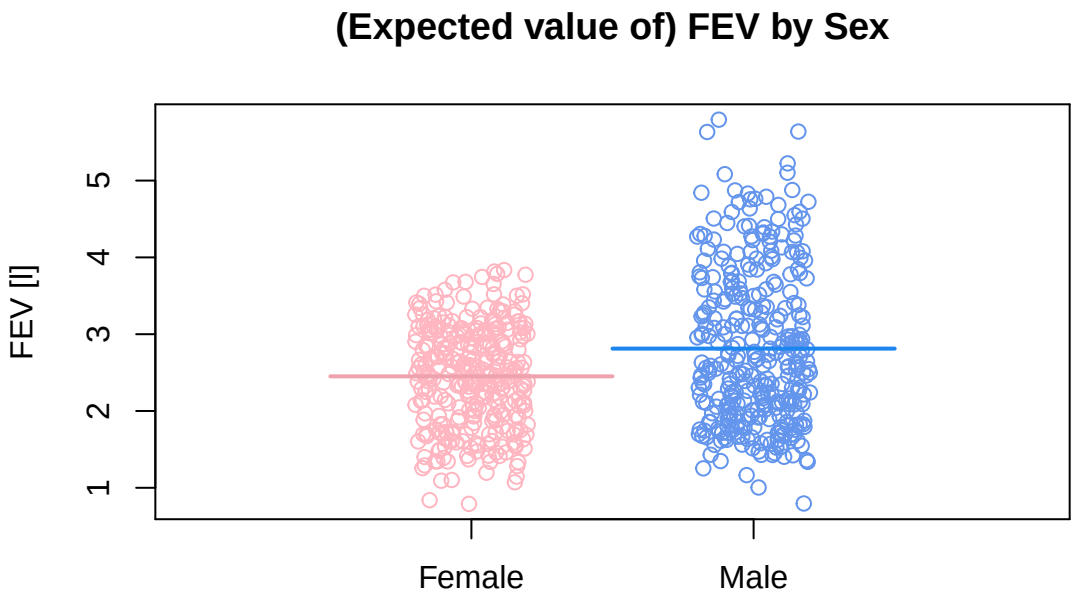
(3.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Sex}_i + \varepsilon_i, \quad i = 1, \ldots, n$

V tomto modelu máme odezvu FEV a $k = 1$. Prediktor Sex je tentokrát kategoriální proměnná o dvou úrovních: chlapec a dívka. Tyto úrovně budeme číselně kódovat jako 0 a 1. Nezvolíme-li si jinak, R přiřadí menší číslo úrovni, která je první v abecedním pořadí. V našem případě půjde o dívku (female). Maticový zápis modelu je

$$\begin{pmatrix} 1.708 \\ 1.724 \\ \ldots \\ 2.795 \\ 3.211 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ \ldots & \ldots \\ 1 & 1 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \ldots \\ \varepsilon_{653} \\ \varepsilon_{654} \end{pmatrix}.$$

Pro lineární prediktor platí $\eta_i = \mathrm{E}(\mathsf{FEV}_i) = \beta_0 + \beta_1 \times \mathsf{Sex}_i$. Koeficient β_0 reprezentuje střední hodnotu FEV pro dívky, zatímco koeficient β_1 vyjádřuje, o kolik litrů vyšší je střední hodnota FEV pro chlapce oproti dívkám. Střední hodnotu FEV pro chlapce můžeme spočítat jako $\beta_0 + \beta_1$.

```
# Coefficient beta_0 represents the expected value of FEV
# for females (coded as 0); coefficient beta_1 represents
# the difference between expected value of FEV for males (coded
# as 1) and females (coded as 0).
# The default order of coding is alphabetical (F comes before M).
# The expected value of FEV for males is beta_0 + beta_1.
```



(4.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Sex}_i + \varepsilon_i, \quad i = 1, \ldots, n$

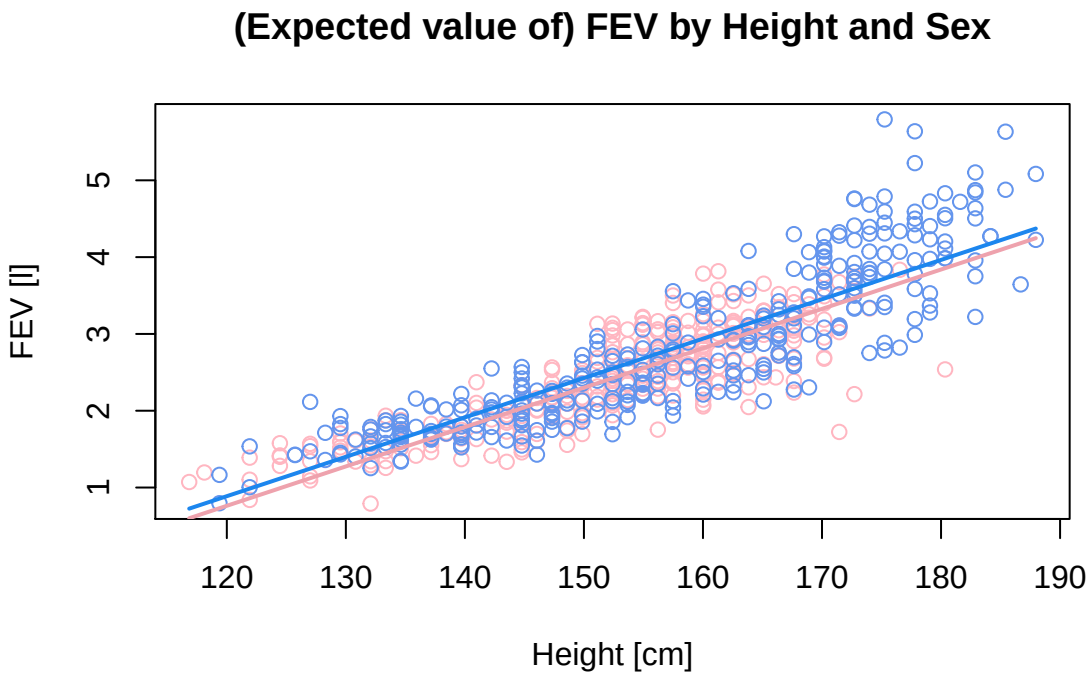
I v tomto modelu máme odezvu FEV, prediktory máme dva ($k = 2$): výška Height a pohlaví Sex. Stejně jako výše je pohlaví kódováno jako 0 a 1: 0 kóduje dívky a 1 kóduje chlapce. Maticový zápis je

$$\begin{pmatrix} 1.708 \\ 1.724 \\ \ldots \\ 2.795 \\ 3.211 \end{pmatrix} = \begin{pmatrix} 1 & 144.78 & 0 \\ 1 & 171.45 & 0 \\ \ldots & \ldots & \ldots \\ 1 & 160.02 & 1 \\ 1 & 168.91 & 0 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \ldots \\ \varepsilon_{653} \\ \varepsilon_{654} \end{pmatrix}.$$

Pro lineární prediktor platí $\eta_i = \mathrm{E}(\mathsf{FEV}_i) = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Sex}_i$. Střední hodnota FEV tedy podle modelu roste s výškou lineárně a přímky pro dívky a chlapce jsou rovnoběžné, obě se směrnicí β_1. Koeficient β_1 vyjádřuje, o kolik litrů je vyšší střední hodnota FEV u dítěte, které je o 1 cm vyšší, porovnáváme-li dvě děti stejného pohlaví. Koeficient β_2 vyjádřuje, o kolik litrů vyšší je střední hodnota FEV pro chlapce oproti stejně vysoké dívce.

```
# Model without interaction:
# Expected value of FEV increases linearly with Height,
# the lines for boys and girls are assumed to be parallel.
# Coefficient beta_1 represents the increase in expected value
# of FEV associated with a 1 cm increase in Height when comparing
```

```
# two children of the same Sex.
# Coefficient beta_2 represents how much higher the expected value
# of FEV is for a boy compared to a girl of the same Height.
```



(5.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Sex}_i + \beta_3 \times (\mathsf{Sex} \times \mathsf{Height})_i + \varepsilon_i, \quad i = 1, \ldots, n$

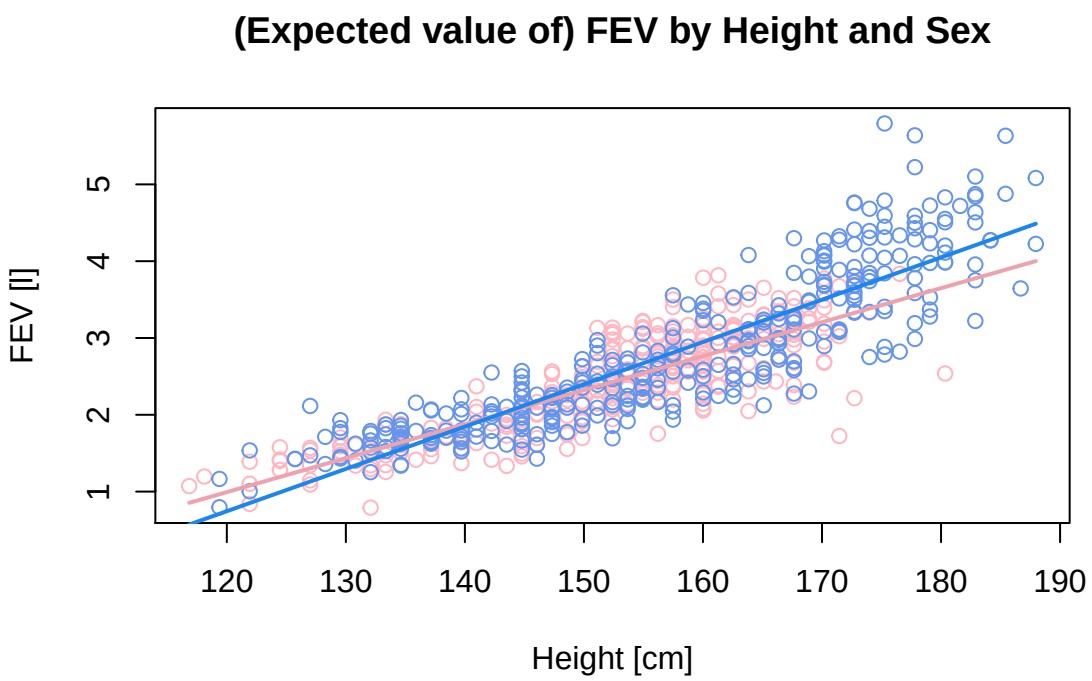
Tento model se od předchozího liší přidáním nového prediktoru, který je funkcí prediktorů z předchozího modelu: výšky Height a pohlaví Sex. V kontextu regresních modelů mu říkáme *interakce*. Jde o funkci Sex × Height, pro všechny dívky má tedy nulovou hodnotu, zatímco pro chlapce obsahuje jejich výšky. V přítomnosti interakce mezi prediktory se na původní prediktory někdy odkazujeme jako na *hlavní efekty*. Máme tedy $k = 3$ a maticový zápis modelu je

$$\begin{pmatrix} 1.708 \\ 1.724 \\ \cdots \\ \cdots \\ 2.795 \\ 3.211 \end{pmatrix} = \begin{pmatrix} 1 & 144.78 & 0 & 0 \\ 1 & 171.45 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ 1 & 160.02 & 1 & 160.02 \\ 1 & 168.91 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \cdots \\ \cdots \\ \varepsilon_{653} \\ \varepsilon_{654} \end{pmatrix}.$$

Pro lineární prediktor platí $\eta_i = \mathrm{E}(\mathsf{FEV}_i) = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Sex}_i + \beta_3 \times (\mathsf{Height} \times \mathsf{Sex})_i$. Střední hodnota FEV tedy podle modelu roste s výškou lineárně, tentokrát ale přímky pro dívky a pro chlapce nemusí být rovnoběžné. Směrnice pro dívky je β_1,

zatímco směrnice pro chlapce je $\beta_1 + \beta_3$. Koeficient β_1 vyjádřuje, o kolik litrů se u dívek zvýší střední hodnota FEV s 1 cm výšky navíc. U chlapců dostaneme obdobnou interpretaci pro koeficient $\beta_1 + \beta_3$. Rozdíl mezi střední hodnotou FEV pro chlapce oproti stejně vysoké dívce v tomto modelu závisí na výšce uvažovaných dětí.

```
# Model with interaction:
# Expected value of FEV increases linearly with Height,
# the lines for boys and girls are not necessarily parallel.
# Coefficient beta_1 represents the increase in expected value
# of FEV associated with a 1 cm increase in Height for girls.
# beta_1 + beta_3 represents the increase in expected value
# of FEV associated with a 1 cm increase in Height for boys.
# The difference in expected values of FEV for boys and girls
# of the same Height here depends on the value of Height.
```



(6.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathbb{I}\{9 < \mathsf{Age}_i \leq 12\} + \beta_3 \times \mathbb{I}\{\mathsf{Age}_i > 12\} + \varepsilon_i, i = 1, \quad \ldots, n$

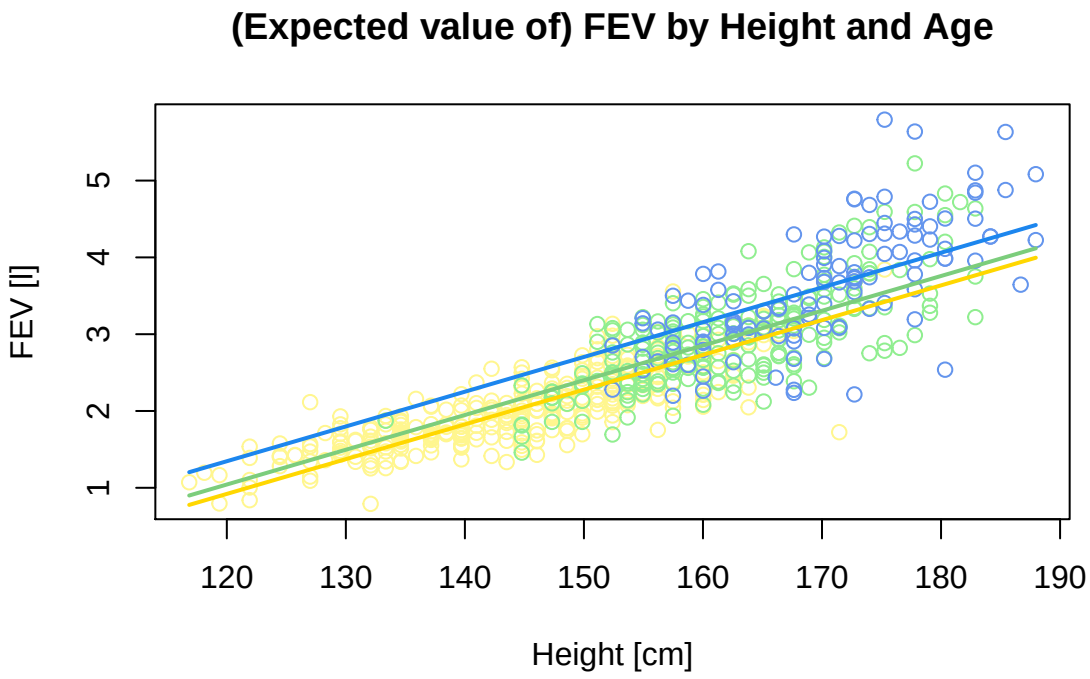
V tomto modelu je odezvou opět FEV. Prediktory jsou prakticky výška Height a věk Age, ale věk je *kategorizovaný* – rozdělený do tří kategorií – čímž se z něj stává kategoriální proměnná o třech úrovních. V lineárním modelu budeme jednu z těchto úrovní (v našem případě věk do 9 let včetně) brát jako *referenční* a zbylé úrovně (věk mezi 9 a 12 lety, věk nad 12 let) s ní budeme srovnávat. Za tímto účelem vytvoříme dva nové prediktory – jeden pro každou úroveň, kterou chceme srovnávat s referenční úrovní – a naplníme je jedničkami

pro pozorování, pro která věk dosahuje dané úrovně, a nulami pro zbylá pozorování. Dostaneme tak $k = 3$ a maticový zápis modelu bude

$$\begin{pmatrix} 1.708 \\ 1.724 \\ \dots \\ \dots \\ 2.795 \\ 3.211 \end{pmatrix} = \begin{pmatrix} 1 & 144.78 & 0 & 0 \\ 1 & 171.45 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ 1 & 160.02 & 0 & 1 \\ 1 & 168.91 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \dots \\ \dots \\ \varepsilon_{653} \\ \varepsilon_{654} \end{pmatrix}.$$

Pro lineární prediktor platí $\eta_i = \mathrm{E}(\mathsf{FEV}_i) = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathbb{I}\{9 < \mathsf{Age}_i \leq 12\} + \beta_3 \times \mathbb{I}\{\mathsf{Age}_i > 12\}$. Střední hodnota FEV tedy podle modelu roste s výškou lineárně a přímky pro jednotlivé věkové kategorie jsou rovnoběžné, všechny se směrnicí β_1. Koeficient β_1 vyjádřuje, o kolik litrů je vyšší střední hodnota FEV u dítěte, které je o 1 cm vyšší, porovnáváme-li dvě děti ze stejné věkové kategorie. Koeficient β_2 vyjádřuje, o kolik litrů vyšší je střední hodnota FEV pro dítě mezi 9 a 12 lety oproti stejně vysokému dítěti do 9 let. Koeficient β_3 vyjádřuje, o kolik litrů vyšší je střední hodnota FEV pro dítě starší 12 let oproti stejně vysokému dítěti do 9 let.

```
# Expected value of FEV increases linearly with Height,
# the lines for the three age categories are parallel.
# Coefficient beta_1 represents the increase in expected value
# of FEV associated with a 1 cm increase in Height when comparing
# two children from the same age category.
# Coefficient beta_2 represents how much higher the expected value
# of FEV is for a child aged between 9 and 12 compared to a child
# of the same Height aged 9 or less.
# Coefficient beta_3 represents how much higher the expected value
# of FEV is for a child aged 12 or more compared to a child
# of the same Height aged 9 or less.
```

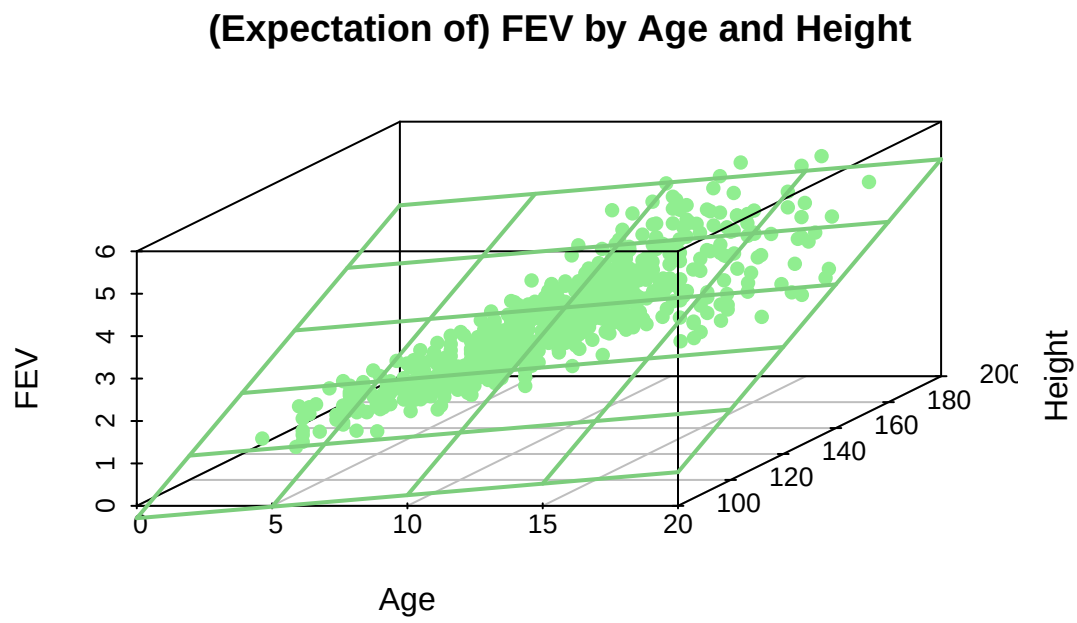


(7.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Age}_i + \varepsilon_i, i = 1, \quad \ldots, n$

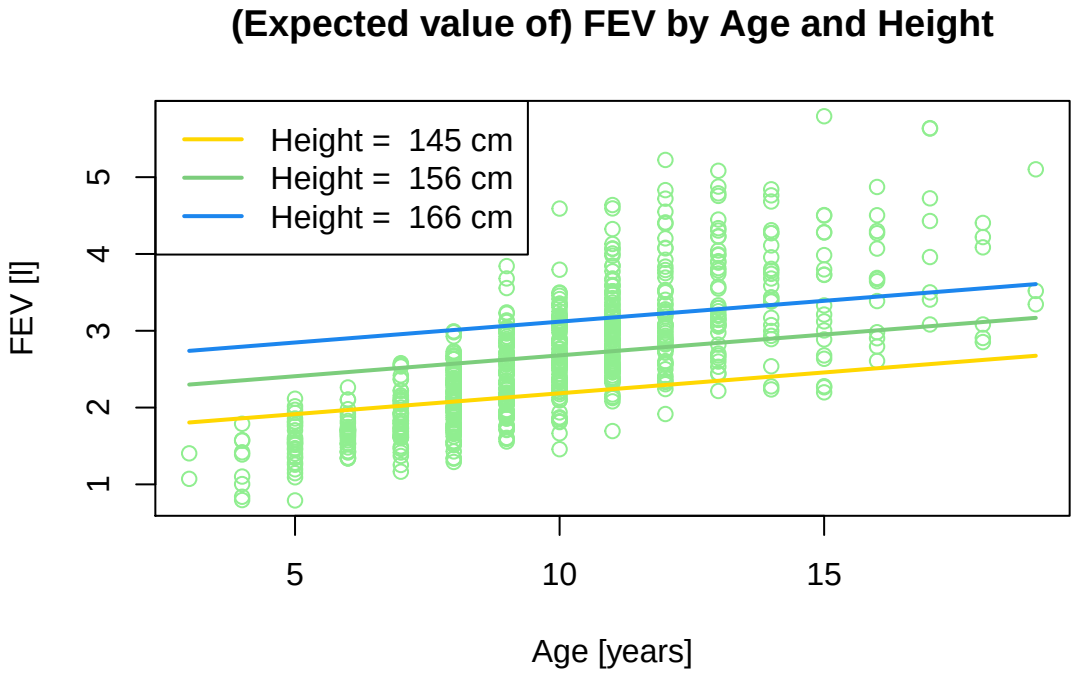
V tomto modelu je odezvou opět FEV a prediktory jsou výška Height a věk Age, tentokrát ale věk nekategorizujeme. Máme tedy $k = 2$ a maticový zápis modelu

$$\begin{pmatrix} 1.708 \\ 1.724 \\ \ldots \\ 2.795 \\ 3.211 \end{pmatrix} = \begin{pmatrix} 1 & 144.78 & 9 \\ 1 & 171.45 & 8 \\ \ldots & \ldots & \ldots \\ 1 & 160.02 & 16 \\ 1 & 168.91 & 15 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \ldots \\ \varepsilon_{653} \\ \varepsilon_{654} \end{pmatrix}.$$

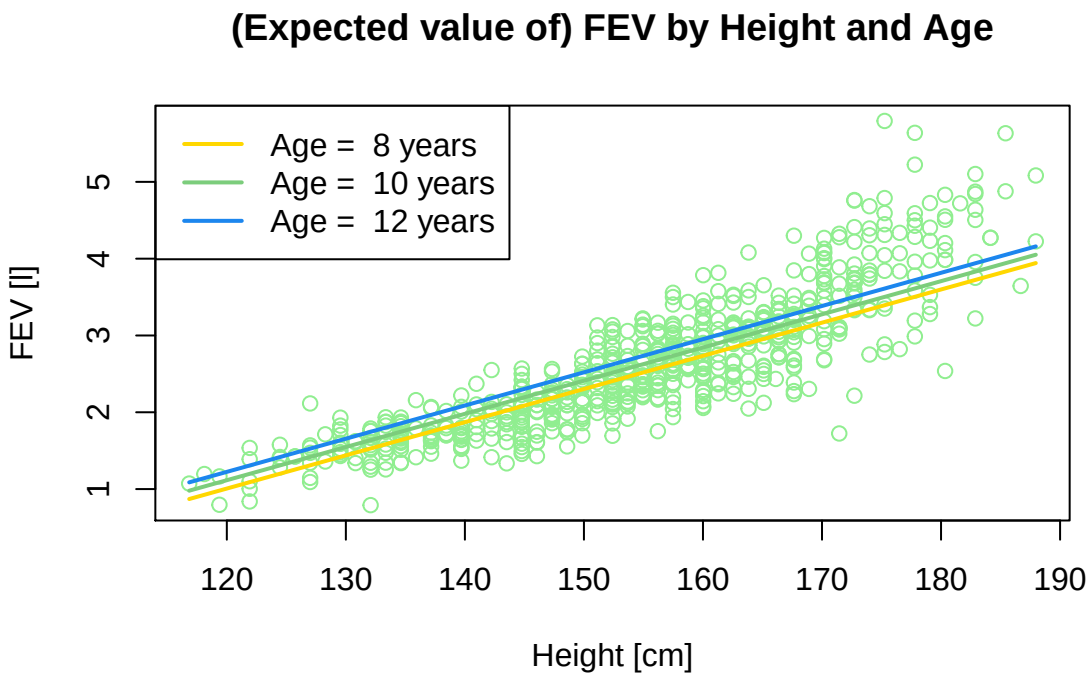
Pro lineární prediktor platí $\eta_i = \mathrm{E}(\mathsf{FEV}_i) = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Age}_i$, střední hodnota FEV podle modelu tedy lineárně roste s věkem i s výškou. Jde vlastně o rovinu, jak je vidět na obrázku níže.



Pro každou pevnou hodnotu výšky roste střední hodnota s věkem lineárně a pro různé hodnoty výšky jsou tyto přímky rovnoběžné, jak je vidět na následujícím obrázku.



Analogická tvrzení i obrázek je možné vykreslit pro závislost střední hodnoty na výšce pro různé úrovně věku.



Při interpretaci koeficientu β_1 uvažujeme dvě děti stejného věku. β_1 vyjádřuje, o kolik litrů je střední hodnota FEV vyšší u dítěte, které je o 1 cm vyšší. Při interpretaci koeficientu β_2 zase uvažujeme dvě děti stejné výšky. β_2 vyjádřuje, o kolik litrů je střední hodnota FEV vyšší u dítěte, které je o 1 rok starší. Koeficient β_0 v modelu reprezentuje střední hodnotu FEV pro dítě s nulovou výškou i věkem: v tomto modelu tedy nemá rozumnou interpretaci.

```
# Coefficient beta_1 represents the increase in E(FEV) associated
# with a 1 cm increase in Height when comparing children of the sa-
# me Age. Coefficient beta_2 represents the increase in E(FEV)
# associated with a 1 year increase in Age when comparing children
# of equal Height. beta_0 represents E(FEV) for a child aged 0 and
# with Height of 0 cm, which makes little sense.
```

Kapitola 5

Lineární model v R

V této kapitole si ukážeme, jak daný lineární model zadáme do R a jak R využijeme ke konstrukci odhadů parametrů modelu, posouzení kvality modelu i vyvození závěrů na základě modelu.

5.1 Zadání modelu do R

Připomeňme si první model z předchozí kapitoly.

