M7222 – 1. cvičení : **GLM01b** (*Porodní hmotnost novorozenců*)

V této části cvičení budeme pracovat s reálnými daty. Popis jednotlivých proměnných vstupních dat je v souboru `novorozenci.txt`, samotná data jsou uložena v souboru nazvaném `novorozenci.dat`.

Nejprve načteme popisný soubor. Tento soubor obsahuje pouze text a ten načteme pomocí příkazu `readLines()` (kterému musí předcházet příkaz `file()` a po něm následuje příkaz `close()`). Protože je příkaz `readLines()` v závorkách, ihned se zobrazí obsah souboru.

```
> fileTxt <- paste(data.library, "novorozenci.txt", sep = "")
> con <- file(fileTxt)
> (popis <- readLines(con))
```

```
 [1] "Porodni hmotnost novorozencu"
 [2] "============================"
 [3] "hmnov     porodni hmotnost novorozence v g"
 [4] "vyska     vyska matky v cm"
 [5] "hmmat     hmotnost matky v kg"
 [6] "prir      vahovy prirustek matky behem tehotenstvi v kg"
 [7] "pohlavi   pohlavi ditete 0=divka, 1=hoch"
 [8] "stav      stav matky, 1=svobodna,2=vdana,3=rozvedena,4=vdova"
 [9] "vzdmat    vzdelani matky 1=zakl,2=vyuc,3=stredosk,4=vysokosk"
[10] "vzdot     vzdelani otce  1=zakl,2=vyuc,3=stredosk,4=vysokosk           "
[11] ""
```

```
> close(con)
```

Samotná data získáme pomocí příkazu `read.table()`.

```
> fileDat <- paste(data.library, "novorozenci.dat", sep = "")
> data <- read.table(fileDat, header = F)
```

Pojmenujeme proměnné a z proměnných, které jsou kategoriální, utvoříme pomocí příkazu `factor()` proměnné typu faktor.

```
> names(data) <- c("hmnov", "vyska", "hmmat", "prir", "pohlavi", "stav", "vzdmat",
    "vzdot")
> data$pohlavi <- factor(data$pohlavi, labels = c("divka", "hoch"))
> data$stav <- factor(data$stav, labels = c("svobodna", "vdana", "rozvedena",
    "vdova"))
> data$vzdmat <- factor(data$vzdmat, labels = c("zakl.", "vyuc.", "stredosk.",
    "vysokosk."))
> data$vzdot <- factor(data$vzdot, labels = c("neuveden", "zakl.", "vyuc.",
    "stredosk.", "vysokosk."))
```

Příkazem `str()` vypíšeme strukturu datového rámce.

```
> str(data)
```

```
,data.frame,:   1476 obs. of  8 variables:
 $ hmnov  : int  3880 3780 3600 3440 3200 3450 3550 3650 3680 3540 ...
 $ vyska  : int  168 168 170 165 163 160 169 177 180 170 ...
 $ hmmat  : int  65 52 57 55 70 53 77 77 66 48 ...
 $ prir   : int  20 15 12 12 21 14 18 21 12 10 ...
 $ pohlavi: Factor w/ 2 levels "divka","hoch": 1 2 2 2 2 1 2 2 1 2 ...
 $ stav   : Factor w/ 4 levels "svobodna","vdana",..: 2 2 2 1 1 2 2 2 2 2 ...
 $ vzdmat : Factor w/ 4 levels "zakl.","vyuc.",..: 2 2 3 1 2 3 2 3 3 3 ...
 $ vzdot  : Factor w/ 5 levels "neuveden","zakl.",..: 3 3 3 2 3 4 5 3 4 3 ...
```

Abychom si udělali nějakou představu o datech, budeme se snažit data popsat pomocí popisných statistik. V knihovně `Hmisc` je celá řada užitečných funkcí, použijeme funkci `describe()`.

```
> library(Hmisc)
> Hmisc::describe(data)
```

```
data

 8  Variables      1476  Observations
---------------------------------------------------------------------------------------------------
hmnov
      n missing  unique    Mean     .05     .10     .25     .50     .75     .90     .95
   1476       0     208    3566    2880    3030    3250    3550    3850    4130    4320

lowest : 2540 2550 2570 2580 2590, highest: 4830 4880 4960 5000 6320
---------------------------------------------------------------------------------------------------
vyska
      n missing  unique    Mean     .05     .10     .25     .50     .75     .90     .95
   1476       0      43   167.1     157     160     163     168     171     175     178

lowest : 147 148 149 150 151, highest: 185 186 187 188 190
---------------------------------------------------------------------------------------------------
hmmat
      n missing  unique    Mean     .05     .10     .25     .50     .75     .90     .95
   1476       0      66   62.57      49      51      55      60      68      77      83

lowest :  30  40  41  42  43, highest: 103 104 105 107 125
---------------------------------------------------------------------------------------------------
prir
      n missing  unique    Mean     .05     .10     .25     .50     .75     .90     .95
   1476       0      35   13.91       7       9      11      14      17      20      22

lowest : -7 -1  0  1  2, highest: 28 29 30 33 35
---------------------------------------------------------------------------------------------------
pohlavi
      n missing  unique
   1476       0       2

divka (684, 46%), hoch (792, 54%)
---------------------------------------------------------------------------------------------------
stav
      n missing  unique
   1476       0       4

svobodna (162, 11%), vdana (1261, 85%), rozvedena (50, 3%), vdova (3, 0%)
---------------------------------------------------------------------------------------------------
```

```
vzdmat
      n missing  unique
   1476       0       4

zakl. (108, 7%), vyuc. (519, 35%), stredosk. (602, 41%), vysokosk. (247, 17%)
-------------------------------------------------------------------------------------------------------
vzdot
      n missing  unique
   1476       0       5

          neuveden zakl. vyuc. stredosk. vysokosk.
Frequency       19    64   624       424       345
%                1     4    42        29        23
-------------------------------------------------------------------------------------------------------
```

Vidíme, že datový soubor obsahuje 4 spojité proměnné a 4 kategoriální proměnné. Spojité proměnné lze ještě lépe popsat pomocí funkce `describe()` z knihovny `psych`.

```
> library(psych)
> psych::describe(data[, 1:4])
```

```
      var    n    mean     sd median trimmed    mad  min  max range skew kurtosis    se
hmnov   1 1476 3565.88 436.69   3550 3553.28 444.78 2540 6320  3780 0.42     0.83 11.37
vyska   2 1476  167.09   6.29    168  167.05   5.93  147  190    43 0.06     0.25  0.16
hmmat   3 1476   62.57  10.79     60   61.50   8.90   30  125    95 1.13     2.06  0.28
prir    4 1476   13.91   4.65     14   13.78   4.45   -7   35    42 0.33     1.05  0.12
```

Abychom se lépe podívali, jakých hodnot nabývají spojité proměnné (například hmotnost novorozence) uvnitř podskupin definovaných pomocí kategoriálních proměnných (například pohlavi), použijeme následující příkaz

```
> with(data, psych::describe.by(hmnov, pohlavi))
```

```
group: divka
  var   n    mean     sd median trimmed    mad  min  max range skew kurtosis    se
1   1 684 3473.74 420.55   3460  3458.2 415.13 2550 4880  2330 0.41     0.17 16.08
------------------------------------------------------------------------------
group: hoch
  var   n    mean     sd median trimmed    mad  min  max range skew kurtosis    se
1   1 792 3645.46 434.98   3635 3635.18 429.95 2540 6320  3780 0.43     1.38 15.46
```

