M0130 – 4. praktikum : **M0130pr04** (*Klasická regrese*)

A. Lineární trend, polynomický trend

Příklad 1

Máme k dispozici průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali). Údaje se vztahují k období 1907 až 1957 a jsou uvedena v cfs.10^{-3}.

Data jsou uložena v souboru `DataNiger.dat`. První řádek obsahuje popis časové řady, pak následuje volný řádek, teprve potom dva sloupce dat: rok a průměrný roční průtok.

Nejprve načteme nadpis.

```
> fileN <- "DataNiger.dat"
> fileDat <- paste(data.library, fileN, sep = "")
> (TXT <- paste(scan(fileDat, what = "", nlines = 1), collapse = " "))
```

```
[1] "Prumerne rocni prutoky vody v rece Nigeru v Coulicoure (Mali) v letech 1907 az 1957"
```

Pak načteme vlastní data do datového rámce, vypíšeme jeho strukturu a prvních šest řádků.

```
> dataNiger <- read.table(fileDat, header = F, skip = 2)
> str(dataNiger)
```

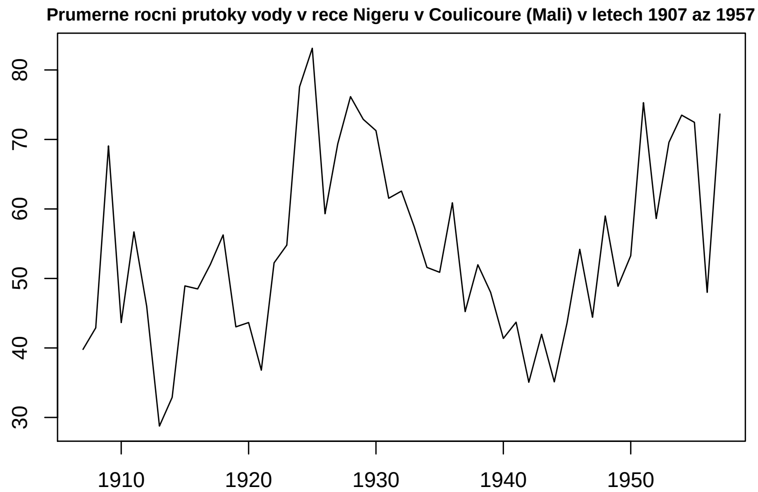
```
'data.frame':        51 obs. of  2 variables:
 $ V1: int  1907 1908 1909 1910 1911 1912 1913 1914 1915 1916 ...
 $ V2: num  39.8 42.9 69.1 43.7 56.7 ...
```

```
> head(dataNiger)
```

```
    V1     V2
1 1907 39.793
2 1908 42.892
3 1909 69.070
4 1910 43.653
5 1911 56.700
6 1912 45.990
```

Z načtených dat vytvoříme časovou řadu a vykreslíme ji.

```
> NigerTS <- ts(dataNiger[, 2], start = 1907, frequency = 1)
> par(mar = c(2, 2, 1, 0) + 0.5)
> plot(NigerTS, main = TXT, cex.main = 0.85)
```



Obrázek 1: Průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

Odhad **lineárního trendu** provedeme pomocí příkazu `lm()`. Pomocí příkazů `summary()` a `anova()` získáme důležité údaje o odhadech neznámých parametrů a o celém modelu.

```
> Time <- time(NigerTS)
> fitModel <- lm(NigerTS ~ Time)
> summary(fitModel)


Call:
lm(formula = NigerTS ~ Time)

Residuals:
    Min      1Q  Median      3Q     Max
-21.667  -9.092  -2.412   7.895  30.325

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -351.6125   239.0573  -1.471   0.1477
Time           0.2101     0.1237   1.698   0.0959 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.01 on 49 degrees of freedom
Multiple R-squared: 0.05556,       Adjusted R-squared: 0.03629
F-statistic: 2.883 on 1 and 49 DF,  p-value: 0.09587


> anova(fitModel)


Analysis of Variance Table

Response: NigerTS
          Df Sum Sq Mean Sq F value  Pr(>F)
Time       1  487.7  487.68  2.8828 0.09587 .
Residuals 49 8289.4  169.17
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

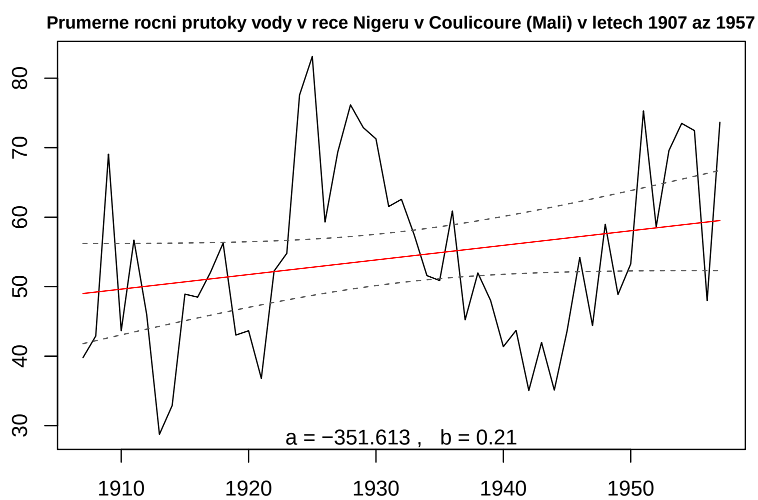
Hodnota *p–value* označená jako `Pr(>|t|)` ukazuje, že lineární trend není významný.

Dalším krokem bude zakreslení regresní přímky, intervalů spolehlivosti kolem regresní přímky a hodnot odhadů parametrů do grafu.