(1.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \varepsilon_i, \quad i = 1, \ldots, n.$

Chceme-li tento model odhadnout pomocí R, požádáme o to následujícím příkazem.

```
> lm(FEV ~ Height, data = fev)
```

Příkazem `lm()` žádáme o odhad **l**ineárního **m**odelu. Prvním argumentem příkazu je takzvaná `formula`, ve které znakem `~` oddělujeme odezvu od prediktoru: na levou stranu píšeme odezvu (`FEV`) a na pravou stranu prediktor (`Height`). Argument `data` nám umožňuje odezvu a prediktory psát tak, aniž bychom u každého specifikovali název `data.frame`, který je obsahuje. Bez využití tohoto argumentu bychom příkaz zadali následovně.

```
> lm(fev$FEV ~ fev$Height)
```

Na oba příkazy R odpoví následujícím výstupem.

```
> lm(FEV ~ Height, data = fev)


Call:
lm(formula = FEV ~ Height, data = fev)
```

```
Coefficients:
(Intercept)       Height
   -5.43268      0.05196
```

Koeficienty β_0 a β_1 se pomocí R odhadly na -5.43 a 0.05. Odhadujeme tedy, že střední hodnota FEV je u dítěte, které je o 1 cm vyšší, vyšší o 0.05 litrů. Pro snazší orientaci v odhadech koeficientů je R vrací jako vektor, jehož složky mají jména. Odhad koeficientu odpovídajícího prediktoru pojmenuje názvem prediktoru, zatímco odhad koeficientu β_0 pojmenuje `Intercept`, jelikož se v grafu závislosti střední hodnoty odezvy na prediktoru jedná o průsečík s osou y ($y-$intercept).

```
# Estimates of beta_0 and beta_1 in the model are -5.43 and 0.05,
# respectively. We estimate that the expected value of FEV
# increases by 0.05 l with a 1 cm increase in a child's height.
```

Model s více prediktory zadáme tak, že je vyjmenujeme na pravé straně od znaku `~` argumentu `formula` a oddělíme od sebe znakem $+$ (neuvažujeme-li mezi nimi interakci). Například model

$$\text{(4.)}\quad \mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Sex}_i + \varepsilon_i, \quad i = 1, \ldots, n$$

zadáme následujícím příkazem.

```
> lm(FEV ~ Height + Sex, data = fev)


Call:
lm(formula = FEV ~ Height + Sex, data = fev)

Coefficients:
(Intercept)       Height      SexMale
   -5.39026      0.05127      0.12512
```

Koeficienty β_0, β_1 a β_2 se pomocí R odhadly na -5.39, 0.05 a 0.13. Odhadujeme tedy, že oproti dítěti stejného pohlaví je střední hodnota FEV u dítěte, které je o 1 cm vyšší, vyšší o 0.05 litrů. U chlapce odhadujeme střední hodnotu FEV o 0.13 litrů vyšší než u stejně vysoké dívky. Název `SexMale` u koeficientu β_2 se skládá ze dvou částí (které ale nejsou odděleny): název prediktoru (`Sex`) a úroveň prediktoru, která je v matici plánu kódována jedničkou (`Male`). Díky tomuto značení víme, že rozdíl ve střední hodnotě FEV o 0.13 je ve prospěch chlapců, nikoliv dívek. Bez tohoto značení by bylo bezpečnější ověřit si kódování v matici plánu pomocí příkazu `model.matrix()`, který vrátí matici plánu modelu v argumentu příkazu.

```
> model.matrix(lm(FEV ~ Height + Sex, data = fev))[1:5, ]

  (Intercept) Height SexMale
1           1 144.78       0
2           1 171.45       0
3           1 138.43       0
4           1 134.62       1
5           1 144.78       1

> fev[1:5, ]

  Age   FEV Height    Sex Smoker
1   9 1.708 144.78 Female    Non
2   8 1.724 171.45 Female    Non
3   7 1.720 138.43 Female    Non
4   9 1.558 134.62   Male    Non
5   9 1.895 144.78   Male    Non
```

```
# Estimates of beta_0, beta_1 and beta_2 in the model are -5.40,
# 0.05, and 0.13, respectively. We estimate that the expected
# value of FEV increases by 0.05 l with a 1 cm increase
# in height for children of the same sex. The difference
# between the expected values of FEV for children of the same
# height is 0.13 in favour of a boy (compared to a girl).
```

Chceme-li do modelu zahrnout interakci, oddělíme od sebe příslušné prediktory znakem $*$ místo $+$. Pro model

(5.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Sex}_i + \beta_3 \times (\mathsf{Height} \times \mathsf{Sex})_i + \varepsilon_i, \quad i = 1, \ldots, n$

bychom použili následující příkaz.

```
> lm(FEV ~ Height * Sex, data = fev)


Call:
lm(formula = FEV ~ Height * Sex, data = fev)

Coefficients:
   (Intercept)          Height         SexMale  Height:SexMale
      -4.31822         0.04426        -1.54563         0.01081
```

Koeficienty β_0, β_1, β_2 a β_3 se pomocí R odhadly na -4.32, 0.04, -1.55 a 0.01. Odhadujeme tedy, že s každým centimetrem výšky je spojený nárůst střední hodnoty FEV u dívek o 0.04 litrů a u chlapců o 0.06 litrů. Název interakce `Height:SexMale` obsahuje názvy prediktorů, mezi kterými interakci uvažujeme, a také úroveň kategoriálního prediktoru, která je v matici plánu kódována jedničkou. Oddělovací znak : se v R používá pro interakci.

```
> model.matrix(lm(FEV ~ Height * Sex, data = fev))[1:5, ]

  (Intercept) Height SexMale Height:SexMale
1           1 144.78       0           0.00
2           1 171.45       0           0.00
3           1 138.43       0           0.00
4           1 134.62       1         134.62
5           1 144.78       1         144.78
```

Stejný model můžeme zadat také explicitním vyjmenováním všech prediktorů včetně interakce oddělených znakem $+$.

```
> lm(FEV ~ Height + Sex + Height:Sex, data = fev)


Call:
lm(formula = FEV ~ Height + Sex + Height:Sex, data = fev)

Coefficients:
   (Intercept)          Height         SexMale  Height:SexMale
      -4.31822         0.04426        -1.54563         0.01081
```

Pro usnadnění zadávání modelů s interakcemi se v argumentu `formula` používá speciální syntaxe, ve které $n-$tá mocnina značí zahrnutí hlavních efektů a všech interakcí až do $n-$tého řádu. Pomocí této syntaxe můžeme výše uvedený model zadat následovně.

```
> lm(FEV ~ (Height + Sex)^2, data = fev)


Call:
lm(formula = FEV ~ (Height + Sex)^2, data = fev)

Coefficients:
   (Intercept)          Height         SexMale  Height:SexMale
      -4.31822         0.04426        -1.54563         0.01081
```

```
# Estimates of beta_0, beta_1, beta_2, and beta_3 in the model
# are -4.30, 0.04, -1.55, and 0.01, respectively. We estimate
# that a 1 cm increase in height is associated with an increase
# in the expected value of FEV by 0.04 l for girls and by 0.06 l
# for boys.
```

Pro interakce vyššího než druhého řádu bychom v modelu museli mít více proměnných. Pro tři proměnné můžeme uvažovat například následující modely s interakcemi.

```
> lm(FEV ~ (Height + Sex)^2 + Age, data = fev)


Call:
lm(formula = FEV ~ (Height + Sex)^2 + Age, data = fev)

Coefficients:
   (Intercept)          Height         SexMale             Age
      -3.09862         0.03199        -1.80829         0.06685
Height:SexMale
       0.01276

> lm(FEV ~ (Height + Sex + Age)^2, data = fev)


Call:
lm(formula = FEV ~ (Height + Sex + Age)^2, data = fev)

Coefficients:
   (Intercept)          Height         SexMale             Age
     -0.763023        0.017919       -0.876319       -0.303109
Height:SexMale      Height:Age     SexMale:Age
      0.005764        0.002256         0.010684

> lm(FEV ~ (Height + Sex + Age)^2 - Height:Age, data = fev)


Call:
lm(formula = FEV ~ (Height + Sex + Age)^2 - Height:Age, data = fev)

Coefficients:
   (Intercept)          Height         SexMale             Age
     -3.329648        0.034311       -1.347481         0.054187
Height:SexMale     SexMale:Age
      0.007958        0.028716
```

```
> lm(FEV ~ (Height + Age + Sex)^3, data = fev)


Call:
lm(formula = FEV ~ (Height + Age + Sex)^3, data = fev)

Coefficients:
       (Intercept)              Height                 Age
         -3.380e+00           3.464e-02           6.126e-02
           SexMale          Height:Age      Height:SexMale
          3.460e+00          -4.464e-05          -2.147e-02
      Age:SexMale  Height:Age:SexMale
         -5.661e-01           3.577e-03
```

Chceme-li jako prediktory využít funkce stávajících prediktorů, můžeme hodnoty těchto nových prediktorů nejprve spočítat a přidat jako další sloupec do příslušného `data.frame`. Ten pak zadáme do argumentu `formula` příkazu `lm()`. Tak bychom postupovali například u modelu

$$\text{FEV}_i = \beta_0 + \beta_1 \times \text{Height}_i + \beta_2 \times \mathbb{I}\{9 < \text{Age}_i \leq 12\} + \beta_3 \times \mathbb{I}\{\text{Age}_i > 12\} + \varepsilon_i, \quad i = 1, \ldots, n. \tag{6.}$$

```
> quantile(fev$Age, probs = c(0:3)/3)

       0% 33.33333% 66.66667%      100%
        3         9        11        19

> fev$Age.cat <- cut(fev$Age, breaks = c(3, 9, 12, 19),
+                    include.lowest = TRUE)
> summary(fev$Age.cat)

  [3,9]  (9,12] (12,19]
    309     228     117

> lm(FEV ~ Height + Age.cat, data = fev)


Call:
lm(formula = FEV ~ Height + Age.cat, data = fev)

Coefficients:
   (Intercept)          Height   Age.cat(9,12]  Age.cat(12,19]
      -4.50925         0.04525         0.12235         0.42566
```

U jednodušších funkcí prediktoru je výhodnější zadat požadovanou funkci přímo uvnitř agrumentu `formula`. V takovém případě je potřeba nový prediktor zadat uvnitř funkce `I()`. Například model

$$\text{FEV}_i = \beta_0 + \beta_1 \times \text{Height}_i + \beta_2 \times \text{Height}_i^2 + \varepsilon_i, \quad i = 1, \ldots, n \tag{2.}$$

zadáme následujícím příkazem.

```
> lm(FEV ~ Height + I(Height^2), data = fev)


Call:
lm(formula = FEV ~ Height + I(Height^2), data = fev)

Coefficients:
(Intercept)       Height  I(Height^2)
  6.0268779   -0.0984444    0.0004891
```

Nakonec si ještě ukážeme zjednodušenou syntaxi pro dva speciální modely. Model bez prediktorů

$$\text{FEV}_i = \beta_0 + \varepsilon_i, \quad i = 1, \ldots, n$$

zadáme následujícím příkazem.

```
> lm(FEV ~ 1, data = fev)


Call:
lm(formula = FEV ~ 1, data = fev)

Coefficients:
(Intercept)
      2.637
```

Naopak model, který jako prediktory obsahuje všechny sloupce `data.frame`, ze kterého pochází odezva, zadáme následovně.

```
> lm(FEV ~ ., data = fev)


Call:
lm(formula = FEV ~ ., data = fev)

Coefficients:
```

```
   (Intercept)             Age          Height         SexMale
      -4.46536         0.01858         0.04253         0.14928
     SmokerNon  Age.cat(9,12]  Age.cat(12,19]
       0.13668         0.11516         0.40769
```

5.2 Celkové charakteristiky modelu

Vraťme se nyní k modelu

(2.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Sex}_i + \varepsilon_i, \quad i = 1, \ldots, n.$

Již jsme viděli, že příkaz `lm()` vrátí odhady koeficientů modelu. Zároveň s odhady spočítá mnoho charakteristik užitečných pro inferenci. Abychom k nim mohli lépe přistupovat, uložíme si vše, co se po zadání příkazu `lm()` napočítalo, například do proměnné `mod`.

```
> mod <- lm(FEV ~ Height + Sex, data = fev)
```

Funkci `model.matrix()`, popsanou výše, nyní můžeme uplatnit přímo na proměnnou `mod`. Funkce `coefficients()` uplatněna na proměnnou `mod` vrátí odhady koeficientů modelu.

```
> model.matrix(mod)[1:5, ]

  (Intercept) Height SexMale
1           1 144.78       0
2           1 171.45       0
3           1 138.43       0
4           1 134.62       1
5           1 144.78       1

> coefficients(mod)

(Intercept)      Height     SexMale
-5.39026319  0.05127185  0.12512339
```

Přehled nejdůležitějších charakteristik, které lze získat z proměnné `mod`, poskytuje příkaz `summary()`.

```
> summary(mod)


Call:
lm(formula = FEV ~ Height + Sex, data = fev)
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-1.6763 -0.2505  0.0001  0.2347  2.0722

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.390263   0.180082 -29.932  < 2e-16 ***
Height       0.051272   0.001167  43.933  < 2e-16 ***
SexMale      0.125123   0.033801   3.702 0.000232 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4265 on 651 degrees of freedom
Multiple R-squared:  0.7587,Adjusted R-squared:  0.758
F-statistic:  1024 on 2 and 651 DF,  p-value: < 2.2e-16
```

První řádek výstupu obsahuje definici modelu. Následují empirické kvantily reziduí $\mathbf{e} = \mathbf{Y} - \mathbf{X}\widehat{\boldsymbol{\beta}}$. Také poslední tři řádky se týkají celého modelu. Obsahují odhad $\hat{\sigma}$ odmocniny z rozptylu σ^2 náhodné chyby $\boldsymbol{\varepsilon}$, koeficienty determinace a test hypotézy o nulovosti všech koeficientů modelu současně.

Odhad $\hat{\sigma}$ je odmocninou z nevychýleného odhadu $\widehat{\sigma^2} = \mathbf{e}^\top\mathbf{e}/(n-p)$. V normálním lineárním modelu má náhodná veličina $(n-p)\widehat{\sigma^2}/\sigma^2$ rozdělení χ^2 s $n-p$ stupni volnosti. Ve výstupu z fuknce `summary()` jsou tyto stupně volnosti uvedeny za odhadem $\hat{\sigma}$. Jedná se o počet pozorování snížený o rozměr vektoru $\boldsymbol{\beta}$, jak snadno ověříme.

```
> nrow(fev) - length(coefficients(mod))   # n-p

[1] 651
```

Koeficient determinace

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2}, \quad \text{kde } \bar{Y} = \sum_{i=1}^n Y_i,$$

reprezentuje proporci variability dat vysvětlenou modelem. Náš model vysvětluje 76 % variability proměnné FEV. Koeficient determinace vzroste pokaždé, když do modelu přidáme prediktor, proto je pro srovnání vnořených modelů vhodnější *adjustovaný (upravený) koeficient determinace*

$$R^2_{\text{adj}} = 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2/(n-p)}{\sum_{i=1}^n (Y_i - \bar{Y})^2/(n-1)}, \tag{5.1}$$

který bere do úvahy i počet prediktorů v modelu. V našem případě je téměř stejný jako koeficient determinace, $R^2_{\text{adj}} = 0.76$. Při velkém počtu dat a malém počtu koeficientů v modelu jsou si totiž hodnoty dělitelů $n-p$ a $n-1$ velmi blízké.

Na posledním řádku máme pozorovanou hodnotu testové statistiky $F-$testu hypotézy $H_0 : (\beta_1, \beta_2)^\top = \mathbf{0}$ proti alternativě $H_1 : (\beta_1, \beta_2)^\top \neq \mathbf{0}$. Stupně volnosti příslušného $F-$rozdělení jsou rozměr testovaného vektoru a stupně volnosti χ^2-rozdělení odhadu rozptylu σ^2 náhodné chyby $\boldsymbol{\varepsilon}$, tedy $p-1$ a $n-p$. Nezamítnutí nulové hypotézy sice neznamená její potvrzení, ale vyvolávalo by pochybnosti o vhodnosti prediktorů pro popis střední hodnoty odezvy. V našem případě však nulovou hypotézu jednoznačně zamítáme ($p-$hodnota $< 10^{-15}$), tudíž tento výsledek pochybnosti o vhodnosti prediktorů nevyvolává.

Pro přístup k jednotlivým výsledkům zobrazeným ve výstupu ze `summary(mod)` potřebujeme znát jejich interní názvy. Ty můžeme najít ve výstupu z funkce `str()` uplatněné na objekt `summary(mod)`.

```
> str(summary(mod))

List of 11
 $ call         : language lm(formula = FEV ~ Height + Sex, data = fev)
 $ terms        :Classes 'terms', 'formula'  language FEV ~ Height + Sex
  .. ..- attr(*, "variables")= language list(FEV, Height, Sex)
  .. ..- attr(*, "factors")= int [1:3, 1:2] 0 1 0 0 0 1
  .. .. ..- attr(*, "dimnames")=List of 2
  .. .. .. ..$ : chr [1:3] "FEV" "Height" "Sex"
  .. .. .. ..$ : chr [1:2] "Height" "Sex"
  .. ..- attr(*, "term.labels")= chr [1:2] "Height" "Sex"
  .. ..- attr(*, "order")= int [1:2] 1 1
  .. ..- attr(*, "intercept")= int 1
  .. ..- attr(*, "response")= int 1
  .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
  .. ..- attr(*, "predvars")= language list(FEV, Height, Sex)
  .. ..- attr(*, "dataClasses")= Named chr [1:3] "numeric" "numeric" "factor"
  .. .. ..- attr(*, "names")= chr [1:3] "FEV" "Height" "Sex"
 $ residuals    : Named num [1:654] -0.3249 -1.6763 0.0127 -0.0791 -0.263 ...
  ..- attr(*, "names")= chr [1:654] "1" "2" "3" "4" ...
 $ coefficients : num [1:3, 1:4] -5.39026 0.05127 0.12512 0.18008 0.00117 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:3] "(Intercept)" "Height" "SexMale"
  .. ..$ : chr [1:4] "Estimate" "Std. Error" "t value" "Pr(>|t|)"
 $ aliased      : Named logi [1:3] FALSE FALSE FALSE
  ..- attr(*, "names")= chr [1:3] "(Intercept)" "Height" "SexMale"
 $ sigma        : num 0.427
 $ df           : int [1:3] 3 651 3
 $ r.squared    : num 0.759
 $ adj.r.squared: num 0.758
 $ fstatistic   : Named num [1:3] 1024 2 651
  ..- attr(*, "names")= chr [1:3] "value" "numdf" "dendf"
 $ cov.unscaled : num [1:3, 1:3] 1.78e-01 -1.14e-03 2.13e-03 -1.14e-03 7.49e-06 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:3] "(Intercept)" "Height" "SexMale"
  .. ..$ : chr [1:3] "(Intercept)" "Height" "SexMale"
 - attr(*, "class")= chr "summary.lm"
```

Z výstupu výše vidíme, že například $\hat{\sigma}$, R^2 a R^2_{adj} získáme pomocí následujících příkazů.

```
> summary(mod)$sigma  # estimate of sigma

[1] 0.4265405

> summary(mod)$r.squared  # coefficient of determination (R^2)

[1] 0.7587368

> summary(mod)$adj.r.squared  # adjusted R^2

[1] 0.7579956
```

5.3 Inference pro jednotlivé koeficienty

Tabulka uprostřed výstupu ze `summary(mod)` obsahuje informace užitečné pro posouzení statistické významnosti a výpočet konfidenčních intervalů pro jednotlivé koeficienty modelu. Uložíme si ji do samostatné proměnné.

```
> tab <- summary(mod)$coefficients
> tab

               Estimate  Std. Error    t value      Pr(>|t|)
(Intercept) -5.39026319 0.180082479 -29.932191 1.821608e-124
Height       0.05127185 0.001167051  43.932823 6.861183e-197
SexMale      0.12512339 0.033800905   3.701776  2.322264e-04
```

Jedná se o matici s pojmenovanými řádky i sloupci.

První sloupec (`Estimate`) obsahuje odhady koeficientů.

```
> tab[, 1]

(Intercept)      Height     SexMale
-5.39026319  0.05127185  0.12512339

> coefficients(mod)

(Intercept)      Height     SexMale
-5.39026319  0.05127185  0.12512339
```

Druhý sloupec (`Std. Error`) obsahuje odmocniny z odhadnutých rozptylů odhadů jednotlivých koeficientů. Z teorie víme, že

$$\widehat{\operatorname{Var}\,\widehat{\boldsymbol{\beta}}} = \widehat{\sigma^2}\,(\mathbf{X}^\top\mathbf{X})^{-1}.$$

```
> tab[, 2]

(Intercept)      Height     SexMale
0.180082479 0.001167051 0.033800905

> X <- model.matrix(mod)
> summary(mod)$sigma * sqrt(diag(solve(t(X) %*% X)))

(Intercept)      Height     SexMale
0.180082479 0.001167051 0.033800905
```

Třetí sloupec matice (`t value`) obsahuje pozorované hodnoty testových statistik pro testy hypotéz $H_0 : \beta_i = 0$ pro jednotlivé složky vektoru koeficientů proti alternativám $H_1 : \beta_i \neq 0$. Testové statistiky jsou

$$T_i = \frac{\hat{\beta}_i}{\sqrt{\widehat{\sigma^2}\,(\mathbf{X}^\top\mathbf{X})^{-1}_{i,i}}},$$

jedná se tedy o podíl prvního a druhého sloupce matice `tab`.

```
> tab[, 3]

(Intercept)      Height     SexMale
 -29.932191   43.932823    3.701776

> tab[, 1]/tab[, 2]

(Intercept)      Height     SexMale
 -29.932191   43.932823    3.701776
```

V normálním lineárním modelu má každá z těchto testových statistik za platnosti příslušné nulové hypotézy $t-$rozdělení s $n - p$ stupni volnosti, $T_i \sim t_{n-p}$. Příslušné $p-$hodnoty najdeme v posledním sloupci matice (`Pr (> |t|)`). Pro jejich výpočet můžeme využít toho, že $p-$hodnota představuje pravděpodobnost, že za platnosti nulové hypotézy má testová statistika hodnotu, kterou jsme pozorovali, nebo hodnoty, které by ještě více svědčily ve prospěch alternativy. V našem případě ve prospěch alternativy svědčí hodnoty testové statistiky vzdálenější od nuly, a to na kladnou i zápornou stranu.

```
> tab[, 4]

  (Intercept)        Height       SexMale
1.821608e-124 6.861183e-197  2.322264e-04
```

```
> 2 * pt(-abs(tab[, 3]), df = summary(mod)$df[2])

  (Intercept)         Height        SexMale
1.821608e-124 6.861183e-197  2.322264e-04
```

Invertováním těchto $t-$testů získáme konfidenční intervaly pro jednotlivé koeficienty

$$\left(\hat{\beta}_i - t_{n-p}(1-\alpha/2)\sqrt{\widehat{\sigma^2}\,(\mathbf{X}^\top\mathbf{X})^{-1}_{i,i}}\ ,\ \hat{\beta}_i + t_{n-p}(1-\alpha/2)\sqrt{\widehat{\sigma^2}\,(\mathbf{X}^\top\mathbf{X})^{-1}_{i,i}}\right).$$

V R je vrací funkce `confint()`.

```
> confint(mod)

                  2.5 %      97.5 %
(Intercept) -5.74387579 -5.03665059
Height       0.04898022  0.05356349
SexMale      0.05875144  0.19149534

> alpha <- 0.05
> df <- summary(mod)$df[2]
> lwr <- tab[, 1] - qt(1 - alpha/2, df = df) * tab[, 2]
> upr <- tab[, 1] + qt(1 - alpha/2, df = df) * tab[, 2]
> data.frame(lwr, upr)

                  lwr         upr
(Intercept) -5.74387579 -5.03665059
Height       0.04898022  0.05356349
SexMale      0.05875144  0.19149534
```

Připomeňme si, že hustota $t-$rozdělení je podobná hustotě standardního normálního rozdělení a s rostoucími stupni volnosti se jí blíží, avšak $t-$rozdělení má těžší chvosty. Kvantily $t-$rozdělení se proto s rostoucími stupni volnosti blíží kvantilům standardního normálního rozdělení, v absolutní hodnotě jsou ale o něco větší. Nemáme-li příliš malý rozsah dat, pro rychlou představu o konfidenčních intervalech díky tomu stačí vzít první sloupec matice a odečíst (resp. přičíst) k němu dvojnásobek ($u_{0.975} \approx 1.96$) druhého sloupce.

Každý z výše uvedených $t-$testů a konfidenčních intervalů se týká jediného koeficientu. Jsou-li splněny předpoklady normálního lineárního modelu, každý z těchto testů má hladinu významnosti α a každý z těchto konfidenčních intervalů má pravděpodobnost pokrytí $1-\alpha$. Pravděpodobnost, že všechny konfidenční intervaly pokryjí odhadované parametry, může být nižší a společná hladina všech testů může být vyšší. Chceme-li kontrolovat hladinu významnosti testu o více koeficientech současně, použijeme místo jednotlivých $t-$testů společný $F-$test, kterému se budeme podrobněji věnovat v části 5.5. Zajímá-li nás společné pokrytí konfidenčních intervalů, nahradíme je konfidenčními regiony nebo konfidenčními pásy, kterým se budeme věnovat v části 5.6.

5.4 Inference pro lineární kombinace koeficientů

Někdy se můžeme vedle jednotlivých koeficientů zajímat také o jejich lineární kombinace. Například v modelu

$$\text{FEV}_i = \beta_0 + \beta_1 \times \text{Height}_i + \beta_2 \times \text{Sex}_i + \beta_3 \times (\text{Sex} \times \text{Height})_i + \varepsilon_i, \quad i = 1, \ldots, n \tag{5.}$$

vyjádřuje koeficient β_1 nárůst střední hodnoty FEV spojený s nárůstem výšky u dívek o 1 cm. Pro chlapce má stejnou interpretaci součet $\beta_1 + \beta_3$.

```
> mod <- lm(FEV ~ Height * Sex, data = fev)
> summary(mod)


Call:
lm(formula = FEV ~ Height * Sex, data = fev)

Residuals:
     Min       1Q   Median       3Q      Max
-1.54654 -0.25282  0.00649  0.25666  2.00491

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    -4.318219   0.297637 -14.508  < 2e-16 ***
Height          0.044262   0.001940  22.815  < 2e-16 ***
SexMale        -1.545629   0.373843  -4.134 4.02e-05 ***
Height:SexMale  0.010810   0.002409   4.487 8.54e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4204 on 650 degrees of freedom
Multiple R-squared:  0.766,Adjusted R-squared:  0.7649
F-statistic: 709.2 on 3 and 650 DF,  p-value: < 2.2e-16
```

Tento efekt odhadneme jako $\hat{\beta}_1 + \hat{\beta}_3 = 0.06$. Odhad rozptylu $\text{Var}(\beta_1 + \beta_3) = \text{Var}\,\beta_1 + \text{Var}\,\beta_3 + 2 \times \text{Cov}(\beta_1, \beta_3)$, který bychom potřebovali ke konstrukci testu nebo konfidenčního intervalu pro $\beta_1 + \beta_3$, ale ve výstupu z funkce `summary()` nenajdeme. Za tímto účelem použijeme funkci `vcov()`, která vrací odhad $\widehat{\sigma^2}\,(\mathbf{X}^\top\mathbf{X})^{-1}$ kovarianční matice odhadů koeficientů v modelu, který je jejím argumentem.

```
> est.var <- vcov(mod)
> est.var
```

```
                (Intercept)        Height       SexMale
(Intercept)     0.088588026 -5.756060e-04 -0.0885880258
Height         -0.000575606  3.763647e-06  0.0005756060
SexMale        -0.088588026  5.756060e-04  0.1397583375
Height:SexMale  0.000575606 -3.763647e-06 -0.0008970667
                Height:SexMale
(Intercept)      5.756060e-04
Height          -3.763647e-06
SexMale         -8.970667e-04
Height:SexMale   5.804094e-06
```

Nyní již můžeme zkonstruovat testovou statistiku pro test hypotézy $H_0 : \beta_1 + \beta_3 = 0$ proti alternativě $H_1 : \beta_1 + \beta_3 \neq 0$ i konfidenční interval pro $\beta_1 + \beta_3$.

```
> est <- sum(coefficients(mod)[c(2,4)])
> est.var.sqrt <- sqrt(sum(diag(est.var)[c(2, 4)])
+                      + 2 * est.var[2, 4])
> c(est, est.var.sqrt) #Estimate and Std.Error for beta_1+beta_3

[1] 0.055072110 0.001428442

> test.stat <- est/est.var.sqrt
> test.p <- 2 * pt(-abs(test.stat), df = summary(mod)$df[2])
> c(test.stat, test.p) # t value and P(>|t|) for beta_1 + beta_3

[1]  3.855396e+01 4.200169e-170

> ci <- est + c(-1, 1) * qnorm(1-alpha/2) * est.var.sqrt
> ci # predidence interval for beta_1 + beta_3

[1] 0.05227242 0.05787181
```

Obecněji, pro test o lineární kombinaci $\mathbf{a}^\top\boldsymbol{\beta} = a_0\beta_0 + a_1\beta_1 + \ldots + a_k\beta_k$ s $H_0 : \mathbf{a}^\top\boldsymbol{\beta} = 0$ a $H_1 : \mathbf{a}^\top\boldsymbol{\beta} \neq 0$, použijeme testovou statistiku

$$T = \frac{\mathbf{a}^\top\widehat{\boldsymbol{\beta}}}{\widehat{\sigma^2}\,\mathbf{a}^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{a}}, \tag{5.2}$$

která má za platnosti normálního lineárního modelu a nulové hypotézy $t-$rozdělení s $n-p$ stupni volnosti. Invertováním testu dostaneme konfidenční interval pro $\mathbf{a}^\top\boldsymbol{\beta}$. V kódu výše bychom změnili jenom výpočet odhadu a odmocniny z jeho odhadnutého rozptylu.

```
> est <- t(a) %*% coefficients(mod)
> est.var.sqrt <- sqrt(t(a) %*% est.var %*% a)
```

5.5 Inference pro více (lineárních kombinací) koeficientů

Souvislost mezi prediktorem výška Height a střední hodnotou odezvy FEV v modelu (5.) prakticky popisují směrnice β_1 a $\beta_1 + \beta_3$. Kdyby výsledky jednorozměrných testů o jejich nulovosti popsané v částech 5.3 a 5.4 nebyly tak jednoznačné, bylo by pro vyčerpávající odpověd na otázku o souvislosti výšky Height se střední hodnotou FEV potřeba testovat nulovost obou směrnic zároveň. Za tímto účelem můžeme využít $F-$test pro testování hypotéz o několika nezávislých lineárních kombinacích koeficientů modelu zároveň. Tvoří-li koeficienty m nezávislých lineárních kombinací řádky matice $\mathbf{A}$, testová statistika pro test hypotézy $H_0 : \mathbf{A}\boldsymbol{\beta} = \mathbf{0}$ proti alternativě $H_1 : \mathbf{A}\boldsymbol{\beta} \neq \mathbf{0}$ má tvar

$$F = \frac{1}{m\,\widehat{\sigma^2}} (\mathbf{A}\widehat{\boldsymbol{\beta}})^\top (\mathbf{A}\,(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{A}^\top)^{-1}\mathbf{A}\widehat{\boldsymbol{\beta}} \tag{5.3}$$

a za platnosti normálního lineárního modelu a nulové hypotézy má $F-$rozdělení s m a $n-p$ stupni volnosti. Následujícím způsobem pak můžeme najednou provést test o nulovosti obou směrnic odpovídajících výšce Height.

```
> A <- rbind(c(0, 1, 0, 0), c(0, 1, 0, 1))
> beta.hat <- coefficients(mod)
> var.beta.hat <- vcov(mod)
> est <- A %*% beta.hat
> est # estimate of the vector (beta_1, beta_1 + beta_3)

           [,1]
[1,] 0.04426220
[2,] 0.05507211

> F <- t(est) %*% solve(A %*% var.beta.hat %*% t(A)) %*% est/2
> p <- 1-pf(F, df1 = 2, df2 = summary(mod)$df[2])
> # test statistic and p-value for testing the hypothesis
> # that both beta_1 = 0 and beta_1 + beta_3 = 0
> c(F, p)

[1] 1003.476    0.000
```

Vzhledem k jednoznačným výsledkům jednotlivých $t-$testů není překvapivé, že také výsledek $F-$testu je jednoznačný. Obecně ale nemusí $F-$test vést ke stejnému výsledku jako jednotlivé $t-$testy. Připomeňme ale, že použijeme-li $F-$test (5.3) k testování hypotézy o jediné lineární

kombinaci koeficientů modelu, jeho testová statistika je kvadrátem testové statistiky $t-$testu z části 5.4. Jelikož kvadrát náhodné veličiny s $t-$rozdělením s n stupni volnosti má $F-$rozdělení s 1 a n stupni volnosti, výsledky testů budou v takovém případě stejné.

Nulovou hypotézu $H_0 : \beta_1 = 0$ a $\beta_1 + \beta_3 = 0$ můžeme ekvivalentně zapsat jako $H_0 : \beta_1 = 0$ a $\beta_3 = 0$. Za její platnosti se tedy model (5.) redukuje na model

(3.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Sex}_i + \varepsilon_i, \quad i = 1, \ldots, n.$

Test hypotézy $H_0 : \beta_1 = 0$ a $\beta_1 + \beta_3 = 0$ proti alternativě $H_1 : \beta_1 \neq 0$ nebo $\beta_1 + \beta_3 \neq 0$ proto můžeme vnímat i jako test o možnosti přechodu od modelu (5.) k modelu (3.). Takovým testům se budeme věnovat v části 6.1.