Chceme–li zkoumat vztah dvou kategoriálních proměnných (například vzdělání matky a vzdělání otce), můžeme vytvořit dvourozměrnou kontingenční tabulku pomocí příkazu `table()` ze základní knihovy funkcí `base`

```
> (tabVZD <- table(data$vzdmat, data$vzdot))
```

```
           neuveden zakl. vyuc. stredosk. vysokosk.
  zakl.           6    30    52        17         3
  vyuc.           9    24   354       110        22
  stredosk.       3     9   196       244       150
  vysokosk.       1     1    22        53       170
```

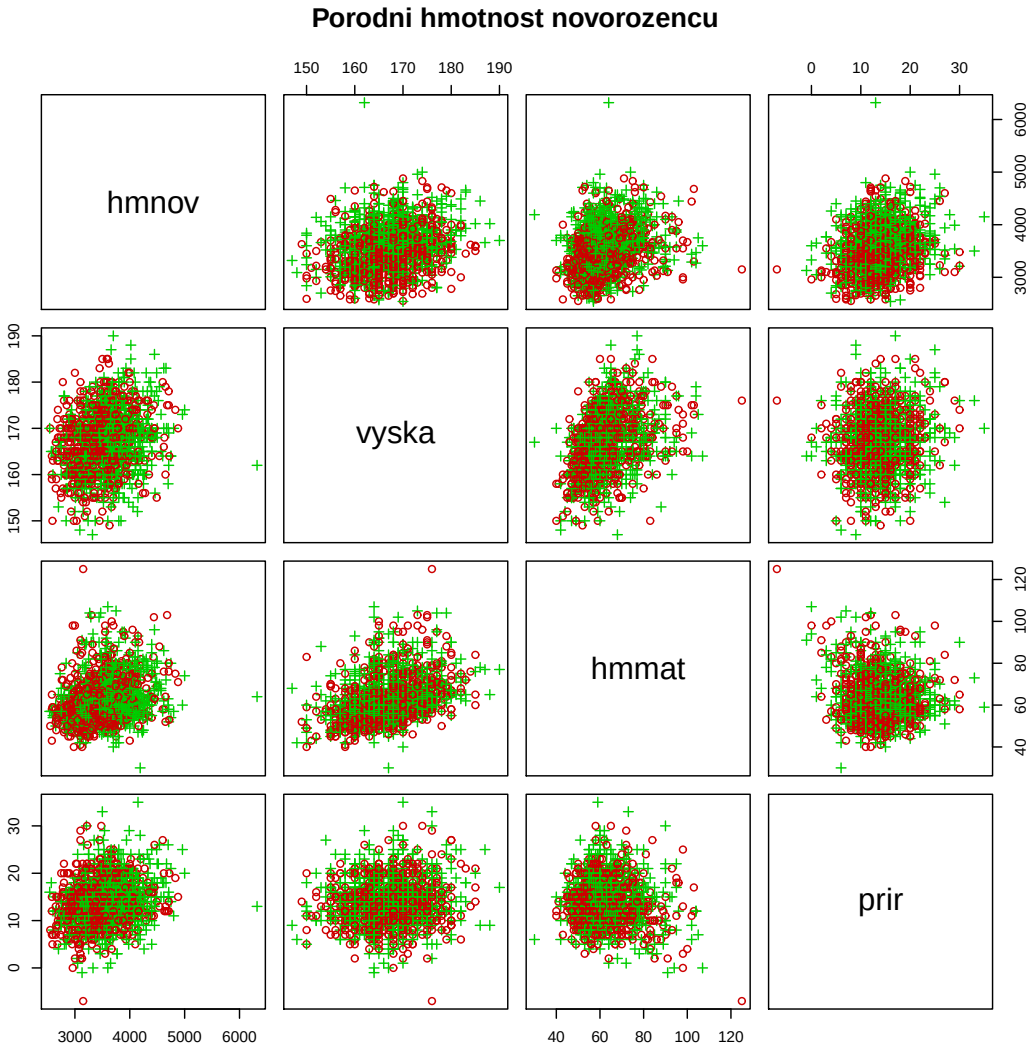
Stejný výsledek dostaneme pomocí příkazu

```
> with(data, table(vzdmat, vzdot))
```

```
          vzdot
vzdmat     neuveden zakl. vyuc. stredosk. vysokosk.
  zakl.           6    30    52        17         3
  vyuc.           9    24   354       110        22
  stredosk.       3     9   196       244       150
  vysokosk.       1     1    22        53       170
```

Grafickou představu o datech získáme pomocí příkazu `pairs()`. Všimněme si, jakým způsobem lze graficky odlišit zjistěné hodnoty pro obě pohlaví. (Pro pochopení příkazů použijte `?unclass`).

```
> pairs(data[1:4], main = popis[1], pch = c(1, 3)[unclass(data$pohlavi)], col = c("red3",
    "green3")[unclass(data$pohlavi)])
```



Obrázek 1: Maticový bodový graf pomocí příkazu `pairs` s rozlišením bodů (kroužek dívka, křížek hoch) i barev (červeně dívka, zeleně hoch).

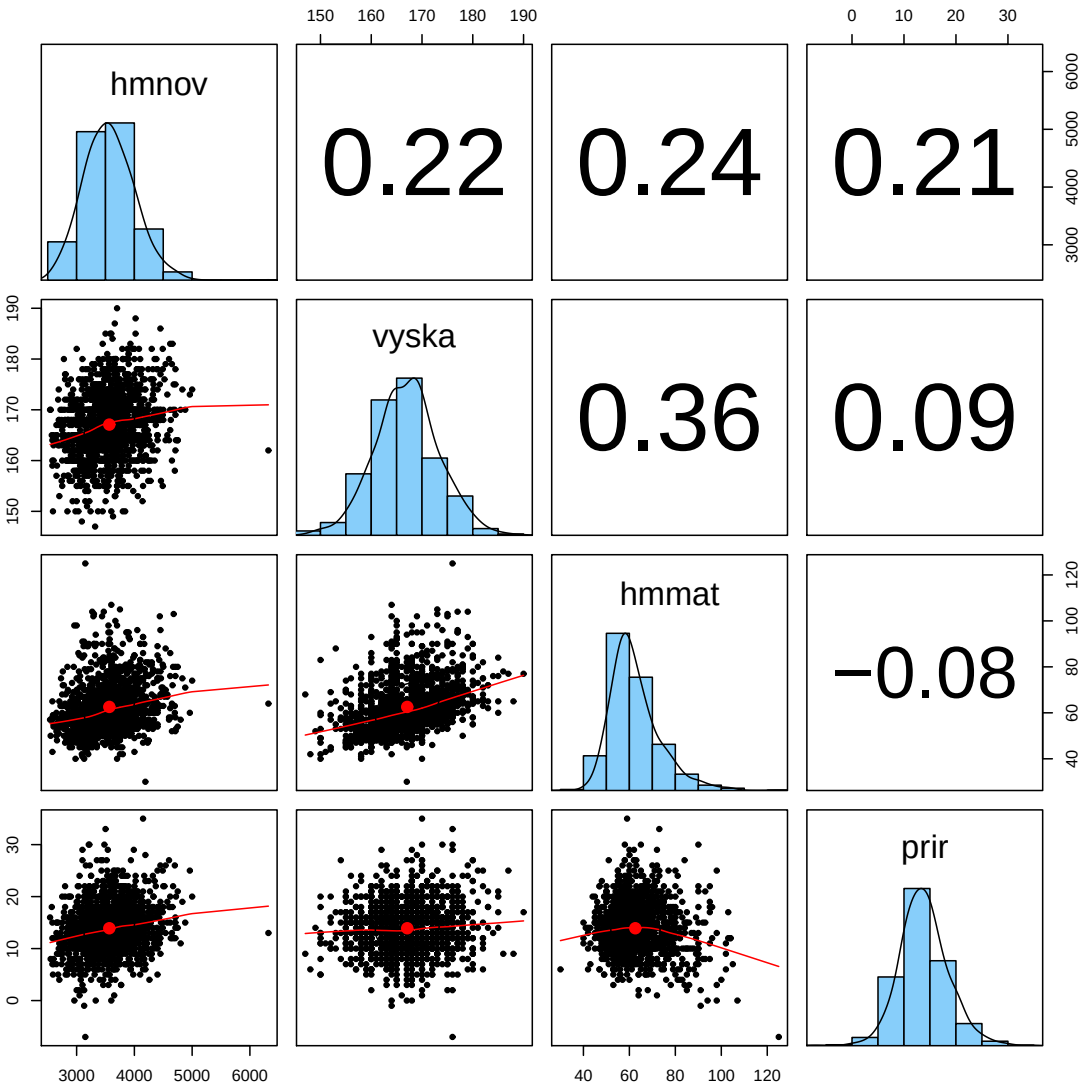
Pomocí příkazu `pairs.panels()` z knihovny `psych` lze získat mnohem zajímavější graf, ve kterém