```
> gridt <- seq(start(NigerTS)[1], end(NigerTS)[1], length.out = 500)
> Pred <- predict(fitModel, data.frame(Time = gridt), se = T)
> str(Pred)
```

```
List of 4
 $ fit           : Named num [1:500] 49 49 49.1 49.1 49.1 ...
  ..- attr(*, "names")= chr [1:500] "1" "2" "3" "4" ...
 $ se.fit        : Named num [1:500] 3.59 3.58 3.57 3.56 3.55 ...
  ..- attr(*, "names")= chr [1:500] "1" "2" "3" "4" ...
 $ df            : int 49
 $ residual.scale: num 13
```

```
> alpha <- 0.05
> CV <- qt(1 - alpha/2, Pred$df)
> par(mar = c(2, 2, 1, 0) + 0.5)
> plot(NigerTS, main = TXT, cex.main = 0.85)
> koefText <- paste("a =", round(coef(fitModel)[1], 3), ",   b =",
     round(coef(fitModel)[2], 3))
> mtext(koefText, side = 1, line = -1)
> matlines(gridt, cbind(Pred$fit, Pred$fit - CV * Pred$se.fit,
     Pred$fit + CV * Pred$se.fit), lty = c(1, 2, 2), col = c("red",
     "gray35", "gray35"))
```



Obrázek 2: Lineární trend pro průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

Chceme-li modelovat i cyklický charakter časové řady, použijeme nejprve polynomický trend třetího řádu.

```
> Time <- time(NigerTS)
> fitModel3 <- lm(NigerTS ~ Time + I(Time^2) + I(Time^3))
> summary(fitModel3)
```

```
Call:
lm(formula = NigerTS ~ Time + I(Time^2) + I(Time^3))

Residuals:
     Min       1Q   Median       3Q      Max 
-22.2179  -7.0170  -0.9496   7.6885  27.1292 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)   
(Intercept) -1.483e+07  4.903e+06  -3.024  0.00403 **
Time         2.302e+04  7.614e+03   3.023  0.00404 **
I(Time^2)   -1.191e+01  3.941e+00  -3.022  0.00405 **
I(Time^3)    2.055e-03  6.800e-04   3.022  0.00406 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 12.14 on 47 degrees of freedom
Multiple R-squared: 0.2108,	Adjusted R-squared: 0.1604 
F-statistic: 4.183 on 3 and 47 DF,  p-value: 0.01050 
```

```
> anova(fitModel3)
```

```
Analysis of Variance Table

Response: NigerTS
          Df Sum Sq Mean Sq F value  Pr(>F)   
Time       1  487.7  487.68  3.3088 0.07528 . 
I(Time^2)  1   16.4   16.37  0.1111 0.74038   
I(Time^3)  1 1345.7 1345.72  9.1304 0.00406 **
Residuals 47 6927.3  147.39                   
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
```

Oba dva lineární modely srovnáme pomocí příkazu `anova()`.

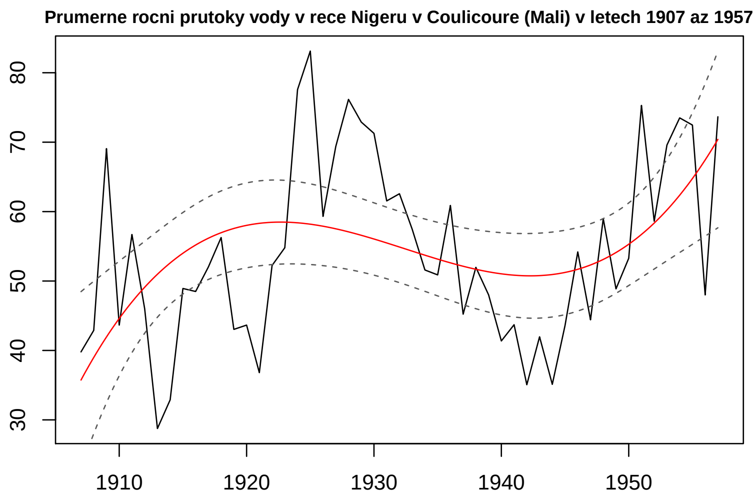
```
> anova(fitModel3, fitModel)
```

```
Analysis of Variance Table

Model 1: NigerTS ~ Time + I(Time^2) + I(Time^3)
Model 2: NigerTS ~ Time
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)  
1     47 6927.3                              
2     49 8289.4 -2   -1362.1 4.6207 0.01472 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
```

Protože p–hodnota srovnávacího testu `Pr(>F)` je menší než 0.05, model s polynomem třetího řadu je významně lepší. Výsledky modelu s polynomem řádu tři opět zakreslíme do grafu.