5.6 Odhady střední hodnoty a predikce na základě modelu

Důležitým speciálním případem lineárních kombinací koeficientů diskutovaných v části 5.4 jsou kombinace odpovídající konkrétním hodnotám prediktorů. Uvažujme opět model (5.) a počítejme odhad střední hodnoty FEV pro dívku vysokou 110 cm. Podle modelu (5.) platí, že

$$\mathsf{FEV} = \beta_0 + \beta_1 \times \mathsf{Height} + \beta_2 \times \mathsf{Sex} + \beta_3 \times (\mathsf{Sex} \times \mathsf{Height}) + \varepsilon$$

a $\mathrm{E}\,\varepsilon = 0$. Pro střední hodnotu FEV pak platí vztah

$$\mathrm{E}(\mathsf{FEV}) = \beta_0 + \beta_1 \times \mathsf{Height} + \beta_2 \times \mathsf{Sex} + \beta_3 \times (\mathsf{Sex} \times \mathsf{Height})\,.$$

Připomeňme, že dívky jsou v proměnné Sex kódovány nulou. Pro střední hodnotu FEV dívky vysoké 110 cm tak dostáváme vztah

$$\mathrm{E}(\mathsf{FEV} \mid \mathsf{Sex} = 0, \mathsf{Height} = 110) = \beta_0 + \beta_1 \times 110 + \beta_2 \times 0 + \beta_3 \times 0 = \beta_0 + 110\,\beta_1.$$

Pro lepší viditelnost hodnot prediktorů, pro které počítáme střední hodnotu odezvy, jsme v rovnici výše použili značení, které je obvyklejší v situacích, kdy jsou prediktory považovány za náhodné veličiny, na jejichž konkrétních hodnotách podmiňujeme.

Střední hodnota odezvy za konkrétních hodnot prediktorů, v našem případě $\mathrm{E}(\mathsf{FEV} \mid \mathsf{Sex} = 0, \mathsf{Height} = 110)$, je tedy lineární kombinací koeficientů modelu. V našem případě tvoří její koeficienty vektor $\mathbf{a} = (1, 110, 0, 0)^\top$. V tomto kontextu se pro koeficienty lineární kombinace používá spíše značení $\mathbf{x} = (1, 110, 0, 0)^\top$ připomínající (generický) řádek regresní matice.

Jak jsme si již rozmysleli v části 5.4, lineární kombinaci $\mathbf{x}^\top\boldsymbol{\beta}$ můžeme odhadnout pomocí $\mathbf{x}^\top\widehat{\boldsymbol{\beta}}$. V našem případě tedy

$$\hat{\mathrm{E}}(\mathsf{FEV} \mid \mathsf{Sex} = 0, \mathsf{Height} = 110) = \hat{\beta}_0 + 110 \times \hat{\beta}_1 = -4.32 + 110 \times 0.04 \approx 0.55.$$

Pro testování hypotézy o konkrétní hodnotě $\mathrm{E}(\mathsf{FEV} \mid \mathsf{Sex} = 0, \mathsf{Height} = 110)$ můžeme použít testovou statistiku (5.2). Jejím invertováním pak dostaneme konfidenční interval pro $\mathrm{E}(\mathsf{FEV} \mid \mathsf{Sex} = 0, \mathsf{Height} = 110)$, jak jsme již také viděli v části 5.4.

```
> # Expected value of FEV for a 110 cm tall girl: beta_0 + 110 *
> # beta_2
> x <- c(1, 110, 0, 0)
> est <- t(x) %*% beta.hat
> est.var.sqrt <- sqrt(t(x) %*% var.beta.hat %*% x)
> # Estimate and Std.Error for beta_0 + 110 * beta_2
> c(est, est.var.sqrt)

[1] 0.55062368 0.08657273

> test.stat <- est/est.var.sqrt
> test.p <- 2 * pt(-abs(test.stat), df = summary(mod)$df[2])
> # t value and P(>|t|) for beta_0 + 110 * beta_2
> c(test.stat, test.p)

[1] 6.360244e+00 3.798466e-10

> ci <- est + c(-1, 1) * qnorm(1 - alpha/2) * est.var.sqrt
> # confidence interval for beta_0 + 110 * beta_2
> ci

[1] 0.3809442 0.7203031
```

Související úlohou je *predikce* hodnoty nového pozorování při konkrétních hodnotách prediktorů. Nové pozorování je náhodná veličina, proto jej nemůžeme *odhadovat*, ale můžeme jej *predikovat*. Pro dívku vysokou 110 cm je podle modelu (5.)

$$\mathsf{FEV} = \beta_0 + \beta_1 \times 110 + \beta_2 \times 0 + \beta_3 \times 0 + \varepsilon = \beta_0 + 110\,\beta_1 + \varepsilon.$$

Odhadneme-li ve vztahu výše koeficienty β_0 a β_1 a predikujeme-li ε jeho střední hodnotou, dostaneme predikci FEV pro dívku vysokou 110 cm ve tvaru

$$(\widehat{\mathsf{FEV}} \,|\, \mathsf{Height} = 110, \mathsf{Sex} = 0) = \hat\beta_0 + 110\,\hat\beta_1,$$

predikce tedy bude stejná jako odhad příslušné střední hodnoty. Rozdíl ale nastane při tvorbě *predikčního intervalu*. Tam musíme vedle variability plynoucí z nahrazení koeficientů v lineární kombinaci jejími odhady vzít do úvahy také variabilitu plynoucí z nahrazení náhodné chyby její střední hodnotou. Celkový rozptyl tedy nebude $\sigma^2\mathbf{x}^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x}$, nýbrž $\sigma^2\mathbf{x}^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x} + \sigma^2$. Predikční interval pak bude

$$\left(\mathbf{x}^\top\widehat{\boldsymbol\beta} - t_{1-\alpha/2}(n-p)\sqrt{\widehat{\sigma^2}\,(1 + \mathbf{x}^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x})}\,,\right.$$
$$\left.\mathbf{x}^\top\widehat{\boldsymbol\beta} + t_{1-\alpha/2}(n-p)\sqrt{\widehat{\sigma^2}\,(1 + \mathbf{x}^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x})}\right).$$

```
> # Prediction interval for FEV for a 110 cm tall girl
> est.var.sqrt <- sqrt(t(x) %*% var.beta.hat %*% x +
+                          (summary(mod)$sigma)^2)
> pi <- est + c(-1, 1) * qnorm(1 - alpha/2) * est.var.sqrt
> pi

[1] -0.2906489  1.3918962
```

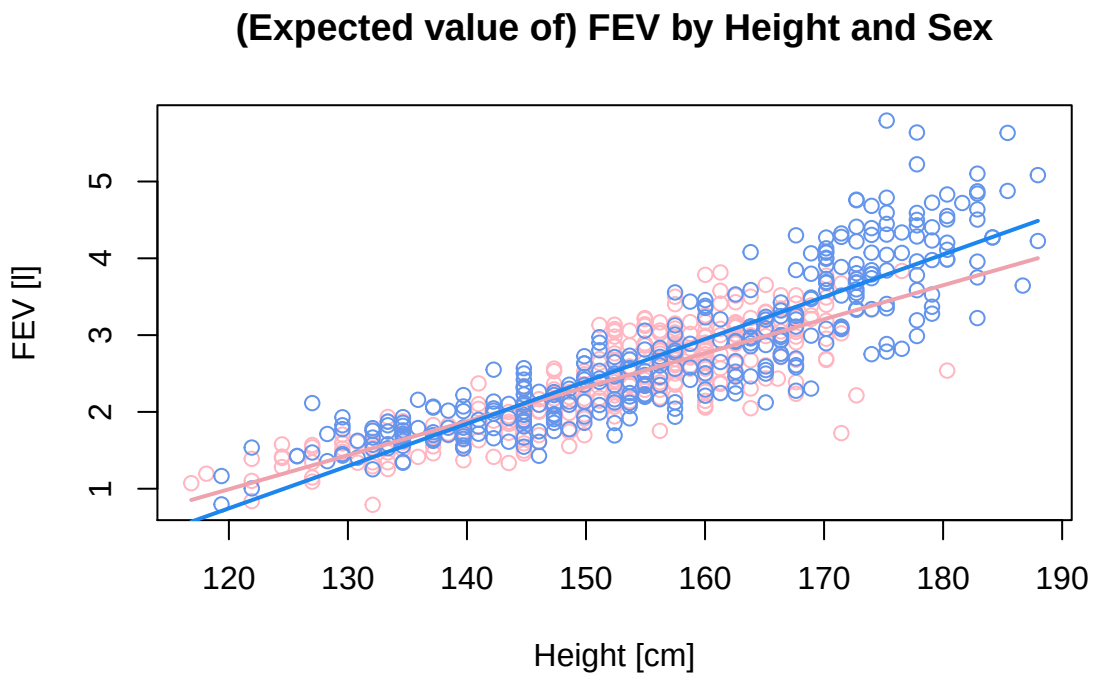
Jelikož záporné hodnoty FEV nedávají smysl, můžeme predikční interval v této sitaci omezit na kladná čísla.

K výpočtu odhadu střední hodnoty i predikce odezvy pro dané hodnoty prediktorů a odpovídajících intervalů můžeme v R využít funkce `predict()`. Jejími argumenty jsou model, na základě kterého chceme odhadovat nebo predikovat, a hodnoty prediktorů, pro které chceme odhady nebo predikce odezvy počítat (ty vkládáme jako `data.frame`, jehož sloupce mají stejné názvy jako prediktory v modelu). Volbou `interval = "confidence"` dále požádáme o konfidenční intervaly, zatímco volbou `interval = "prediction"` požádáme o predikční intervaly. Nepožádáme-li o interval, funkce vrátí jenom odhad střední hodnoty/predikci.

```
> # estimate and confidence interval for the expected value
> # of FEV for a 110 cm tall girl
> predict(mod,
+         newdata = data.frame(Height = 110, Sex = "Female"),
+         interval = "confidence")

        fit       lwr       upr
1 0.5506237 0.3806277 0.7206196

> # prediction and prediction interval for FEV for a 110 cm
> # tall girl
> predict(mod,
+         newdata = data.frame(Height = 110, Sex = "Female"),
+         interval = "prediction")

        fit        lwr      upr
1 0.5506237 -0.2922183 1.393466

> # estimated expected value/prediction of FEV for a 110 cm
> # tall girl
> predict(mod,
+         newdata = data.frame(Height = 110, Sex = "Female"))

        1
0.5506237
```

V argumentu `newdata` funkce `predict()` můžeme zadat `data.frame` s více řádky, což se hodí u vykreslování odhadů nebo predikcí na základě modelu jako funkcí prediktorů.

```
> xx <- seq(from = min(fev$Height), to = max(fev$Height),
+           length = 501)
> # a grid o heights for which to compute estimates/predictions
>
> yy1 <- predict(mod,
+                newdata = data.frame(Height = xx,
+                                     Sex = "Female"))
> # estimates/predictions for girls on the grid of heights
> yy2 <- predict(mod,
+                newdata = data.frame(Height = xx, Sex = "Male"))
> # estimates/predictions for boys on the grid of heights
>
> plot(fev$FEV ~ fev$Height,
+      main = "(Expected value of) FEV by Height and Sex",
+      xlab = "Height [cm]", ylab = "FEV [l]", type = "n")
> # (empty) plot of FEV against Height
> with(fev,
+      points(FEV[Sex == "Female"] ~ Height[Sex == "Female"],
+             col = "lightpink"))
> # pink points for girls
> with(fev,
+      points(FEV[Sex == "Male"] ~ Height[Sex == "Male"],
+             col = "cornflowerblue"))
> # blue points for boys
> lines(xx, yy1, col = "lightpink2", lwd = 2)
> # estimated expected/predicted FEV by Height for girls
> lines(xx, yy2, col = "dodgerblue2", lwd = 2)
```

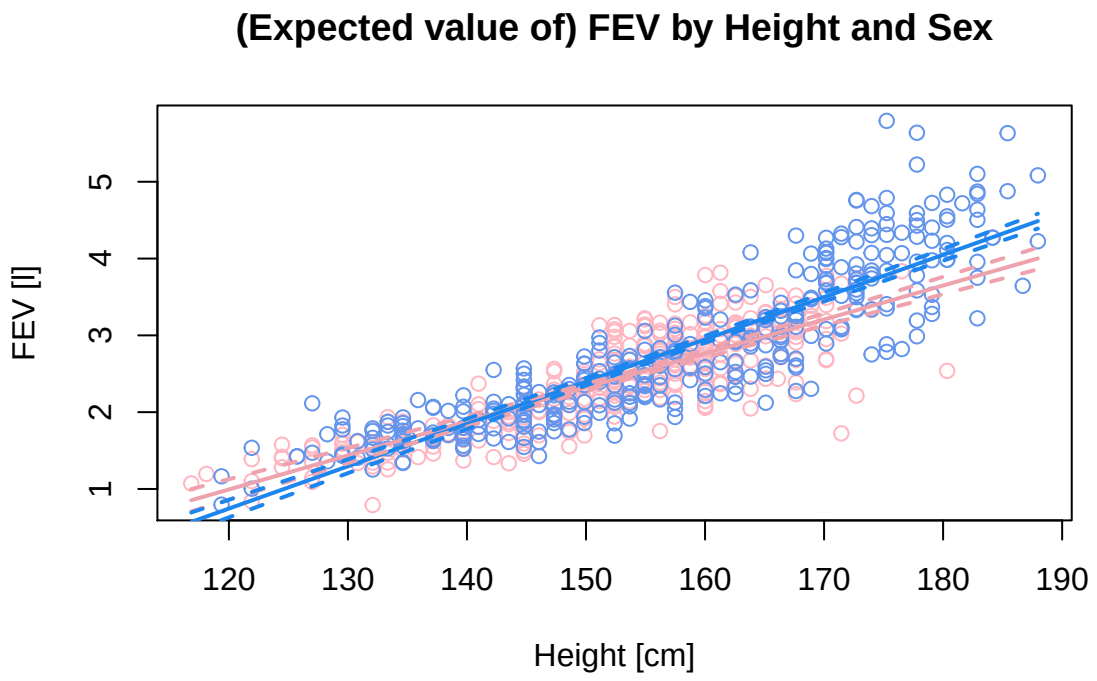



```
> # estimated expected/predicted FEV by Height for boys
```

Přidání konfidenčních intervalů je s funkcí `predict()` přímočaré.

```
> # confidence intervals for the fitted lines
> yy1.conf <- predict(mod,
+                 newdata = data.frame(Height = xx,
+                                      Sex = "Female"),
+                 interval = "confidence")
> yy2.conf <- predict(mod,
+                 newdata = data.frame(Height = xx,
+                                      Sex = "Male"),
+                 interval = "confidence")
>
> plot(fev$FEV ~ fev$Height,
+      main = "(Expected value of) FEV by Height and Sex",
+      xlab = "Height [cm]", ylab = "FEV [l]", type = "n")
> with(fev,
+      points(FEV[Sex == "Female"] ~ Height[Sex == "Female"],
+             col = "lightpink"))
> with(fev,
+      points(FEV[Sex == "Male"] ~ Height[Sex == "Male"],
```

```
+              col = "cornflowerblue"))
> lines(xx, yy1.conf[, 1], col = "lightpink2", lwd = 2)
> # estimated expected value of FEV by Height for girls
> lines(xx, yy2.conf[, 1], col = "dodgerblue2", lwd = 2)
> # estimated expected value of FEV by Height for boys
> lines(xx, yy1.conf[, 2], col = "lightpink2", lwd = 2, lty = 2)
> # lower limits of the confidence intervals for girls
> lines(xx, yy2.conf[, 2], col = "dodgerblue2", lwd = 2, lty = 2)
> # lower limits of the confidence intervals for boys
> lines(xx, yy1.conf[, 3], col = "lightpink2", lwd = 2, lty = 2)
> # upper limits of the confidence intervals for girls
> lines(xx, yy2.conf[, 3], col = "dodgerblue2", lwd = 2, lty = 2)
```

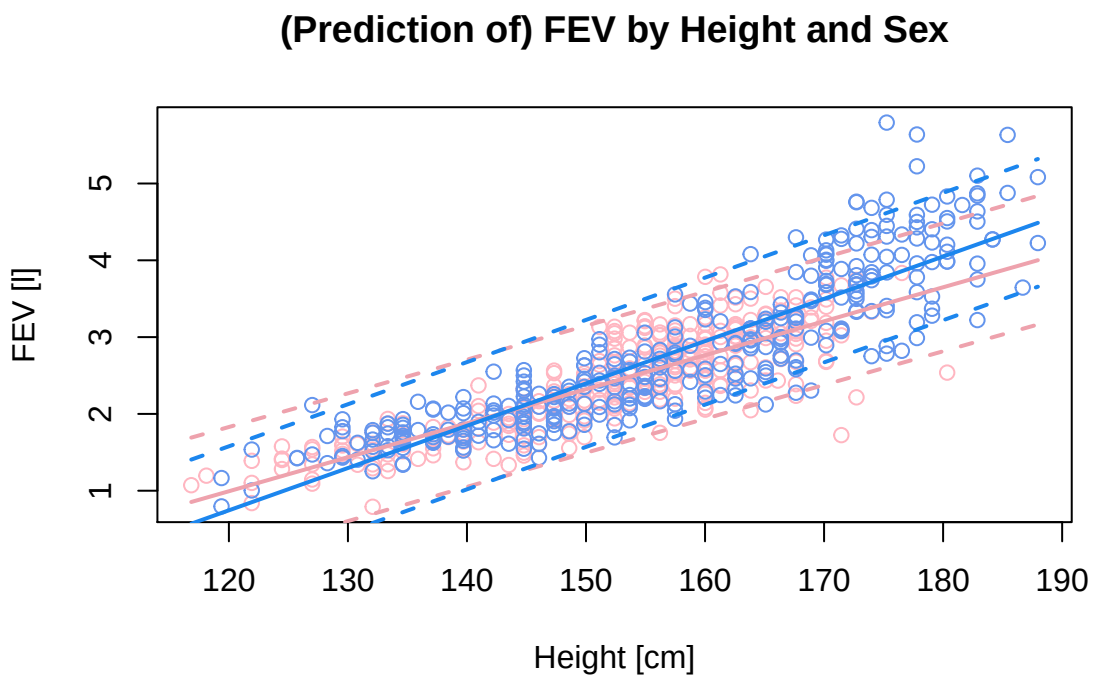



```
> # upper limits of the confidence intervals for boys
```

Stejně tak přidání predikčních intervalů.

```
> # prediction intervals for the fitted lines
> yy1.pred <- predict(mod,
+                     newdata = data.frame(Height = xx,
+                                          Sex = "Female"),
```

```
+                 interval = "prediction")
> yy2.pred <- predict(mod,
+                 newdata = data.frame(Height = xx,
+                                      Sex = "Male"),
+                 interval = "prediction")
>
> plot(fev$FEV ~ fev$Height,
+      main = "(Prediction of) FEV by Height and Sex",
+      xlab = "Height [cm]", ylab = "FEV [l]", type = "n")
> with(fev,
+      points(FEV[Sex == "Female"] ~ Height[Sex == "Female"],
+             col = "lightpink"))
> with(fev,
+      points(FEV[Sex == "Male"] ~ Height[Sex == "Male"],
+             col = "cornflowerblue"))
> lines(xx, yy1.pred[, 1], col = "lightpink2", lwd = 2)
> # prediction of FEV by Height for girls
> lines(xx, yy2.pred[, 1], col = "dodgerblue2", lwd = 2)
> # prediction of FEV by Height for boys
> lines(xx, yy1.pred[, 2], col = "lightpink2", lwd = 2, lty = 2)
> # lower limits of the prediction intervals for girls
> lines(xx, yy2.pred[, 2], col = "dodgerblue2", lwd = 2, lty = 2)
> # lower limits of the prediction intervals for boys
> lines(xx, yy1.pred[, 3], col = "lightpink2", lwd = 2, lty = 2)
> # upper limits of the prediction intervals for girls
> lines(xx, yy2.pred[, 3], col = "dodgerblue2", lwd = 2, lty = 2)
```




```
> # upper limits of the prediction intervals for boys
```

Konfidenční intervaly jsou poměrně úzké díky velkému rozsahu výběru ($n = 654$). To neplatí pro predikční intervaly, v jejichž šířce hraje významnou roli rozptyl σ^2 náhodné chyby ε, který se s rozsahem dat nemění.

Vykreslené konfidenční (i predikční) intervaly jsou *bodové*, jejich $(1-\alpha)\,100\,\%$ pokrytí je tedy zaručeno pro každou hodnotu výšky Height a pohlaví Sex zvlášť. Společné pokrytí by zaručovaly *konfidenční pásy*. Konzervativní konfidenční pásy můžeme odvodit ze Scheffého věty a poté spočítat a vykreslit v R. Podle Scheffého věty je pravděpodobnost, že

$$\left\{\mathbf{b}^\top(\mathbf{A}\,\widehat{\boldsymbol{\beta}} - \mathbf{A}\,\boldsymbol{\beta})\right\}^2 \leq m\,F_{1-\alpha}(m, n-p)\,\widehat{\sigma^2}\;\mathbf{b}^\top\mathbf{A}\,(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{A}^\top\mathbf{b}$$

rovna $1-\alpha$ pro všechna $\mathbf{b} \in \mathbb{R}^m$ současně, je-li $\mathbf{A}$ matice typu $m \times p$ plné hodnosti. V modelu (5.) je vektor koeficientů $(\beta_0, \beta_1, \beta_2, \beta_3)^\top$. Přímka popisující závislost střední hodnoty FEV na výšce Height má pro dívky předpis $y = \beta_0 + \beta_1 x$ a pro chlapce předpis $y = (\beta_0 + \beta_2) + (\beta_1 + \beta_3)\,x$. Pro konfidenční pásy pro přímku pro dívky můžeme v Scheffého větě použít

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1 \\ x \end{pmatrix}, \; x \in \mathbb{R},$$

zatímco pro konfidenční pásy pro přímku pro chlapce můžeme použít

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1 \\ x \end{pmatrix}, \; x \in \mathbb{R}.$$

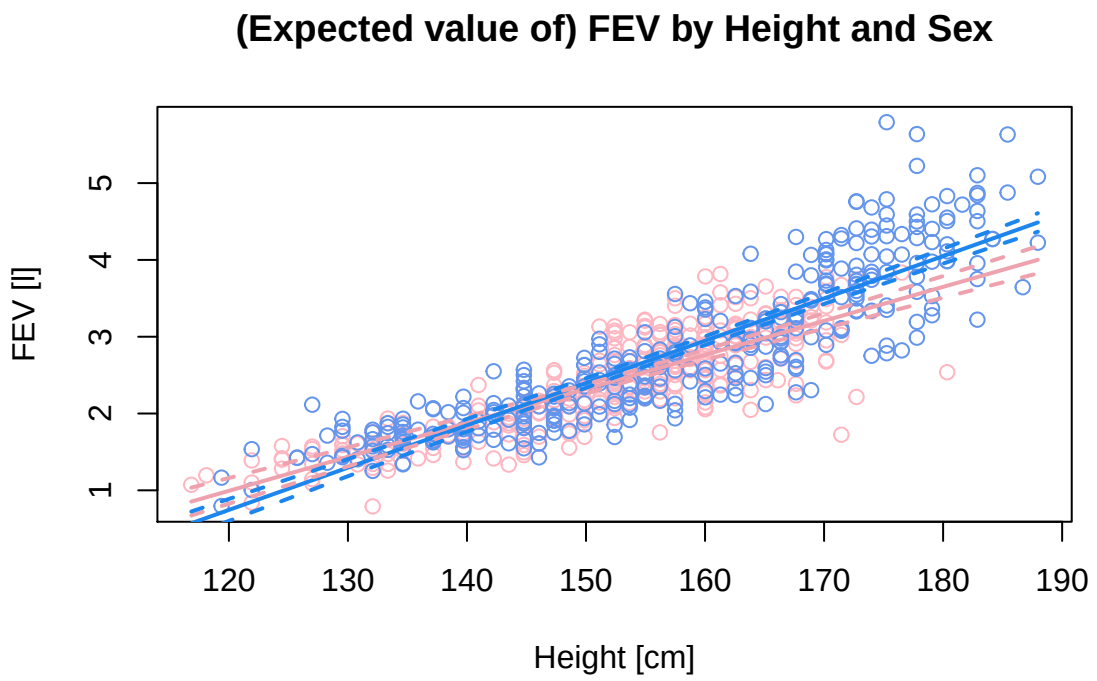
Pokrytí je pak alespoň $100 \times (1-\alpha)\,\%$ pro celou přímku současně (pro každou přímku zvlášť). Konzervativní konfidenční pásy pro obě přímky mají tvar

$$\left(\mathbf{x}^\top\widehat{\boldsymbol{\beta}} - \sqrt{2\,F_{1-\alpha}(2,n-p)\,\widehat{\sigma^2}\,\mathbf{x}^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x}}\,,\right.$$
$$\left.\mathbf{x}^\top\widehat{\boldsymbol{\beta}} + \sqrt{2\,F_{1-\alpha}(2,n-p)\,\widehat{\sigma^2}\,\mathbf{x}^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x}}\right),$$

kde $\mathbf{x} = (1, x, 0, 0)^\top$, $x \in \mathbb{R}$, pro dívky a $\mathbf{x} = (1, x, 1, x)^\top$, $x \in \mathbb{R}$, pro chlapce. Konfidenční pásy pro přímku tedy vypadají podobně jako konfidenční intervaly: místo násobení kvantilem $t_{1-\alpha/2}(n-p)$ se použije násobení $\sqrt{2\,F_{1-\alpha}(2,n-p)}$.

```
> # confidence bands for the fitted lines
> df2 <- summary(mod)$df[2]
> yy1.conf.b <- yy1.conf[, 1] + (yy1.conf - yy1.conf[, 1]) /
+   qt(1 - alpha / 2, df = df2) *
+   sqrt(2 * qf(1 - alpha, df1 = 2, df2 = df2))
> yy2.conf.b <- yy2.conf[, 1] + (yy2.conf - yy2.conf[, 1]) /
+   qt(1 - alpha / 2, df = df2) *
+   sqrt(2 * qf(1 - alpha, df1 = 2, df2 = df2))
>
> plot(fev$FEV ~ fev$Height,
+      main = "(Expected value of) FEV by Height and Sex",
+      xlab = "Height [cm]", ylab = "FEV [l]", type = "n")
> with(fev,
+      points(FEV[Sex == "Female"] ~ Height[Sex == "Female"],
+             col = "lightpink"))
> with(fev,
+      points(FEV[Sex == "Male"] ~ Height[Sex == "Male"],
+             col = "cornflowerblue"))
> lines(xx, yy1.conf[, 1], col = "lightpink2", lwd = 2)
> # estimated expected value of FEV by Height for girls
> lines(xx, yy2.conf[, 1], col = "dodgerblue2", lwd = 2)
> # estimated expected value of FEV by Height for boys
> lines(xx, yy1.conf.b[, 2], col = "lightpink2",
+       lwd = 2, lty = 2)
> # lower limits of the confidence bands for girls
> lines(xx, yy2.conf.b[, 2], col = "dodgerblue2",
+       lwd = 2, lty = 2)
> # lower limits of the confidence bands for boys
> lines(xx, yy1.conf.b[, 3], col = "lightpink2",
+       lwd = 2, lty = 2)
> # upper limits of the confidence bands for girls
```

```
> lines(xx, yy2.conf.b[, 3], col = "dodgerblue2",
+       lwd = 2, lty = 2)
```




```
> # upper limits of the confidence bands for boys
```

Jelikož rozdíl mezi $t_{1-\alpha/2}(n-p)$ a $\sqrt{2\,F_{1-\alpha}(2,n-p)}$ není velký,

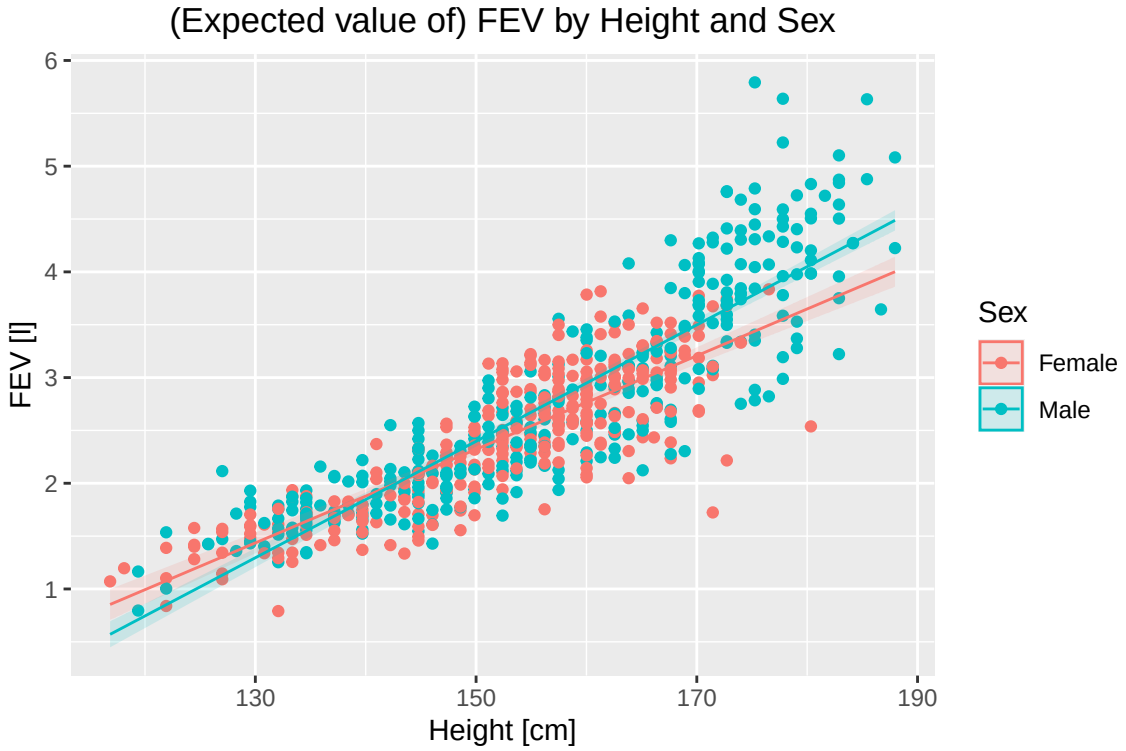
```
> qt(1 - alpha/2, df = df2)

[1] 1.96362

> sqrt(2 * qf(1 - alpha, df1 = 2, df2 = df2))

[1] 2.453398
```

nejsou konfidenční pásy výrazně širší než konfidenční intervaly.

