

Semestrální práce 4ST431

Artem Vitkov

2025-06-08

Semestrální práce z předmětu Časové řady

1. Úvod

Tato práce se zabývá analýzou časové řady ekonomického charakteru – konkrétně denní národní spotřeby elektřiny ve Velké Británii. Spotřeba elektřiny je klíčovým ekonomickým ukazatelem, neboť její vývoj odráží ekonomické cykly, sezónní variace a společenské změny (např. svátky a volné dny).

Cílem této práce je aplikovat metody Box-Jenkinsovy metodologie (ARIMA, SARIMA, ARFIMA) k modelování dynamiky této časové řady, identifikovat vhodný model pro predikci budoucí spotřeby elektřiny a posoudit jeho predikční schopnosti.

Data pocházejí od společnosti National Grid ESO, která je systémovým operátorem elektrické soustavy ve Velké Británii. Zpracovávaná data jsou k dispozici od roku 2009 do roku 2024 a zachycují jak spotřebu elektřiny (proměnná `nd` – National Demand), tak indikátor svátků (`is_holiday`).

Zdroj: https://www.kaggle.com/datasets/albertovidalrod/electricity-consumption-uk-20092022?select=historic_demand_2009_2024_noNaN.csv

1.1 Knihovny a vstupní data byla načtena následujícím kódem:

```
# =====  
# R Script: Time Series Processing & Modeling  
# Based on provided test template and topics list  
# =====  
  
# --- 0) Load required libraries ---  
library(tidyverse) # data manipulation  
  
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --  
## v dplyr      1.1.4      v readr      2.1.5  
## v forcats    1.0.0      v stringr    1.5.0  
## v ggplot2    3.5.1      v tibble     3.2.1  
## v lubridate  1.9.3      v tidyr      1.3.1  
## v purrr      1.0.2  
## -- Conflicts ----- tidyverse_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag()     masks stats::lag()  
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors  
  
library(lubridate) # data aggregation  
library(readxl)    # read Excel files  
library(ggplot2)    # visualization  
library(tseries)    # ADF, Jarque-Bera, etc.
```

```

## Registered S3 method overwritten by 'quantmod':
##   method      from
##   as.zoo.data.frame zoo

library(FinTS)      # ARCH LM Test

## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##   as.Date, as.Date.numeric

library(forecast)   # ACF/PACF, STL decomposition, auto.arima, forecasting

##
## Attaching package: 'forecast'
##
## The following object is masked from 'package:FinTS':
##
##   Acf

library(future.apply) # parallel computing

## Loading required package: future
##
## Attaching package: 'future'
##
## The following object is masked from 'package:tseries':
##
##   value

library(fracdiff)    # ARFIMA
library(rugarch)     # GARCH models

## Loading required package: parallel
##
## Attaching package: 'rugarch'
##
## The following object is masked from 'package:purrr':
##
##   reduce
##
## The following object is masked from 'package:stats':
##
##   sigma

library(Metrics)     # MAE, MSE, RMSE

##
## Attaching package: 'Metrics'
##
## The following object is masked from 'package:forecast':
##
##   accuracy

```

```

library(urca)          # Detailed ADF regression outputs
library(dynlm)         # Dynamic linear models for manual ADF regressions
library(uroot)         # provides hegy.test()          # HEGY seasonal unit-root test
library(Matrix)        # pokročilý maticový výpočet

##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
library(xts)           # transformace ts

##
## ##### Warning from 'xts' package #####
## #
## # The dplyr lag() function breaks how base R's lag() function is supposed to #
## # work, which breaks lag(my_xts). Calls to lag(my_xts) that you type or      #
## # source() into this session won't work correctly.                          #
## #
## # Use stats::lag() to make sure you're not using dplyr::lag(), or you can add #
## # conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop         #
## # dplyr from breaking base R's lag() function.                             #
## #
## # Code in packages is not affected. It's protected by R's namespace mechanism #
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning. #
## #
## #####
##
## Attaching package: 'xts'
##
## The following objects are masked from 'package:dplyr':
##
##     first, last
library(knitr)         # Tabulka výsledků

# Načtení dat včetně proměnné is_holiday
data <- read.csv("historic_demand_2009_2024_noNaN.csv")

```

1.2 Popis dat a jejich příprava

Původní data jsou dostupná na půlhodinové frekvenci, což znamená 48 záznamů za každý den. Pro potřeby této analýzy byla data agregována na denní průměrné hodnoty, což významně zjednodušuje analýzu dlouhodobých sezónních a trendových komponent.

Díky této přípravě můžeme přistoupit k analýze stacionarity a sezónnosti časové řady, která je popsána v následující kapitole.

Výsledná denní časová řada byla definována s týdenní sezónností (frekvence = 7 dní):

```

set.seed(6945)

data$datetime <- as.POSIXct(data$settlement_date)

```

```
# Agregace na denní průměrné hodnoty
daily_data <- data %>%
  mutate(date = as.Date(settlement_date)) %>%
  group_by(date) %>%
  summarise(nd = mean(nd),
            is_holiday = max(is_holiday))

# Definice denní časové řady s týdenní sezónností (frekvence=7)
ts_daily <- ts(daily_data$nd, frequency = 7, start = 1)
```

1.3 Vizualní analýza časové řady

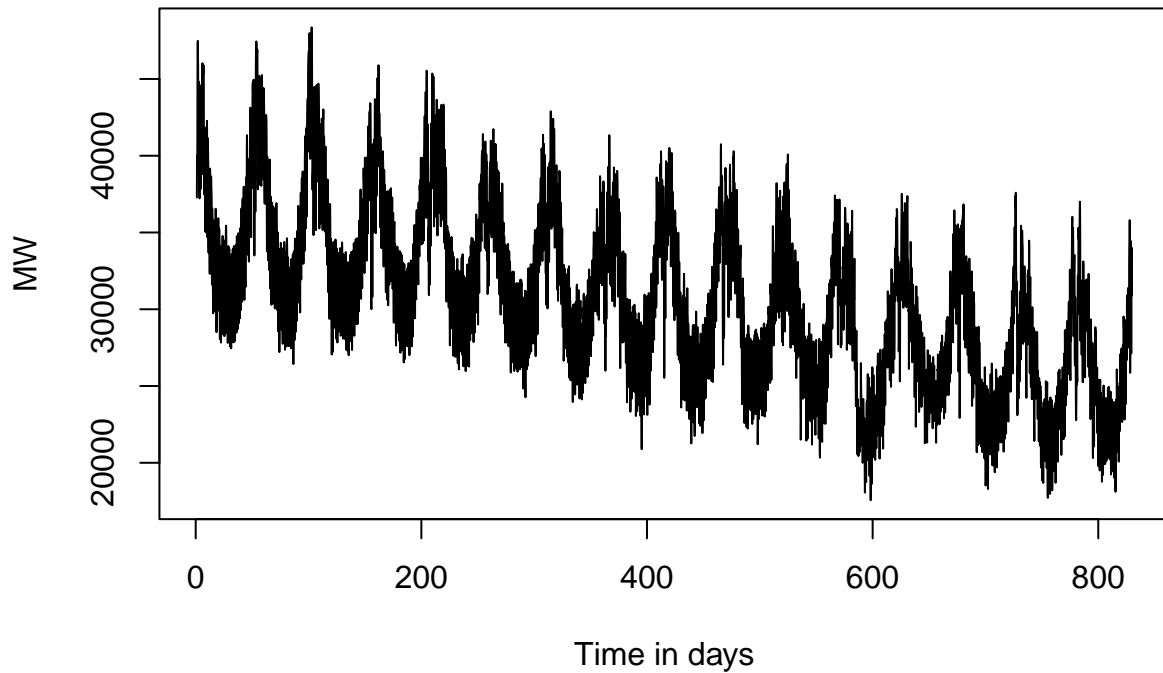
Na grafu je zobrazen vývoj denní národní spotřeby elektřiny ve Velké Británii za období od roku 2009 do roku 2024.

Z grafu lze pozorovat:

- Jasný sezónní charakter, který se projevuje opakovanými výkyvy v krátkých intervalech (týdenní cyklus).
- Klesající dlouhodobý trend, což může indikovat změny v energetické politice, efektivnější technologie, ekonomické faktory nebo změny v demografické struktuře.
- Výraznou volatilitu, která poukazuje na časté krátkodobé výkyvy způsobené specifickými událostmi, jako jsou svátky, extrémní počasí nebo technické výpadky.

```
par(mfrow=c(1,1))
# Time series plot
plot(ts_daily, main="National Demand of Electricity in the United Kingdom 2009-2024", ylab="MW", xlab =
```

National Demand of Electricity in the United Kingdom 2009–2024



1.4 Rozdělení dat na trénovací a testovací sadu

Pro posouzení schopnosti modelů efektivně předpovídat budoucí hodnoty byla původní denní časová řada rozdělena na dvě části:

1. Trénovací sada obsahuje prvních 95 dat (5512 dní).
2. Testovací sada zahrnuje zbývajících 5 dat (284 dní).

```
train_size_daily <- floor(0.95 * length(ts_daily))
train_ts_daily <- window(ts_daily, end=c(floor(time(ts_daily)[train_size_daily]),
                                         cycle(ts_daily)[train_size_daily]))

test_ts_daily <- window(ts_daily, start=c(floor(time(ts_daily)[train_size_daily])+1,
                                           cycle(ts_daily)[train_size_daily]+1))

length(train_ts_daily)

## [1] 5512

length(test_ts_daily)

## [1] 284
```

2. Analýza sezónnosti

Pro detailní posouzení týdenní sezónnosti v časové řadě národní spotřeby elektřiny byla provedena sezónní vizualizace (Seasonal Plot). Tento graf zobrazuje jednotlivé týdny, což umožňuje snadnou identifikaci pravidelných sezónních patternů.