```
> Pred3 <- predict(fitModel3, data.frame(Time = gridt), se = T)
> alpha <- 0.05
> CV <- qt(1 - alpha/2, Pred3$df)
> par(mar = c(2, 2, 1, 0) + 0.5)
> plot(NigerTS, main = TXT, cex.main = 0.85)
> matlines(gridt, cbind(Pred3$fit, Pred3$fit - CV * Pred3$se.fit,
     Pred3$fit + CV * Pred3$se.fit), lty = c(1, 2, 2), col = c("red",
     "gray35", "gray35"))
```



Obrázek 3: Polynomický trend třetího řádu pro průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

Protože odhad parametru u polynomu třetího řádu je statisticky významný, budeme uvažovat model vyššího řádu, řekněme 6. řádu.

```
> fitModel6 <- lm(NigerTS ~ Time + I(Time^2) + I(Time^3) + I(Time^4) +
     I(Time^5) + I(Time^6))
> summary(fitModel6)


Call:
lm(formula = NigerTS ~ Time + I(Time^2) + I(Time^3) + I(Time^4) +
    I(Time^5) + I(Time^6))

Residuals:
     Min       1Q   Median       3Q      Max
-22.2179  -7.0170  -0.9496   7.6885  27.1292

Coefficients: (3 not defined because of singularities)
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.483e+07  4.903e+06  -3.024  0.00403 **
Time         2.302e+04  7.614e+03   3.023  0.00404 **
I(Time^2)   -1.191e+01  3.941e+00  -3.022  0.00405 **
I(Time^3)    2.055e-03  6.800e-04   3.022  0.00406 **
I(Time^4)           NA         NA      NA       NA
I(Time^5)           NA         NA      NA       NA
I(Time^6)           NA         NA      NA       NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 12.14 on 47 degrees of freedom
Multiple R-squared: 0.2108,       Adjusted R-squared: 0.1604
F-statistic: 4.183 on 3 and 47 DF,  p-value: 0.01050
```

Vidíme, že s polynomem tak vysokého stupně jsou spojeny numerické problémy. Proto provedeme centrování času a výpočet provedeme znovu.

```
> delta <- 0.5 * (start(NigerTS)[1] + end(NigerTS)[1])
> TimeC <- Time - delta
> fitModel6 <- lm(NigerTS ~ TimeC + I(TimeC^2) + I(TimeC^3) + I(TimeC^4) +
     I(TimeC^5) + I(TimeC^6))
> summary(fitModel6)
```

```
Call:
lm(formula = NigerTS ~ TimeC + I(TimeC^2) + I(TimeC^3) + I(TimeC^4) +
    I(TimeC^5) + I(TimeC^6))

Residuals:
     Min       1Q   Median       3Q      Max
-19.3877  -5.4541  -0.9673   5.1213  21.0478

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.469e+01  2.872e+00  22.523  < 2e-16 ***
TimeC       -1.757e+00  3.909e-01  -4.494 5.02e-05 ***
I(TimeC^2)  -2.385e-01  5.367e-02  -4.444 5.90e-05 ***
I(TimeC^3)   1.045e-02  2.371e-03   4.408 6.63e-05 ***
I(TimeC^4)   8.809e-04  2.268e-04   3.884 0.000341 ***
I(TimeC^5)  -1.165e-05  3.209e-06  -3.631 0.000733 ***
I(TimeC^6)  -8.461e-07  2.526e-07  -3.350 0.001666 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.36 on 44 degrees of freedom
Multiple R-squared: 0.5608,       Adjusted R-squared: 0.5009
F-statistic: 9.364 on 6 and 44 DF,  p-value: 1.278e-06
```

```
> anova(fitModel6)
```

```
Analysis of Variance Table

Response: NigerTS
           Df Sum Sq Mean Sq F value    Pr(>F)
TimeC       1  487.7  487.68  5.5664 0.0228113 *
I(TimeC^2)  1   16.4   16.37  0.1869 0.6676180
I(TimeC^3)  1 1345.7 1345.72 15.3601 0.0003066 ***
I(TimeC^4)  1  934.1  934.08 10.6616 0.0021221 **
I(TimeC^5)  1 1155.0 1155.02 13.1834 0.0007329 ***
I(TimeC^6)  1  983.3  983.29 11.2233 0.0016656 **
Residuals  44 3854.9   87.61
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Oba dva lineární modely srovnáme pomocí příkazu `anova()`.

```
> anova(fitModel6, fitModel3)
```