K vykreslení obrázků můžeme použít také knihovnu `ggplot2`.

```
> newdat <- expand.grid(Height = xx,
+                       Sex = factor(c("Female", "Male")))
> newdat <- cbind(newdat, FEV = NA, Lw = NA, Up = NA)
```

```
> newdat[, c(3:5)] <- rbind(yy1.conf, yy2.conf)
>
> ggplot(data = fev,
+        mapping = aes(x = Height, y = FEV, color = Sex)) +
+   geom_point() +
+   geom_ribbon(data = newdat,
+               aes(ymin = Lw, ymax = Up,
+                   fill = Sex, color = NULL),
+               alpha = .15) +
+   geom_line(data = newdat) +
+   xlab("Height [cm]") + ylab("FEV [l]") +
+   ggtitle("(Expected value of) FEV by Height and Sex") +
+   theme(plot.title = element_text(hjust = 0.5))
```



Kapitola 6

Výběr modelu

V předchozí kapitole jsme se zabývali inferencí v daném lineárním modelu. Nemáme-li model předem daný, zpravidla je se součástí analýzy dat i výběr vhodného modelu, což je v lineárních modelech ekvivalentní výběru vhodné regresní matice. Výběr modelu je komplexní úloha, na kterou se můžeme dívat z několika úhlů. Některé aspekty jsou přitom důležitější než jiné v závislosti na tom, k čemu má výsledný model sloužit. Jediný správný model obvykle neexistuje, proto nebývá užitečné jej hledat. Užitečnější bývá pohlížet na modely jako na aproximace skutečnosti a hledat takovou, která nám umožní zodpovědět otázku, kvůli které data analyzujeme. Na zodpovězení různých otázek je možné použít i různé modely. Závěry, které na jejich základě můžeme udělat, by ale neměly být kvalitativně odlišné. Dospějeme-li k závěru, že data podpořují modely s kvalitativně odlišnými závěry, je vhodné zvážit, zda je na základě takových dat možné zodpovědět na položené otázky.

V následujících částech si popíšeme několik faktorů, které mohou do výběru modelu vstoupit.

6.1 Srovnání vnořených modelů pomocí testování

V části 5.5 jsme se věnovali $F-$testu hypotézy o několika lineárních kombinacích složek vektoru koeficientů $\boldsymbol{\beta}$. Nyní se na tento problém podíváme jako na problém testování možnosti přechodu od většího modelu k menšímu. Připomeňme si, že o modelech $\mathbf{Y} = \mathbf{X}_\mathrm{b}\boldsymbol{\beta}_\mathrm{b} + \boldsymbol{\varepsilon}$ a $\mathbf{Y} = \mathbf{X}_\mathrm{s}\boldsymbol{\beta}_\mathrm{s} + \boldsymbol{\varepsilon}$ mluvíme jako o *vnořených*, je-li lineární obal sloupců regresní matice menšího modelu podprostorem lineárního obalu sloupců regresní matice většího modelu. O menším modelu pak mluvíme jako o *podmodelu* většího modelu. Tento vztah mezi lineárními obaly sloupců regresních matic nastane, když jsou sloupce regresní matice podmodelu lineárními kombinacemi sloupců regresní matice modelu. Platnost podmodelu pak implikuje konkrétní hodnoty pro konkrétní lineární kombinace příslušných koeficientů v modelu. Na testování možnosti přechodu od modelu k podmodelu se proto můžeme dívat jako na testování hypotézy o těchto lineárních kombinacích koeficientů v modelu.

V kontextu testování možnosti přechodu od modelu k podmodelu se testová statistika (5.3)

používá spíše ve tvaru

$$F = \frac{(||\mathbf{e}_s||^2 - ||\mathbf{e}_b||^2)/m}{||\mathbf{e}_b||^2/(n-p)}, \tag{6.1}$$

kde $\mathbf{e}_b = \mathbf{Y} - \mathbf{X}_b\widehat{\boldsymbol{\beta}_b}$ je vektor reziduí v modelu a $\mathbf{e}_s = \mathbf{Y} - \mathbf{X}_s\widehat{\boldsymbol{\beta}_s}$ je vektor reziduí v podmodelu. Za platnosti podmodelu normálního lineárního modelu má testová statistika $F-$rozdělení s m a $n-p$ stupni volnosti, kde m je rozdíl mezi dimenzí lineárního obalu sloupců regresní matice modelu a dimenzí lineárního obalu sloupců regresní matice podmodelu. Mají-li obě regresní matice plnou hodnost, jedná se o rozdíl mezi počtem koeficientů v modelu a v podmodelu.

Vraťme se nyní ke srovnání modelů

(5.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Height}_i + \beta_2 \times \mathsf{Sex}_i + \beta_3 \times (\mathsf{Sex} \times \mathsf{Height})_i + \varepsilon_i, \quad i = 1, \ldots, n$

a

(3.) $\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Sex}_i + \varepsilon_i, \quad i = 1, \ldots, n,$

které jsme již v části 5.5 provedli pomocí testové statistiky (5.3). Test o možnosti přechodu od modelu k podmodelu provedeme v R pomocí funkce `anova()`.

```
> mod.b <- lm(FEV ~ Height * Sex, data = fev)
> mod.s <- lm(FEV ~ Sex, data = fev)
>
> anova(mod.s, mod.b)

Analysis of Variance Table

Model 1: FEV ~ Sex
Model 2: FEV ~ Height * Sex
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1    652 469.60
2    650 114.88  2    354.71 1003.5 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Hodnota testové statistiky (sloupec `F` ve výstupu výše) a $p-$hodnoty (sloupec `Pr(>F)`) jsou, podle očekávání, stejné (až na zaokrouhlení) jako v části 5.5. Zbytek výstupu obsahuje jednotlivé stavební kameny výpočtu testové statistiky podle vzorce (6.1): normy reziduí jednotlivých modelů $\mathbf{e}_b$ a $\mathbf{e}_s$ (sloupec `RSS`), jejich rozdíl (sloupec `Sum of Sq.`), rozdíl m v počtu koeficientů modelu a podmodelu (sloupec `Df`) a stupně volnosti χ^2-rozdělení odhadů rozptylů náhodných chyb v jednotlivých modelech (sloupec `Res.Df`).

Pro korektní provedení testu je nutné, aby oba modely, `model.b` i `model.s`, byly v R odhadnuty na stejných datech. Někdy se stane, že v datech pro některá pozorování chybí informace o některých prediktorech. Taková pozorování R nepoužije k odhadu modelu. Stane-li se u některých pozorování, že chybějící hodnoty se objeví v prediktorech modelu, ale už ne

v prediktorech podmodelu, budou tato pozorování použita k odhadu podmodelu ale ne k odhadu modelu. V takovém případě na dotaz `anova(mod.s,mod.b)` odpoví hlášením chyby. Té se můžeme vyhnout tak, že z dat, ze kterých se má odhadnout podmodel, vynecháme pozorování, u kterých chybí informace o prediktorech modelu, například pomocí funkce `na.omit()`.

6.2 Srovnání modelů pomocí kritérií

Kromě testování lze modely srovnávat i pomocí kritérií, jako jsou adjustovaný koeficient determinace, kterému jsme se věnovali v části 5.2, anebo Akaikeho či bayesovské informační kritérium. Na rozdíl od testování lze pomocí kritérií srovnávat i modely, které nejsou vnořené. Stále ale platí, že srovnání má smysl jenom tehdy, byly-li modely odhadnuty na stejných datech. Na tuto podmínku si v případě kritérií musíme dávat větší pozor než u testování, jelikož o kritéria můžeme žádat pro každý model zvlášť. pak „neví“, které modely chceme srovnávat, a nemůže nás upozornit na případný problém s nestejnými daty.

Z části 5.2 víme, že adjustovaný koeficient determinace (5.1) pro model `model` získáme pomocí příkazu `summary(model)$adj.r.squared` a že větší hodnota kritéria znamená lepší model (z hlediska tohoto kritéria).

Ve vztahu (5.1) pro adjustovaný koeficient determinace je kvalita modelu reprezentována vzdáleností predikce na základě modelu od pozorované odezvy $||\mathbf{e}||^2 = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$. Tato kvantita se nutně zvýší, kdykoliv do modelu přidáme prediktor, který není lineární kombinací prediktorů, které již v modelu jsou. Proto je ve vztahu pro adjustovaný koeficient determinace tato vzdálenost penalizována počtem p koeficientů modelu (dělí se $n - p$).

Na zohlednění odhadnuté přesnosti predikce a velikosti modelu je postaveno také *Akaikeho* a *bayesovské informační kritérium*, (AIC) a BIC. Odhad přesnosti predikce je v obou případech reprezentován maximalizovanou věrohodností a penalizace má tvar násobku počtu všech parametrů modelu. V případě AIC jde o dvojnásobek počtu parametrů, zatímco v případě BIC násobíme počet parametrů logaritmem počtu pozorování: penalizace je tedy vyšší pro vyšší rozsahy dat. Znaménka jsou u obou kritérií nastavena tak, že menší hodnota kritéria znamená lepší model (z hlediska daného kritéria). AIC mívá tendenci vybírat spíše větší modely a doporučuje se k němu přihlížet spíše v situacích, kdy nás zajímá predikce na základě modelu. BIC, naopak, častěji preferuje menší modely a lze jej využít v situaci, kdy nás spíše zajímá, které prediktory opravdu souvisejí se střední hodnotou odezvy. Obě kritéria lze použít i pro jiné než lineární modely. O Akaikeho a bayesovské informační kriterium pro model `model` zažádáme v pomocí příkazů `AIC(model)` a `BIC(model)`.

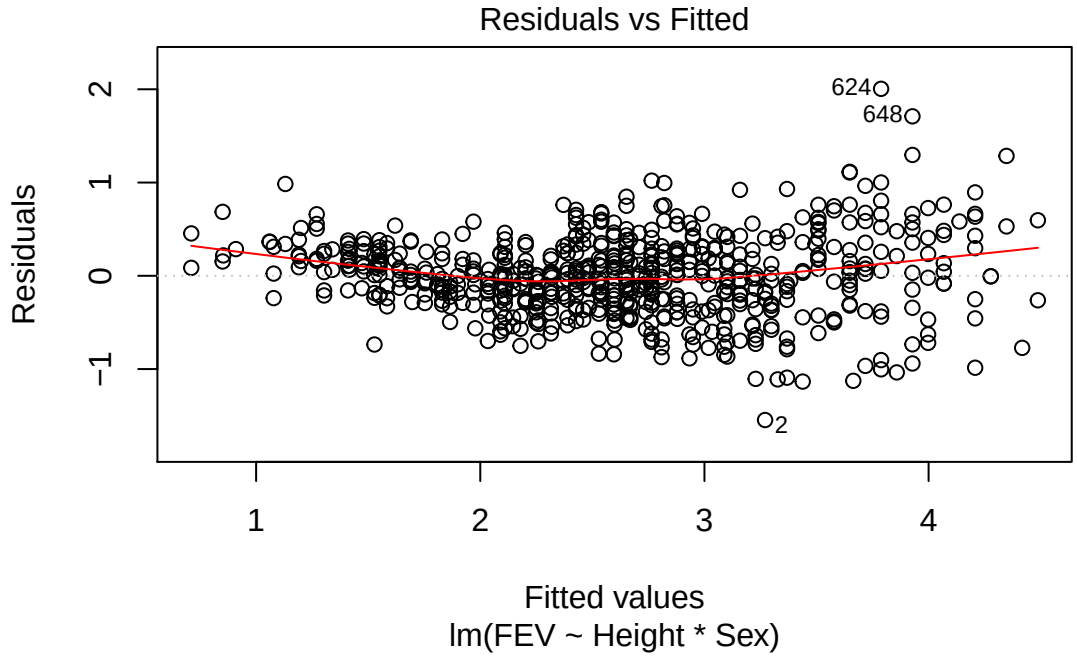
6.3 Diagnostika modelu

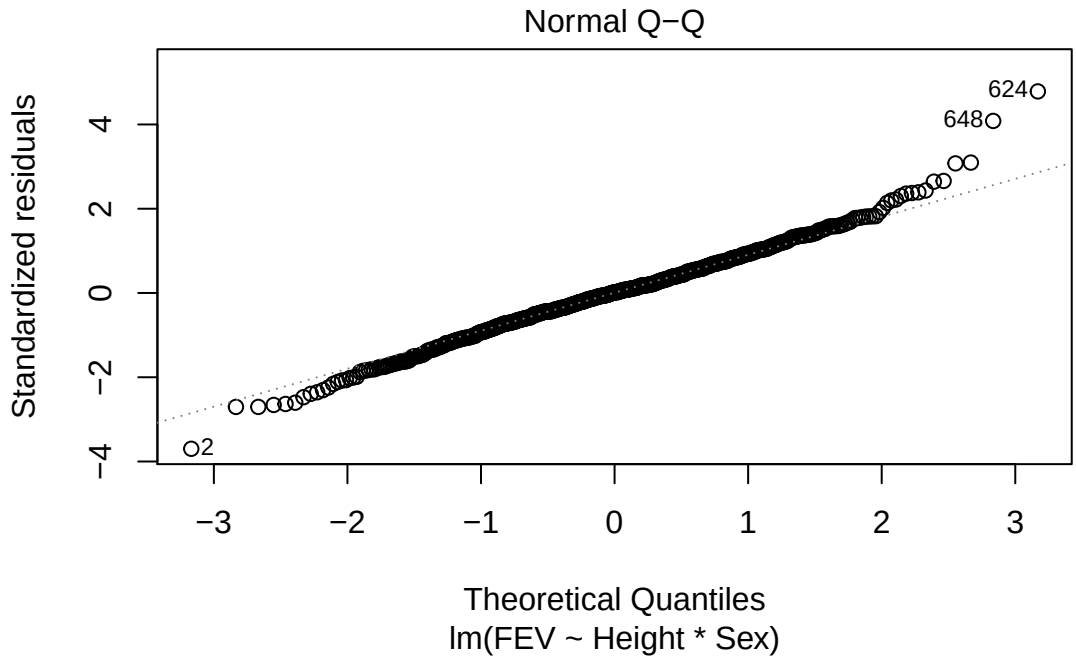
Inference na základě modelu je spolehlivá jenom tehdy, když model platí, tj. když jsou splněny jeho předpoklady. Připomeňme si, že u lineárního modelu předpokládáme, že $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, náhodná chyba $\boldsymbol{\varepsilon}$ má nulovou střední hodnotu, $\mathrm{E}\,\boldsymbol{\varepsilon} = \mathbf{0}$, a rozptyl $\mathrm{Var}\,\boldsymbol{\varepsilon} = \sigma^2\mathbf{I}$. V normálním lineárním modelu navíc požadujeme normalitu náhodné chyby.

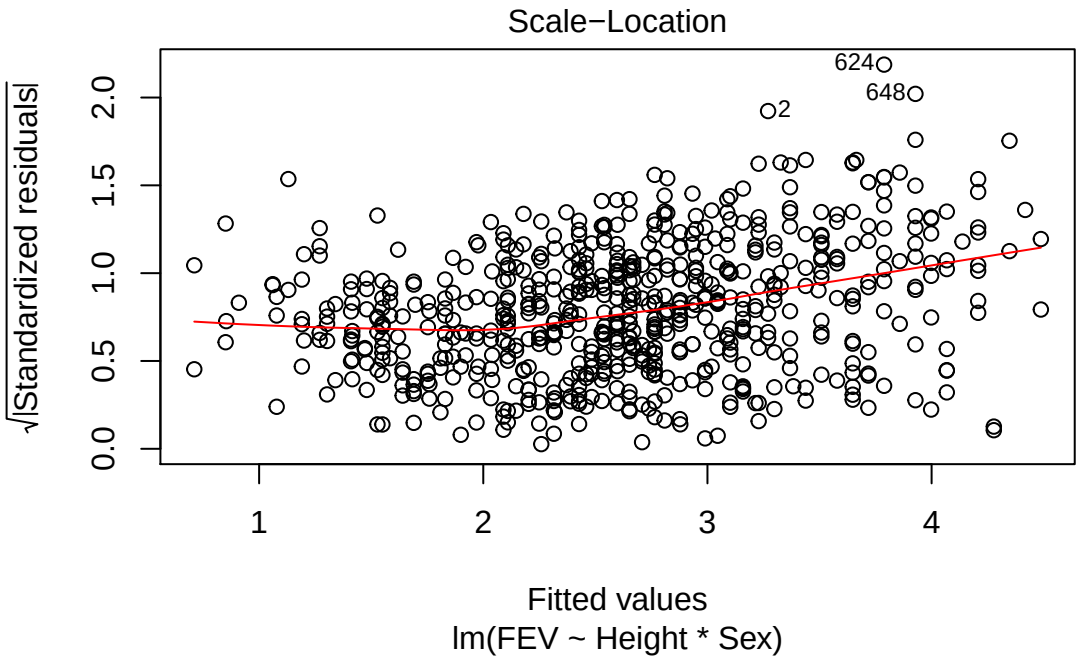
Splnění předpokladů lineárního modelu nelze ověřovat přímo, protože $\boldsymbol{\beta}$ neznáme a $\boldsymbol{\varepsilon}$ nepozorujeme. Proto při ověřování předpokladů místo rovnice $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ pracujeme s rovnicí $\mathbf{Y} = \mathbf{X}\widehat{\boldsymbol{\beta}} + \mathbf{e}$, která obsahuje známé složky. Jelikož lze předpoklady lineárního modelu formulovat na náhodnou chybu $\boldsymbol{\varepsilon}$, jejich splnění ověřujeme především na vektoru reziduí $\mathbf{e}$.

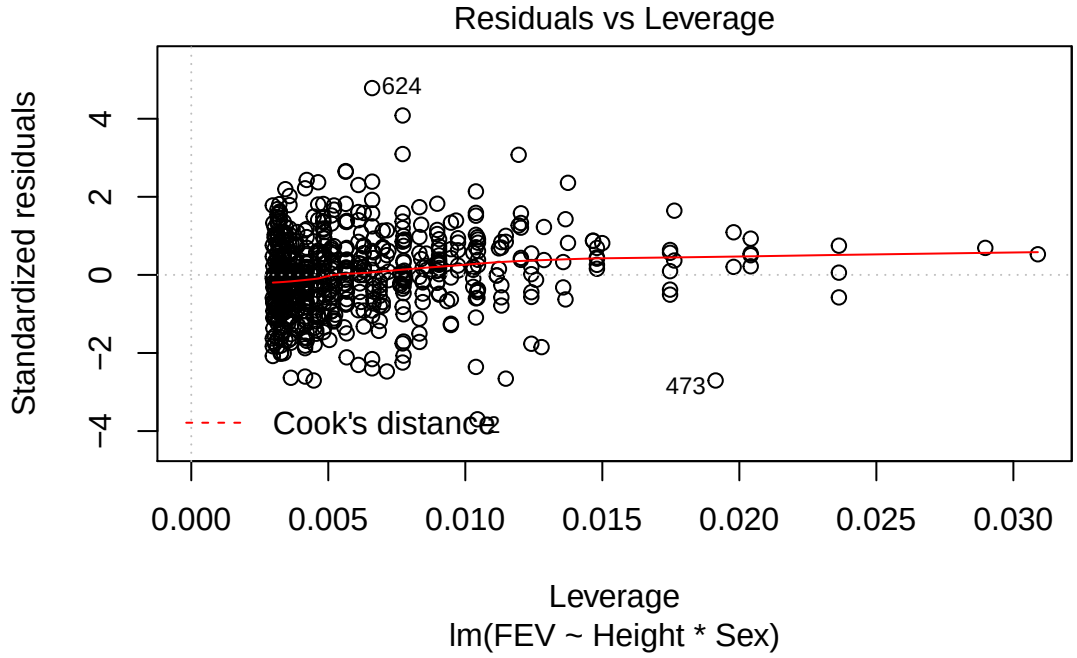
U ověřování předpokladů je výhodné pracovat s diagnostickými grafy. Na nich můžeme vidět, zda se nějaký předpoklad modelu zdá být porušen, a také jakým způsobem, což nás může navést na vhodné řešení problému. Nejdůležitější diagnostické grafy vrací v R funkce `plot()` uplatněna na lineární model, u kterého chceme splnění předpokladů ověřovat.