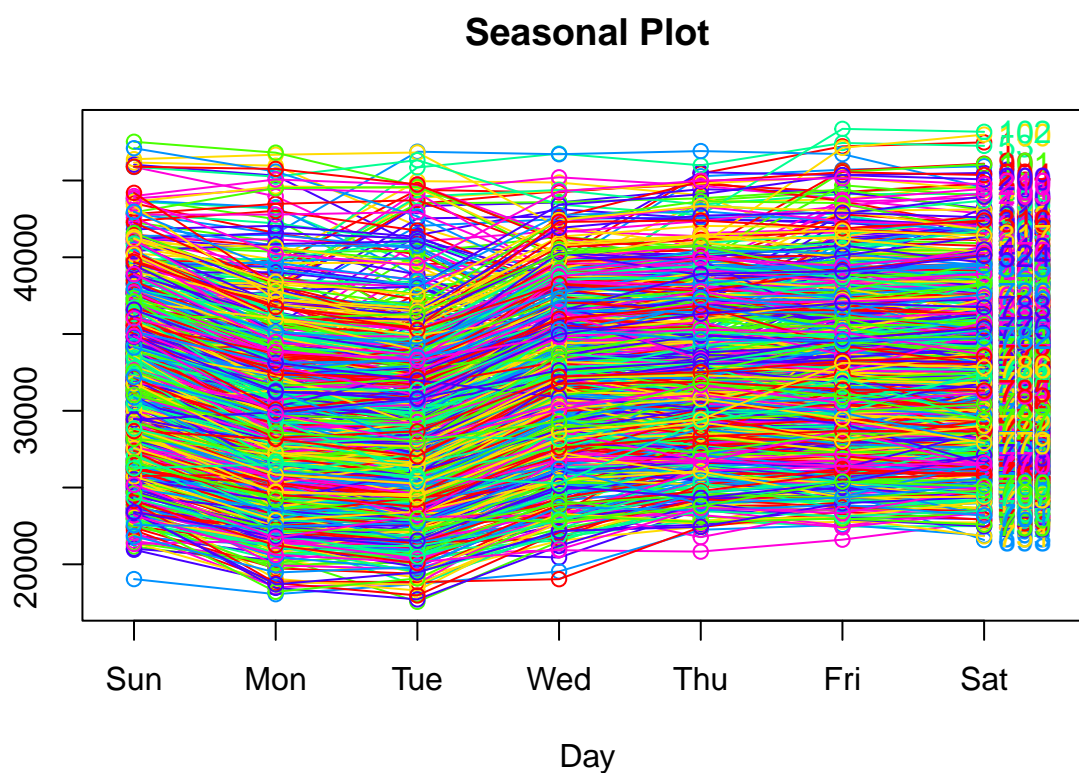
2.1 Grafická analýza

Z grafu je zřejmé, že:

- Spotřeba elektřiny je nejnižší v neděli a během pracovního týdne postupně narůstá, vrcholí v pracovní dny a klesá opět o víkendu.
- Týdenní sezónnost je zřejmá a opakuje se pravidelně každý týden.

Tento graf naznačuje významnou týdenní sezónnost, která bude dále zohledněna při modelování a odstraňování sezónních efektů.

```
# Seasonal plot
dates <- cycle(train_ts_daily)
seasonplot(train_ts_daily, col=rainbow(7), year.labels=TRUE, main="Seasonal Plot")
```



2.2 STL dekompozice

Pro hlubší porozumění struktury časové řady byla provedena STL dekompozice (Seasonal-Trend Decomposition based on Loess), která rozkládá původní řadu na tři komponenty: sezónní složku, trendovou složku a náhodnou složku (rezidua).

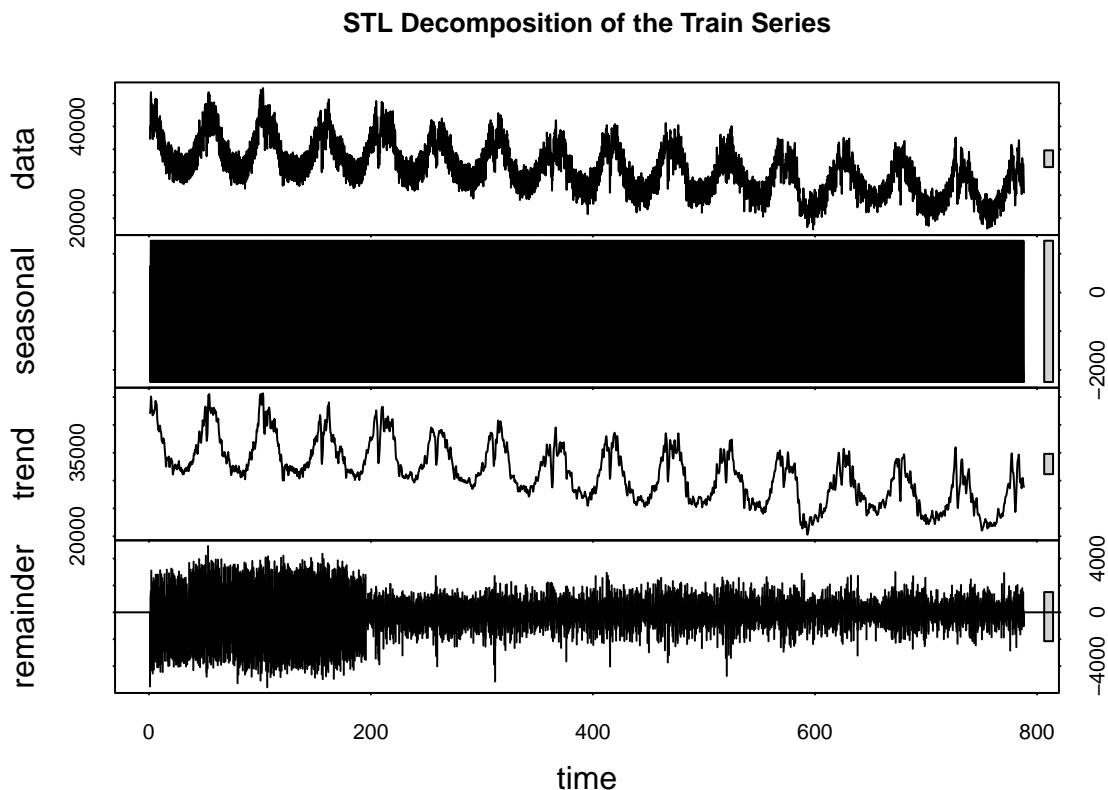
Z grafu je zřejmé:

Sezónní složka: Identifikuje pravidelné týdenní cykly, které jsou konstantní v průběhu celého období.

Trendová složka: Vykazuje dlouhodobě klesající tendenci, což naznačuje postupné snížení národní spotřeby elektřiny v čase.

Reziduální složka: Neobsahuje viditelné vzorce, což značí, že většina systematické variability byla zachycena sezónní a trendovou složkou.

```
stl_decomp <- stl(train_ts_daily, s.window="periodic")  
plot(stl_decomp, main="STL Decomposition of the Train Series")
```



2.3. Testování sezónnosti

2.3.1 Deterministický F-test (ANOVA):

Testujeme nulovou hypotézu, že týdenní průměry spotřeby elektřiny jsou stejné.

Zde na jakékoli rozumné hladině významnosti zamítáme nulovou hypotézu, což indikuje statisticky významnou deterministickou sezónní složku.

```
# --- Seasonality Tests ---  
# Deterministic F-test for seasonality  
# H0: no seasonal effect (means equal across months); H1: at least one month differs  
f_mod <- lm(train_ts_daily ~ factor(cycle(train_ts_daily)))  
anova(f_mod)
```

```
## Analysis of Variance Table  
##  
## Response: train_ts_daily  
##  
##          Df      Sum Sq   Mean Sq F value    Pr(>F)  
## factor(cycle(train_ts_daily))    6 1.0185e+10 1697511018  58.558 < 2.2e-16 ***  
## Residuals                    5505 1.5958e+11   28988357  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.3.2 Kruskal-Wallisův neparametrický test:

Je testována hypotéza, že distribuce hodnot v jednotlivých dnech týdne jsou stejné (test mediánů).

Zamítáme nulovou hypotézu, což potvrzuje, že mediány spotřeby elektřiny se napříč týdnem významně liší.

```
# Nonparametric Kruskal-Wallis test  
# H0: distribution of series across seasons is identical; H1: distributions differ  
kruskal.test(train_ts_daily ~ factor(cycle(train_ts_daily)))
```

```
##  
##  Kruskal-Wallis rank sum test  
##  
## data:  train_ts_daily by factor(cycle(train_ts_daily))  
## Kruskal-Wallis chi-squared = 321.57, df = 6, p-value < 2.2e-16
```

2.3.3 Test plovoucí sezónnosti (meziroční variabilita):

Ověřujeme, zda se průměrná spotřeba elektřiny mění v jednotlivých týdnech napříč roky.

Zamítáme nulovou hypotézu, což značí významné změny sezónních průměrů v čase, tedy plovoucí sezónnost.

Tyto výsledky jasně indikují přítomnost silné sezónní komponenty, která se navíc mění v průběhu času, což neumožní nám používat dummy-proměnné pro modelování sezónnosti. Proto budeme muset použít stochastické modely pomocí Box-Jenkinsovy metodologie.

```
# Moving seasonality test (Between-year variability)  
# H0: no between-year effect; H1: number of airline passenger means differ across years  
weeks <- floor(time(train_ts_daily))
```



```
f_mod2 <- lm(train_ts_daily ~ factor(weeks))
anova(f_mod2)

## Analysis of Variance Table
##
## Response: train_ts_daily
##              Df      Sum Sq   Mean Sq F value    Pr(>F)
## factor(weeks) 787 1.4469e+11 183848674  34.633 < 2.2e-16 ***
## Residuals    4724 2.5077e+10   5308439
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.4. Testování stacionarity

Pro ověření stacionarity časové řady byly použity testy jednotkových kořenů:

2.4.1 Augmented Dickey-Fullerův test (ADF):

Testuje přítomnost jednotkového kořene (nulová hypotéza: řada je nestacionární).