diagonála – obsahuje histogram spojité proměnné spolu s ádrovým odhadem hustoty,

dolní trojúhelníková část matice – obsahuje dvourozměrné bodové grafy, kterými je proložena *LOWESS* křivka (vážená polynomiální regrese),

horní trojúhelníková část matice – obsahuje hodnoty výběrových korelačních koeficientů.

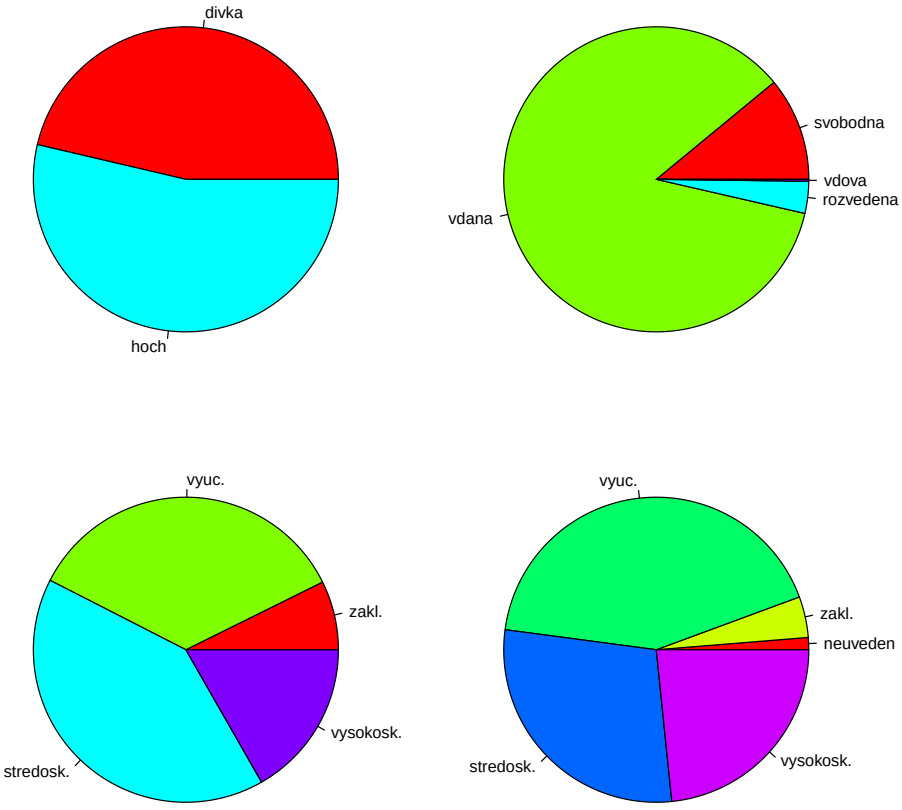
```
> pairs.panels(data[1:4], hist.col = "lightskyblue")
```



Obrázek 2: Graf vytvořený pomocí příkazu `pairs.panels` s histogramy a jádrovým odhadem hustoty na diagonále, s váženou polynomiální regresí (*LOWESS* křivka) v dolní části a s výběrovými korelačními koeficienty v horní části grafu.

Pro kategoriální proměnné se hodí graf `pie()`.

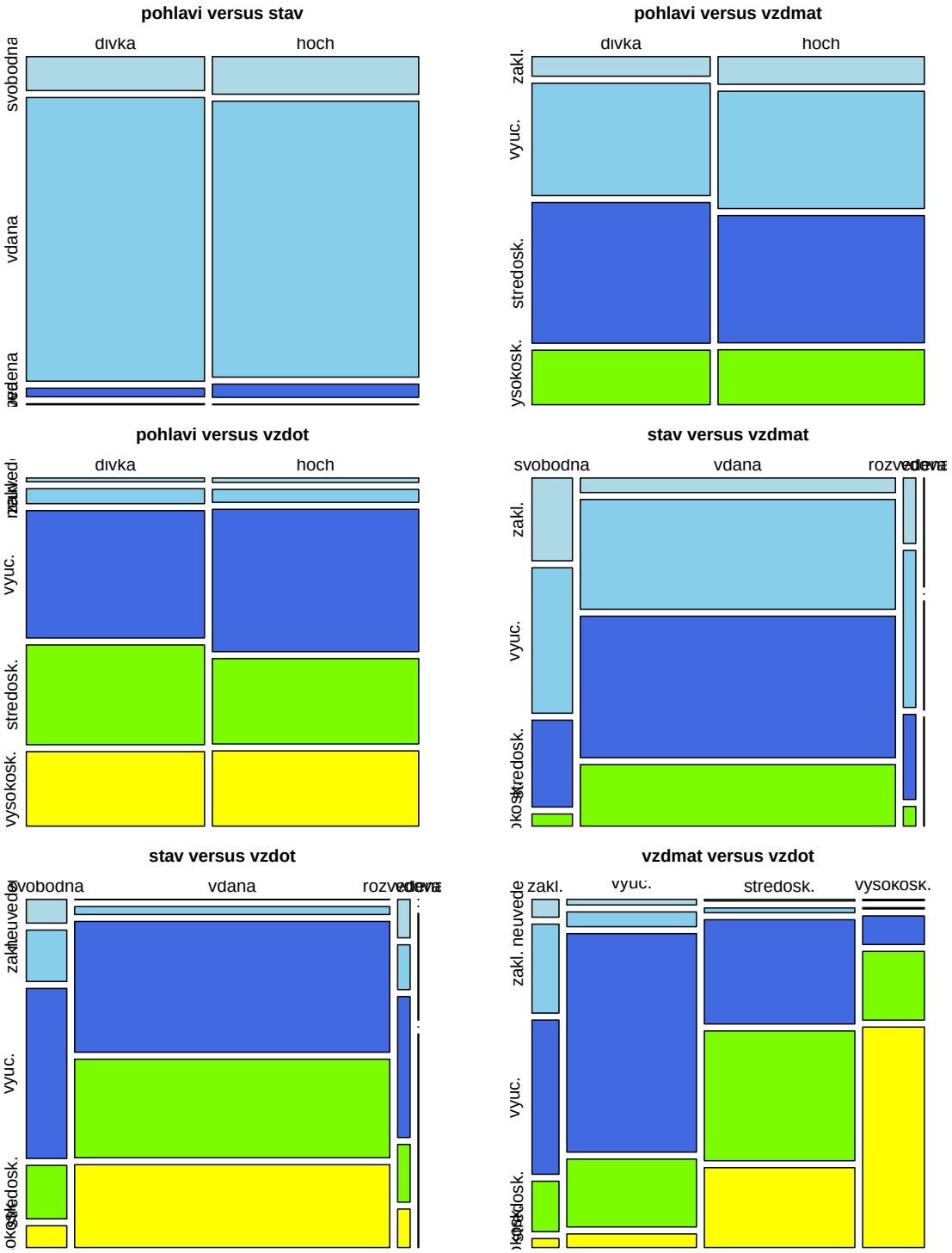
```
> par(mfrow = c(2, 2), mar = c(1, 1, 1, 2))
> for (i in 5:8) pie(table(data[, i]), col = rainbow(length(levels(data[, i]))))
```