```
> plot(mod)
```




Normal Q−Q
Standardized residuals
624
648
2
−4
−2
0
2
4
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(FEV ~ Height * Sex)


Scale−Location
√|Standardized residuals|
624
648
2
0.0
0.5
1.0
1.5
2.0
1
2
3
4
Fitted values
lm(FEV ~ Height * Sex)



První z diagnostických grafů vykresluje rezidua $\mathbf{e}$ proti odhadnutým středním hodnotám $\hat{\mathbf{Y}}$. Červená křivka je neparametrický odhad závislosti $\mathbf{e}$ na $\hat{\mathbf{Y}}$. Z definice jsou tyto vektory na sebe kolmé a za platnosti normálního lineárního modelu jde o realizace nezávislých náhodných vektorů. Na grafu by proto neměly být patrné žádné trendy. Různé trendy, které na tomto grafu může být vidět, mohou naznačovat porušení různých předpokladů modelu.

Nejdůležitějším (pro platnost inference) předpokladem lineárního modelu je nulovost střední hodnoty náhodné chyby $\boldsymbol{\varepsilon}$ nebo ekvivalentně správný tvar regresní matice $\mathbf{X}$. Za platnosti tohoto předpokladu mají i rezidua $\mathbf{e}$ nulovou střední hodnotu. Je-li tento předpoklad, naopak, porušen, rezidua nulovou střední hodnotu mít nemusí a na grafu reziduí proti odhadnutým středním hodnotám se mohou objevit nějaké trendy. Červená křivka v prvním grafu se v takovém případě může výrazněji odchylovat od nuly.

Máme-li podezření, že jsme v modelu nezachytili závislost střední hodnoty odezvy na nějakém (potenciálním) prediktoru anebo že jsme nezvolili správnou formu závislosti na nějakém prediktoru (například je potřeba transformace prediktoru), můžeme si rezidua vykreslit také proti tomuto prediktoru. Tvar závislosti, který na grafu případně uvidíme, můžeme poté zkusit zahrnout do modelu. Po každém takovém rozšíření modelu je potřeba znovu zkontrolovat diagnostiku modelu.

Dalším problémem, kterého si případně můžeme všimnout již na prvním grafu, je závislost rozptýlení reziduí na odhadnutých středních hodnotách $\hat{\mathbf{Y}}$. To naznačuje problém s předpokladem o stejných rozptylech náhodných chyb pro jednotlivá pozorování (*homoskedasticitě*). Na ověřování tohoto předpokladu je zaměřený **třetí graf**. Ten místo reziduí $\mathbf{e}$ používá *standardizovaná*

rezidua

$$r_i = \frac{e_i}{\sqrt{\widehat{\sigma^2}(1 - h_{ii})}},$$

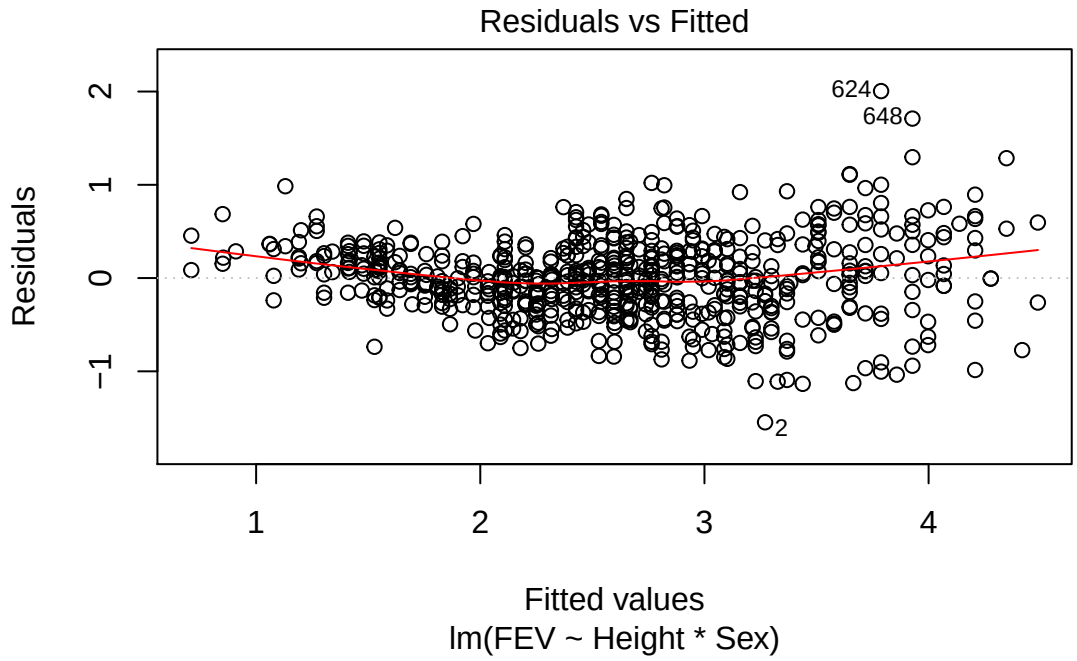
kde $h_{i,i}$ jsou diagonální prvky projekční matice $\mathbf{H} = \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top$. Standardizovaná rezidua r_i mají, na rozdíl od reziduí e_i, za platnosti modelu stejné (jednotkové) rozptyly. Standardizovaná rezidua se na třetím grafu vykreslí proti odhadnutým středním hodnotám $\hat{\mathbf{Y}}$. Pro zvýraznění případné závislosti se na $y-$novou osu vykresluje odmocnina z absolutní hodnoty reziduí a graf se prokládá neparametrickým odhadem jejich závislosti na $\hat{\mathbf{Y}}$ (červená křivka). Výraznější odchylky od (kladné) horizontální přímky naznačují problémy s homoskedasticitou. Podobně jako u střední hodnoty můžeme i u rozptylu vyhodnotit případnou závislost na prediktorech pomocí grafů standardizovaných reziduí proti prediktorům.

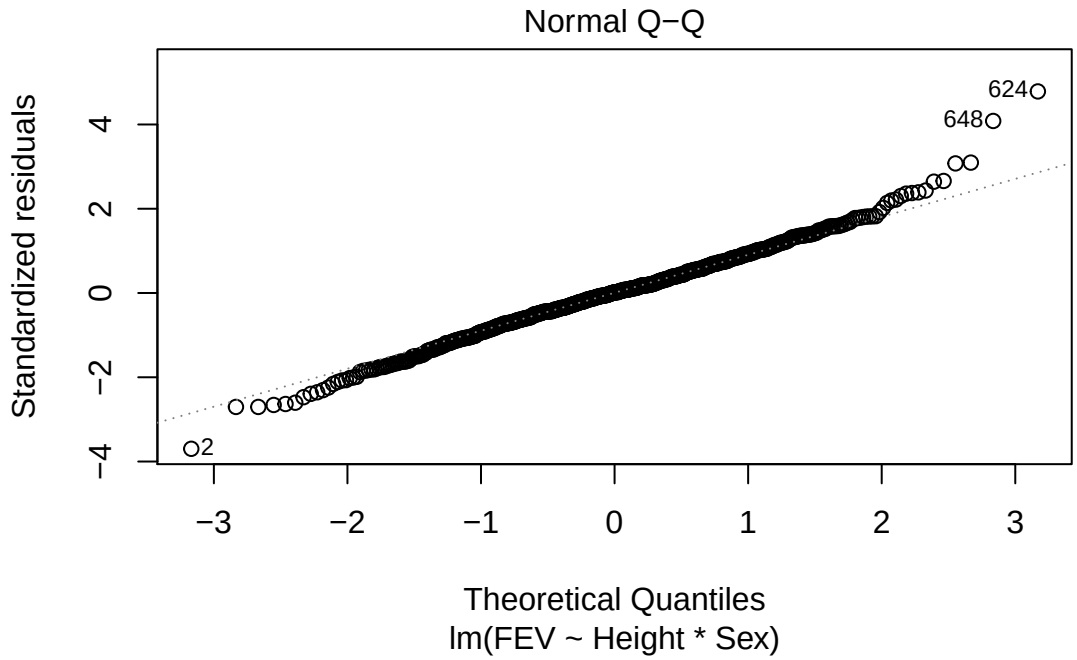
Vedle homoskedasticity předpokládá lineární model i nekorelovanost náhodných chyb příslouchajících jednotlivým pozorováním. Porušení tohoto předpokladu často vyžaduje využití pokročilejšího modelu, například časové řady, plyne-li závislost z časové následnosti mezi pozorováními, anebo modelu s náhodnými efekty, je-li závislost dána opakovanými pozorováními na stejných objektech. Detailní diagnostika závislosti se pak provede nástroji obvyklými v kontextu těchto pokročilejších modelů.

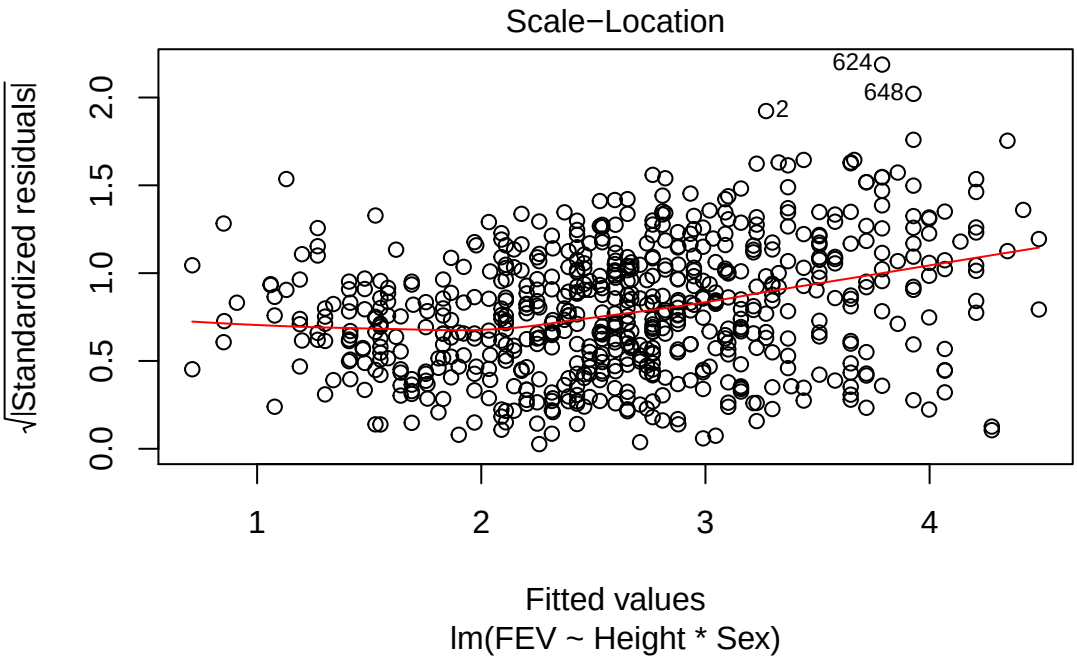
Druhý graf slouží k ověření normality náhodných chyb. I tady se používají spíše standardizovaná rezidua, která mají za platnosti modelu přibližně stejné normální rozdělení. Jedná se o klasický Q-Q graf, a proto by se za platnosti předpokladů body neměly výrazně odchylovat od proložené přímky, vyskytovat se od ní jenom na jednu stranu (problémy se šikmostí rozdělení) anebo ve tvaru písmene S (problémy se špičatostí rozdělení). Mírná odchýlení od proložené přímky jsou ale vpořádku. Názorné ukázky Q-Q grafů pro data různých rozsahů pochazející z normálního i z jiných rozdělení je možné nalézt například v příloze bakalářské práce [4].

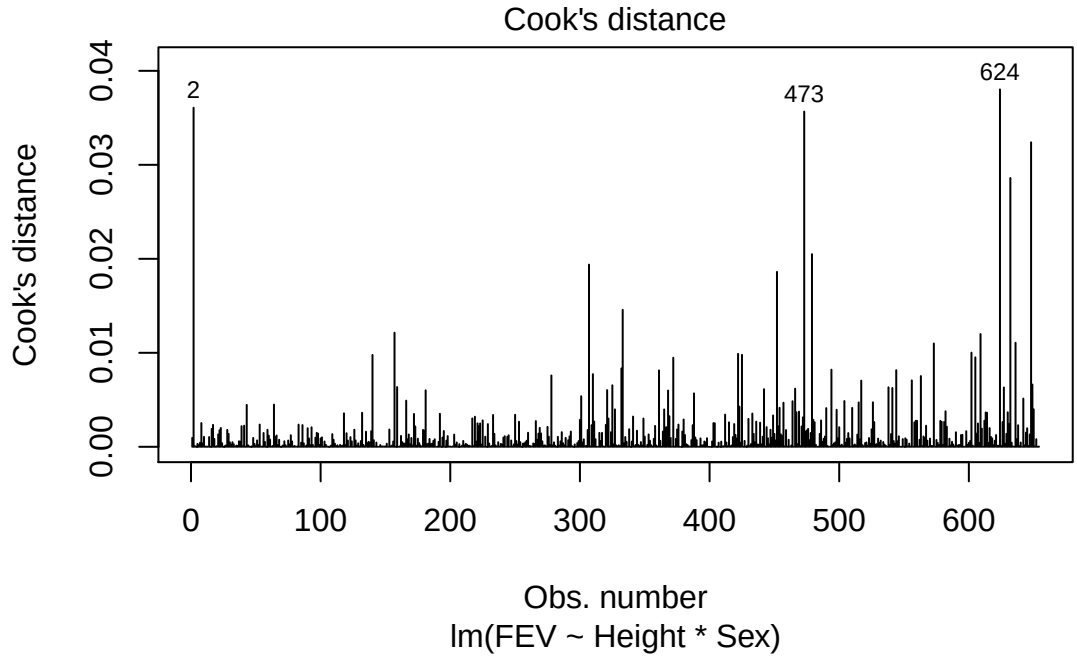
Poslední graf slouží k detekci vlivných pozorování. *Leverages* $h_{i,i}$ vykresleny na $x-$ové ose kvantifikují potenciál pozorování ovlivnit odhad modelu. *Cookova vzdálenost*, která je do grafu vnesena pomocí vrstevnic, kvantifikuje, jestli k ovlivnění odhadu opravdu dochází. U pozorování s vysokými hodnotami těchto ukazatelů (u leverages se, v závislosti od zdroje, uvádí p/n, $2p/n$, anebo $3p/n$, kde p je počet koeficientů v modelu a n je počet pozorování; u Cookovy vzdálenosti 0.5 anebo 1) je potřeba ověřit, zda jsou hodnoty odezvy i prediktorů správně zadané, a rozmyslet si, zda jsou tato pozorování reprezentativní pro populaci, na kterou chceme výsledky modelu zobecňovat. Pro podrobnější diagnostiku vlivných pozorování je v možné si v R vyžádat rozšířenou nabídku diagnostických grafů.

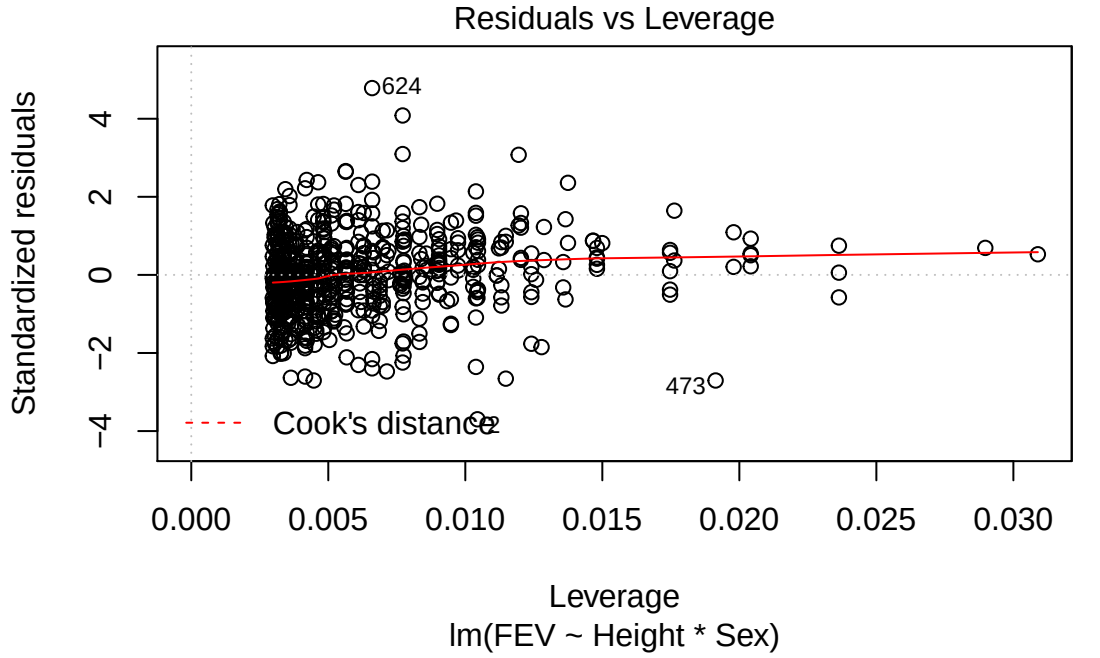
```
> plot(mod, which = c(1:6))
```

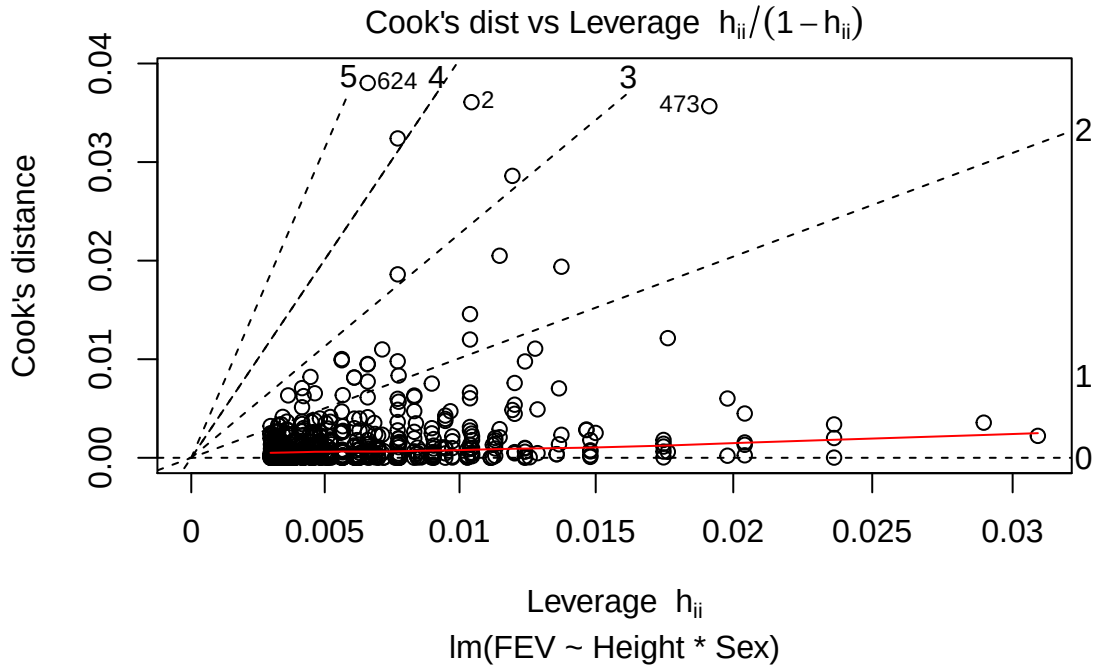

Residuals vs Fitted
Residuals
Fitted values
lm(FEV ~ Height * Sex)
624
648
2


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(FEV ~ Height * Sex)
624
648
2


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
624
648
2
1
2
3
4
Fitted values
lm(FEV ~ Height * Sex)


Cook's distance
Cook's distance
0.00
0.01
0.02
0.03
0.04
2
473
624
0
100
200
300
400
500
600
Obs. number
lm(FEV ~ Height * Sex)



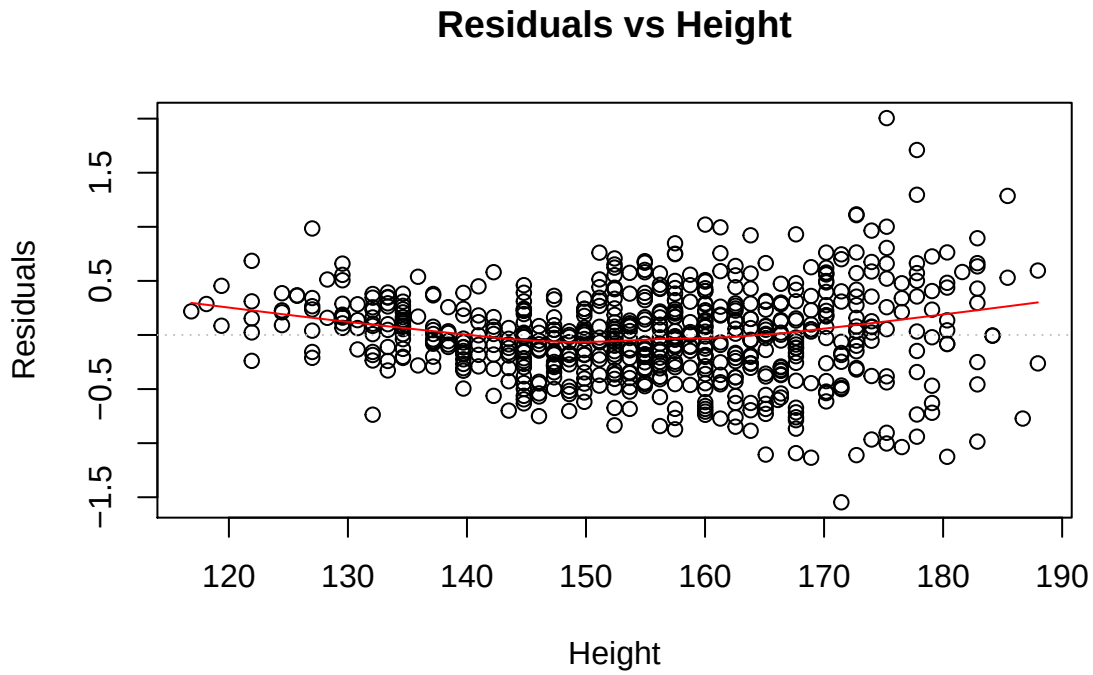


Na diagnostických grafech pro model

$$\text{FEV}_i = \beta_0 + \beta_1 \times \text{Height}_i + \beta_2 \times \text{Sex}_i + \beta_3 \times (\text{Height} \times \text{Sex})_i + \varepsilon_i, \quad i = 1, \ldots, n \tag{5.}$$

vidíme problémy již s volbou regresní matice. Tvar závislosti na prvním diagnostickém grafu naznačuje nepodchycení nějaké kvadratické závislosti. Podobný trend najdeme na grafu reziuí e proti prediktoru výška Height.

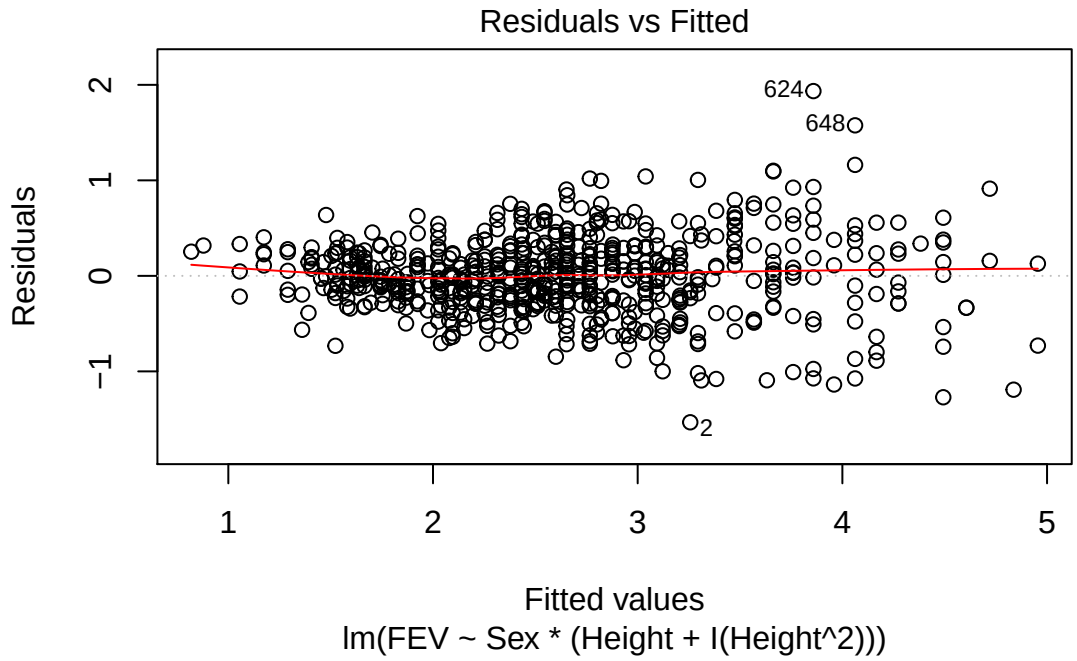
```
> e <- residuals(mod)
> plot(e ~ fev$Height, main = "Residuals vs Height",
+      xlab = "Height", ylab = "Residuals")
> abline(h = 0, col = "grey", lty = 3)
> lines(lowess(e ~ fev$Height), col = "red")
```

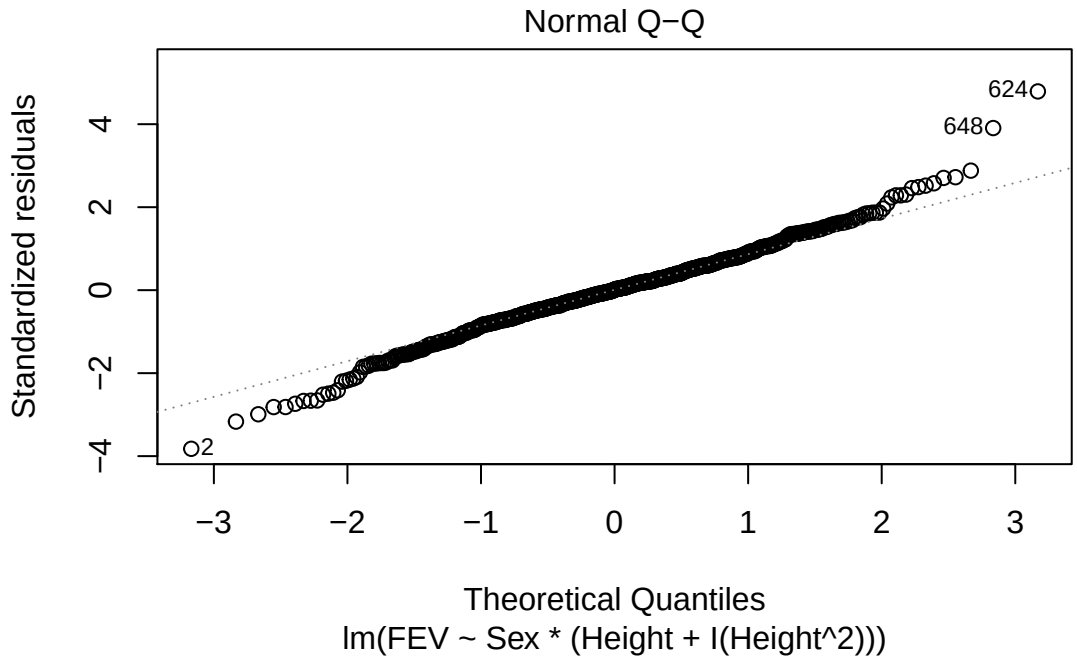


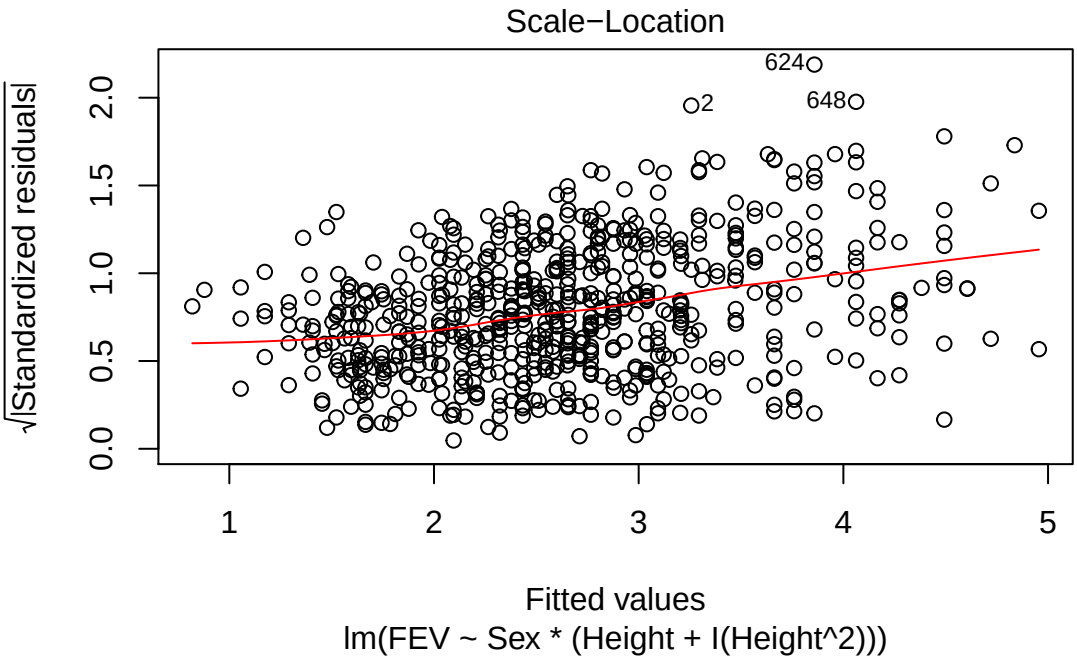
Můžeme proto do modelu zkusit přidat kvadratickou závislost střední hodnoty FEV na výšce Height

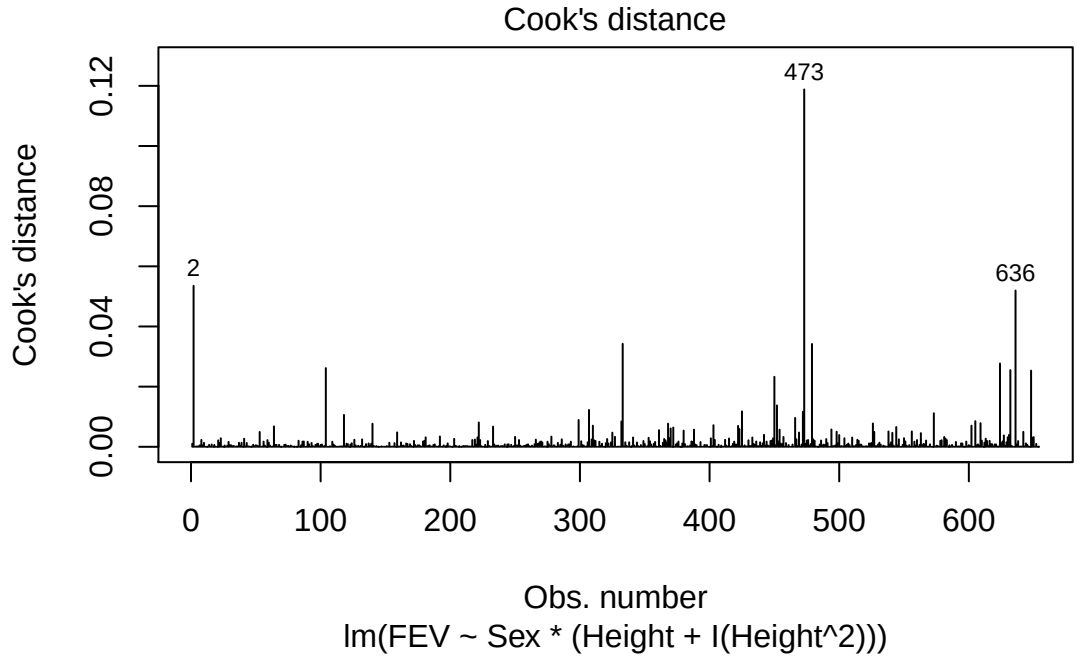
$$\mathsf{FEV}_i = \beta_0 + \beta_1 \times \mathsf{Sex}_i + \beta_2 \times \mathsf{Height}_i + \beta_3 \times (\mathsf{Height} \times \mathsf{Sex})_i + \\ + \beta_4 \times \mathsf{Height}_i^2 + \beta_5 \times (\mathsf{Height}^2 \times \mathsf{Sex})_i + \varepsilon_i, \quad i = 1, \ldots, n.$$

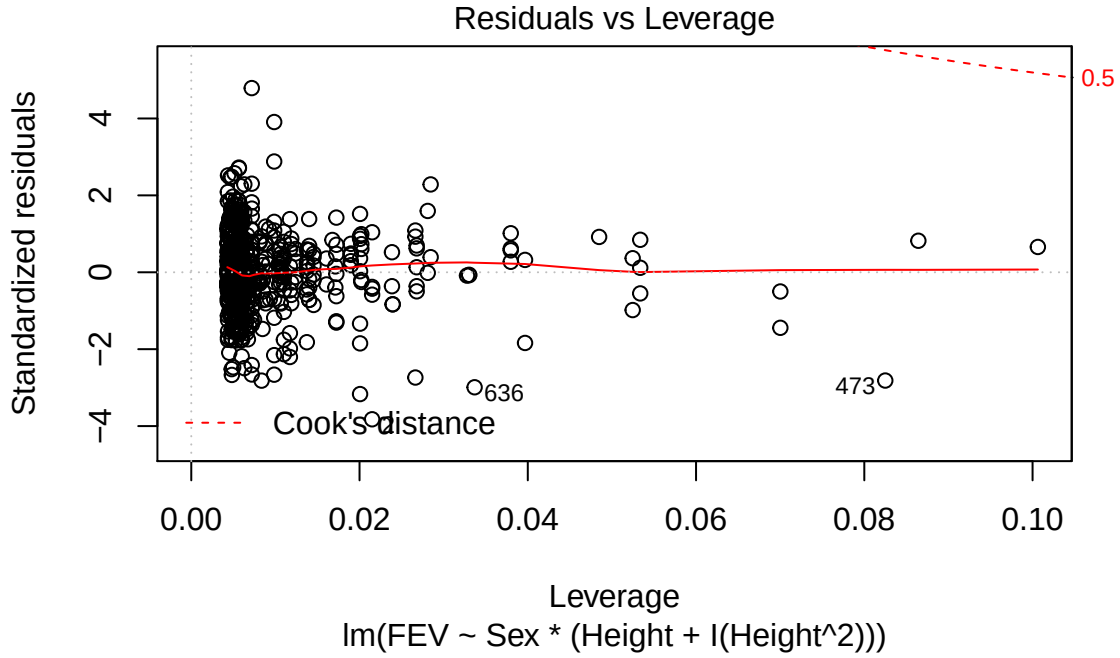
```
> mod <- lm(FEV ~ Sex * (Height + I(Height^2)), data = fev)
> plot(mod, which = c(1:6))
```

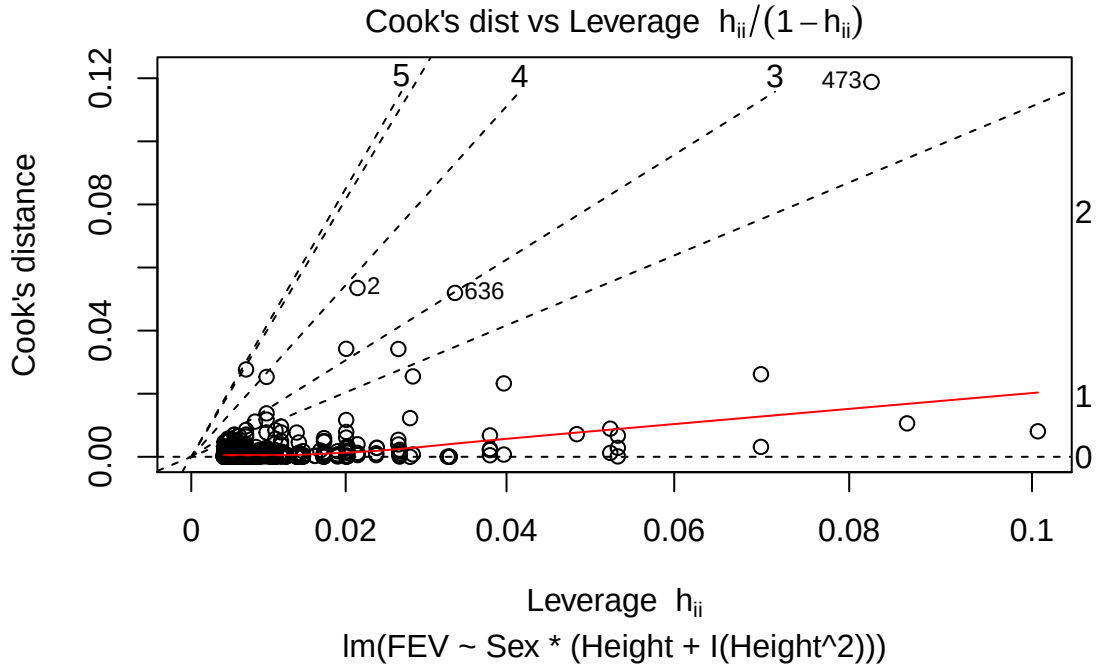

Residuals vs Fitted
Residuals
624
648
2
−1
0
1
2
1
2
3
4
5
Fitted values
lm(FEV ~ Sex * (Height + I(Height^2)))


Normal Q−Q
Standardized residuals
624
648
2
−4
−2
0
2
4
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(FEV ~ Sex * (Height + I(Height^2)))


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
624
2
648
1
2
3
4
5
Fitted values
lm(FEV ~ Sex * (Height + I(Height^2)))


Cook's distance
Cook's distance
0.00
0.04
0.08
0.12
473
2
636
0
100
200
300
400
500
600
Obs. number
lm(FEV ~ Sex * (Height + I(Height^2)))





Dianostické grafy rozšířeného modelu již nenaznačují problémy se střední hodnotou náhodných chyb. Naopak zůstal problém s homoskedasticitou a s normalitou dat. Při takto velkém rozsahu

výběru ($n = 654$) a absenci pozorování s výrazným potenciálem ovlivnit odhad modelu není normalita klíčová, jelikož v takových případech lze platnost inference doložit pomocí centrální limitní věty. Homoskedasticita je ale pro platnost testů a konfidenčních intervalů potřeba. Její nesplnění můžeme řešit například nahrazením lineárního modelu některým z jeho zobecnění. Ty jsou již nad rámec tohoto textu.

6.4 Multikolinearita

Multikolinearita, tedy vztah blízký lineární závislosti mezi prediktory, může působit problémy se stabilitou i statistickou kvalitou odhadů koeficientů i v jinak platných modelech. Závislost mezi dvojicemi prediktorů můžeme odhalit pomocí grafů, na kterých vykreslujeme jeden prediktor proti druhému, například pomocí funkce `pairs()`, kterou jsme používali ve 2. kapitole. Na závislosti nás také může upozornit vysoká výběrová korelace mezi dvojicemi (kvantitativních) prediktorů.

```
> cor(fev[, c(1, 3)])

              Age    Height
Age    1.0000000 0.7919436
Height 0.7919436 1.0000000
```

Složitější závislosti zahrnující více prediktorů snadněji odhalíme pomocí *variance inflation factor* $VIF_j = 1/(1 - R_j^2)$, kde R_j^2 je koeficient determinace v lineárním modelu, kde je $j-$tý prediktor z původního modelu odezvou a zbylé prediktory z původního modelu jsou prediktory i v tomto modelu. Hodnoty větší než pět se považují za problematické. Odpovídá-li jednomu prediktoru v původním modelu více koeficientů, použije se místo *variance inflation factor* VIF jeho zobecnění *generalized variance inflation factor* $gVIF$ pocházející z lineární regrese s mnohorozměrnou odezvou. V R o tyto charakteristiky požádáme funkcí `vif()` z knihovny `car`.

```
> library(car)
> vif(mod)

             Sex           Height      I(Height^2)      Sex:Height
       11654.353         1137.165         1213.311        52072.429
Sex:I(Height^2)
       15254.133
```

U proměnných, které jsou z definice propojeny (například interakce a polynomy) lze, pochopitelně, závislost očekávat.

```
> vif(lm(FEV ~ Sex + Height, data = fev))

     Sex   Height
1.025946 1.025946

> vif(lm(FEV ~ Sex + Age, data = fev))

    Sex     Age
1.00085 1.00085
```

U polynomů lze problém řešit pomocí funkce `poly()`, která místo klasické parametrizace polynomů využívá *ortogonální polynomy* daného řádu. Odhady střední hodnoty a tedy i celkové charakteristiky modelu budou stejné, jako kdybychom použili klasický polynom stejného řádu. Sloupce regresní matice odpovídající prediktoru, na kterém má střední hodnota odezvy záviset polynomiálně, jsou ale na sebe kolmé, což zaručuje stabilitu odhadu.