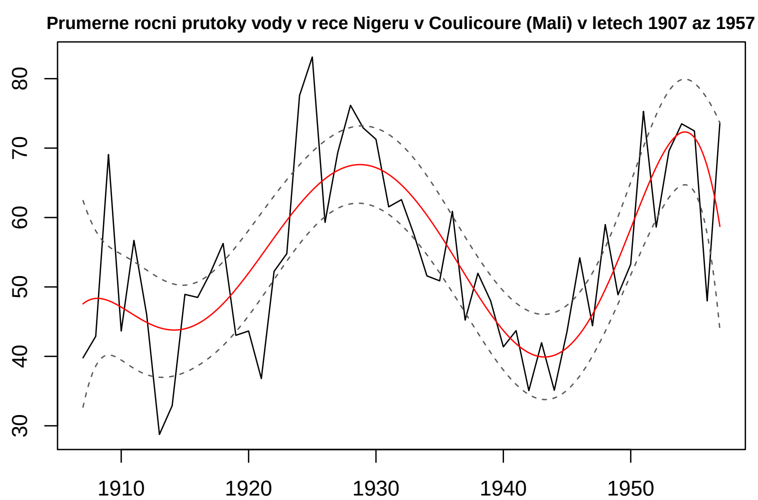
```
Analysis of Variance Table

Model 1: NigerTS ~ TimeC + I(TimeC^2) + I(TimeC^3) + I(TimeC^4) + I(TimeC^5) +
    I(TimeC^6)
Model 2: NigerTS ~ Time + I(Time^2) + I(Time^3)
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1     44 3854.9
2     47 6927.3 -3   -3072.4 11.690 9.241e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Protože p–hodnota srovnávacího testu označená jako `Pr(>F)` je menší než 0.05, je model s polynomem šestého stupně významně lepší než model s polynomem třetího řádu.

Výsledky opět zakreslíme do grafu.

```
> gridtC <- gridt - delta
> Pred6 <- predict(fitModel6, data.frame(TimeC = gridtC), se = T)
> alpha <- 0.05
> CV <- qt(1 - alpha/2, Pred6$df)
> par(mar = c(2, 2, 1, 0) + 0.5)
> plot(NigerTS, main = TXT, cex.main = 0.85)
> matlines(gridt, cbind(Pred6$fit, Pred6$fit - CV * Pred6$se.fit,
     Pred6$fit + CV * Pred6$se.fit), lty = c(1, 2, 2), col = c("red",
     "gray35", "gray35"))
```



Obrázek 4: Polynomický trend šestého řádu pro průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

B. Odhady počtu regresních koeficientů:

Mějme **regresní model** plné hodnosti (např. polynomický regresní model):

$$\boxed{\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \quad \wedge \quad h(\mathbf{X}) = h(\mathbf{X}'\mathbf{X}) = p+1 \quad \wedge \quad n > k = p+1 \quad \wedge \quad \boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2 \mathbf{I}_n)}$$

Odhad neznámých parametrů $\boldsymbol{\beta}$ provedeme *metodou nejmenších čtverců*:

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1} \mathbf{X}'\mathbf{Y}$$

Pro **odhad počtu regresních koeficientů** se nabízejí dvě intuitivní metody

- **Od nejmenšího k nejvyššímu počtu regresorů**:

  Začneme od $k = 1$ a postupně zvyšujeme počet regresorů o jeden a testujeme hypotézu $\boxed{H_0 : \beta_k = 0 \text{ proti } H_1 : \beta_k \neq 0}$ pomocí statistiky $T_k = \frac{\hat{\beta}_k - \beta_k}{\sqrt{s_k^2 v_{kk}}} \sim t(n-k)$,
  kde $(\mathbf{X}'\mathbf{X})^{-1} = (v_{ij})_{i,j=1,\dots,k}$ (viz. Anděl, 1993).

  Jestliže H_0 **zamítneme** $\Rightarrow$ **zvyšujeme** počet regresorů.

- **Od maximálního počtu regresorů dolů**:

  Nejprve zvolme $k = k_{max}$. Testujeme opět $\boxed{H_0 : \beta_k = 0 \text{ proti } H_1 : \beta_k \neq 0}$ pomocí statistiky T_k.

  Jestliže H_0 **nezamítneme** $\Rightarrow$ **snižujeme** počet regresorů.

I když těmto metodám byla věnována velká pozornost a byly propracovány do podrobností (viz. Anderson, 1971), žádných uspokojivých výsledků se pomocí nich nedosáhlo. Vyložme jiný přístup k tomuto problému, který byl nalezen poměrně nedávno.

Penalizační metoda odhadu počtu regresních koeficientů

Předpokládejme, že $\boxed{k_0}$ je skutečný počet regresních parametrů.
Lze ukázat, že platí pro $k < k_0$: $E(s_k^2) > \sigma^2$
pro $k \geq k_0$: $E(s_k^2) = \sigma^2$.

Otázkou však zůstává, jak z grafu hodnot $\boxed{s_k^2}$ určit právě tu hodnotu k_0, od níž počínaje již graf dostává vodorovný charakter.

Ukázalo se, že místo s_k^2 je třeba uvažovat $\boxed{A_k = s_k^2(1 + kw_n),}$ kde w_n je tzv. *penalizační funkce* a udává jakousi relativní cenu, kterou je třeba zaplatit za přidání každého dalšího parametru.

Hodnota w_n

- nesmí být příliš velká - aby nezkreslila klesající charakter s_k^2 pro $k < k_0$;
- nesmí být příliš malá - aby z hodnot s_k^2 oscilujících kolem σ^2 vytvořila pro $k \geq 0$ rostoucí posloupnost;

Za odhad $\boxed{\hat{k}}$ se bere hodnota $k \in \{0, 1, \ldots, k_{max}\}$, pro kterou A_k **nabývá svého minima**, kde k_{max} je maximální počet parametrů, které jsme ochotni uvažovat a o němž jsme si jisti, že $k_0 \leq k_{max}$.

Za dosti obecných podmínek týkajících se rozumné volby hodnot x_{ij} lze ukázat (Geweke a Meese(1981), Anděl a kol.(1981)):

pro $\quad n \to \infty \quad$ pokud $w_n > 0 \quad \wedge \quad w_n \longrightarrow 0 \quad \wedge \quad nw_n \longrightarrow \infty \quad \Rightarrow$

$$\boxed{\hat{k} \to k_0 \quad \text{podle pravděpodobnosti.}}$$

V praxi se osvědčilo volit $\boxed{w_n = \frac{1}{\sqrt[4]{n}}}$, tj. $A_k = s_k^2(1 + \frac{k}{\sqrt[4]{n}})$.

Další kriteria pro určení počtu regresních koeficientů

Akaikeovo informační kritérium	$AIC_k = \ln s_k^2 + \frac{2k}{n}$	nadhodnocuje k_0
Swarz (1978) a Rissanen (1978)	$SR_k = \ln s_k^2 + \frac{k \ln n}{n}$	
Hannan a Quinn (1979)	$HQ_k = \ln s_k^2 + \frac{2kc \ln \ln n}{n}$	$c > 1$, obvykle $c = 2$ nebo 3

Příklad 1 (pokračování)

Nejprve vytvoříme funkci, která vypočítá jednotlivá kritéria a výsledky vynese do grafu.