Zamítáme nulovou hypotézu; časová řada je na první pohled stacionární, avšak kvůli vizuální přítomnosti trendu a sezónnosti provedeme další ověření.

```
# --- Stationarity Tests ---
# Augmented Dickey-Fuller Test
adf0 <- adf.test(train_ts_daily, alternative="stationary")

## Warning in adf.test(train_ts_daily, alternative = "stationary"): p-value
## smaller than printed p-value

print(adf0)

##
## Augmented Dickey-Fuller Test
##
## data: train_ts_daily
## Dickey-Fuller = -5.6014, Lag order = 17, p-value = 0.01
## alternative hypothesis: stationary
```

2.4.2 KPSS test:

Otestujeme zase nulovou hypotézu o stacionaritě časové řady.

p-hodnota je numerická nula, pak nulová hypotéza je na základě tohoto faktu námi zamítnuta, což indikuje přítomnost jednotkového kořene a, tím pádem, nestacionaritu procesu.

```
# KPSS Test (confirmation)
# H0: series is stationary; H1: has unit root
kpss0 <- kpss.test(train_ts_daily)

## Warning in kpss.test(train_ts_daily): p-value smaller than printed p-value

print(kpss0)

##
## KPSS Test for Level Stationarity
##
## data: train_ts_daily
```

```
## KPSS Level = 22.723, Truncation lag parameter = 10, p-value = 0.01
```

2.4.3 Další ověření pomocí regresní formulace ADF testu:

Odhadnutá hodnota parametru $\rho(\phi - 1)$: -0.0989 , p-hodnota: ≈ 0

Odpovídající hodnota ϕ : 0.9011

Odhad AR(1) přímo na úrovni dat: ϕ : 0.9133, p-hodnota: ≈ 0

Regresní koeficient AR(1) a p-hodnota ADF-testu naznačuje, že proces je stacionární, byť s dlouhou přetrvávající pamětí.

```
# --- Extract estimates ---
ur0 <- ur.df(train_ts_daily, type="drift", selectlags="AIC")
lags0 <- ur0@lags
model_adf0 <- dynlm(diff(train_ts_daily) ~ L(train_ts_daily, 1) + L(diff(train_ts_daily), 1:lags0))
# rho estimate and p-value
coeff_rho0 <- coef(model_adf0)["L(train_ts_daily, 1)"]
pval_rho0 <- summary(model_adf0)$coefficients["L(train_ts_daily, 1)", "Pr(>|t|)"]
cat("ADF rho (phi-1):", round(coeff_rho0,4), "p-value:", round(pval_rho0,4), "\n")

## ADF rho (phi-1): -0.0989 p-value: 0
phi0 <- coeff_rho0 + 1
cat("Calculated phi:", round(phi0,4), "\n")

## Calculated phi: 0.9011
```

Vzhledem k protichůdným výsledkům (ADF vs. KPSS) jsme se rozhodli ČŘ následně diferencovat:

1. Nejprve provedeme sezónní diferenciaci s periodou 7 (týdenní sezónnost).
2. Poté použijeme první diferenci pro odstranění trendu.

```
# Direct AR(1) regression on levels
model_ar1 <- dynlm(train_ts_daily ~ L(train_ts_daily, 1))
phi_hat <- coef(model_ar1)["L(train_ts_daily, 1)"]
pval_phi_hat <- summary(model_ar1)$coefficients["L(train_ts_daily, 1)", "Pr(>|t|)"]
cat("Direct AR(1) phi_hat:", round(phi_hat,4), "p-value:", round(pval_phi_hat,4), "\n")

## Direct AR(1) phi_hat: 0.9133 p-value: 0
rho_hat <- phi_hat - 1
cat("Corresponding rho_hat:", round(rho_hat,4), "\n")

## Corresponding rho_hat: -0.0867
```

2.4.4 Další ověření pomocí regresní formulace ADF testu:

Dickey-Fuller statistika: -28.892 , p-hodnota: < 0.01

Závěr: Po diferenciaci je řada jednoznačně stacionární.

```
# If non-stationary, difference once and re-test
# Sezónní diferenciaci na periodě 7
ts_diff_season <- diff(train_ts_daily, lag = 7)
```

```

# Další první diferenciaci pro odstranění trendu
ts_diff <- diff(ts_diff_season, differences = 1)

# ADF test na stacionaritě
adf_result <- adf.test(ts_diff, alternative = "stationary")

## Warning in adf.test(ts_diff, alternative = "stationary"): p-value smaller than
## printed p-value
print(adf_result)

##
## Augmented Dickey-Fuller Test
##
## data: ts_diff
## Dickey-Fuller = -28.892, Lag order = 17, p-value = 0.01
## alternative hypothesis: stationary
# p-hodnota < 0.05 indikuje stacionaritě po dvojím diferencování

```

2.4.5 Regresní formulace ADF po diferenciaci:

Parametr ρ (diff): -1.2984 , p-hodnota: ≈ 0

Tento výsledek již jednoznačně indikuje stacionaritě diferencované řady.

Shrnutí pro podkapitolu: výsledky testů potvrzují, že časová řada vyžaduje jak sezónní, tak první běžnou diferenciaci, aby byla dosažena stacionarizace nutná pro korektní aplikaci Box-Jenkinsové metodologie.

```

# Confirmation on first-differenced series
ur1 <- ur.df(ts_diff, type="drift", selectlags="AIC")
lags1 <- ur1@lags
model_adf1 <- dynlm(diff(ts_diff) ~ L(ts_diff, 1) + L(diff(ts_diff), 1:lags1))
coeff_rho1 <- coef(model_adf1)["L(ts_diff, 1)"]
pval_rho1 <- summary(model_adf1)$coefficients["L(ts_diff, 1)", "Pr(>|t|)"]
cat("ADF rho (diff):", round(coeff_rho1,4), "p-value:", round(pval_rho1,4), "\n")

## ADF rho (diff): -1.2984 p-value: 0

```

2.5 Korelogramy ACF a PACF

Korelogramy ACF a PACF pro původní řadu naznačují silnou autokorelaci a nutnost diferenciaci.

Dokonce zde můžeme pozorovat hyperbolický průběh ACF, který je vlastností procesu s dlouhou pamětí. Toto teoreticky by mohlo naznačovat vhodnost modelování této ČŘ pomocí ARFIMA modelů, avšak tuto skutečnost ověříme v další části analýzy.

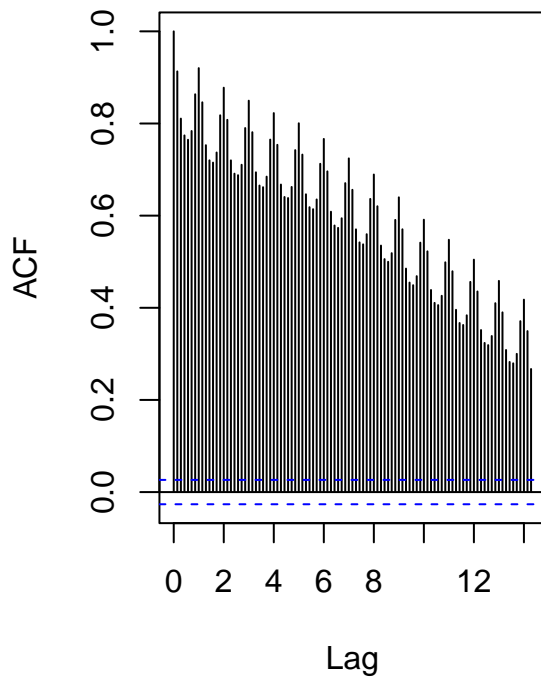
Také u PACF lze pozorovat zbytky cyklické složky, kterou ale nebudeme zohledňovat v rámci našeho modelování.

```

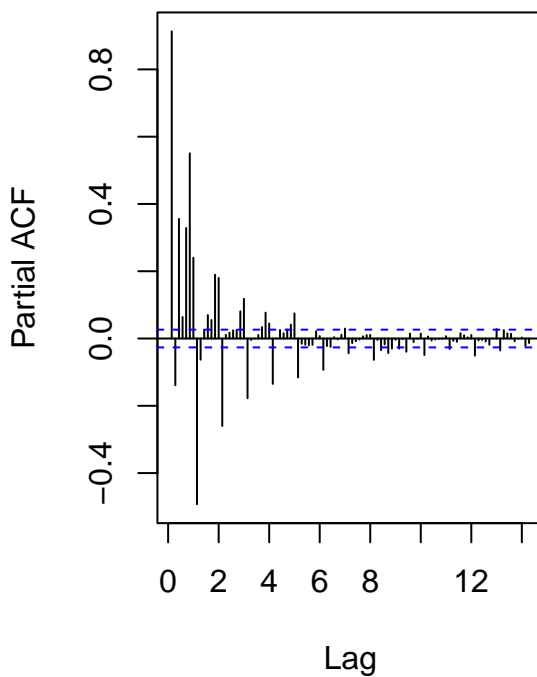
# --- ACF & PACF Plots ---
par(mfrow = c(1, 2))
acf(train_ts_daily, lag.max = 100, main = "ACF původní řady")
pacf(train_ts_daily, lag.max = 100, main = "PACF původní řady")

```

ACF puvodní rady



PACF puvodní rady

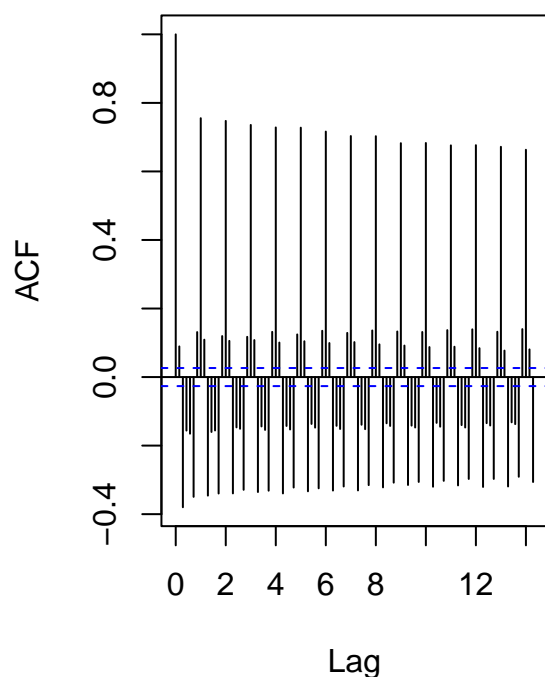


Po provedení první diferenciace se výrazně snížila autokorelace, ale neodstranila se vůbec sezónnost, což naznačuje, že další modelování trendu a sezonality by bylo vhodné provádět právě na diferencované řadě.