Obrázek 3: Koláčové grafy pomocí příkazu `pie` pro kategoriální proměnné.

Pro grafické znázornění kontingenčních tabulek použijeme příkaz `mosaicplot()`.

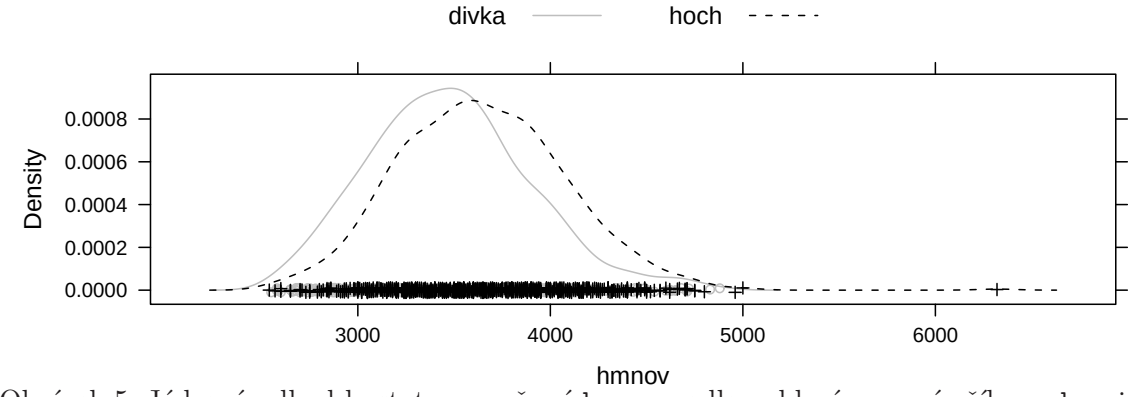
```
> par(mfrow = c(3, 2), mar = c(0, 2, 3, 2) + 0.1)
> for (i in 5:7) {
     for (j in (i + 1):8) mosaicplot(table(data[, i], data[, j]), cex.axis = 1.25,
         main = paste(names(data)[i], "versus", names(data)[j]), col = c("lightblue",
             "skyblue", "RoyalBlue", "LawnGreen", "Yellow", "Gold1", "Navy",
             "violetred4", "Plum"))
 }
```



Obrázek 4: Mosaikové grafy pomocí příkazu `mosaicplot` pro dvojice kategoriálních proměnných.

Znovu se vrátíme ke spojitým proměnným a zaměříme se na jejich rozdělení podle jednotlivých kategoriálních proměnných. U porodní hmotnosti novorozence má smysl zkoumat rozdělení proměnné `hmnov` ve vztahu k pohlaví.

```
> print(densityplot(~hmnov, groups = pohlavi, data = data, n = 1024,
     par.settings = simpleTheme(col = c("gray", "black"), lty = c(1,
         2), pch = c(1, 3)), auto.key = list(columns = 2)))
```

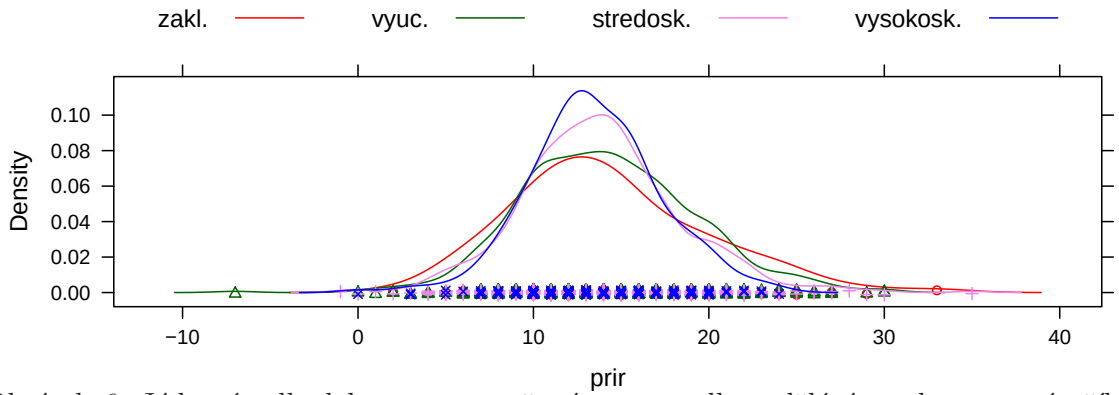


Obrázek 5: Jádrový odhad hustoty proměnné `hmnov` podle pohlaví pomocí příkazu `densityplot`.

Z grafu je patrné, že porodní hmotnost u chlapců je vyšší než u děvčat.

Pokud se zaměříme na váhový přírůstek matky během těhotenství, zkoumejme proměnnou `prir` například podle vzdělání matky.

```
> print(densityplot(~prir, groups = vzdmat, data = data, n = 1024,
     par.settings = simpleTheme(lty = 1, col = c("red", "darkgreen",
         "violet", "blue"), pch = c(1:4)), auto.key = list(columns = 4)))
```