```
> summary(lm(FEV ~ Height + I(Height^2), data = fev))


Call:
lm(formula = FEV ~ Height + I(Height^2), data = fev)

Residuals:
     Min       1Q   Median       3Q      Max
-1.80103 -0.22928 -0.00255  0.21924  1.99699

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.027e+00  1.503e+00   4.009 6.79e-05 ***
Height      -9.844e-02  1.963e-02  -5.015 6.83e-07 ***
I(Height^2)  4.891e-04  6.372e-05   7.675 6.07e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4127 on 651 degrees of freedom
Multiple R-squared:  0.7741,Adjusted R-squared:  0.7734
F-statistic:  1115 on 2 and 651 DF,  p-value: < 2.2e-16

> summary(lm(FEV ~ poly(Height, 2), data = fev))


Call:
lm(formula = FEV ~ poly(Height, 2), data = fev)
```

```
Residuals:
     Min       1Q   Median       3Q      Max
-1.80103 -0.22928 -0.00255  0.21924  1.99699

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       2.63678    0.01614 163.376  < 2e-16 ***
poly(Height, 2)1 19.23502    0.41274  46.604  < 2e-16 ***
poly(Height, 2)2  3.16778    0.41274   7.675 6.07e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4127 on 651 degrees of freedom
Multiple R-squared:  0.7741,Adjusted R-squared:  0.7734
F-statistic:  1115 on 2 and 651 DF,  p-value: < 2.2e-16
```

Funkce `vif()` chápe ortogonální polynom jako jediný prediktor, kterému odpovídá více koeficientů.

```
> vif(lm(FEV ~ poly(Height, 2) + Sex, data = fev))

                    GVIF Df GVIF^(1/(2*Df))
poly(Height, 2) 1.092802  2        1.022434
Sex             1.092802  1        1.045372
```

Kapitola 7

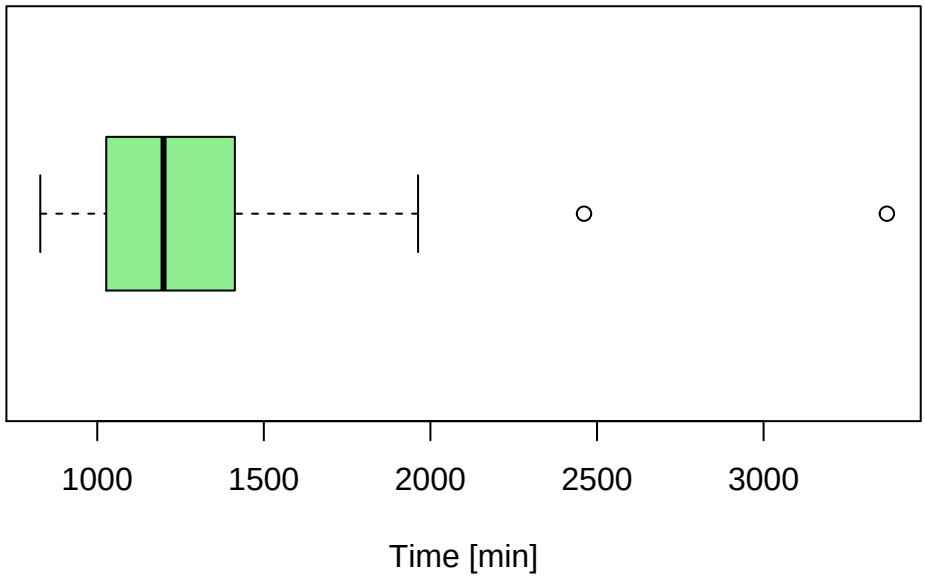
Ilustrační analýza časů přeplavání jezera

V této kapitole budeme pomocí lineárních modelů zkoumat, zda čas, za který plavci překonali jezero Ontario, souvisí s jejich věkem a pohlavím. Nejprve si připomeňme data `swim` z části 2.1.

```
> str(swim)

'data.frame': 65 obs. of  8 variables:
 $ Name     : Factor w/ 54 levels "Angela Kondrak",..: 32 24 7 5 22 22 11 16 15 1 ...
 $ Sex      : Factor w/ 2 levels "F","M": 1 2 1 2 2 2 1 1 1 1 ...
 $ Age      : num  16 36 28 57.3 41 ...
 $ Start.Day: int  8 23 12 23 26 2 17 30 16 22 ...
 $ Month    : Ord.factor w/ 3 levels "July"<"Aug"<"Sep": 3 1 2 2 2 3 2 2 2 2 ...
 $ Year     : int  1954 1956 1956 1957 1957 1961 1974 1974 1975 1976 ...
 $ Time.min.: num  1255 1273 1131 1501 1115 ...
 $ Direction: Factor w/ 3 levels "NS","NSN","SN": 3 3 3 3 3 3 3 1 3 3 ...
```

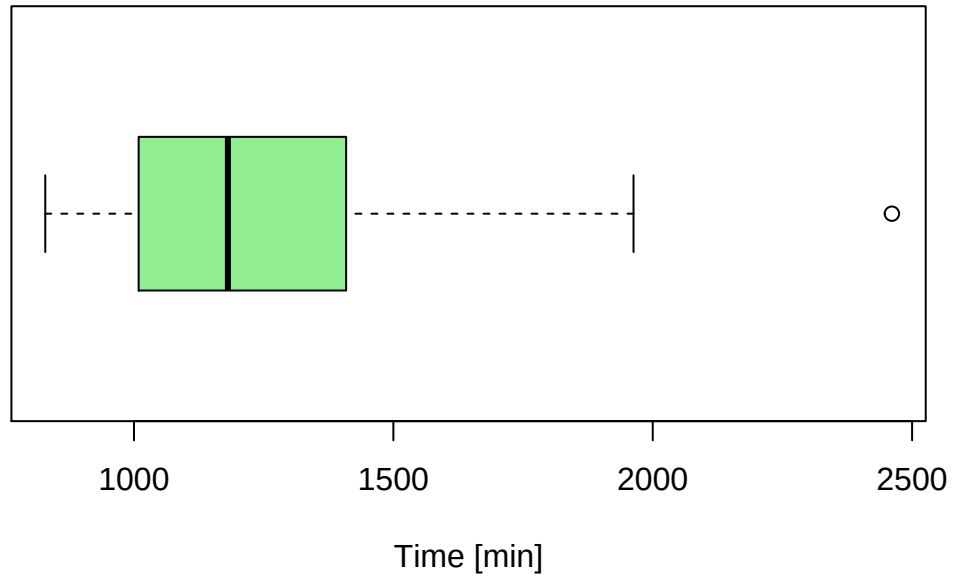
Naší závislou proměnnou bude Time.min. udávající dobu proplavání jezera v minutách. Pro snadnější práci si ji přejmenujeme na Time a podíváme se na ni.

```
> names(swim)[7] <- "Time"
> with(swim,
+      boxplot(Time, xlab="Time [min]",
+              col="lightgreen", horizontal = TRUE)
+      )
```



Většina plavců jezero překonala do 1 500 minut, zhruba čtvrtina potřebovala více času, dva plavci plavali dokonce 2 500 a i více než 3 000 minut. Tento nejdelší čas ale patří Vicki Keith, která v roce 1987 jezero proplavala obousměrně. Její čas proto nebudeme uvažovat.

```
> swim <- swim[swim$Time < 3000, ]
> swim$Direction <- factor(swim$Direction)
> with(swim,
+      boxplot(Time, xlab = "Time [min]",
+              col = "lightgreen", horizontal = TRUE)
+      )
```



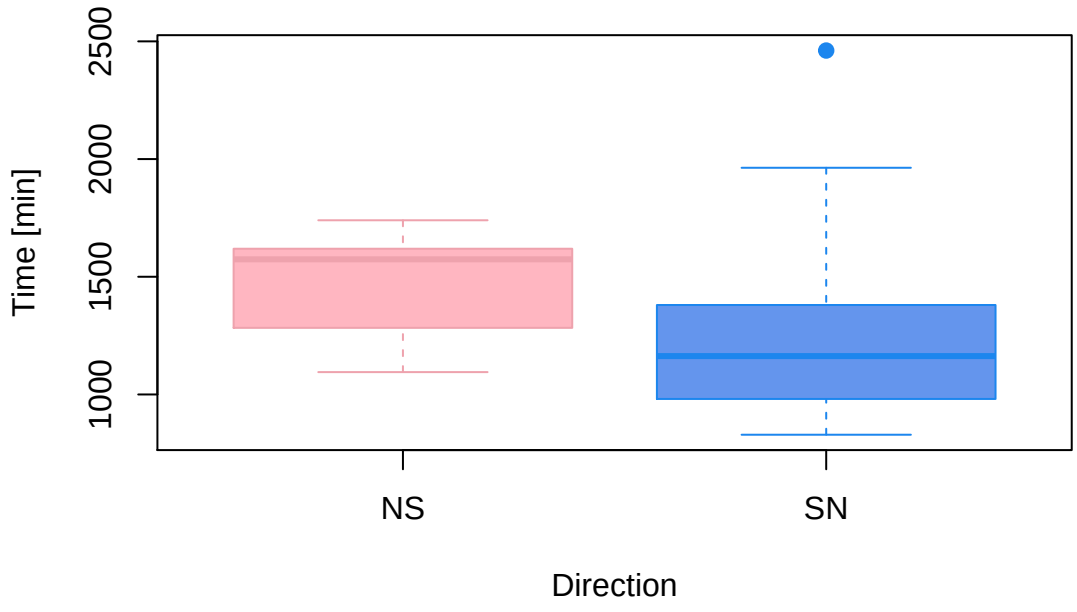
Nyní se na data podíváme podrobněji.

```
> head(swim)

            Name Sex      Age Start.Day Month Year Time Direction
1  Marilyn Bell   F 16.00000         8   Sep 1954 1255        SN
2  John Jaremey   M 36.00000        23  July 1956 1273        SN
3 Brenda Fisher   F 28.00000        12   Aug 1956 1131        SN
4    Bill Sadlo   M 57.31781        23   Aug 1957 1501        SN
5     Jim Woods   M 41.00000        26   Aug 1957 1115        SN
6     Jim Woods   M 45.00000         2   Sep 1961 1027        SN

> summary(swim)

               Name    Sex          Age           Start.Day
 Colleen Shields: 3   F:38   Min.   :14.05   Min.   : 1.00
 Kim Middleton  : 3   M:26   1st Qu.:20.75   1st Qu.: 8.00
 Vicki Keith    : 3          Median :28.00   Median :13.50
 Jim Woods      : 2          Mean   :31.11   Mean   :14.14
 John Scott     : 2          3rd Qu.:38.00   3rd Qu.:19.75
 Kim Lumsdon    : 2          Max.   :66.57   Max.   :31.00
 (Other)        :49
  Month         Year           Time      Direction
```

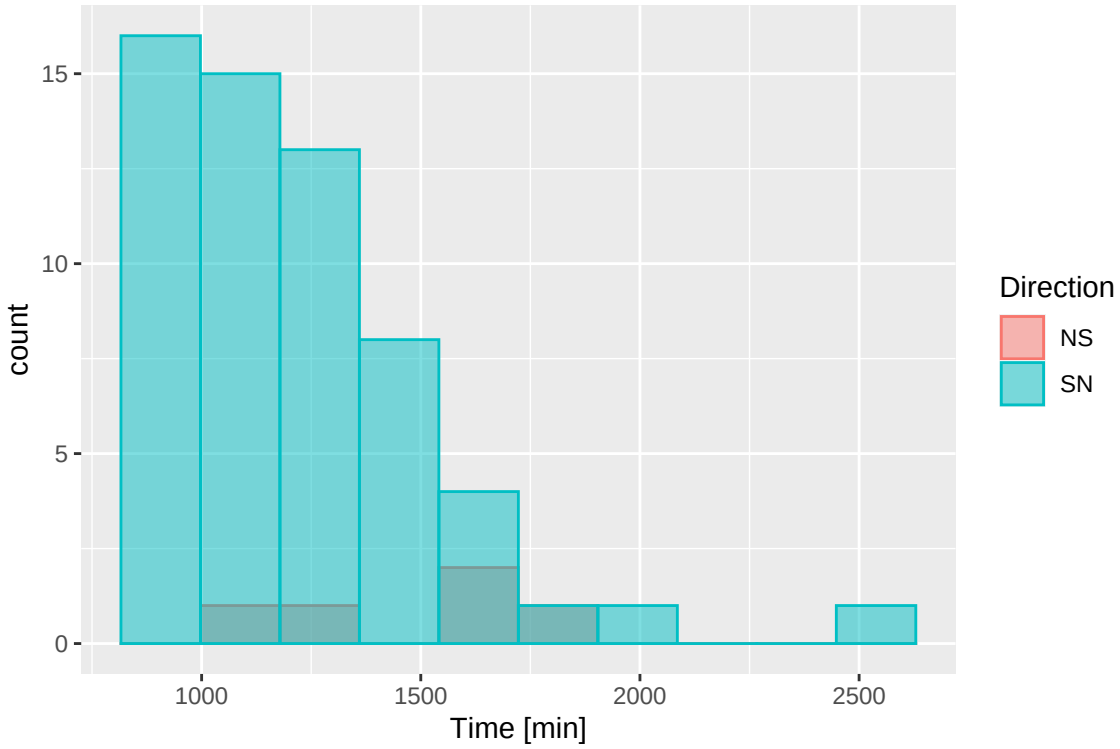
```
 July: 6   Min.   :1954   Min.   : 829   NS: 5
 Aug :49   1st Qu.:1979   1st Qu.:1018   SN:59
 Sep : 9   Median :1993   Median :1181
           Mean   :1992   Mean   :1246
           3rd Qu.:2007   3rd Qu.:1407
           Max.   :2018   Max.   :2461
```

Můžeme si všimnout, že většina plavců se rozhodla plavat z jihu na sever (Direction `SN`).

```
> with(swim,
+      boxplot(Time ~ Direction, ylab = "Time [min]",
+              col = c("lightpink", "cornflowerblue"),
+              border = c("lightpink2", "dodgerblue2"),
+              pch = 19)
+ )
```



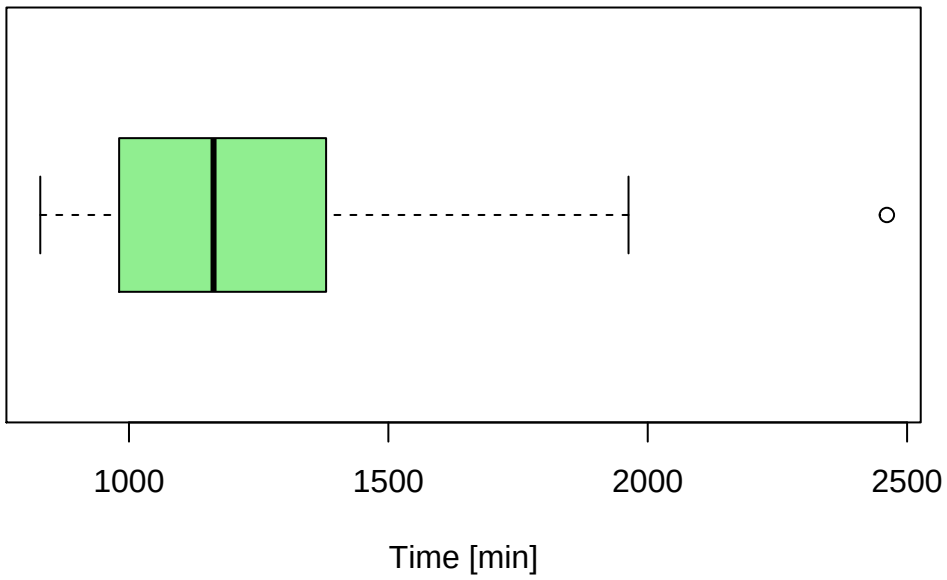

```
> ggplot(data = swim, aes(x = Time, color = Direction)) +
+   geom_histogram(position = "identity",
+                  aes(fill = Direction, color = Direction),
+                  bins = 10, alpha = 0.5) +
+   xlab("Time [min]")
```



Tento směr se opravdu jeví jako rychlejší. V obráceném směru plavalo jenom pět plavců, odhad vlivu směru na délku plavání by tak nemusel být spolehlivý. Nás primárně zajímá závislost délky plavání na věku a pohlaví, proto se omezíme jenom na data o plavcích, kteří plavali z jihu na sever.

```
> swim <- swim[swim$Direction == "SN", -8]
> summary(swim)

               Name     Sex          Age           Start.Day
 Colleen Shields: 3   F:33   Min.   :14.05   Min.   : 1.00
 Jim Woods      : 2   M:26   1st Qu.:19.50   1st Qu.: 8.00
 John Scott     : 2          Median :28.00   Median :12.00
 Kim Lumsdon    : 2          Mean   :31.28   Mean   :13.73
 Vicki Keith    : 2          3rd Qu.:39.00   3rd Qu.:18.50
 Angela Kondrak : 1          Max.   :66.57   Max.   :31.00
 (Other)        :47
  Month          Year           Time
 July: 5   Min.   :1954   Min.   : 829
 Aug :46   1st Qu.:1978   1st Qu.: 981
 Sep : 8   Median :1993   Median :1163
           Mean   :1993   Mean   :1228
           3rd Qu.:2008   3rd Qu.:1380
           Max.   :2018   Max.   :2461
```

```
> with(swim,
+      boxplot(Time, xlab = "Time [min]",
+              col = "lightgreen", horizontal = TRUE)
+ )
```



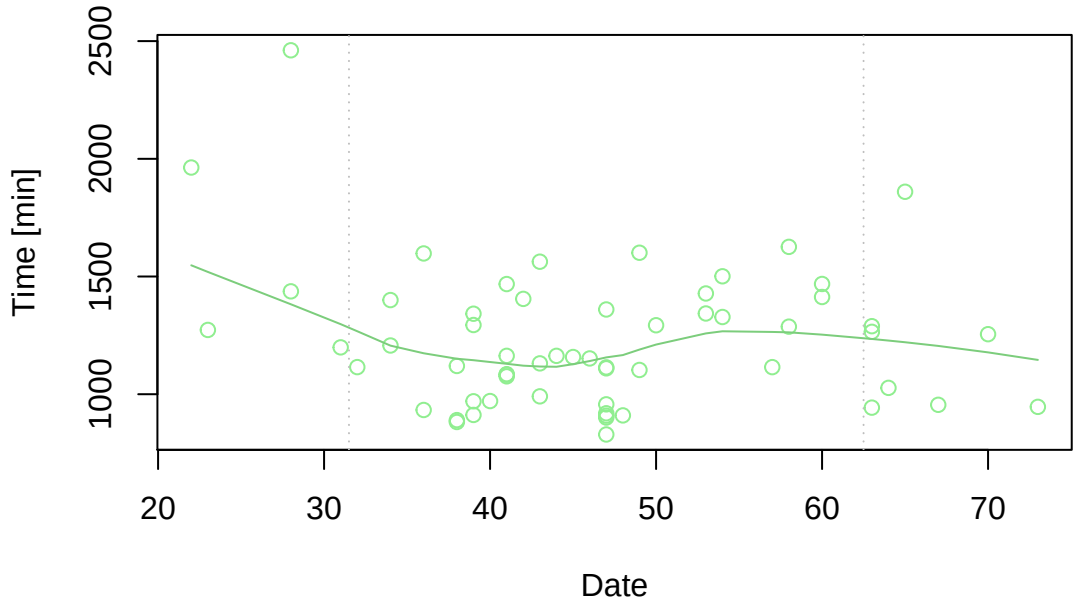
Proměnné Month a Start.Day společně obsahují informaci o datu, kdy plavec zahájil úspěšný pokus o přeplavání jezera. Datum můžeme také uvažovat jako kvantitativní prediktor, například tak, že budeme uvažovat, kolik dnů uběhlo od začátku července v den, kdy plavec zahájil svůj pokus.

```
> swim$Date <- NA
> swim$Date[swim$Month == "July"] <- 0
> swim$Date[swim$Month == "Aug"] <- 31
> swim$Date[swim$Month == "Sep"] <- 62
> swim$Date <- swim$Date + swim$Start.Day
```

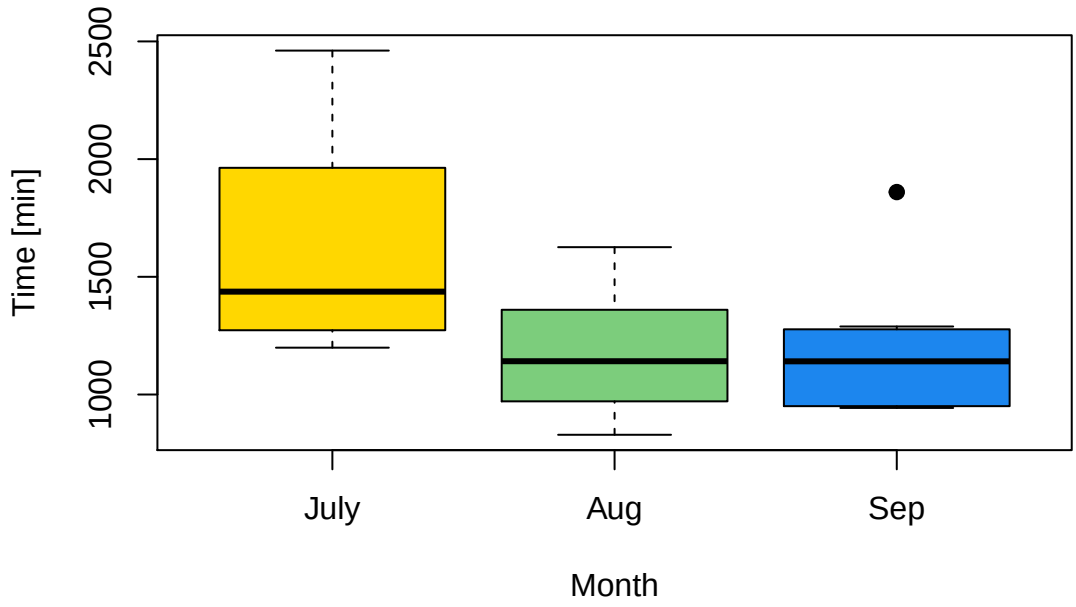
Nyní se můžeme podívat na souvislost mezi časem Time a datem plavby Date.

```
> with(swim, plot(Time ~ Date, ylab = "Time [min]",
+                 col="lightgreen"))
> with(swim, lines(lowess(Time ~ Date), col = "palegreen3"))
```

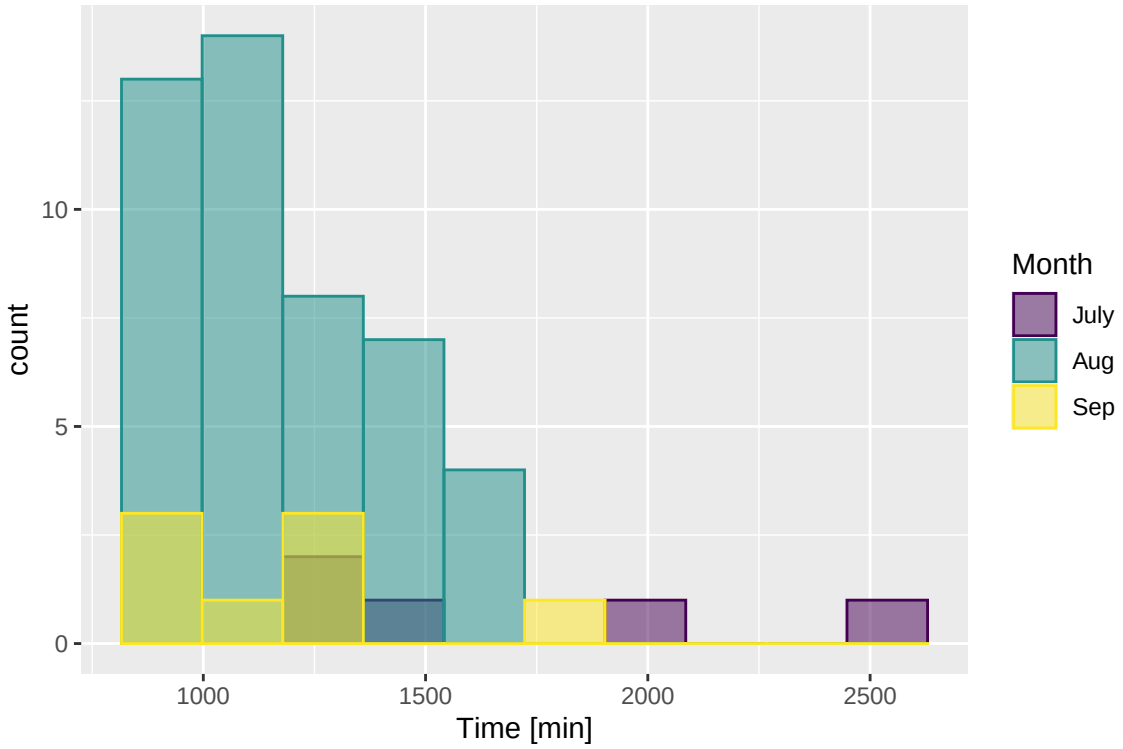
```
> abline(v = 31.5, col = "grey", lty = 3)
> abline(v = 62.5, col = "grey", lty = 3)
```




```
> with(swim,
+      boxplot(Time ~ Month, ylab = "Time [min]",
+              col = c("gold", "palegreen3", "dodgerblue2"),
+              pch = 19)
+ )
```




```
> ggplot(data = swim, aes(x = Time, color = Month)) +
+   geom_histogram(position = "identity",
+                  aes(fill = Month, color = Month),
+                  bins = 10, alpha = 0.5) +
+   xlab("Time [min]")
```



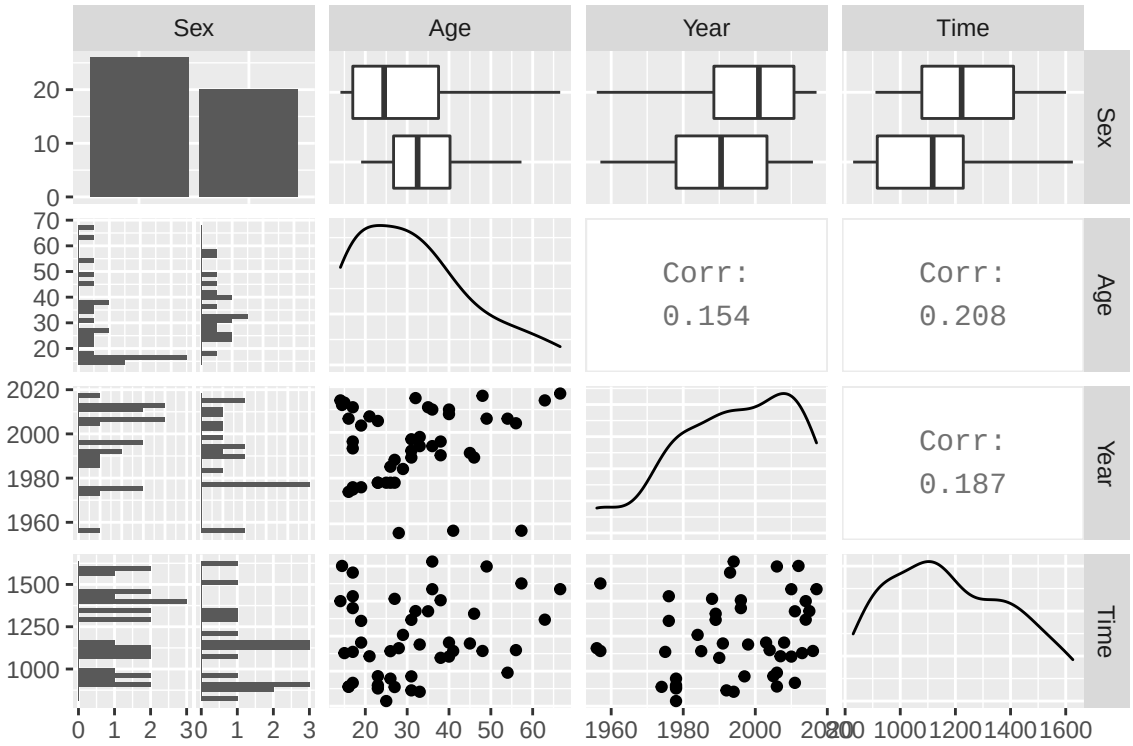
Na grafech výše vidíme, že nejvíce úspěšných pokusů o překonání jezera se uskutečnilo v srpnu, což může souviset s vhodnějším počasím. Časy přeplavání jezera v červenci a v září, kdy bylo úspěšných pokusů o poznání méně, vypadají rozptýlenější než v srpnové časy. Červencové časy navíc vypadají delší. Z podobných důvodů jako u směru plavání může i u data plavání být rozumné omezit se u analýzy vlivu věku a pohlaví na čas přeplavání jezera na srpnové pokusy, kdy časy plavání nejsou natolik ovlivněny nesouvisejícími faktory, na odhadnutí jejichž efektů nemáme dostatek dat.