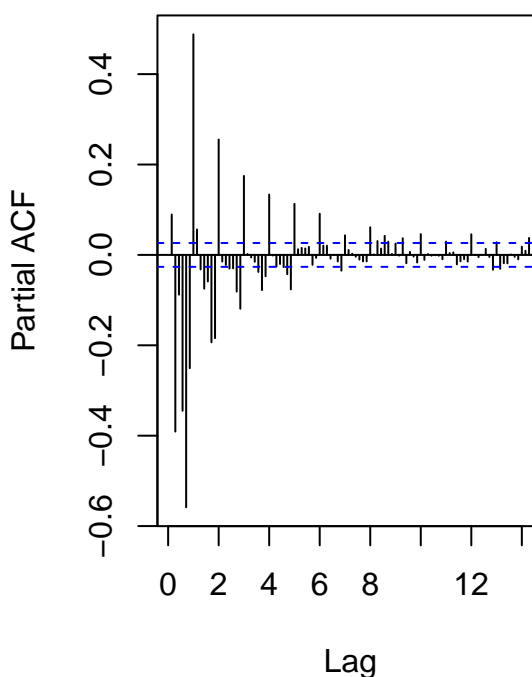
```
# Další první diferenciace pro odstranění sezonnosti
ts_diff <- diff(train_ts_daily, differences = 1)

# Korelogramy
par(mfrow = c(1, 2))
acf(ts_diff, lag.max = 100, main = "ACF diferencované řady")
pacf(ts_diff, lag.max = 100, main = "PACF diferencované řady")
```

ACF diferencované rady



PACF diferencované rady



3. Modelování časové řady

V této kapitole budeme porovnávat smysluplnost a přiměřenost tří navržených stochastických modelů pro predikci.

Model 1: SARIMA(2,0,2)(0,1,1)[7]

První model byl identifikován automatickým algoritmem na základě Akaikeho informačního kritéria (AIC). Zvolený model je SARIMA(2,0,2)(0,1,1) se sezónností s periodou 7 dní.

Všechny odhadnuté parametry (ar1, ar2, ma1, ma2, sma1) jsou statisticky významné ($p < 0.05$), což naznačuje správnou specifikaci modelu.

AIC hodnota je poměrně vysoká (93841.34), což naznačuje možnost dalšího zlepšení modelu.

```
model_stl_arima <- auto.arima(train_ts_daily, seasonal=TRUE, stepwise=FALSE, approximation=FALSE)
summary(model_stl_arima)
```

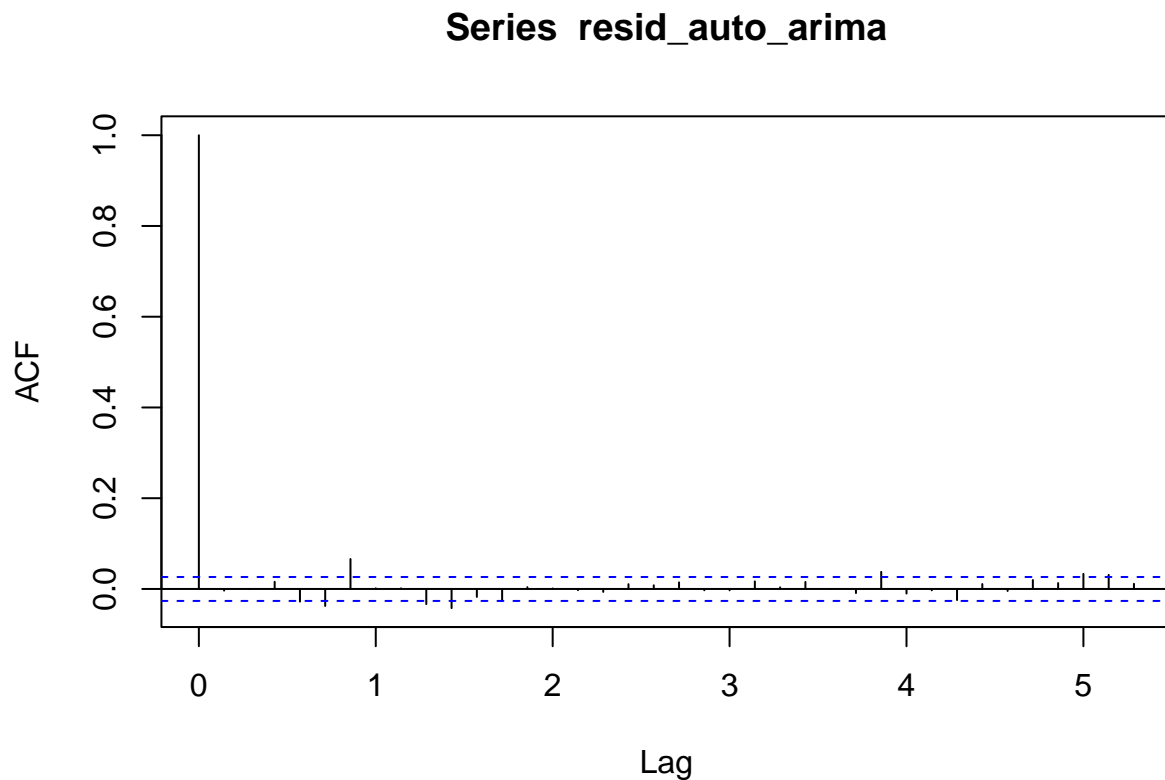
```
## Series: train_ts_daily
## ARIMA(2,0,2)(0,1,1)[7]
##
## Coefficients:
##      ar1      ar2      ma1      ma2      sma1
##      1.1445 -0.1887 -0.3120 -0.1586 -0.7870
## s.e.  0.0962  0.0877  0.0953  0.0197  0.0101
##
## sigma^2 = 1477950: log likelihood = -46914.67
## AIC=93841.34  AICc=93841.36  BIC=93881.02
```

```
##
## Training set error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -6.402562 1214.386 886.4157 -0.1329913 2.922781 0.5629532
##           ACF1
## Training set -0.003715366

# další testy
resid_auto_arima <- residuals(model_stl_arima)
```

V korelogramu ACF je přítomna statisticky významná a dokonalá korelace v nultém lagu, kterou nijak nebudeme intepretovat.

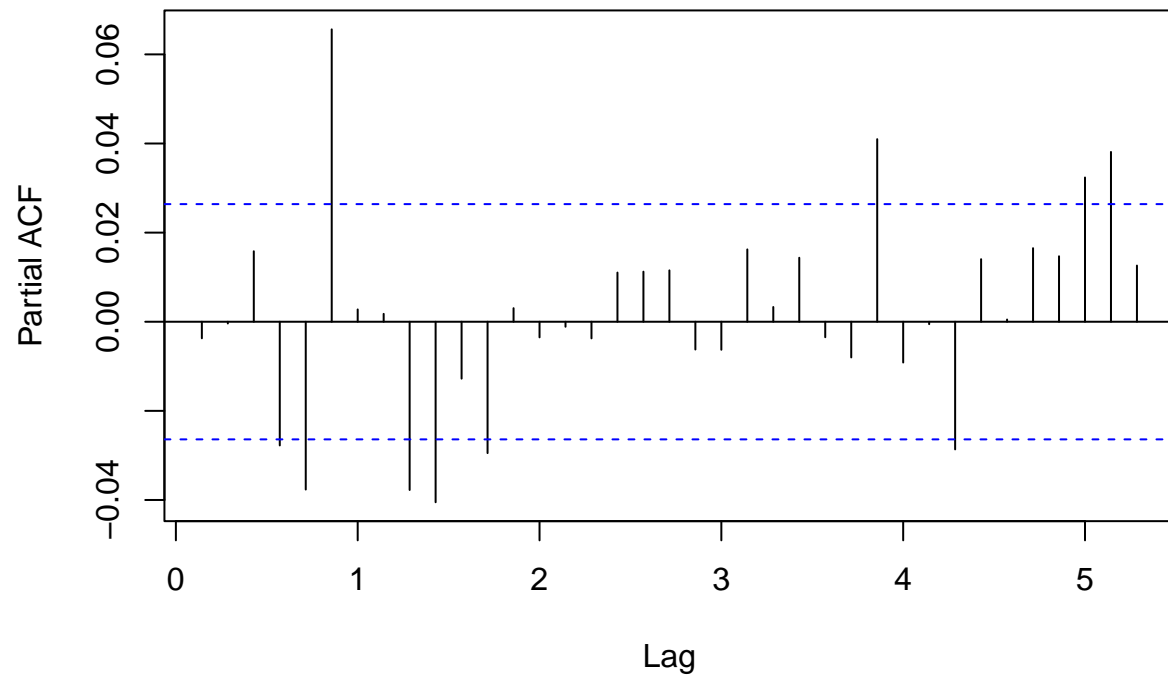
```
# Korelogramy reziduí ARIMA modelu
acf(resid_auto_arima)
```



Korelogram PACF reziduí ukazuje, že většina korelací je již uvnitř mezi konfidenčních intervalů. Přesto, že jsou korelace velmi nízké, existují signifikantní autokorelace na některých periodách, což svědčí o potřebě dalšího vylepšení modelu, například za použití GARCH modelování volatility.