```
> fndptr <- function(y, x = (1:length(y)) - 0.5 * (1 + length(y)),
     alpha = 0.05, maxdgr = 10, mindgr = 1) {
     x <- x - mean(x)
     vh <- NULL
     vsk <- NULL
     vak <- NULL
     vaick <- NULL
     vsrk <- NULL
     vhqk2 <- NULL
     vhqk3 <- NULL
     ny <- length(y)
     maxdgr <- min(maxdgr, ny - 2)
     mindgr <- max(mindgr, 1)
     for (k in mindgr:maxdgr) {
         if (k == 1)
             pomf <- as.formula("y~x")
         else pomf <- as.formula(paste("y~x", paste("+I(x^", 2:k,
             ")", sep = "", collapse = ""), sep = ""))
         m <- lm(pomf)
         CI <- confint(m)[k + 1, ]
         if ((0 >= CI[1]) & (0 <= CI[2]))
```

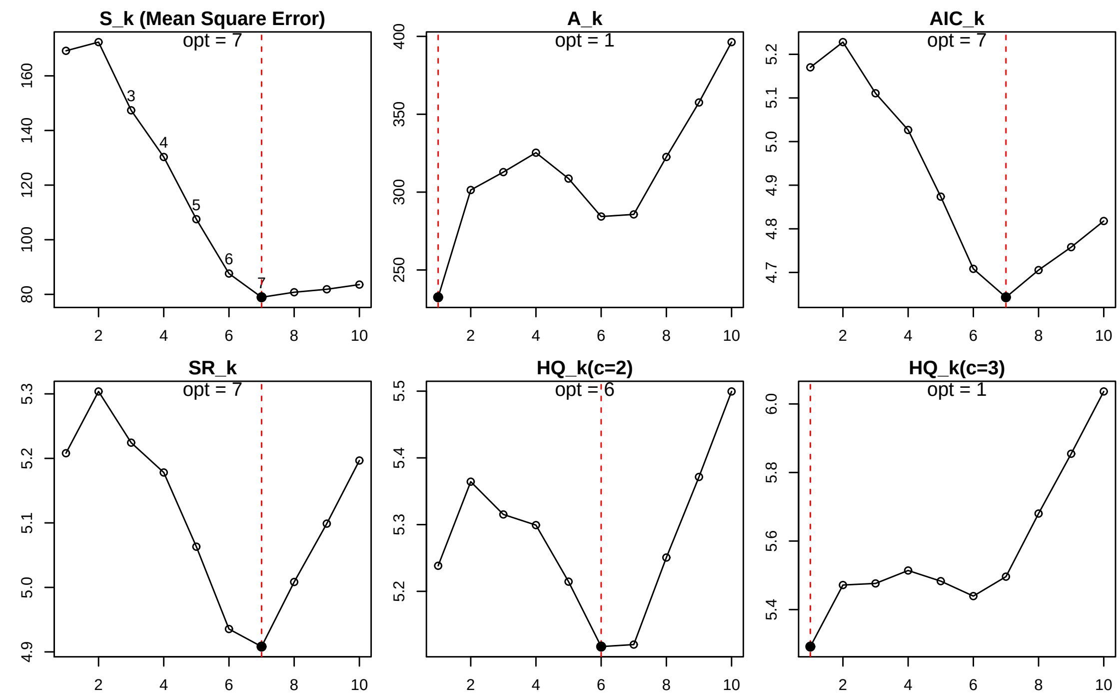
```
            hyp <- 0
        else hyp <- 1
        sk1 <- sum(m$residuals^2)/(ny - k - 1)
        vh <- c(vh, hyp)
        vsk <- c(vsk, sk1)
        vak <- c(vak, sk1 * (1 + k * (ny^(-0.25))))
        vaick <- c(vaick, log(sk1) + (2 * k/ny))
        vsrk <- c(vsrk, log(sk1) + k * log(ny)/ny)
        vhqk2 <- c(vhqk2, log(sk1) + 4 * k * log(log(ny))/ny)
        vhqk3 <- c(vhqk3, log(sk1) + 6 * k * log(log(ny))/ny)
    }
    vd <- mindgr:maxdgr
    L = vh > 0
    vysl <- data.frame(vd, vh, vsk, vak, vaick, vsrk, vhqk2,
        vhqk3)
    names(vysl) <- c("dgr", "significant", "S_k", "A_k", "AIC_k",
        "SR_k", "HQ_k(c=2)", "HQ_k(c=3)")
    TXT <- c("S_k (Mean Square Error)", "A_k", "AIC_k", "SR_k",
        "HQ_k(c=2)", "HQ_k(c=3)")
    par(mfrow = c(2, 3), mar = c(2, 2, 1, 0) + 0.5)
    for (kk in 1:6) {
        plot(vysl[, 1], vysl[, kk + 2], type = "o", main = TXT[kk],
            xlab = "k", ylab = "")
        if (kk == 1)
            text(vysl[L, 1], vysl[L, kk + 2], labels = vysl[L,
                1], pos = 3)
        ind <- which.min(vysl[, kk + 2])
        abline(v = vysl[ind, 1], lty = 2, col = "red")
        points(vysl[ind, 1], vysl[ind, kk + 2], pch = 19, cex = 1.25)
        mtext(paste("opt =", vysl[ind, 1]), cex = 0.85, line = -1)
    }
    return(vysl)
}
```

Pro průměrné roční průtoky vody v řece Nigeru se budeme snažit pomocí funkce `fndptr` zjistit optimální počet regresorů, tj. optimální stupeň polynomu.

Implicitně funkce nastavuje minimální stupeň polynomu 1 a maximální stupeň polynomu 10.