```
> swim <- swim[swim$Month == "Aug", -c(4, 5, 8)]
> summary(swim)

                Name     Sex          Age              Year
 Colleen Shields : 3   F:26   Min.   :14.05   Min.   :1956
 John Scott      : 2   M:20   1st Qu.:19.50   1st Qu.:1980
 Kim Lumsdon     : 2          Median :30.00   Median :1995
 Angela Kondrak  : 1          Mean   :31.66   Mean   :1994
 Annaleise Carr  : 1          3rd Qu.:39.50   3rd Qu.:2008
 Ashleigh Beacham: 1          Max.   :66.57   Max.   :2017
 (Other)         :36
      Time
 Min.   : 829
 1st Qu.: 976
 Median :1142
```

```
 Mean   :1187
 3rd Qu.:1356
 Max.   :1626
```

Nyní se podíváme na vztahy mezi zbylými proměnnými.

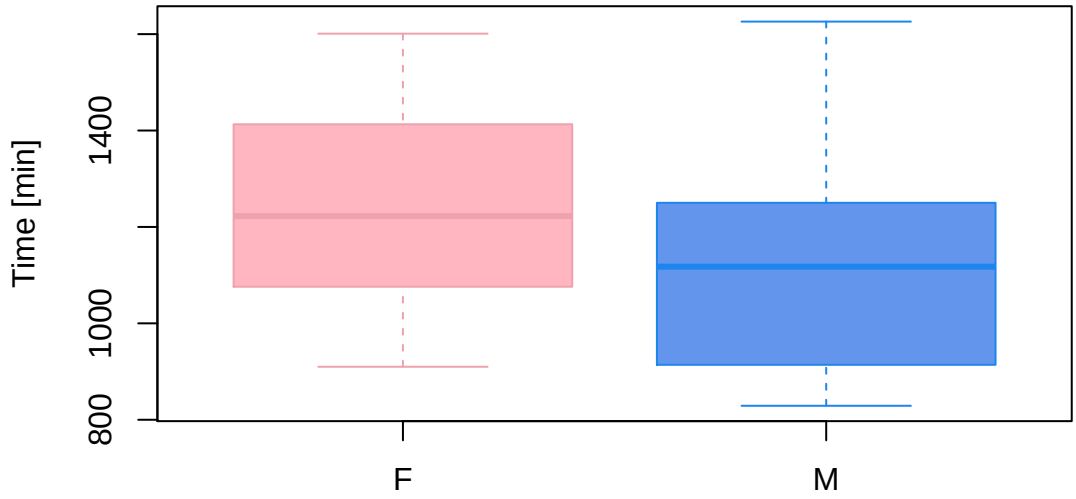
```
> ggpairs(swim[, -1])
```



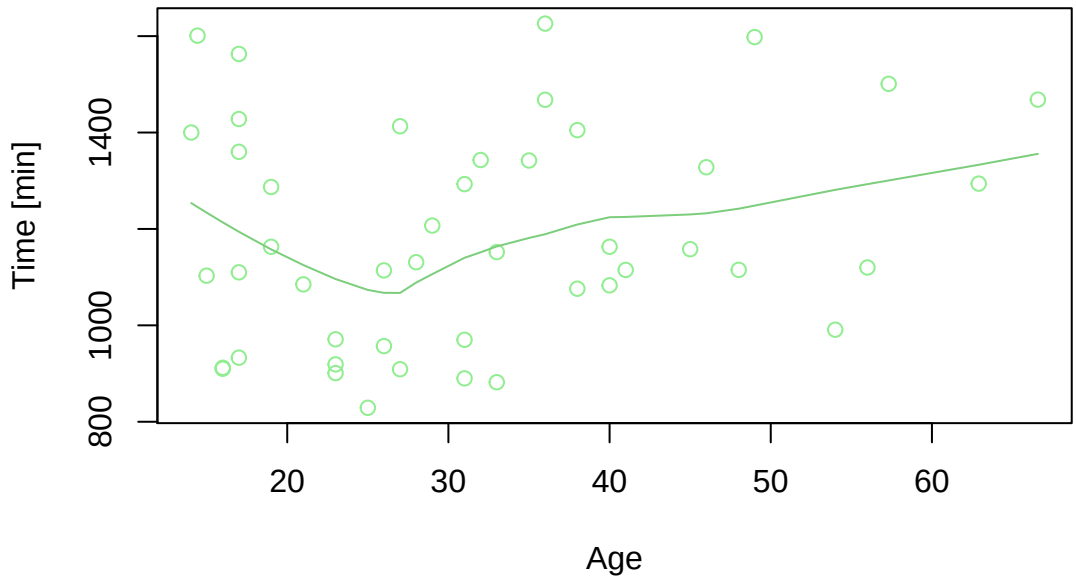
Grafy nenaznačují žádné zřejmé problémy s daty ani žádné výrazné závislosti. Nejspíše se tedy nemusíme obávat multikolinearity, ale také nemůžeme očekávat vysvětlení podstatnější části variability časů přeplavání jezera pomocí věku a pohlaví plavce anebo roku, ve kterém svůj pokus uskutečnil.

Podíváme se ještě podrobněji na závislost mezi plánovanou odezvou a prediktory.

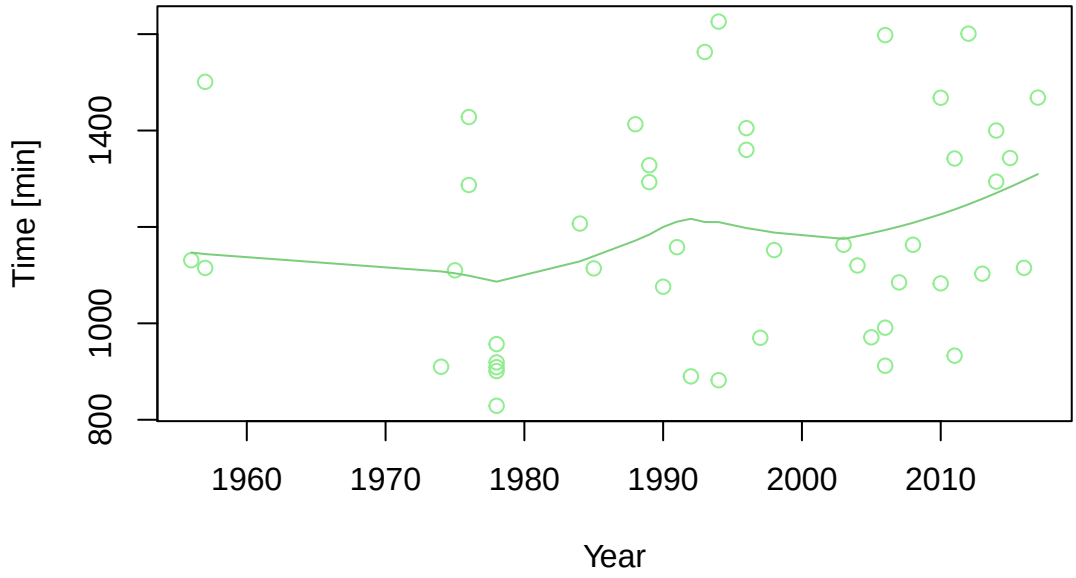
```
> with(swim,
+      boxplot(Time ~ Sex, ylab = "Time [min]", xlab = "",
+              col = c("lightpink", "cornflowerblue"),
+              border = c("lightpink2", "dodgerblue2"),
+              pch = 19)
+ )
```




```
> with(swim, plot(Time ~ Age, ylab = "Time [min]",
+                 col="lightgreen"))
> with(swim, lines(lowess(Time ~ Age), col = "palegreen3"))
```




```
> with(swim, plot(Time ~ Year, ylab = "Time [min]",
+                 col="lightgreen"))
> with(swim, lines(lowess(Time ~ Year), col = "palegreen3"))
```



Grafy nenaznačují žádné složité závislosti, proto se podíváme na jednoduchý model

$$\mathsf{Time}_i = \beta_0 + \beta_1 \times \mathsf{Sex}_i + \beta_2 \times \mathsf{Age}_i + \beta_3 \times \mathsf{Year}_i + \varepsilon_i, \quad i = 1, \ldots, n.$$

```
> model.1 <- lm(Time ~ Sex + Age + Year, data = swim)
> summary(model.1)


Call:
lm(formula = Time ~ Sex + Age + Year, data = swim)

Residuals:
   Min     1Q Median     3Q    Max
-355.2 -156.6  -48.1  172.4  491.5

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1298.371   4063.641  -0.320   0.7509
SexM         -123.696     68.047  -1.818   0.0762 .
Age             4.084      2.447   1.669   0.1026
Year            1.208      2.042   0.592   0.5571
```

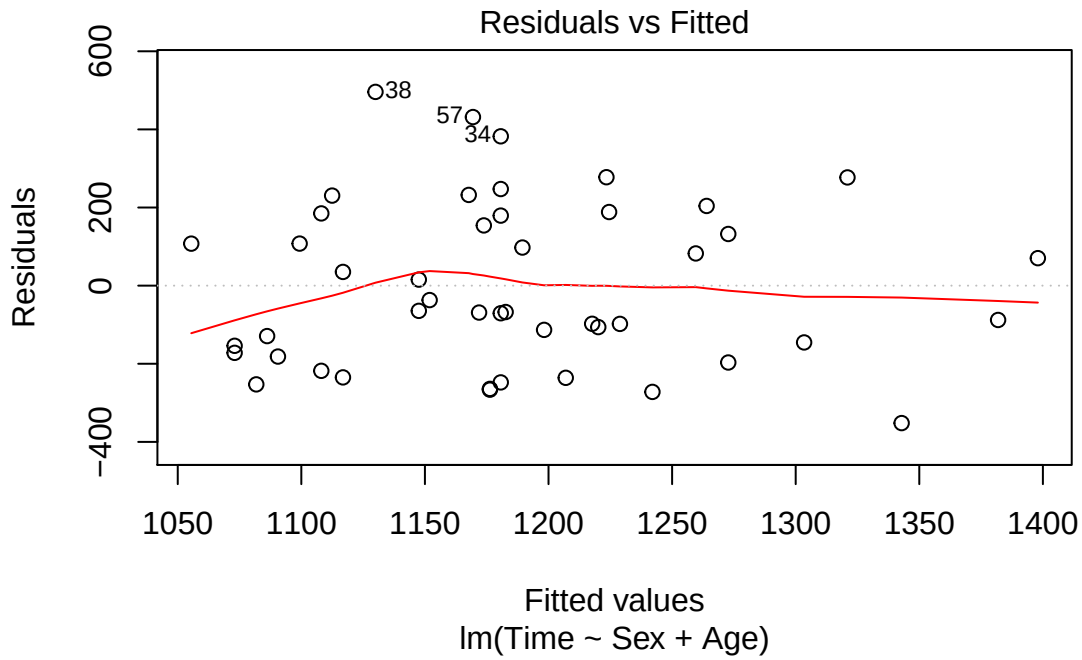
```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 216.6 on 42 degrees of freedom
Multiple R-squared:  0.1356,Adjusted R-squared:  0.07388
F-statistic: 2.197 on 3 and 42 DF,  p-value: 0.1026
```

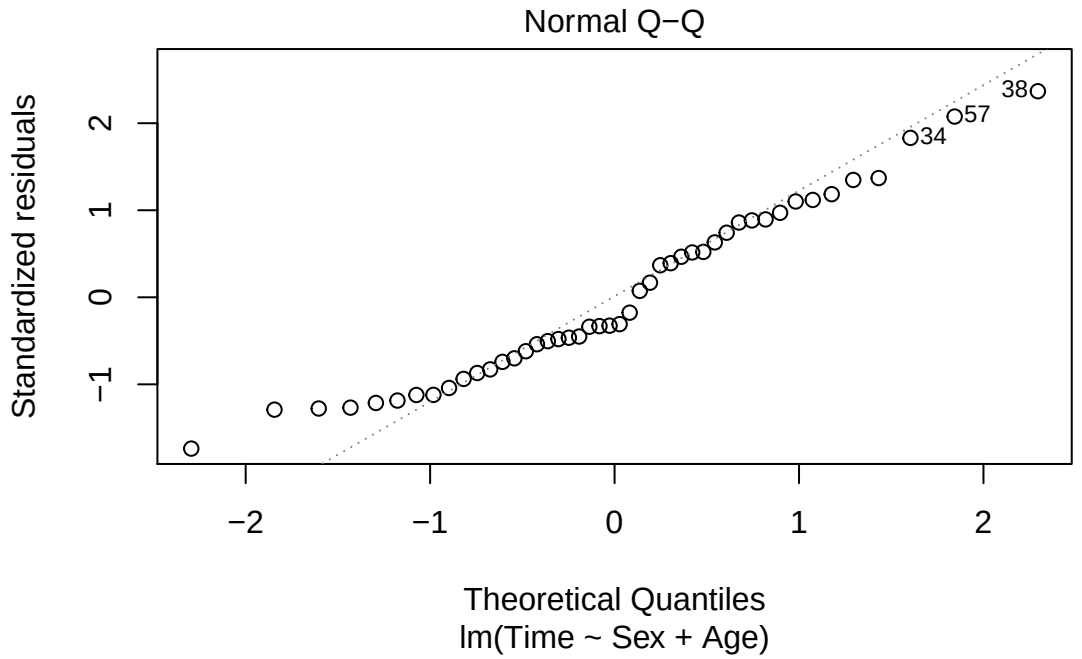
Model vysvětluje jen malé procento variability dat, což jsme ale očekávali. Test nulovosti všech koeficientů zároveň vyšel nevýznamný, prediktory tedy buď to nemají velký vliv na střední hodnotu odezvy, anebo data nemají dostatečnou sílu na prokázání tak malých efektů. Zkusíme ještě vynechat nevýznamný prediktor Year, abychom se soustředili na prediktory, které nás zajímají.

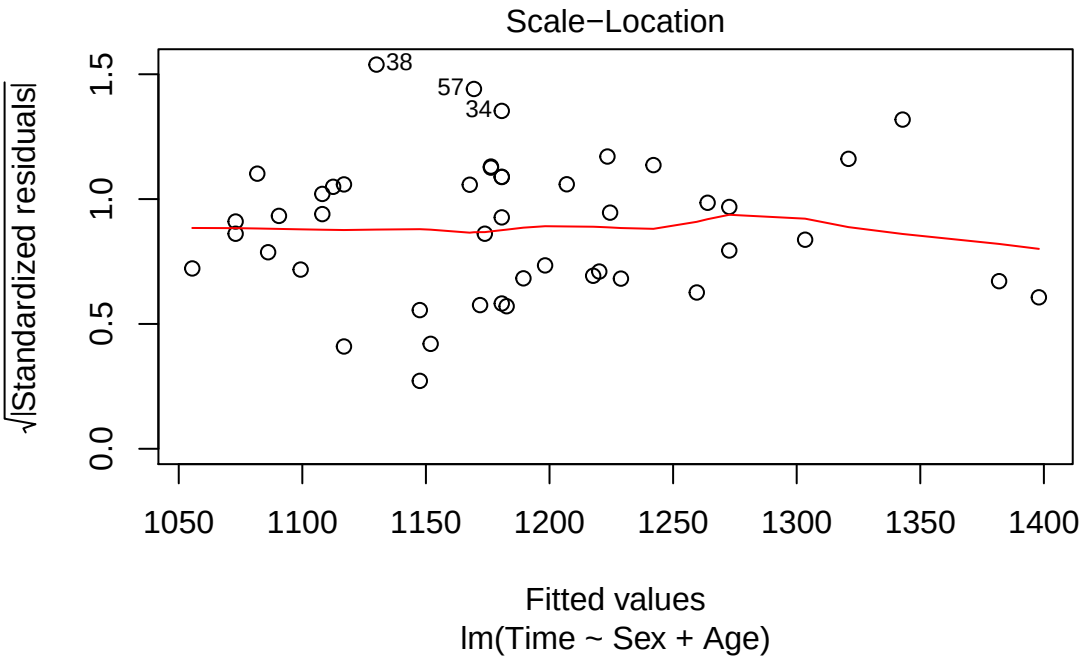
```
> model.2 <- lm(Time ~ Sex + Age, data = swim)
> summary(model.2)


Call:
lm(formula = Time ~ Sex + Age, data = swim)

Residuals:
    Min      1Q  Median      3Q     Max
-351.86 -167.51  -66.03  173.03  496.01

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1106.163     81.241  13.616   <2e-16 ***
SexM        -133.967     65.298  -2.052   0.0463 *
Age            4.383      2.376   1.845   0.0720 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 215 on 43 degrees of freedom
Multiple R-squared:  0.1284,Adjusted R-squared:  0.08788
F-statistic: 3.168 on 2 and 43 DF,  p-value: 0.05208
```

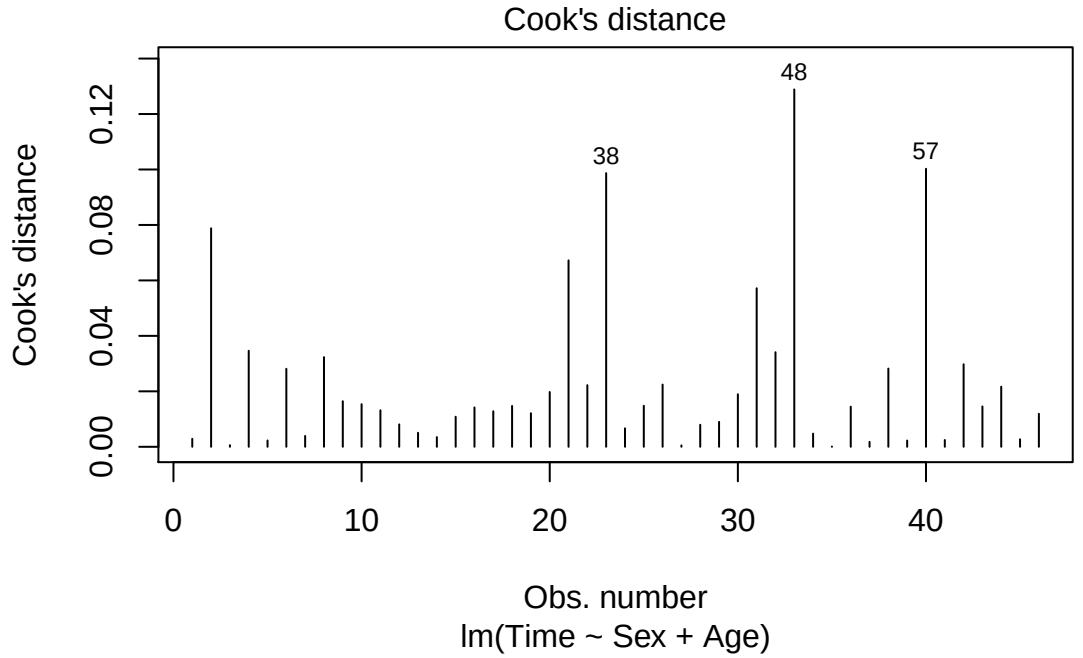
I tento model vysvětluje jen malé procento variability dat, adjustovaný koeficient determinace se ale mírně zlepšil a test o nulovosti obou prediktorů současně je nyní na hranici signifikance. Také testy o nulovosti jednotlivých prediktorů dávají přesvědčivější výsledky, což může být docíleno soustředěním veškeré síly na dva prediktory. Abychom se na výsledky testů mohli spolehnout, musí model splňovat předpoklady. Podíváme se proto na diagnostiku modelu.

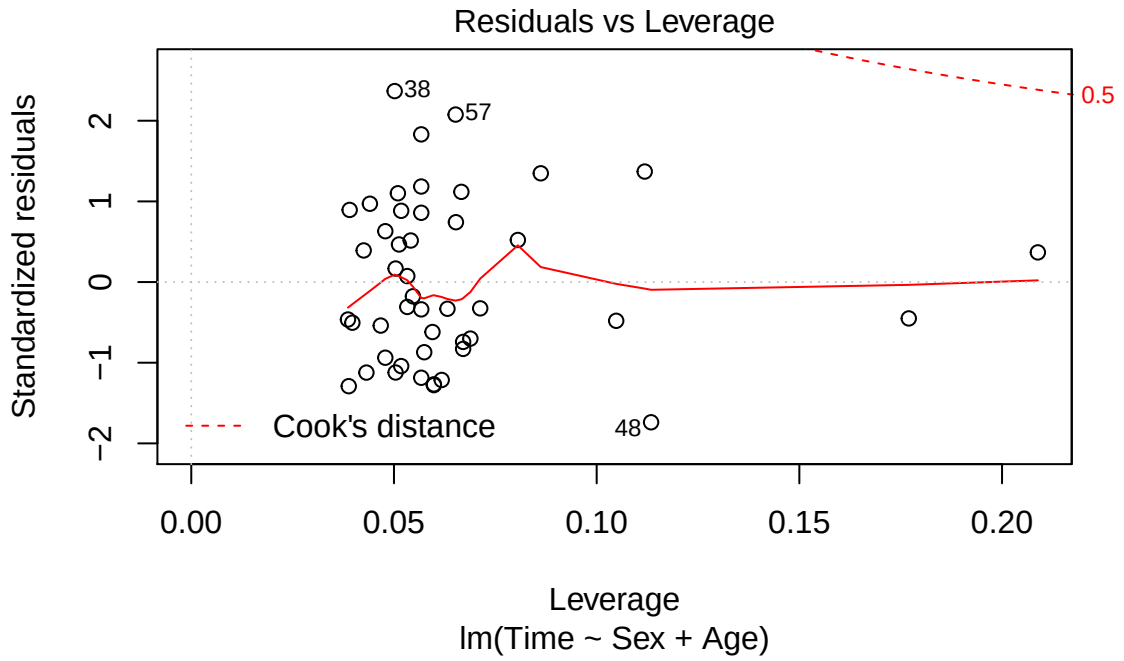
```
> plot(model.2, which = c(1:6))
```

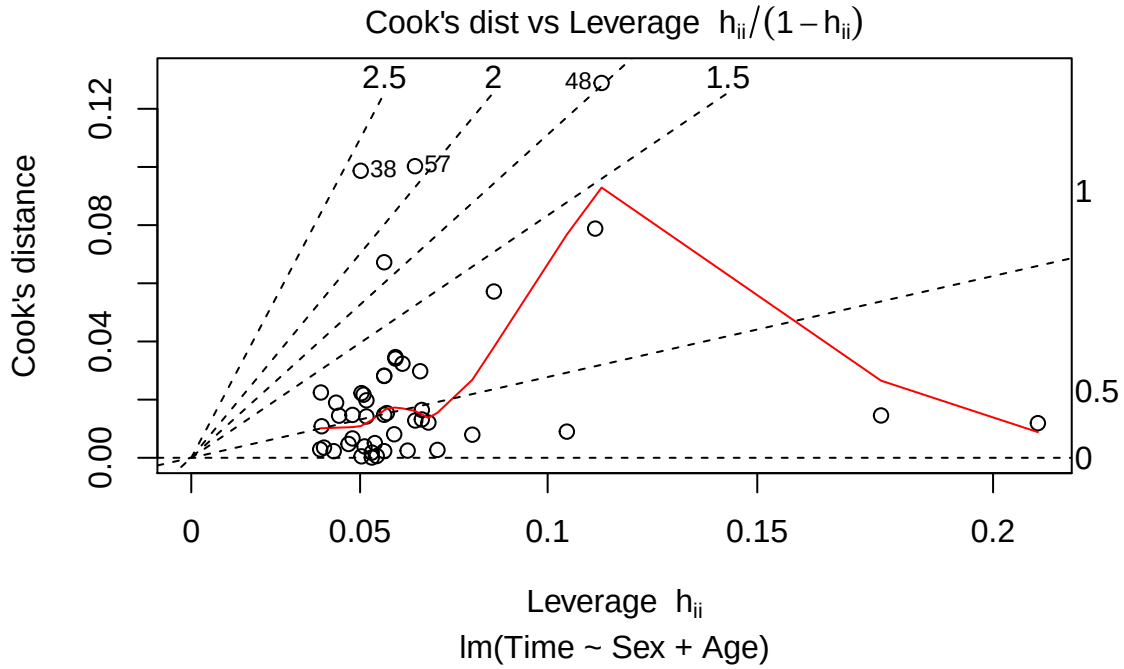





Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
38
57
34
1050
1100
1150
1200
1250
1300
1350
1400
Fitted values
lm(Time ~ Sex + Age)


Cook's distance
Cook's distance
0.00
0.04
0.08
0.12
48
38
57
0
10
20
30
40
Obs. number
lm(Time ~ Sex + Age)

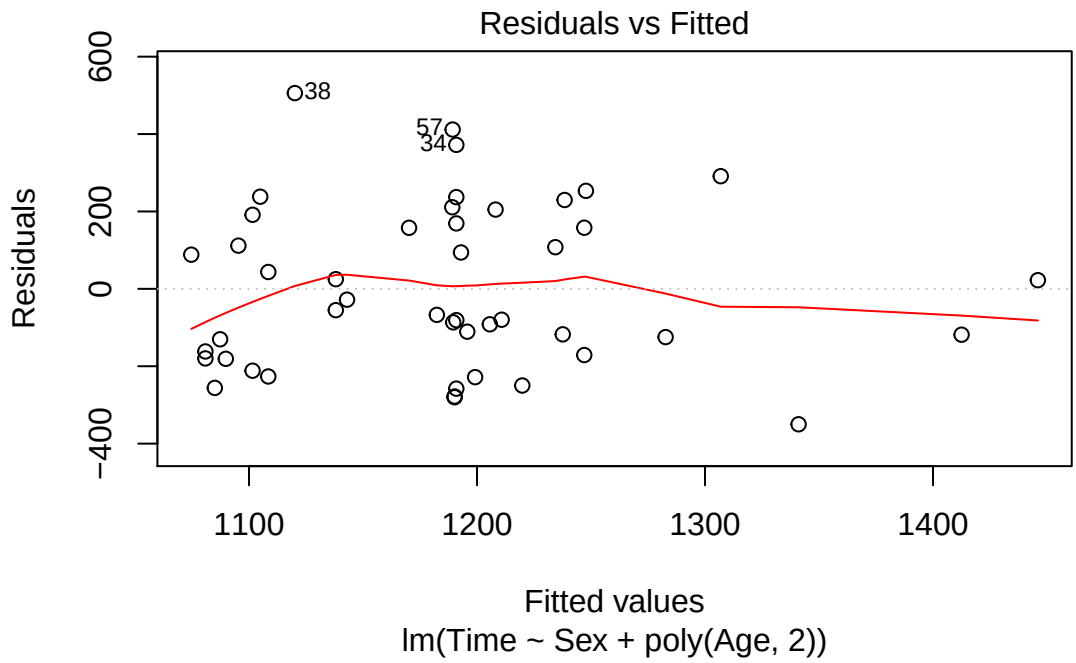


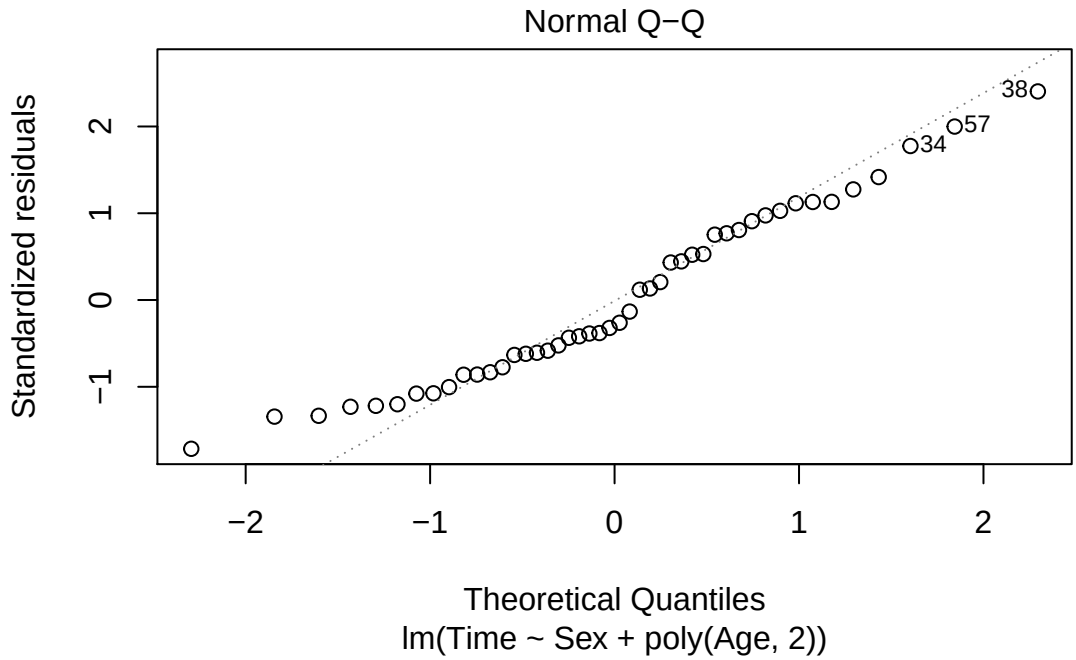


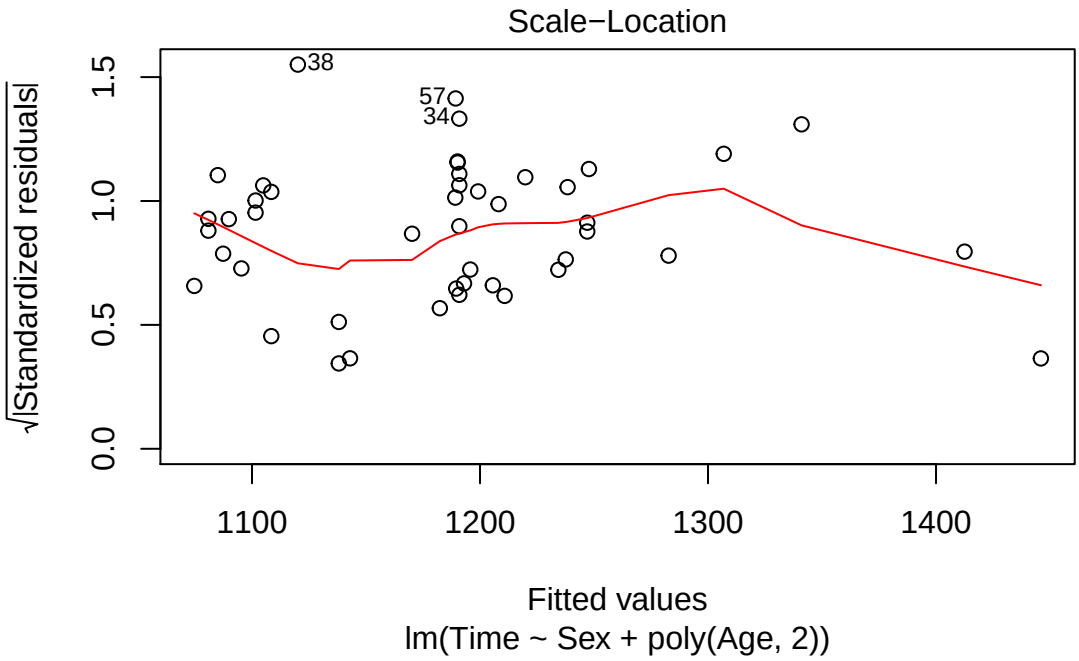
Na diagnostických grafech nevidíme žádné zásadní problémy, menší problémy se vyskytují jenom u nejrychlejších časů.

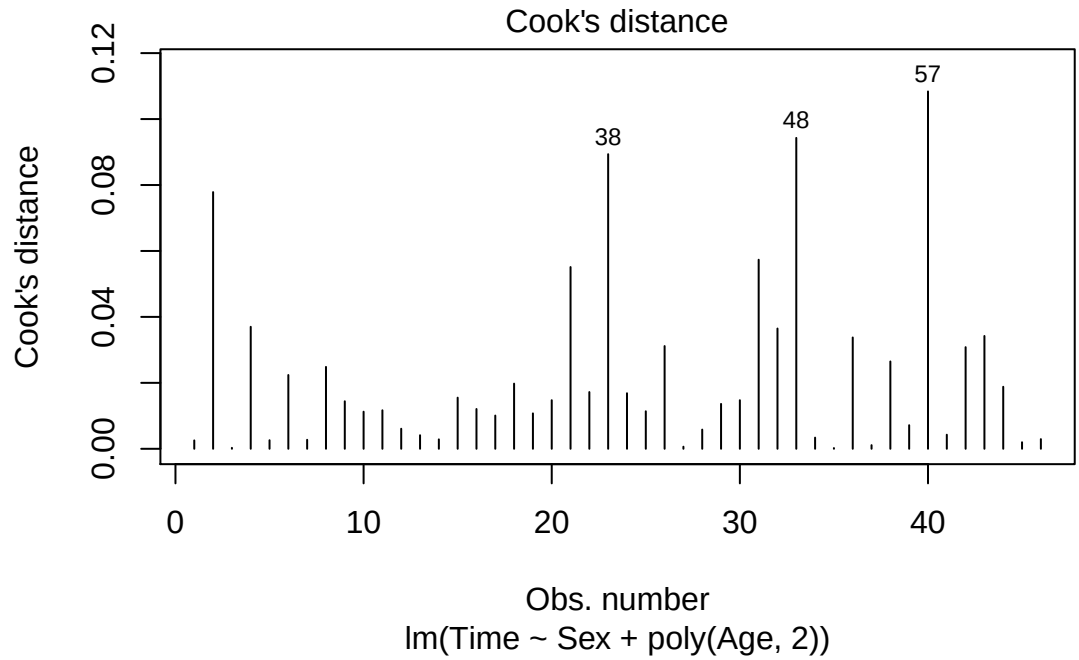
Můžeme ještě vyzkoušet, zda si nepolepšíme zahrnutím kvadratické závislosti na věku, kterou by mohl naznačovat již graf z deskriptivní fáze analýzy.

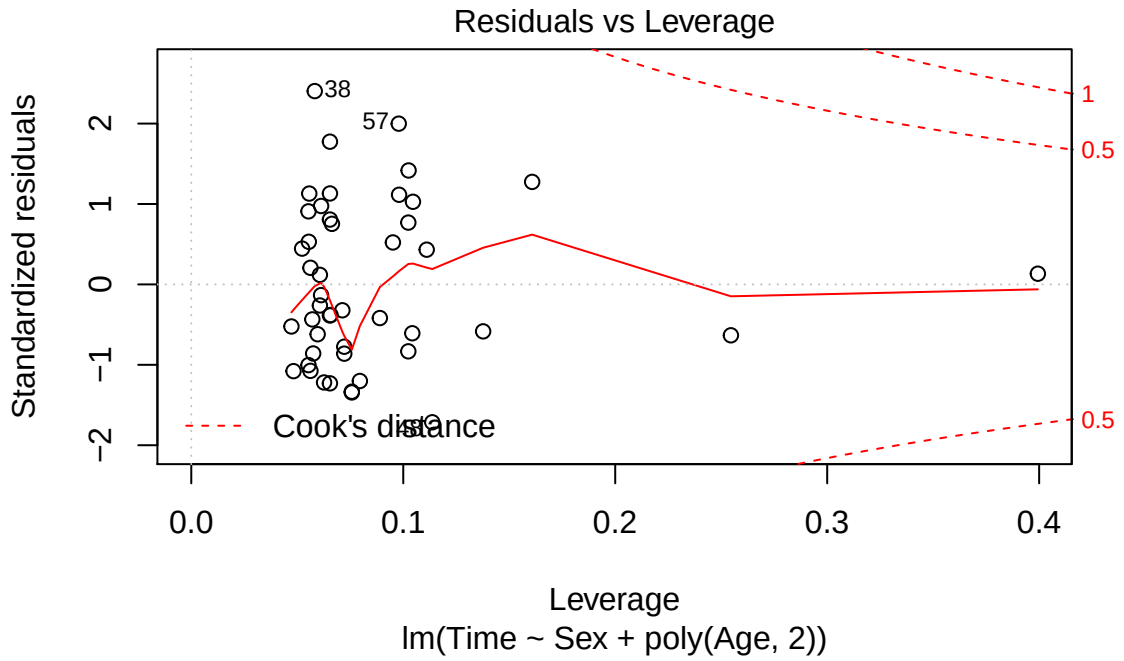
```
> model.3 <- lm(Time ~ Sex + poly(Age, 2), data = swim)
> summary(model.3)