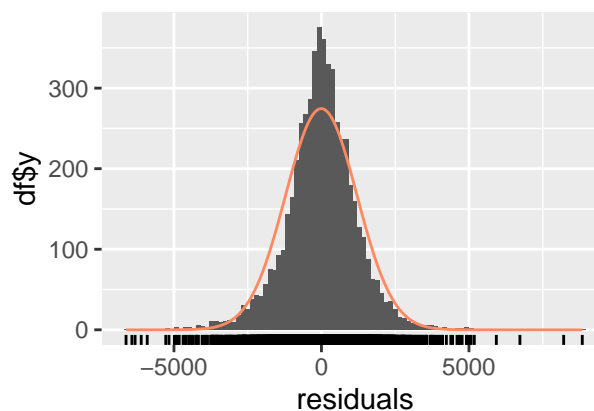
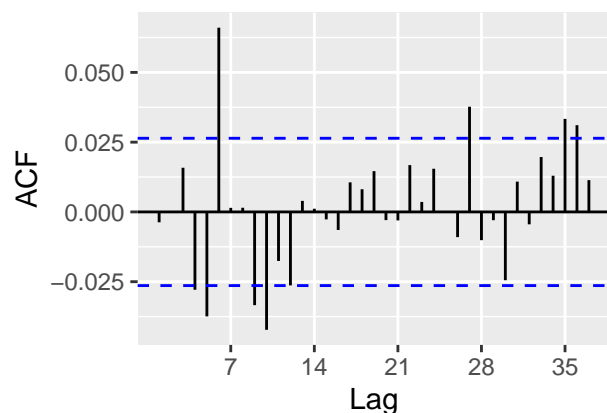
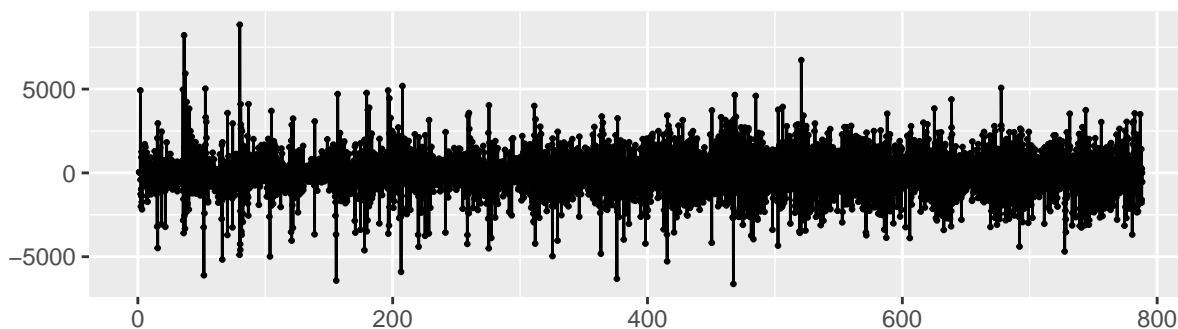
```
pacf(resid_auto_arima)
```

Series resid_auto_arima



```
checkresiduals(resid_auto_arima)
```

Residuals



```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 59.208, df = 14, p-value = 1.615e-07
##
## Model df: 0.   Total lags used: 14
```

Diagnostika reziduí:

- Ljung-Boxův test potvrdil přítomnost zbývající autokorelace v reziduích (p-hodnota ≈ 0), což ukazuje na ne zcela dokonalé očištění řady od korelací.
- ARCH LM-test výrazně zamítá nulovou hypotézu o absenci heteroskedasticity (p-hodnota ≈ 0). To naznačuje, že rezidua mají měnící se rozptyl v čase a potvrzuje přítomnost ARCH efektů.
- Jarque-Bera test reziduí ukazuje, že rezidua se výrazně odchyľují od normality (p-hodnota ≈ 0), což je patrné také na vizuálním QQ plotu.

```
Box.test(resid_auto_arima, lag=7, type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  resid_auto_arima
## X-squared = 37.547, df = 7, p-value = 3.693e-06
```



```

ArchTest(resid_auto_arima, lags=7)

##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  resid_auto_arima
## Chi-squared = 345.35, df = 7, p-value < 2.2e-16

jarque.bera.test(resid_auto_arima)

##
##  Jarque Bera Test
##
## data:  resid_auto_arima
## X-squared = 2119.2, df = 2, p-value < 2.2e-16

```

Přestože byl model SARIMA schopen dobře popsat hlavní dynamiku ČŘ, diagnostika reziduí poukazuje na zbývající struktury v rozptylu, které je potřeba řešit, například prostřednictvím modelu GARCH. Model SARIMA samotný je vhodný jako základ pro další rozšíření analýzy.

GJR-GARCH(3,3) model reziduí (sstd distribuce):

a. GJR-GARCH(3,3) úspěšně stabilizoval volatilitu reziduí. Parametry omega, alpha1 a beta1 jsou statisticky významné, zatímco vyšší řady parametrů (alpha2, alpha3, beta2, beta3) jsou zanedbatelné.

b. Modelování asymetrických efektů (gamma parametry) je významné, což potvrzuje přítomnost tzv. leverage efektu (záporné šoky mají větší dopad než kladné).

Rozdělení reziduí bylo úspěšně zachyceno pomocí skewed studentovy distribuce (Q-Q plot ukazuje dobrou shodu).

```

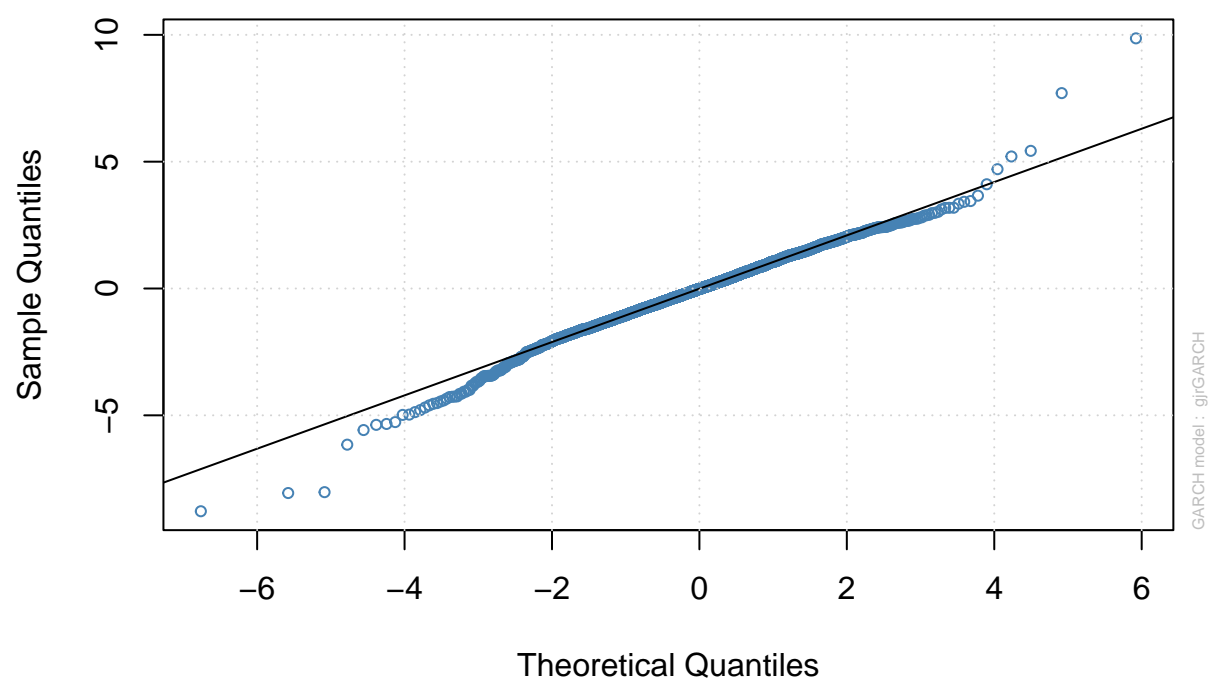
spec_garch_updated <- ugarchspec(
  variance.model = list(model="gjrGARCH", garchOrder=c(1,1)),
  mean.model = list(armaOrder=c(2,2), include.mean=TRUE),
  distribution.model = "sstd"
)

fit_garch_updated <- ugarchfit(spec_garch_updated, resid_auto_arima, solver='hybrid')

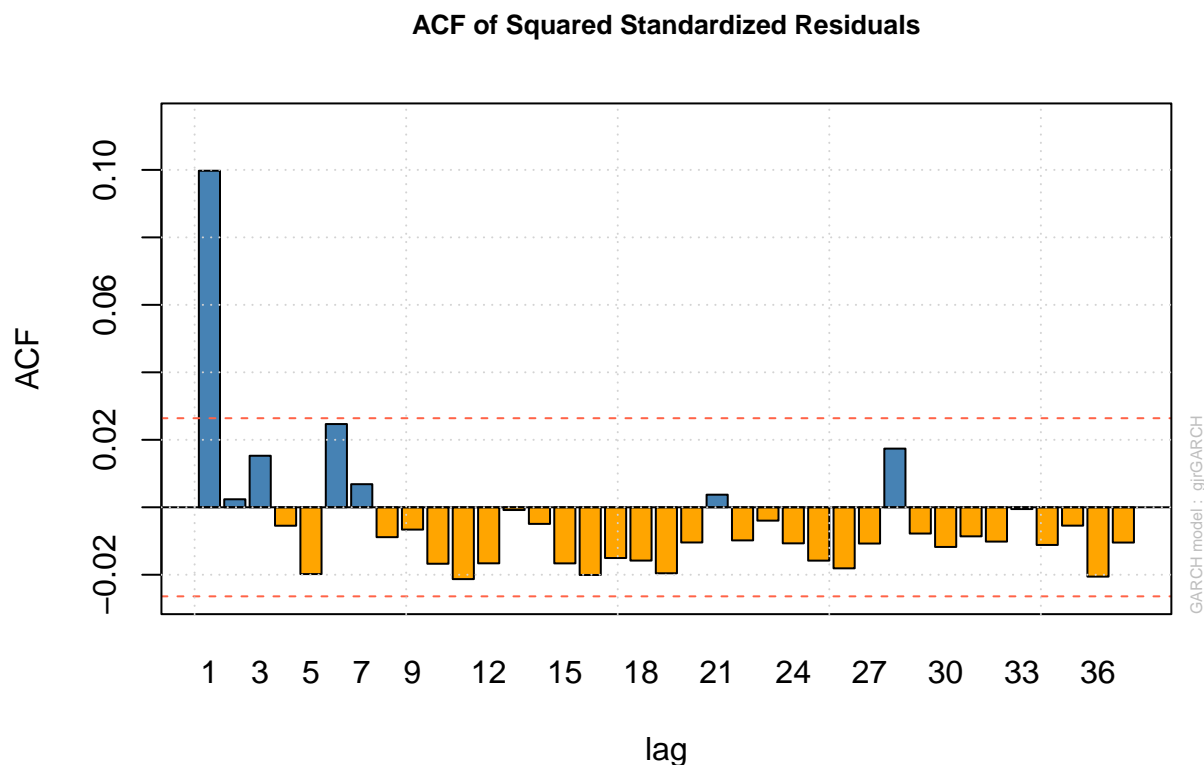
plot(fit_garch_updated, which = 9) # QQ plot standardizovaných reziduí

```

sstd – QQ Plot



```
plot(fit_garch_updated, which = 11) # Standardizovaná ACF kvartátů reziduí
```



Diagnostika reziduí GJR-GARCH(1,1):

- a. Rezidua GJR-GARCH modelu již nevykazují výrazný problém s autokorelací ani heteroskedasticitou (potvrzeno Ljung-Box testy a ARCH LM testem s přijatelnými p-hodnotami).
- b. Q-Q plot indikuje mírné odchylky v ocasech rozdělení, ale celkově je distribuce reziduí adekvátně zachycena.