```
> fndptr(NigerTS)
```

```
   dgr significant       S_k      A_k    AIC_k     SR_k HQ_k(c=2) HQ_k(c=3)
1    1           0 169.17139 232.4759 5.170128 5.208007  5.238293  5.291983
2    2           0 172.35465 301.3460 5.227986 5.303743  5.364316  5.471696
3    3           1 147.38939 312.8501 5.110725 5.224362  5.315220  5.476291
4    4           1 130.28745 325.3034 5.026606 5.178122  5.299266  5.514027
5    5           1 107.51558 308.6789 4.873714 5.063109  5.214539  5.482991
6    6           1  87.61154 284.3186 4.708207 4.935480  5.117197  5.439339
7    7           1  78.92988 285.6805 4.643070 4.908222  5.120225  5.496057
8    8           0  80.77111 322.5695 4.705345 5.008376  5.250665  5.680188
9    9           0  81.85967 357.5490 4.757948 5.098858  5.371433  5.854646
10  10           0  83.57424 396.3117 4.817892 5.196681  5.499542  6.036446
```



Obrázek 5: Penalizační kritéria pro výběr vhodného stupně polynomu pro průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

Z výsledků je zřejmé, že bychom se měli rozhodnout mezi polynomem šestého či sedmého stupně.

Model s polynomem šestého stupně jsme již počítali, proto na závěr ještě vypočítáme model, ve kterém použijeme polynom stupně sedm.

```
> fitModel7 <- lm(NigerTS ~ TimeC + I(TimeC^2) + I(TimeC^3) + I(TimeC^4) +
     I(TimeC^5) + I(TimeC^6) + I(TimeC^7))
> summary(fitModel7)
```

```
Call:
lm(formula = NigerTS ~ TimeC + I(TimeC^2) + I(TimeC^3) + I(TimeC^4) +
    I(TimeC^5) + I(TimeC^6) + I(TimeC^7))

Residuals:
     Min       1Q   Median       3Q      Max
-18.5824  -6.2406  -0.8423   6.3083  16.8680

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.469e+01  2.726e+00  23.730  < 2e-16 ***
TimeC       -2.765e+00  5.582e-01  -4.953 1.18e-05 ***
I(TimeC^2)  -2.385e-01  5.094e-02  -4.682 2.85e-05 ***
I(TimeC^3)   2.449e-02  6.231e-03   3.931 0.000303 ***
I(TimeC^4)   8.809e-04  2.152e-04   4.092 0.000184 ***
I(TimeC^5)  -5.944e-05  2.001e-05  -2.971 0.004848 **
I(TimeC^6)  -8.461e-07  2.397e-07  -3.530 0.001006 **
```