Call:
lm(formula = Time ~ Sex + poly(Age, 2), data = swim)

Residuals:
    Min      1Q  Median      3Q     Max
-350.03 -168.75  -61.16  166.29  505.89

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    1238.12      44.98  27.529   <2e-16 ***
SexM           -118.30      72.75  -1.626   0.1114
poly(Age, 2)1   394.32     222.47   1.772   0.0836 .
poly(Age, 2)2   121.47     239.50   0.507   0.6147
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 216.8 on 42 degrees of freedom
Multiple R-squared:  0.1337,Adjusted R-squared:  0.07184
F-statistic: 2.161 on 3 and 42 DF,  p-value: 0.1069

> plot(model.3, which = c(1:6))
```

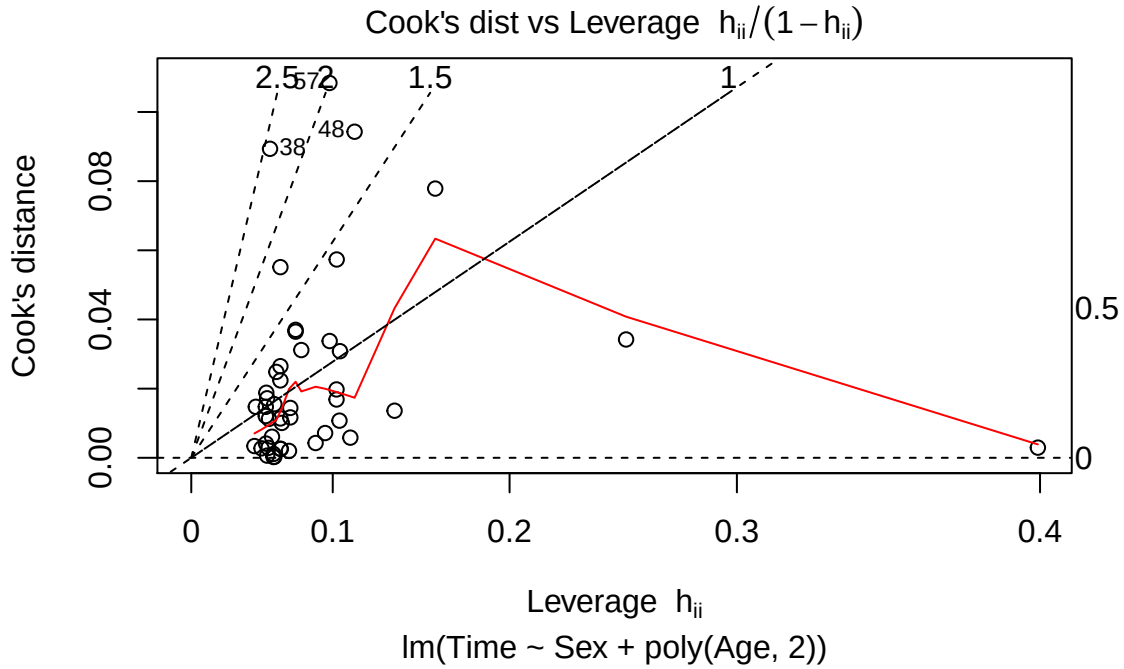

Residuals vs Fitted
Residuals
600
200
0
−400
38
57
34
1100
1200
1300
1400
Fitted values
lm(Time ~ Sex + poly(Age, 2))


Normal Q−Q
Standardized residuals
2
1
0
−1
38
57
34
−2
−1
0
1
2
Theoretical Quantiles
lm(Time ~ Sex + poly(Age, 2))


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
38
57
34
1100
1200
1300
1400
Fitted values
lm(Time ~ Sex + poly(Age, 2))


Cook's distance
Cook's distance
0.00
0.04
0.08
0.12
38
48
57
0
10
20
30
40
Obs. number
lm(Time ~ Sex + poly(Age, 2))

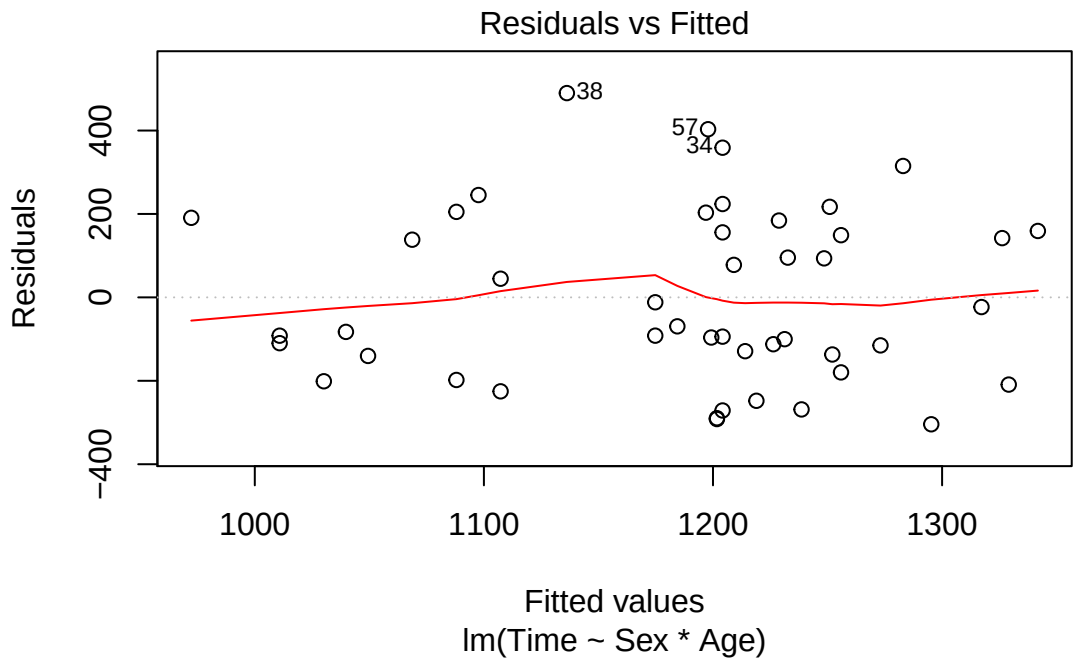


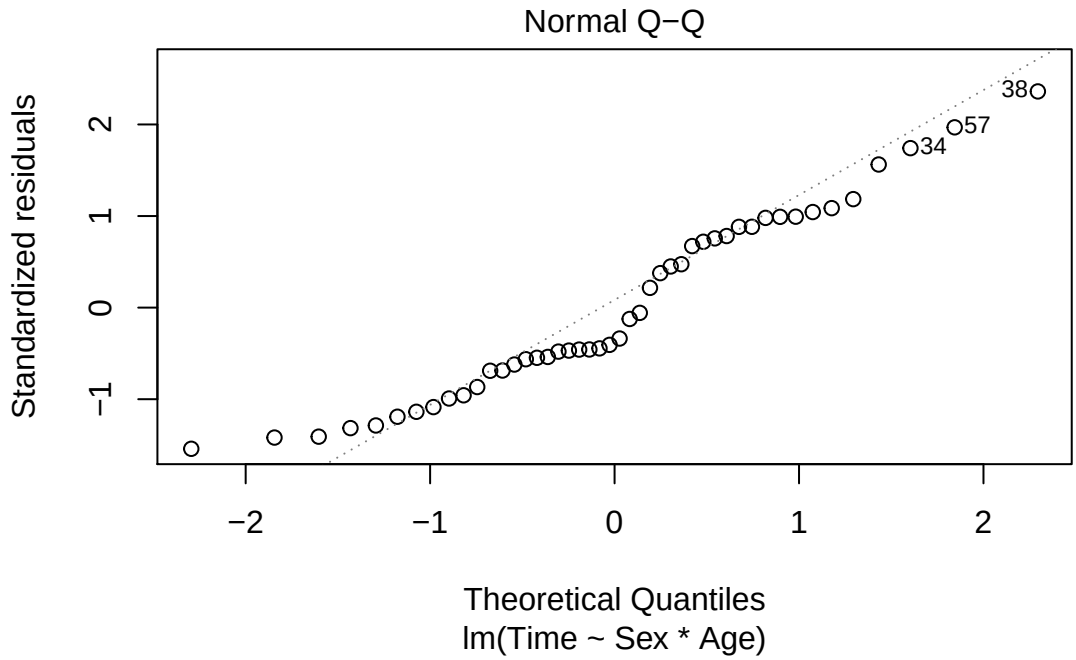


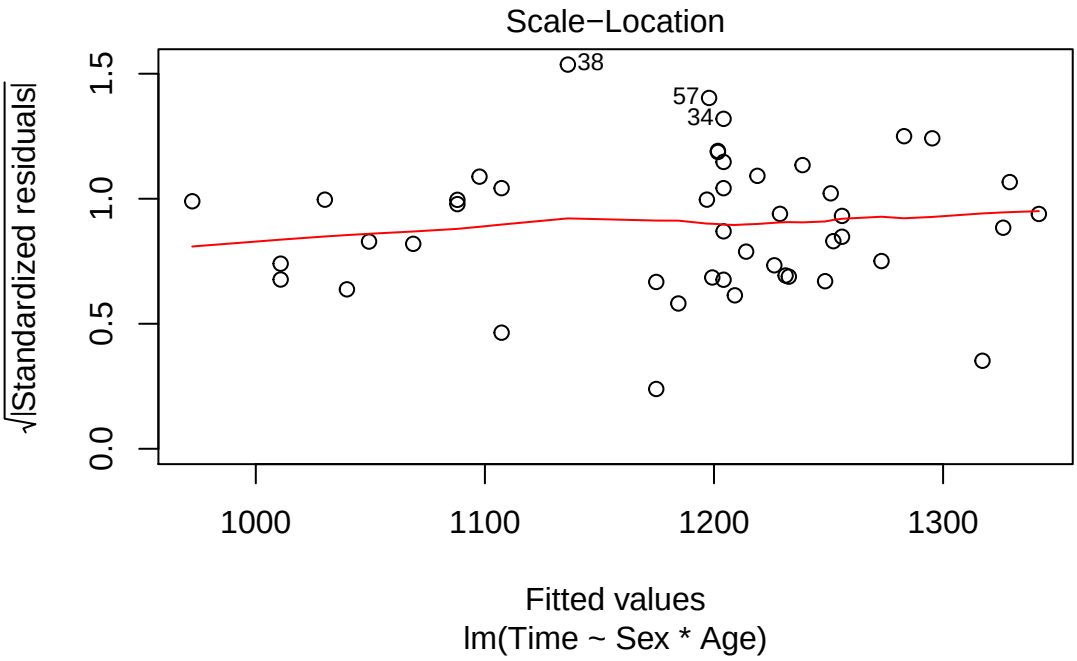
Kvadratický efekt není signifikantní a ani diagnostické grafy se nezlepšily. Zůstaneme proto u lineární závislosti.

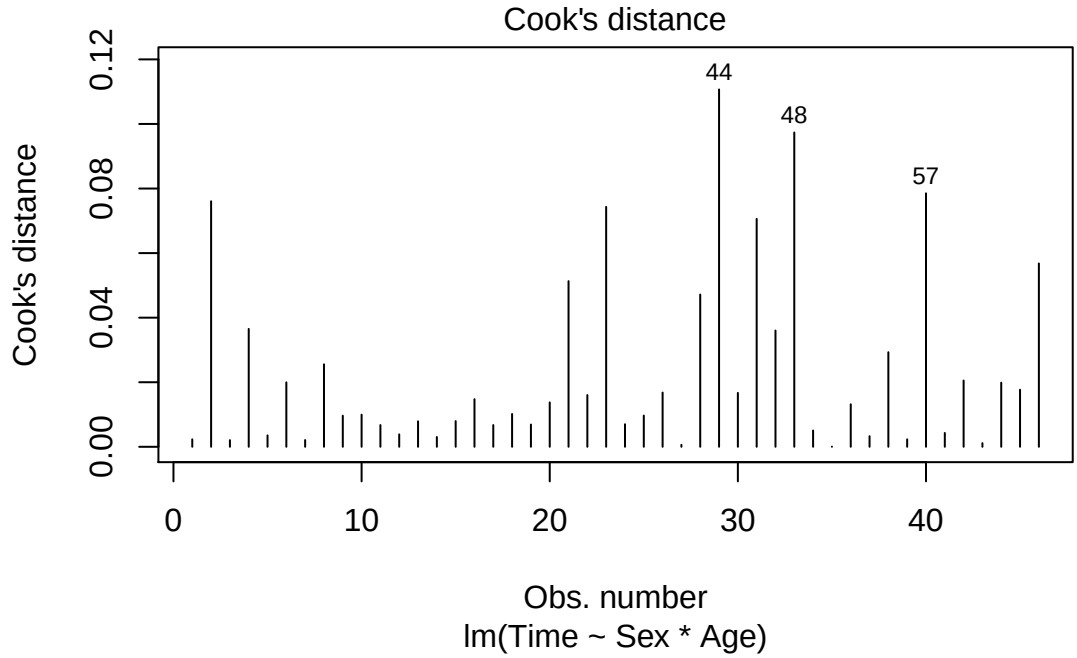
Nakonec ještě můžeme prozkoumat potřebu interakce mezi věkem a pohlavím.

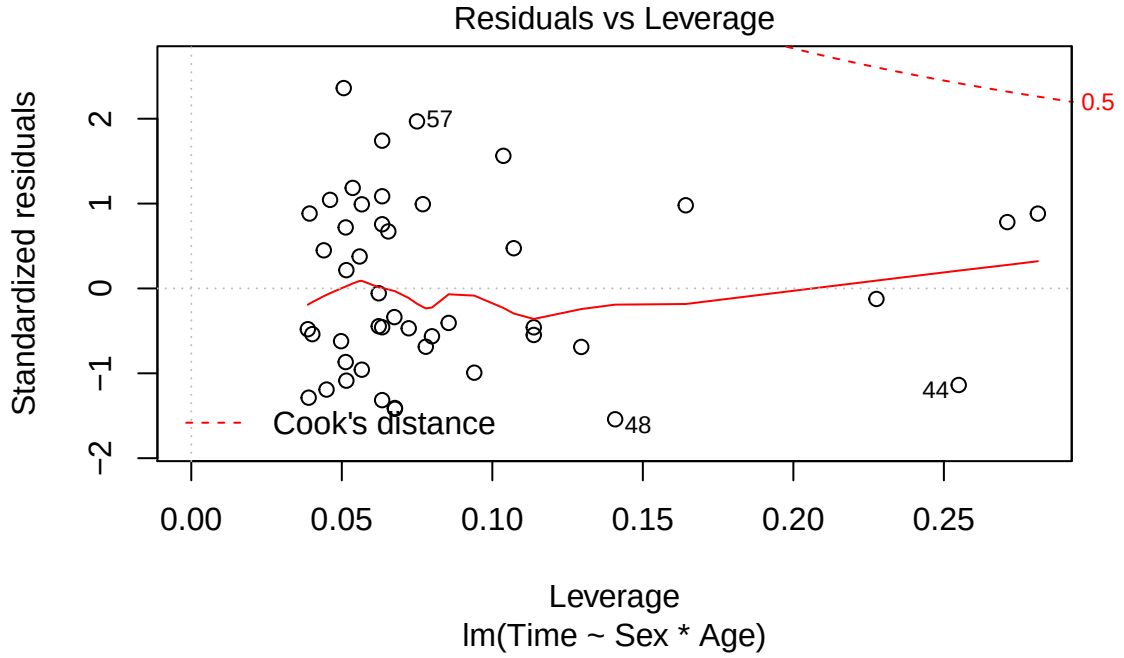
```
> model.4 <- lm(Time ~ Sex * Age, data = swim)
> summary(model.4)


Call:
lm(formula = Time ~ Sex * Age, data = swim)

Residuals:
    Min      1Q  Median      3Q     Max
-304.27 -139.56  -76.11  158.36  489.78

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1162.319     90.578  12.832    4e-16 ***
SexM        -373.270    188.598  -1.979   0.0544 .
Age            2.462      2.750   0.895   0.3757
SexM:Age       7.182      5.317   1.351   0.1840
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 212.9 on 42 degrees of freedom
Multiple R-squared:  0.1647,Adjusted R-squared:  0.105
F-statistic:  2.76 on 3 and 42 DF,  p-value: 0.05391

> plot(model.4, which = c(1:6))
```

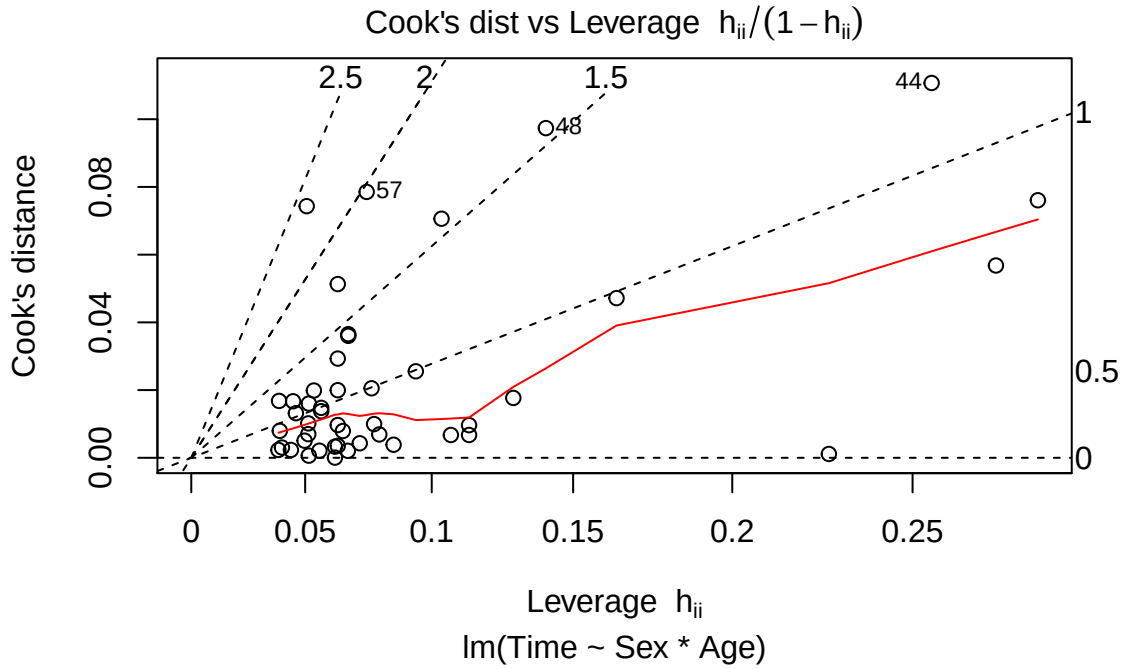

Residuals vs Fitted
Residuals
Fitted values
lm(Time ~ Sex * Age)
38
57
34
−400
0
200
400
1000
1100
1200
1300


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Time ~ Sex * Age)
38
57
34
−1
0
1
2
−2
−1
0
1
2


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
38
57
34
1000
1100
1200
1300
Fitted values
lm(Time ~ Sex * Age)


Cook's distance
Cook's distance
0.00
0.04
0.08
0.12
44
48
57
0
10
20
30
40
Obs. number
lm(Time ~ Sex * Age)





Ani toto rozšíření se neukázalo jako nevyhnutelné. Diagnostické grafy se o něco málo zlepšily, ale ani u menšího modelu nebyly výrazně problematické. Interakce přitom není ani hraničně

statisticky významná.

Na základě dat tedy pro čas přeplavání jezerem Ontario ve směru z jihu na sever v měsíci srpnu vybereme model

$$\mathsf{Time}_i = \beta_0 + \beta_1 \times \mathsf{Sex}_i + \beta_2 \times \mathsf{Age}_i + \varepsilon_i, \quad i = 1, \ldots, n.$$

Připomeňme si jeho výsledky.

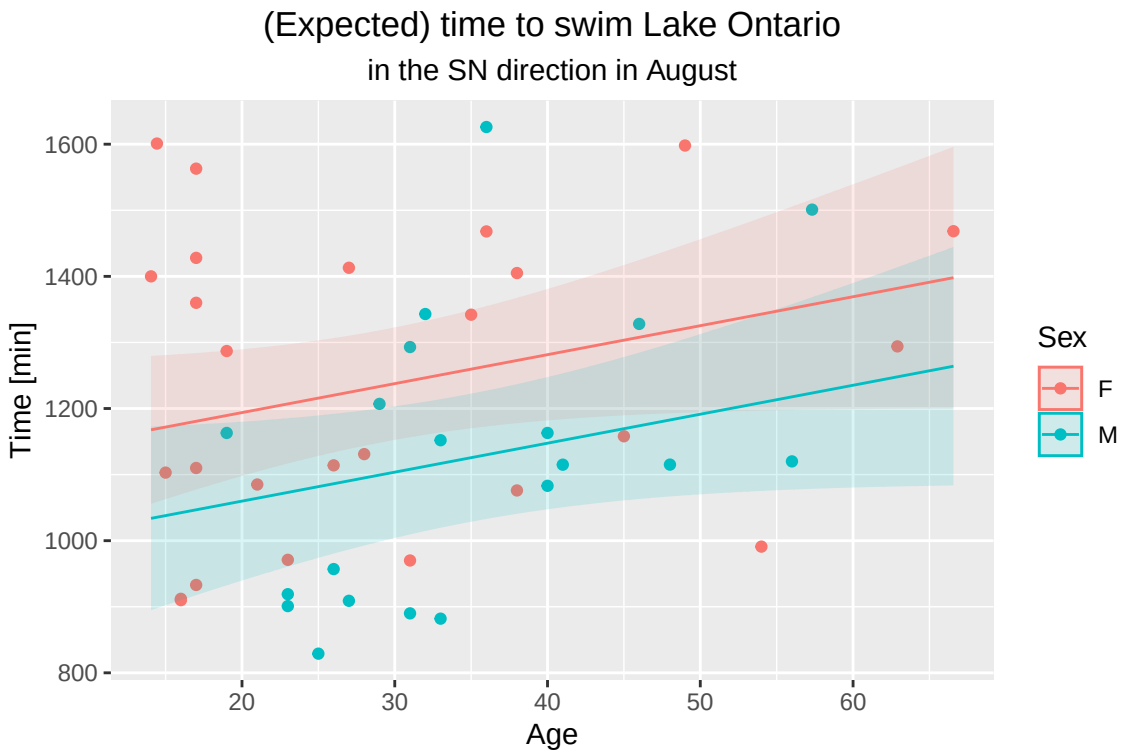
```
> summary(model.2)


Call:
lm(formula = Time ~ Sex + Age, data = swim)

Residuals:
    Min      1Q  Median      3Q     Max
-351.86 -167.51  -66.03  173.03  496.01

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1106.163     81.241  13.616   <2e-16 ***
SexM        -133.967     65.298  -2.052   0.0463 *
Age            4.383      2.376   1.845   0.0720 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 215 on 43 degrees of freedom
Multiple R-squared:  0.1284,Adjusted R-squared:  0.08788
F-statistic: 3.168 on 2 and 43 DF,  p-value: 0.05208
```

Na základě modelu můžeme říci, že muži jezero přeplavou v průměru o 134 minut rychleji než ženy stejného věku, a plavcům stejného pohlaví trvá přeplavání jezera s každým rokem věku v průměru o 4 minuty více. Věk a pohlaví ale vysvětlují pouhých $13\,\%$ variability mezi časy přeplavání jezera. Podstatnější část variability mezi výkony jednotlivých plavců tedy nesouvisí s jejich věkem ani pohlavím.

Vliv pohlaví jsme prokázali pětiprocentní hladině, zatímco vlik věku jenom na desetiprocentní. I když diagnostické grafy nedávají důvod na pochybnosti o platnosti testů, těmto poměrně hraničním výsledkům nemůžeme připisovat plnou váhu, jelikož jsme si na deskriptivních statistikách pro jména plavců mojli všimnout, že dva plavci přeplavali jezero dvakrát a jeden třikrát. Jejich časy nemůžeme považovat za nezávislé, jak to vyžadují předpoklady lineárního modelu. I když se nejedná o velký počet závislostí, u celkového počtu 46 dat by bylo korektnější použít místo standardních testů pro lineární modely robustnější testy, které platí i v případě závislosti mezi pozorováními. Ty by $p-$hodnoty pravděpodobně o něco málo posunuly.

Na závěr ještě můžeme výsledky modelu vykreslit.

```
> xx <- seq(min(swim$Age), max(swim$Age), length=501)
> newdat <- expand.grid(Age = xx,
+                       Sex = factor(c("F", "M")))
> newdat <- cbind(newdat, Time = NA, Lw = NA, Up = NA)
> yy.conf <- predict(model.2, newdata = newdat,
+                    interval = "confidence")
> newdat[, c(3:5)] <- yy.conf
>
> ggplot(data = swim,
+        mapping = aes(x = Age, y = Time, color = Sex)) +
+   geom_point() +
+   geom_ribbon(data = newdat,
+               aes(ymin = Lw, ymax = Up,
+                   fill = Sex, color = NULL),
+               alpha = .15) +
+   geom_line(data = newdat) +
+   xlab("Age") + ylab("Time [min]") +
+   ggtitle("(Expected) time to swim Lake Ontario",
+           subtitle = "in the SN direction in August") +
+   theme(plot.title = element_text(hjust = 0.5),
+         plot.subtitle = element_text(hjust = 0.5))
```



Literatura

[1] Ilustrační data: Childhood respiratory disease. Dostupná z http://www.statsci.org/data/general/fev.html. cit. 29. 1. 2021.

[2] Ilustrační data: Successful Lake Ontario swims (Toronto). Dostupná z http://www.soloswims.com/swims.htm. cit. 29. 1. 2021.

[3] Anděl, J. (2007). *Základy matematické statistiky*. Matfyzpress, Druhé vydání.

[4] Bubeníček, J. (2019). Shapirův-Wilkův test. Bakalářská práce, Masarykova Univerzita.

[5] Faraway, J. J. (2014). *Linear Models with R*. Chapman & Hall/CRC, Druhé vydání.

[6] R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing.

[7] Wood, S. (2006). *Generalized Additive Models: An Introduction with R*. Chapman & Hall/CRC.

[8] Zvára, K. (2008). *Regrese*. Matfyzpress.