GJR-GARCH(1,1) model se skewed t-distribucí reziduí poměrně dobře zachycuje volatilitu spotřeby elektřiny, včetně asymetrické reakce na negativní události a výskytu extrémních hodnot.

Navzdory drobným přetrvávajícím problémům s nestabilitou parametrů a mírnou reziduální autokorelací jde o nejvhodnější model pro řízení rizik a rozhodování v energetickém sektoru.

Máme zde relativně velkou chybu predikce, protože jsme nedokázali zachytit dlouhodobou cyklickou složku svým modelem SARIMA, který je vhodný jen pro krátké predikce.

```
MSE <- mean((test_ts_daily - final_forecast_mean)^2)
MAE <- mean(abs(test_ts_daily - final_forecast_mean))
RMSE <- sqrt(MSE)

predikce_tab <- data.frame(
  MSE = round(MSE,2),
  MAE = round(MAE,2),
  RMSE = round(RMSE,2)
)
```

```
kable(predikce_tab, caption="Predikční chyby SARIMA + GJR-GARCH")
```

Table 1: Predikční chyby SARIMA + GJR-GARCH

MSE	MAE	RMSE
27265287	4633.18	5221.62

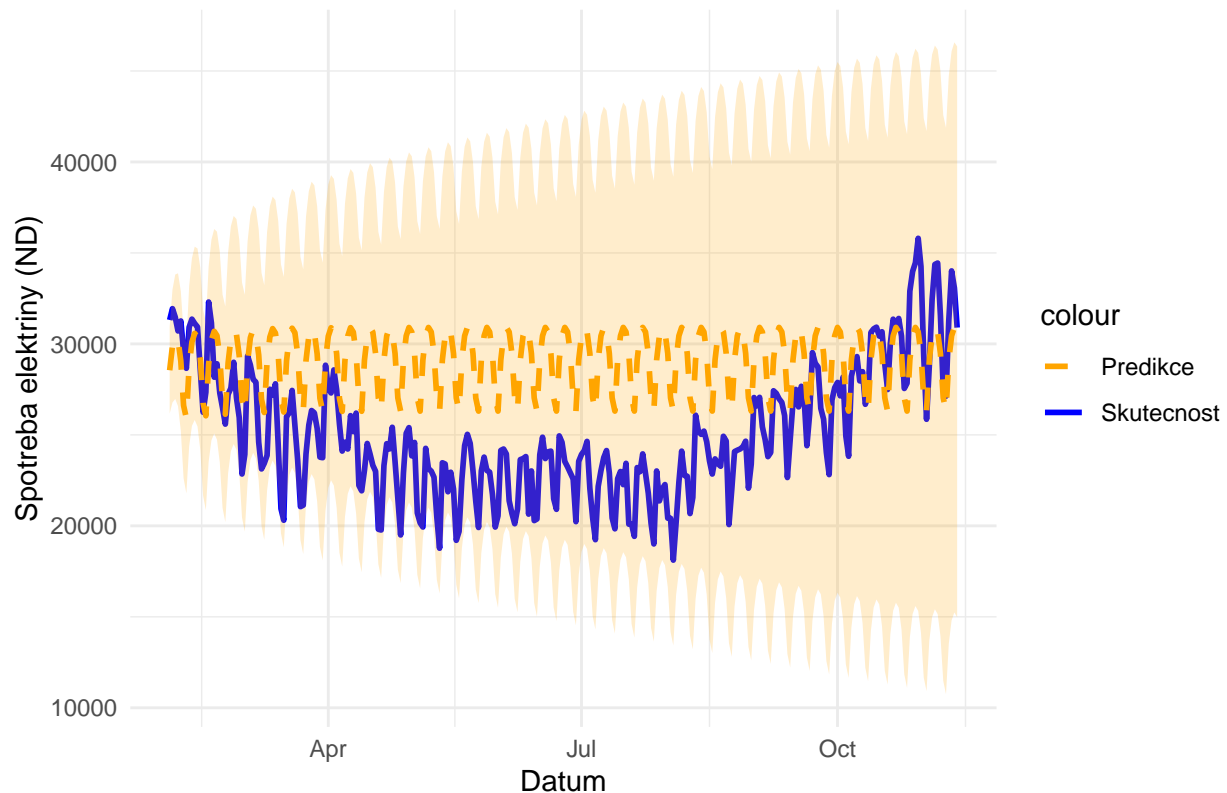
```
train_end_date <- as.Date("2009-01-01") + length(train_ts_daily) - 1
forecast_dates <- seq.Date(from = train_end_date + 1, by = "day", length.out = h)

pred_df <- data.frame(
  Datum = forecast_dates,
  Skutecnost = as.numeric(test_ts_daily),
  Predikce = as.numeric(final_forecast_mean),
  Lower = as.numeric(final_forecast_lower),
  Upper = as.numeric(final_forecast_upper)
)

ggplot(pred_df, aes(x = Datum)) +
  geom_line(aes(y = Skutecnost, color = "Skutečnost"), size = 1) +
  geom_line(aes(y = Predikce, color = "Predikce"), size = 1, linetype = "dashed") +
  geom_ribbon(aes(ymin = Lower, ymax = Upper), alpha = 0.2, fill = "orange") +
  labs(title = "Finální predikce SARIMA(2,0,2)(0,1,1)[7] + GJR-GARCH(1,1)",
    y = "Spotřeba elektřiny (ND)", x = "Datum") +
  scale_color_manual(values = c("Skutečnost" = "blue", "Predikce" = "orange")) +
  theme_minimal()

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

Finální predikce SARIMA(2,0,2)(0,1,1)[7] + GJR-GARCH(1,1)



Aditivní kombinace SARIMA(2,0,2)(0,1,1)[7] a GJR-GARCH(1,1) modelu vykazuje hodnotu $RMSE = 5221.62$, což představuje přibližně 17.5 průměrné hodnoty časové řady.

Z výsledného grafu je patrné, že kombinovaný model zachycuje hlavní trendy a sezónní cykly relativně dobře, avšak některé extrémní hodnoty a nečekané změny spotřeby zůstávají obtížně predikovatelné.

Celkově lze tento model považovat za vhodný pro řízení dlouhodobých kapacit, avšak vyžaduje opatrnost při plánování v období očekávaných extrémních změn v ekonomice nebo počasí.

Model 2 SARIMA (3,1,4)(1,1,1)[7]:

Model vykazuje relativně nízkou hodnotu informačních kritérií ($AIC = 93811.06$), což indikuje dobrou shodu s daty.

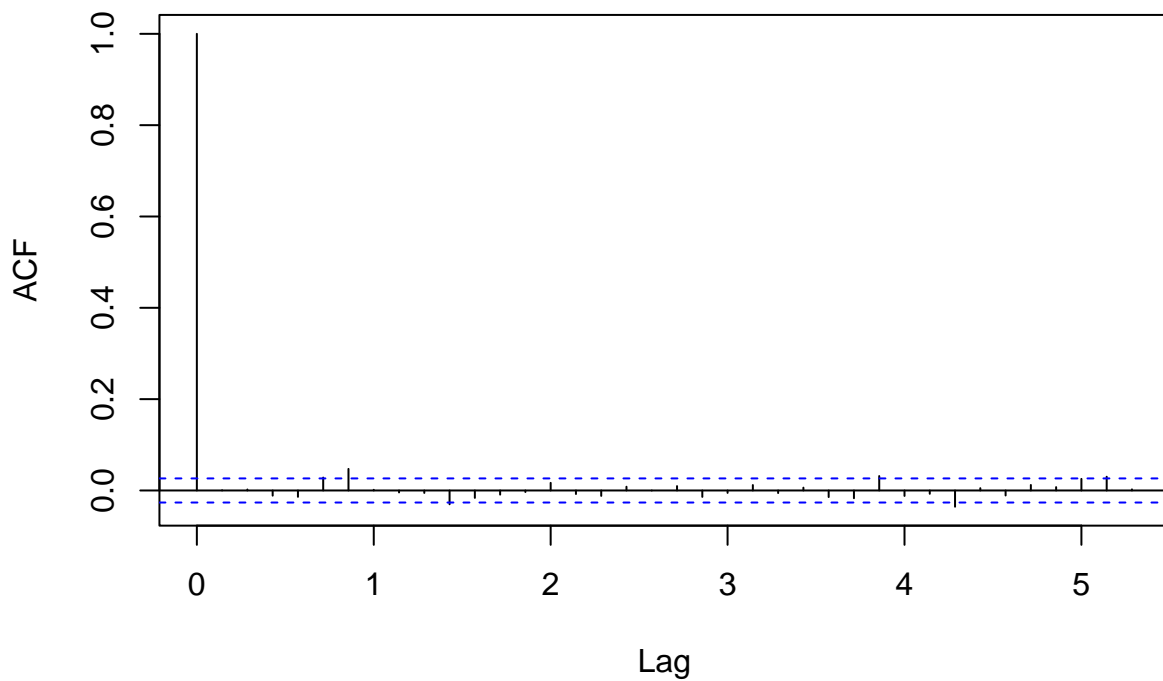
Parametry modelu jsou všechny statisticky významné, což potvrzuje správnost zvoleného modelu.