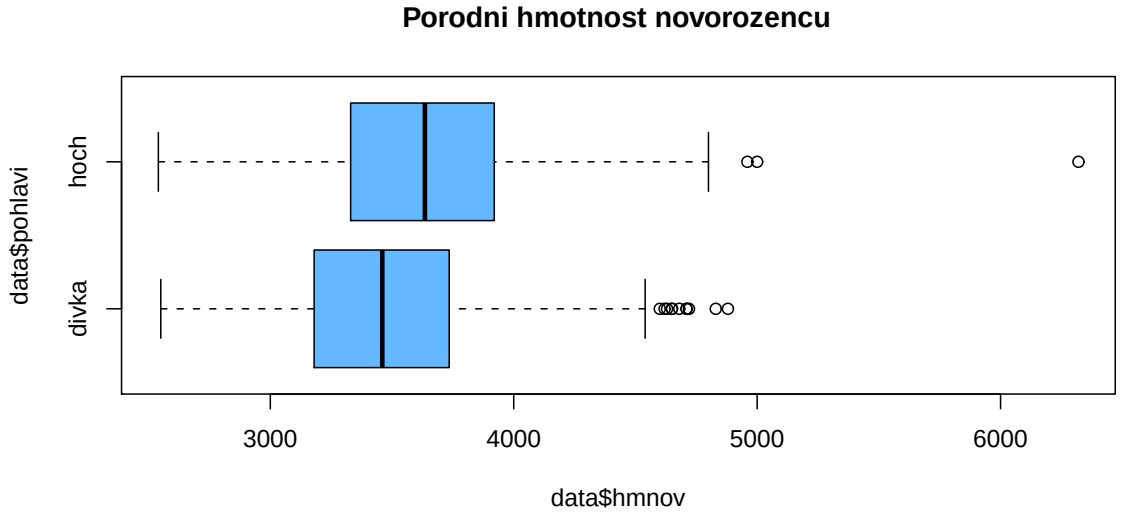
Obrázek 6: Jádrový odhad hustoty proměnné `prir` podle vzdělání matky pomocí příkazu `densityplot`.

I když rozdělení váhových přírůstků pro jednotlivé skupiny není úplně totožné, přesto se významně neliší.

Analýza rozptylu proměnné `hmnov` vzhledem k pohlaví dítěte

Závisle proměnnou, která nás bude zajímat, je porodní hmotnost novorozenců `hmnov`. Dříve než vytvoříme regresní model, pomocí příkazu `plot()` si prohlédněme, jak to vypadá s variabilitou porodní hmotnosti pro jednotlivá pohlaví

```
> plot(data$hmnov ~ data$pohlavi, main = popis[1], horizontal = TRUE,
     col = "steelblue1")
```



Obrázek 7: Krabicový graf pro `hmnov` pro jednotlivá pohlaví pomocí příkazu `plot`.

Podle grafu variabilita porodní hmotnosti u obou pohlaví se neliší. Zkusme tento fakt ověřit také pomocí Bartletova testu

```
> bartlett.test(hmnov ~ pohlavi, data = data)


    Bartlett test of homogeneity of variances

data:  hmnov by pohlavi
Bartlett,s K-squared = 0.8322, df = 1, p-value = 0.3616
```

To, co naznačovaly krabicové grafy, ukázal i výsledek Bartletova testu. Protože p–hodnota není menší než 0.05, rozdílnost v rozptylech jednotlivých pohlaví se nepotvrdila. Můžeme tedy uvažovat klasický regresní model.

Analýza rozptylu pomocí příkazu `lm()`

Uvažujme klasický ANOVA1 model

$$\boxed{Y_{jk} = \mu_j + \varepsilon_{jk} = \mu + \alpha_j + \varepsilon_{jk}}, \text{ kde } j = 1, \ldots, a \ (a = 2),\ k = 1, \ldots, n_j.$$

K výpočtům použijeme příkazy

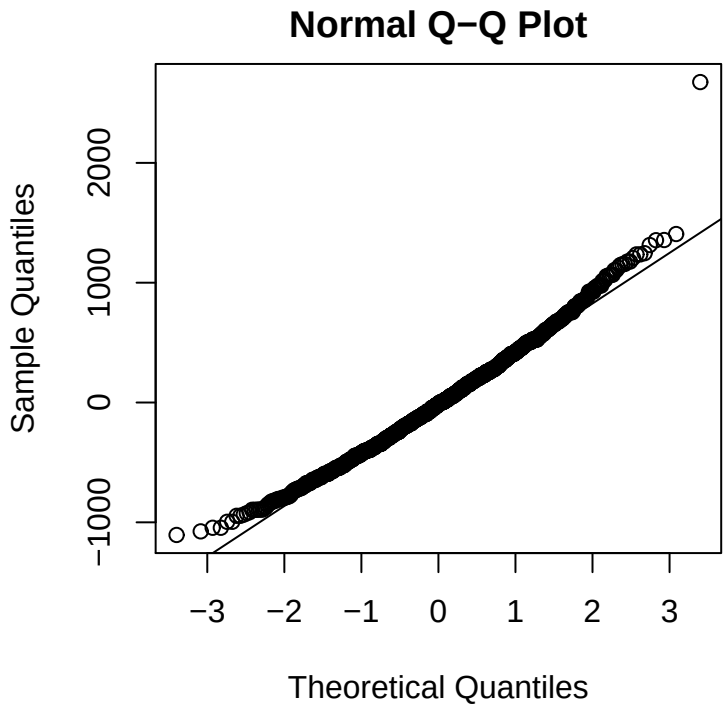
```
> m.lm <- lm(hmnov ~ pohlavi, data = data)
> summary(m.lm)
```

```
Call:
lm(formula = hmnov ~ pohlavi, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-1105.46  -303.74   -13.74   266.26  2674.54

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  3473.74      16.38  212.09  < 2e-16 ***
pohlavihoch   171.72      22.36    7.68 2.88e-14 ***
---
Signif. codes:  0 ,***, 0.001 ,**, 0.01 ,*, 0.05 ,., 0.1 , , 1

Residual standard error: 428.4 on 1474 degrees of freedom
Multiple R-squared: 0.03848,    Adjusted R-squared: 0.03782
F-statistic: 58.98 on 1 and 1474 DF,  p-value: 2.88e-14
```

Vidíme, že pohlaví je statisticky významnou veličinou.

U klasického regresního modelu se předpokládá normalita závisle proměnné. Protože jde o nezápornou náhodnou veličinu, nemusí mít normální rozdělení. Normalitu reziduí ověříme nejprve graficky pomocí kvantil-kvantil grafu pro normální rozdělení

```
> par(mfrow = c(1, 1), mar = c(5, 5, 2, 0) + 0.1)
> qqnorm(resid(m.lm))
> qqline(resid(m.lm))
```



Obrázek 8: Grafické ověření normnality.

Protože většina bodů leží nad přímkou, normalita reziduí je ohrožena.

Ověřme normalitu reziduí také pomocí testů, které lze najít v knihovně `nortest`

`ad.test`	Anderson-Darling test
`cvm.test`	Cramer-von Mises test
`lillie.test`	Lilliefors (Kolmogorov-Smirnov) test
`pearson.test`	Pearson chi-square test
`sf.test`	Shapiro-Francia test

```
> library(nortest)
> ad.test(resid(m.lm))


    Anderson-Darling normality test

data:  resid(m.lm)
A = 1.5239, p-value = 0.0006316


> cvm.test(resid(m.lm))


    Cramer-von Mises normality test

data:  resid(m.lm)
W = 0.2136, p-value = 0.003624


> lillie.test(resid(m.lm))


    Lilliefors (Kolmogorov-Smirnov) normality test

data:  resid(m.lm)
D = 0.0263, p-value = 0.01813


> pearson.test(resid(m.lm))


    Pearson chi-square normality test

data:  resid(m.lm)
P = 69.5908, p-value = 0.0004492


> sf.test(resid(m.lm))


    Shapiro-Francia normality test

data:  resid(m.lm)
W = 0.9886, p-value = 1.011e-08
```

Vidíme, že všechny testy zamítají normalitu reziduí.