```
I(TimeC^7)   4.575e-08  1.893e-08   2.417 0.019986 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.884 on 43 degrees of freedom
Multiple R-squared: 0.6133,       Adjusted R-squared: 0.5504
F-statistic: 9.743 on 7 and 43 DF,  p-value: 3.327e-07
```

```
> anova(fitModel7)
```

```
Analysis of Variance Table

Response: NigerTS
           Df Sum Sq Mean Sq F value    Pr(>F)
TimeC       1  487.7  487.68  6.1787 0.0168955 *
I(TimeC^2)  1   16.4   16.37  0.2075 0.6510563
I(TimeC^3)  1 1345.7 1345.72 17.0496 0.0001644 ***
I(TimeC^4)  1  934.1  934.08 11.8343 0.0013049 **
I(TimeC^5)  1 1155.0 1155.02 14.6335 0.0004174 ***
I(TimeC^6)  1  983.3  983.29 12.4578 0.0010061 **
I(TimeC^7)  1  460.9  460.92  5.8397 0.0199862 *
Residuals  43 3394.0   78.93
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Oba dva lineární modely srovnáme pomocí příkazu `anova()`.

```
> anova(fitModel7, fitModel6)
```

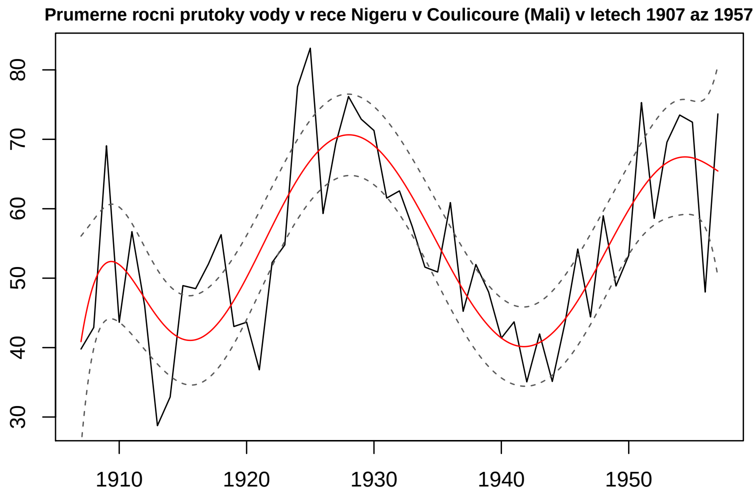
```
Analysis of Variance Table

Model 1: NigerTS ~ TimeC + I(TimeC^2) + I(TimeC^3) + I(TimeC^4) + I(TimeC^5) +
    I(TimeC^6) + I(TimeC^7)
Model 2: NigerTS ~ TimeC + I(TimeC^2) + I(TimeC^3) + I(TimeC^4) + I(TimeC^5) +
    I(TimeC^6)
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)
1     43 3394.0
2     44 3854.9 -1   -460.92 5.8397 0.01999 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Protože p–hodnota srovnávacího testu označená jako `Pr(>F)` je menší než 0.05, je model s polynomem sedmého stupně významně lepší než model s polynomem šestého řádu.

Výsledky opět zakreslíme do grafu.