```
sarima_custom <- Arima(train_ts_daily,
                        order = c(3,1,4),
                        seasonal = list(order = c(1,1,1), period = 7))
summary(sarima_custom)
```

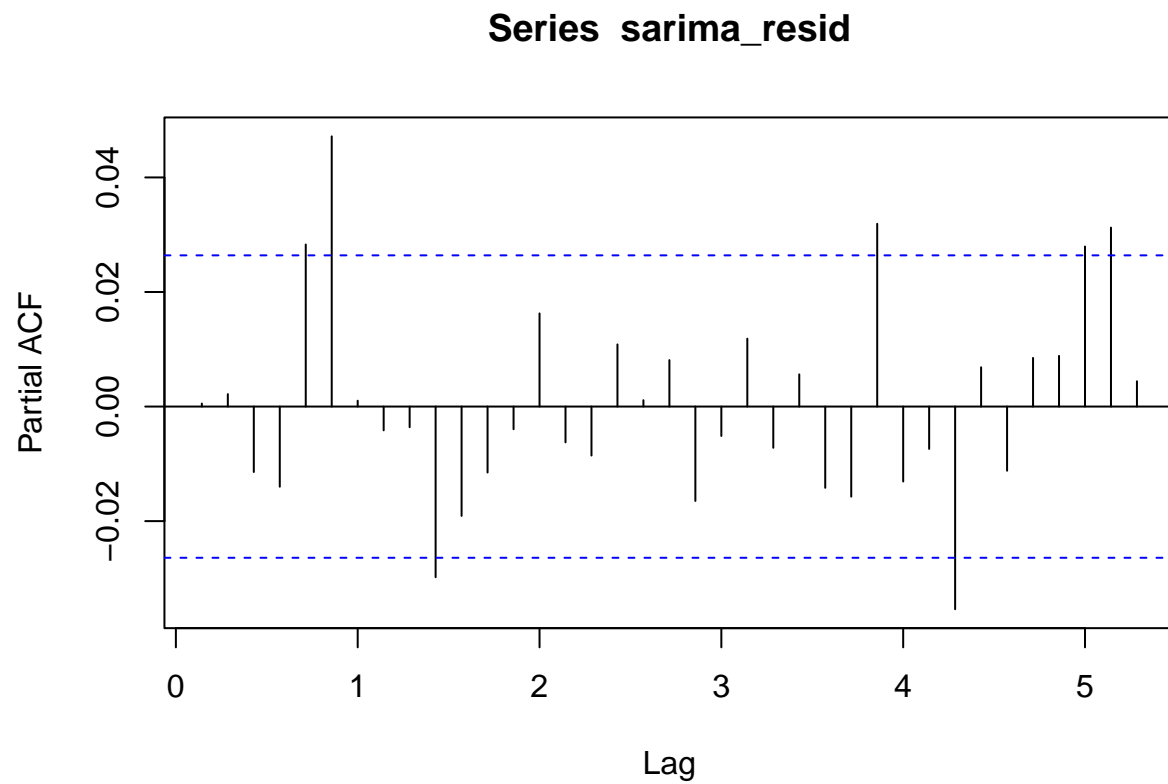
```
## Series: train_ts_daily
## ARIMA(3,1,4)(1,1,1)[7]
##
## Coefficients:
##          ar1      ar2      ar3      ma1      ma2      ma3      ma4      sar1
```

```
##      -0.5254  0.4964  0.4835  0.3610  -0.8083  -0.5582  0.1185  0.0834
## s.e.   0.0801  0.0517  0.0440  0.0805   0.0548   0.0585  0.0211  0.0197
##      sma1
##      -0.8028
## s.e.   0.0120
##
## sigma^2 = 1472818:  log likelihood = -46895.53
## AIC=93811.06  AICc=93811.1  BIC=93877.2
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -4.909917 1211.724 884.6798 -0.08192676 2.91846 0.5618507
##              ACF1
## Training set 0.000529748
sarima_resid <- residuals(sarima_custom)
# Korelogramy reziduů SARIMA modelu
acf(sarima_resid)
```

Series sarima_resid

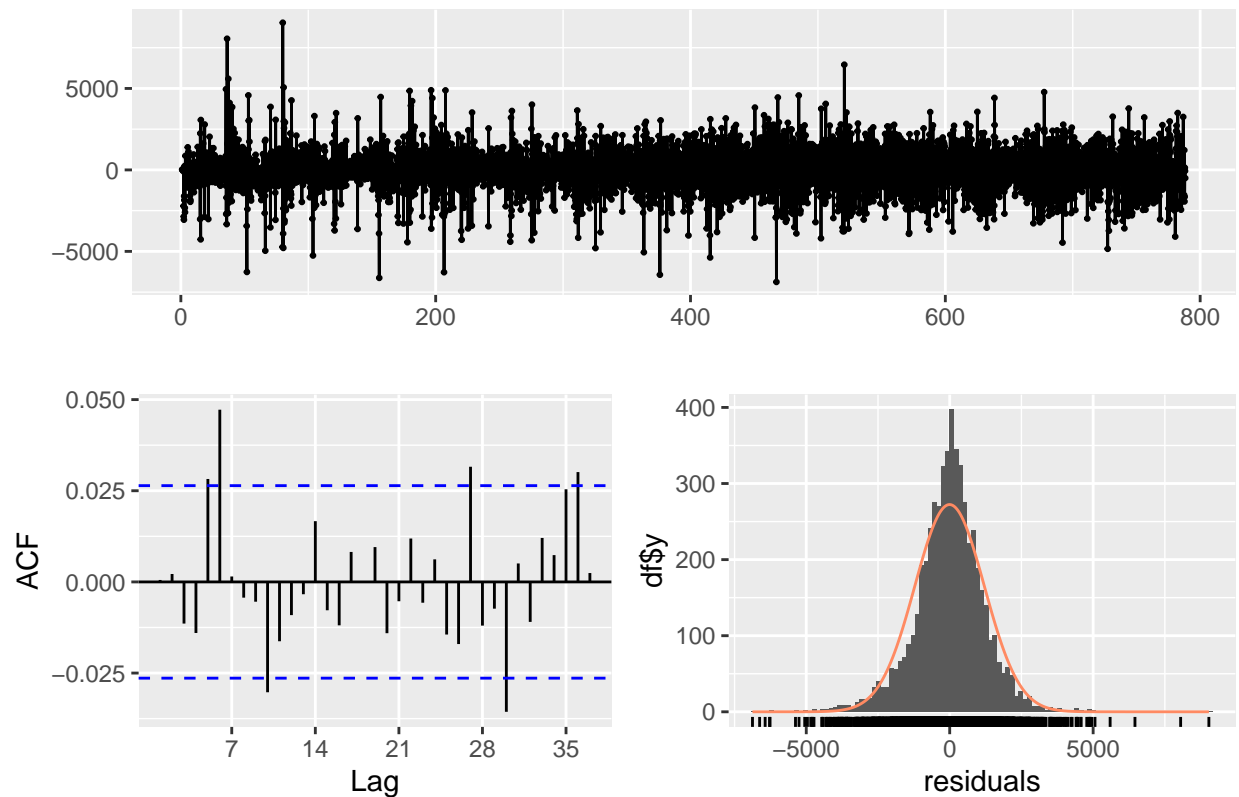


```
pacf(sarima_resid)
```



Korelogram reziduů naznačuje, že rezidua obsahuje významnou autokorelaci v náhodilých zpožděních.
`checkresiduals(sarima_resid)`

Residuals



```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 27.395, df = 14, p-value = 0.0171
##
## Model df: 0.   Total lags used: 14
```

Diagnostika reziduí SARIMA modelu

Rezidua vykazují významnou autokorelaci (Ljung-Box test $p\text{-value} = 0.0097$).

Test heteroskedasticity (ARCH LM-test) vykazuje silnou přítomnost ARCH efektů ($p\text{-value} \approx 0$).

Jarque-Bera test zamítl normalitu reziduí ($p\text{-value} \approx 0$).

Tyto výsledky naznačují přítomnost podmíněné heteroskedasticity a odůvodňují použití GJR-GARCH modelu.

```
Box.test(sarima_resid, lag=7, type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  sarima_resid
## X-squared = 18.548, df = 7, p-value = 0.009727
```



```
ArchTest(sarima_resid, lags=7)
```

```
##  
## ARCH LM-test; Null hypothesis: no ARCH effects  
##  
## data: sarima_resid  
## Chi-squared = 355.59, df = 7, p-value < 2.2e-16
```