Jako další spojité rozdělení exponenciálního typu přichází v úvahu gamma rozdělení. Dvouparametrické gamma rozdělení $\boxed{Y \sim G(\alpha, \beta)}$, kde $\alpha > 0, \beta > 0$, se nejčastěji definuje pomocí hustoty takto

$$f(y) = \frac{1}{\Gamma(\alpha)\beta^{\alpha}} y^{\alpha-1} e^{-\frac{y}{\beta}} \qquad y > 0,$$

přičemž $EY = \alpha\beta$ a $DY = \alpha\beta^2$.

Protože regresní model odhaduje střední hodnotu , provedeme následující reparametrizaci

$$\mu = \alpha\beta \qquad \Rightarrow \qquad \beta = \frac{\mu}{\alpha},$$

a

$$f(y) = \frac{1}{\Gamma(\alpha)} \left(\frac{\alpha}{\mu}\right)^{\alpha} y^{\alpha-1} e^{-\frac{\alpha}{\mu}y} = \exp\left\{ \frac{y\left(-\frac{1}{\mu}\right) - \ln\mu}{\frac{1}{\alpha}} + \alpha\ln\alpha + (\alpha-1)\ln y - \ln\Gamma(\alpha) \right\}$$

přičemž $EY = \mu$ a $DY = \frac{\mu^2}{\alpha}$.

Budeme se snažit ověřit, že rozdělení porodní hmotnosti novorozence je gamma rozdělení. Nejprve z dat provedeme MLE–odhady neznámých parametrů a pak porovnáme jádrový odhad hustoty s hustotou pro gamma rozdělení, a to pro každé pohlaví dítěte zvlášť.

Budeme–li hledat MLE–odhady neznámých parmetrů μ a α na základě náhodného výběru $\mathbf{Y} = (Y_1, \ldots, Y_n)^{\mathrm{T}}$, pak budeme maximalizovat logaritmus věrohodnostní funkce

$$l(\mu, \alpha; y_1, \ldots, y_n) = -\frac{\alpha}{\mu}\sum_{i=1}^{n} y_i - n\alpha\ln\mu + n\alpha\ln\alpha + (\alpha-1)\sum_{i=1}^{n}\ln y_i - n\ln\Gamma(\alpha)$$

Dříve než použijeme příkaz `maxLik()` z knihovny téhož jména, provedeme nejprve určité pomocné akce, a to vytvoříme funkci `logL()` pro výpočet logaritmu věrohodnostní funkce, rozdělíme porodní hmotnosti dívek a hochů a pomocí momentové metody nachystáme počáteční odhady parametrů μ a α.

```
> logL <- function(param) {
    alpha <- param[1]
    mu <- param[2]
    return(-alpha * sum(x)/mu - length(x) * alpha * log(mu) + length(x) *
        alpha * log(alpha) + (alpha - 1) * sum(log(x)) - length(x) *
        log(gamma(alpha)))
 }
> L1 <- data$pohlavi == "divka"
> x1 <- data$hmnov[L1]
> x2 <- data$hmnov[!L1]
> mu01 <- mean(x1)
> mu02 <- mean(x2)
> alpha01 <- mu01^2/var(x1)
> alpha02 <- mu02^2/var(x2)
```

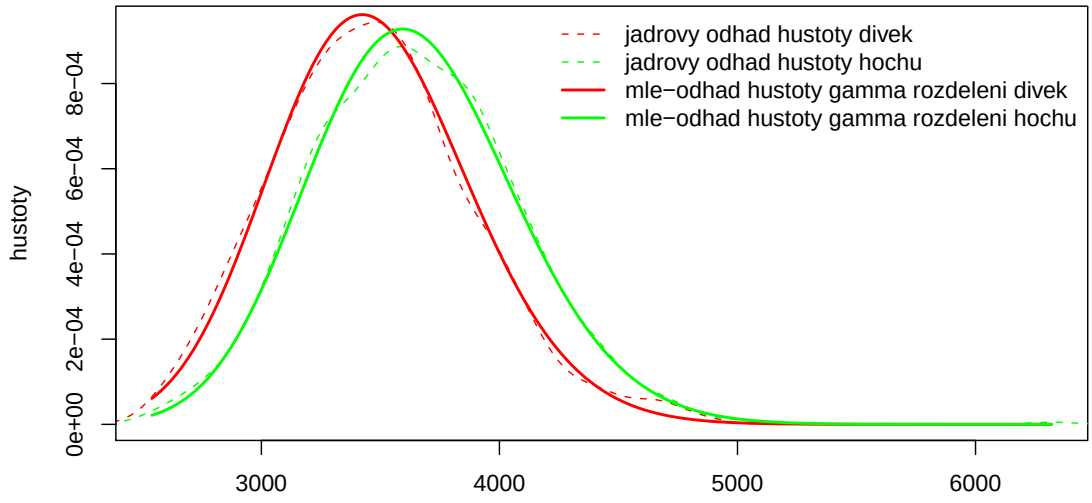
Následně provedeme MLE–odhad parametrů gamma rozdělení pomocí příkazu `maxLik()`, a to zvlášť pro dívky a chlapce.

```
> library(maxLik)
> x <- x1
> (mle.est1 <- maxLik(logL, start = c(alpha01, mu01)))


Maximum Likelihood estimation
Newton-Raphson maximisation, 10 iterations
Return code 2: successive function values within tolerance limit
Log-Likelihood: -5094.348 (2 free parameter(s))
Estimate(s): 69.29492 3473.746


> x <- x2
> (mle.est2 <- maxLik(logL, start = c(alpha02, mu02)))


Maximum Likelihood estimation
Newton-Raphson maximisation, 23 iterations
Return code 2: successive function values within tolerance limit
Log-Likelihood: -5927.116 (2 free parameter(s))
Estimate(s): 71.05562 3645.46
```

A nyní již můžeme porovvnat jádrové odhady hustot (tzv. neparametrické odhady hustot) s parametrickými odhady hustot (příkaz `dgamma()`), ve kterých jsme neznámé parametry nahradili MLE–odhady.

Při použití příkazu `dgamma()` musíme být opatrní. S využitím nápovědy `?dgamma` totiž zjistíme, že tento příkaz používá první tvar hustoty gamma rozdělení s parametry $\boxed{\alpha}$ (parametr `shape`) a $\boxed{\beta = \frac{\alpha}{\mu}}$ (parametr `scale`)

```
> dens1 <- density(x1, n = 512)
> dens2 <- density(x2, n = 512)
> ylim <- c(0, max(c(dens1$y, dens2$y)))
> ab <- range(data$hmnov)
> xx <- seq(ab[1], ab[2], length = 512)
> par(mar = c(3, 5, 0, 0) + 0.1)
> plot(dens1$x, dens1$y, xlim = ab, ylim = ylim,
     col = "red", type = "l", lty = 2, xlab = "x",
     ylab = "hustoty")
> lines(dens2$x, dens2$y, col = "green", type = "l",
     lty = 2)
> lines(xx, dgamma(xx, shape = mle.est1$estimate[1],
     scale = mle.est1$estimate[2]/mle.est1$estimate[1]),
     col = "red", lty = 1, lwd = 2)
> lines(xx, dgamma(xx, shape = mle.est2$estimate[1],
     scale = mle.est2$estimate[2]/mle.est2$estimate[1]),
     col = "green", lty = 1, lwd = 2)
> legend(x = "topright", bty = "n", col = c("red",
     "green", "red", "green"), lty = c(2, 2, 1,
     1), lwd = c(1, 1, 2, 2), legend = c("jadrovy odhad hustoty divek",
     "jadrovy odhad hustoty hochu", "mle-odhad hustoty gamma rozdeleni divek",
     "mle-odhad hustoty gamma rozdeleni hochu"))
```



Obrázek 9: Grafické ověření gamma rozdělení.