```
> Pred7 <- predict(fitModel7, data.frame(TimeC = gridtC), se = T)
> alpha <- 0.05
> CV <- qt(1 - alpha/2, Pred7$df)
> par(mar = c(2, 2, 1, 0) + 0.5)
> plot(NigerTS, main = TXT, cex.main = 0.85)
> matlines(gridt, cbind(Pred7$fit, Pred7$fit - CV * Pred7$se.fit,
     Pred7$fit + CV * Pred7$se.fit), lty = c(1, 2, 2), col = c("red",
     "gray35", "gray35"))
```

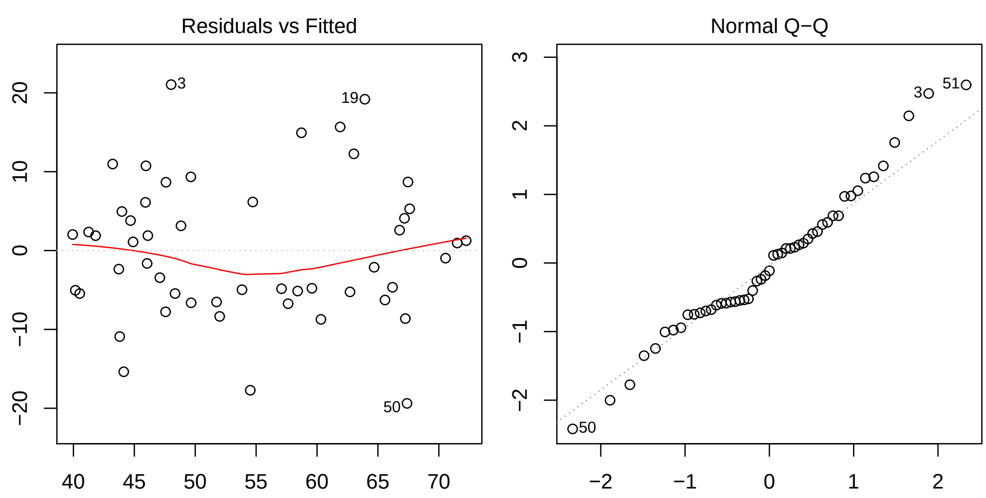


Obrázek 6: Polynomický trend sedmého řádu pro průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

Kvalitu modelu lze posoudit pomocí analýzy reziduí, kterou provedeme pro oba modely, tj. pro polynomy stupně šest a sedm.

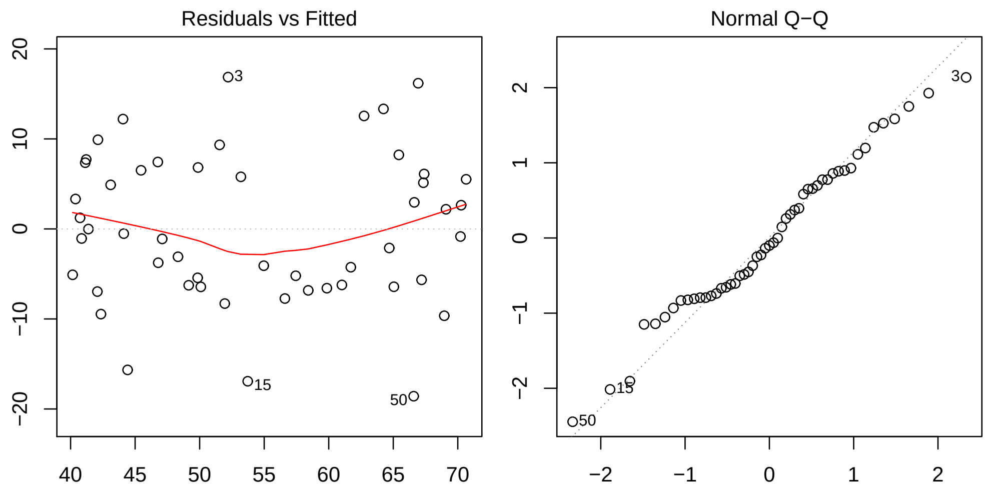
Pomocí příkazu plot, ve kterém bude argumentem odhadnutý model, můžeme vykreslit celou řadu grafů. Vybereme si první a druhý.

```
> par(mfrow = c(1, 2), mar = c(2, 2, 1, 0) + 0.5)
> plot(fitModel6, which = c(1, 2))
```



Obrázek 7: Vybrané grafy z analýzy reziduí pro polynomický trend šestého řádu pro průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

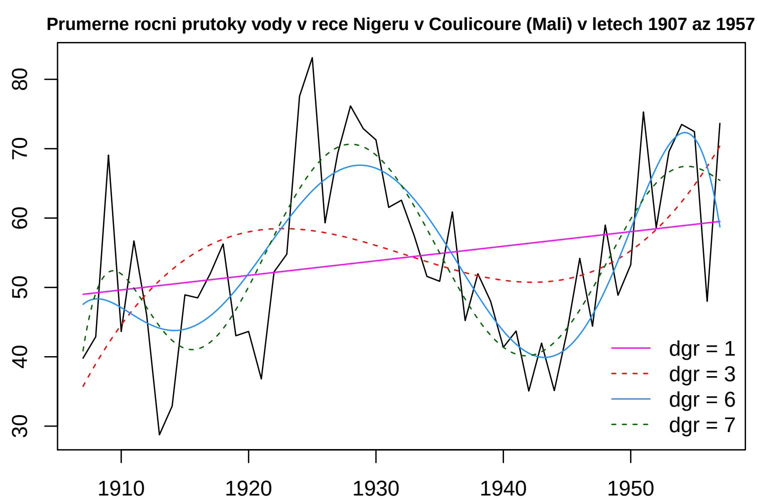
```
> par(mfrow = c(1, 2), mar = c(2, 2, 1, 0) + 0.5)
> plot(fitModel7, which = c(1, 2))
```



Obrázek 8: Vybrané grafy z analýzy reziduí pro polynomický trend sedmého řádu pro průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

Na závěr všechny odhadnuté trendy zakreslíme do jediného grafu.

```
> par(mar = c(2, 2, 1, 0) + 0.5)
> plot(NigerTS, main = TXT, cex.main = 0.85)
> matlines(gridt, cbind(Pred$fit, Pred3$fit, Pred6$fit, Pred7$fit),
     lty = c(1, 2, 1, 2), col = c("magenta", "red", "dodgerblue",
         "darkgreen"))
> legend("bottomright", legend = paste("dgr =", c(1, 3, 6, 7)),
     bty = "n", lty = c(1, 2, 1, 2), col = c("magenta", "red",
         "dodgerblue", "darkgreen"))
```



Obrázek 9: Polynomické trendy řádu jedna, tři, šest a sedm pro průměrné roční průtoky vody v řece Nigeru v Coulicouro (Mali).

C. Úkol:

Pro časové řady, které jste si našli ve druhém praktiku, vytvořte, vykreslete a analyzujte vhodné modely polynomického trendu.