Graf ukazuje, že veličina porodní hmotnost novorozenců má gamma rozdělení, neboť neparametrické jádrové odhady hustot a parametrické odhady hustot se téměř neliší.

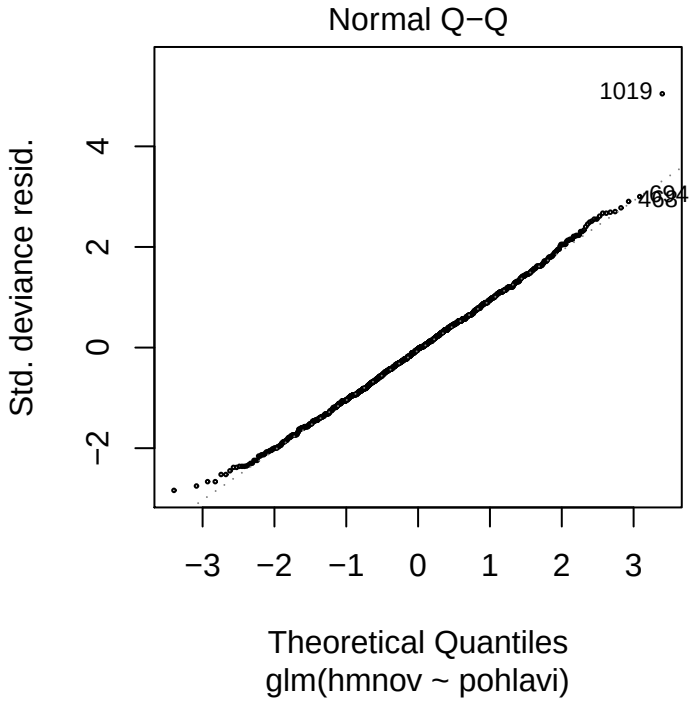
Analýza rozptylu pomocí příkazu `glm()`

Uvažujme nyní GLM model se závisle proměnnou `hmnov` (gamma rozdělení) podle pohlaví s kanonickou linkovací funkcí.

```
> m.glm <- glm(hmnov ~ pohlavi, data = data, family = Gamma(link = "inverse"))
> summary(m.glm)


Call:
glm(formula = hmnov ~ pohlavi, family = Gamma(link = "inverse"),
    data = data)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.34081  -0.08707  -0.00396   0.07350   0.60569

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.879e-04  1.322e-06 217.706  < 2e-16 ***
pohlavihoch -1.356e-05  1.766e-06  -7.677 2.94e-14 ***
---
Signif. codes:  0 ,***, 0.001 ,**, 0.01 ,*, 0.05 ,., 0.1 , , 1

(Dispersion parameter for Gamma family taken to be 0.01443159)

    Null deviance: 21.920  on 1475  degrees of freedom
Residual deviance: 21.066  on 1474  degrees of freedom
AIC: 22049

Number of Fisher Scoring iterations: 4
```

Vidíme, že pohlaví je statisticky významnou veličinou. Vhodnost modelu ověříme pomocí Q-Q (kvantil-kvantil) grafu pro rezidua.

```
> plot(m.glm, which = 2, cex = 0.25)
```



Obrázek 10: Grafické ověření normality reziduí modelu `m.glm` pomocí Q-Q grafu.

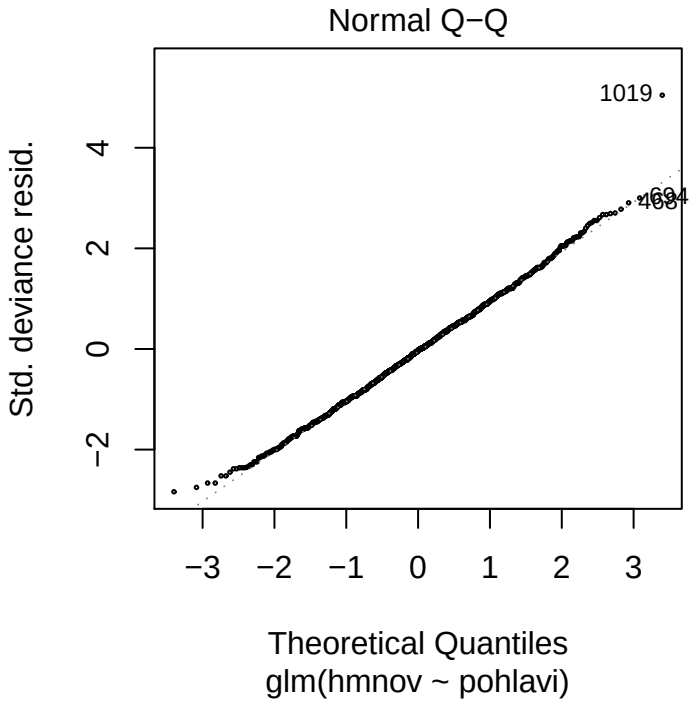
Porovnáme–li Q-Q graf reziduí modelu `m.lm` s Q-Q grafem reziduí modelu `m.glm`, vidíme že oproti klasickému regresnímu modelu je model s gamma rozdělením určitě vhodnější. Pro úplnost ještě použijme jinou linkovací funkci, a to logaritmus.

```
> m.glm.log <- glm(hmnov ~ pohlavi, data = data, family = Gamma(link = log))
> summary(m.glm.log)


Call:
glm(formula = hmnov ~ pohlavi, family = Gamma(link = log), data = data)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.34081  -0.08707  -0.00396   0.07350   0.60569

Coefficients:
            Estimate Std. Error  t value Pr(>|t|)
(Intercept) 8.152986   0.004593 1774.956  < 2e-16 ***
pohlavihoch 0.048250   0.006271    7.695 2.58e-14 ***
---
```

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

(Dispersion parameter for Gamma family taken to be 0.01443159)

    Null deviance: 21.920  on 1475  degrees of freedom
Residual deviance: 21.066  on 1474  degrees of freedom
AIC: 22049

Number of Fisher Scoring iterations: 4
```

Vidíme, že pohlaví je i v tomto modelu statisticky významnou veličinou. Vhodnost modelu opět ověříme pomocí Q-Q (kvantil-kvantil) grafu pro rezidua.

```
> plot(m.glm.log, which = 2, cex = 0.25)
```



Obrázek 11: Grafické ověření normality reziduí modelu `m.glm.log` pomocí Q-Q grafu.

Porovnáme–li Q-Q graf reziduí modelu `m.glm` s kanonickou linkovací funkcí s Q-Q grafem reziduí modelu `m.glm.log` s logaritmickou linkovací funkcí, vidíme že se grafy téměř neliší.

Analýza kovariance

Protože na hmotnost novorozenců mohou mít vliv i další veličiny, opět využijeme stepwise procedur pro výběr vhodných kovariát. Ponecháme logaritmickou linkovací funkci. V příkazu `step()` uvedeme vedle nulového modelu i nejbohatší model, který chceme uvažovat. Pokud neuvedeme jinak, stepwise procedura bude na základě AIC kritéria přidávat a ubírat proměnné tak dlouho, dokud další změna nepřinese zlepšení AIC kritéria. Každý krok je komentován.

```
> model.step <- step(mNULL <- glm(hmnov ~ 1, data, family = Gamma(link = log)),
     scope = ~(vyska + hmmat + prir) * pohlavi * stav * vzdmat)


Start:  AIC=22105.8
hmnov ~ 1

          Df Deviance   AIC
+ hmmat    1   20.595 22020
+ vyska    1   20.867 22038
+ prir     1   20.894 22039
+ pohlavi  1   21.066 22051
+ vzdmat   3   21.749 22100
<none>         21.920 22106
+ stav     3   21.855 22108

Step:  AIC=22015.58
hmnov ~ hmmat

          Df Deviance   AIC
+ prir     1   19.366 21931
+ pohlavi  1   19.785 21960
+ vyska    1   20.162 21987
+ vzdmat   3   20.401 22008
<none>         20.595 22016
+ stav     3   20.548 22018
- hmmat    1   21.920 22108

Step:  AIC=21926.57
hmnov ~ hmmat + prir

          Df Deviance   AIC
+ pohlavi  1   18.677 21877
+ vyska    1   19.092 21908
+ vzdmat   3   19.106 21913
+ stav     3   19.264 21925
<none>         19.366 21927
- prir     1   20.595 22017
- hmmat    1   20.894 22040

Step:  AIC=21875.01
hmnov ~ hmmat + prir + pohlavi

                Df Deviance   AIC
+ vyska          1   18.367 21853
+ vzdmat         3   18.391 21859
+ stav           3   18.565 21872
+ hmmat:pohlavi  1   18.620 21873
<none>               18.677 21875
+ prir:pohlavi   1   18.677 21877
- pohlavi        1   19.366 21927
- prir           1   19.785 21960
- hmmat          1   20.152 21988

Step:  AIC=21852.24
hmnov ~ hmmat + prir + pohlavi + vyska

                Df Deviance   AIC
```

```
+ vzdmat         3   18.152 21841
+ hmmat:pohlavi  1   18.311 21850
+ stav           3   18.268 21850
<none>               18.367 21852
+ vyska:pohlavi  1   18.362 21854
+ prir:pohlavi   1   18.367 21854
- vyska          1   18.677 21875
- pohlavi        1   19.092 21908
- hmmat          1   19.223 21918
- prir           1   19.314 21925

Step:  AIC=21840.79
hmnov ~ hmmat + prir + pohlavi + vyska + vzdmat

                 Df Deviance   AIC
+ hmmat:pohlavi   1   18.093 21838
<none>               18.152 21841
+ vyska:pohlavi   1   18.148 21843
+ prir:pohlavi    1   18.151 21843
+ stav            3   18.108 21843
+ vyska:vzdmat    3   18.126 21845
+ pohlavi:vzdmat  3   18.135 21846
+ hmmat:vzdmat    3   18.136 21846
+ prir:vzdmat     3   18.151 21847
- vzdmat          3   18.367 21852
- vyska           1   18.391 21858
- pohlavi         1   18.895 21898
- hmmat           1   19.073 21913
- prir            1   19.163 21920

Step:  AIC=21837.95
hmnov ~ hmmat + prir + pohlavi + vyska + vzdmat + hmmat:pohlavi

                 Df Deviance   AIC
<none>               18.093 21838
+ prir:pohlavi    1   18.091 21840
+ vyska:pohlavi   1   18.092 21840
+ stav            3   18.050 21841
- hmmat:pohlavi   1   18.152 21841
+ vyska:vzdmat    3   18.064 21842
+ hmmat:vzdmat    3   18.076 21843
+ pohlavi:vzdmat  3   18.079 21843
+ prir:vzdmat     3   18.092 21844
- vzdmat          3   18.311 21850
- vyska           1   18.331 21855
- prir            1   19.110 21918
```

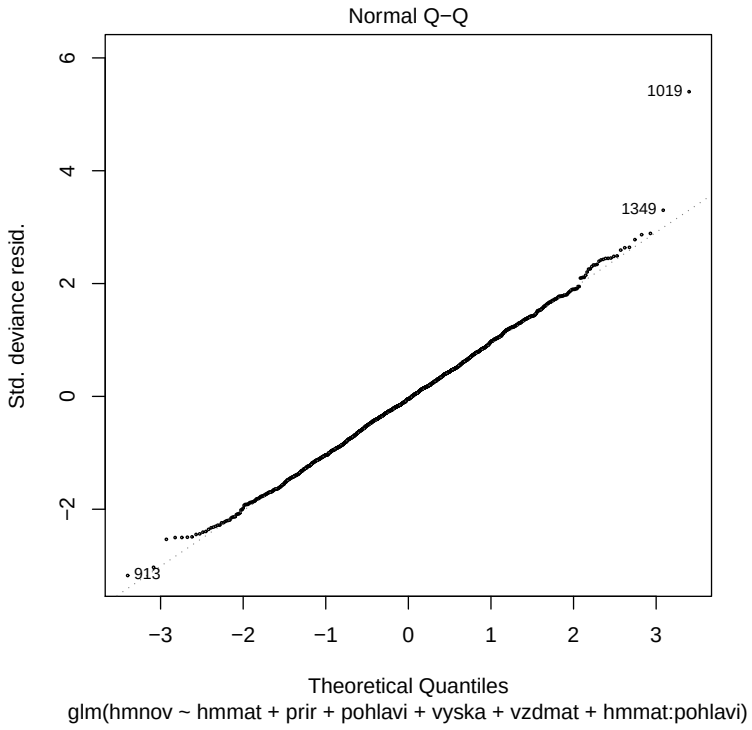
```
> summary(model.step)
```

```
Call:
glm(formula = hmnov ~ hmmat + prir + pohlavi + vyska + vzdmat +
    hmmat:pohlavi, family = Gamma(link = log), data = data)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.35326  -0.07980  -0.00501   0.06846   0.60100
```

```
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       7.4839852  0.0806730  92.769  < 2e-16 ***
hmmat             0.0031312  0.0004049   7.733 1.93e-14 ***
prir              0.0057855  0.0006349   9.113  < 2e-16 ***
pohlavihoch       0.1192340  0.0342156   3.485 0.000507 ***
vyska             0.0022001  0.0005039   4.366 1.35e-05 ***
vzdmatvyuc.       0.0166699  0.0118231   1.410 0.158767
vzdmatstredosk.   0.0301628  0.0117368   2.570 0.010270 *
vzdmatvysokosk.   0.0456199  0.0129720   3.517 0.000450 ***
hmmat:pohlavihoch -0.0011837  0.0005390  -2.196 0.028250 *
---
Signif. codes:  0 ,***, 0.001 ,**, 0.01 ,*, 0.05 ,., 0.1 , , 1

(Dispersion parameter for Gamma family taken to be 0.01244901)

    Null deviance: 21.920  on 1475  degrees of freedom
Residual deviance: 18.093  on 1467  degrees of freedom
AIC: 21838

Number of Fisher Scoring iterations: 4
```

Na závěr vykresleme Q-Q (kvantil-kvantil) grafu pro rezidua.

```
> plot(model.step, which = 2, cex = 0.25)
```



Obrázek 12: Grafické ověření normality reziduí modelu `model.step` pomocí Q-Q grafu.