```
jarque.bera.test(sarima_resid)
```

```
##  
## Jarque Bera Test  
##  
## data: sarima_resid  
## X-squared = 2153.6, df = 2, p-value < 2.2e-16
```

GJR-GARCH model na reziduích SARIMA:

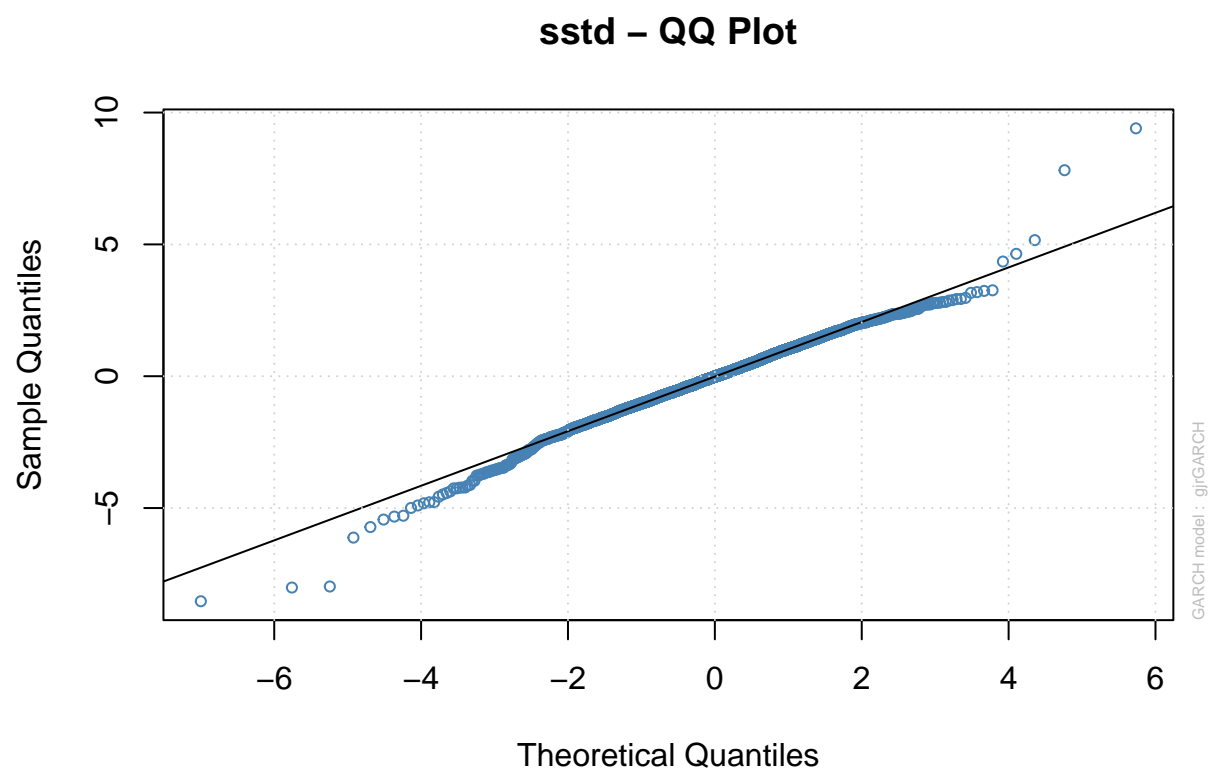
Model GJR-GARCH(1,1) efektivně zachycuje volatilitu reziduí, přičemž parametry α , β a γ jsou statisticky významné.

Zejména koeficient asymetrického efektu γ (1.1413) potvrzuje výraznou přítomnost asymetrického efektu ('leverage effect') v sérii reziduí.

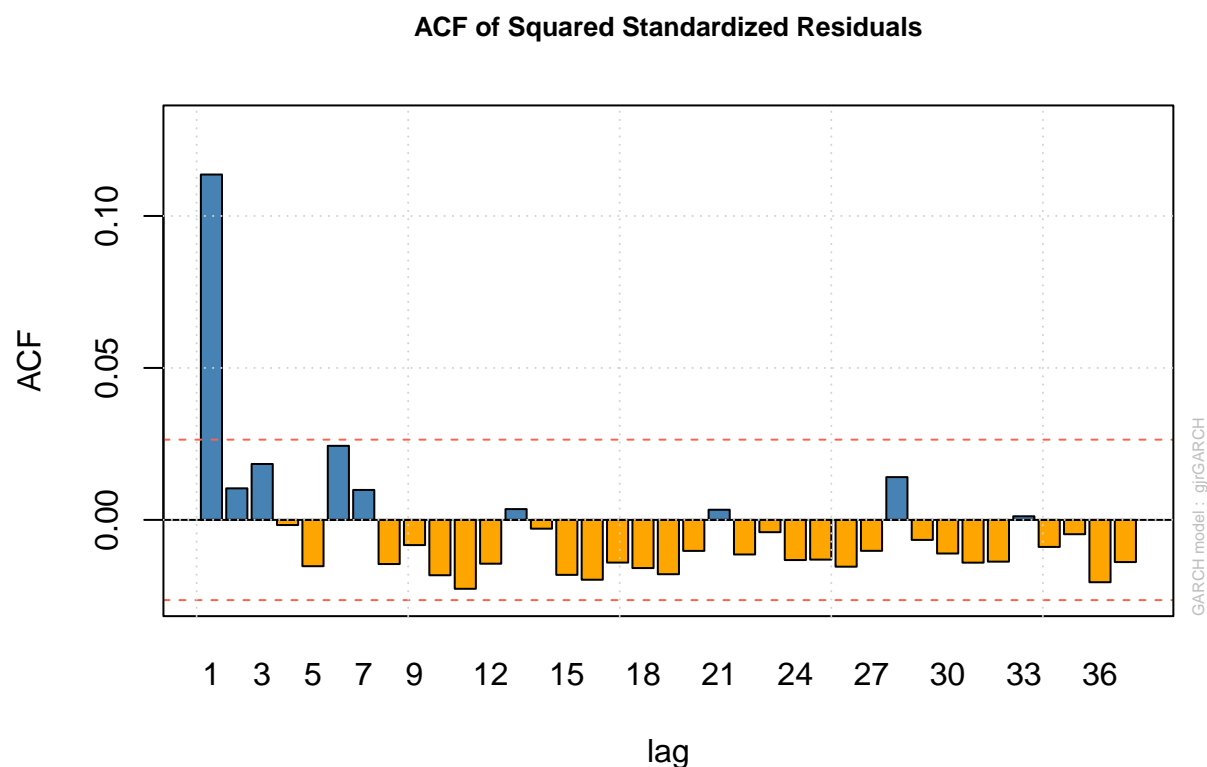
Výsledky Ljung-Box testů potvrzují, že standardizovaná rezidua stále obsahují určitou autokorelaci.

Přítomnost ARCH efektů je statisticky významná.

```
spec_sarima_garch <- ugarchspec(  
  variance.model = list(model="gjrGARCH", garchOrder=c(1,1)),  
  mean.model = list(armaOrder=c(3,4), include.mean=TRUE),  
  distribution.model="sstd"  
)  
  
fit_garch_sarima <- ugarchfit(spec_sarima_garch, sarima_resid, solver='hybrid')  
  
plot(fit_garch_sarima, which = 9) # QQ plot standardizovaných reziduí
```



```
plot(fit_garch_sarima, which = 11) # Standardizovaná ACF kvartátů reziduí
```



Druhý model poskytuje dobrou rovnováhu mezi složitostí a schopností modelovat strukturu volatility a dynamiku reziduí. I přes dílčí nedostatky (mírná autokorelace reziduí) vykazuje robustní schopnost zachytit základní charakteristiky časové řady.

Tento model je kvalitnější než Model 1 z hlediska chyby bodové predikce. Má o ≈ 10 nižší chybu odhadu, než porovnatelný additivní model SARIMA(2,0,2)(0,1,1)[7] + GJR-GARCH(1,1).

```
MSE <- mean((test_ts_daily - final_forecast_sarima_mean)^2)
MAE <- mean(abs(test_ts_daily - final_forecast_sarima_mean))
RMSE <- sqrt(MSE)

predikce_tab_sarima <- data.frame(
  MSE = round(MSE,2),
  MAE = round(MAE,2),
  RMSE = round(RMSE,2)
)

kable(predikce_tab_sarima, caption="Predikční chyby SARIMA(3,1,4)(1,1,1)[7] + GJR-GARCH(1,1)")
```

Table 2: Predikční chyby SARIMA(3,1,4)(1,1,1)[7] + GJR-GARCH(1,1)

MSE	MAE	RMSE
21431040	4102.72	4629.37

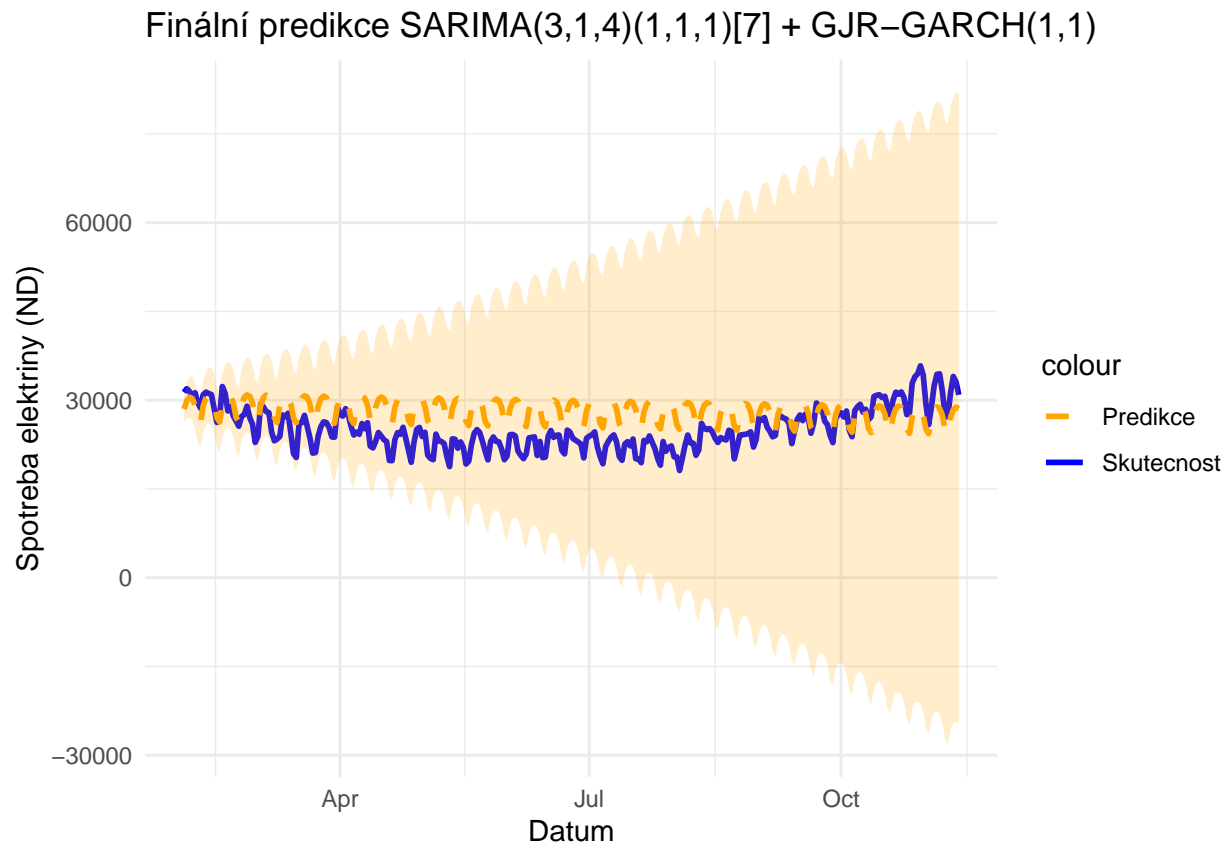
Grafická interpretace finální predikce naznačuje, že model zachytil hlavní trend a sezónní strukturu dat, nicméně intervalové predikce se směrem do budoucnosti znatelně rozšiřují. To může být způsobeno zvýšenou nejistotou dlouhodobějších predikcí typických pro heteroskedastické modely.

Model SARIMA(3,1,4)(1,1,1)[7] doplněný o GJR-GARCH(1,1) tedy představuje vhodnou volbu díky schopnosti modelovat podmíněnou volatilitu a sezónní dynamiku, ale zároveň vykazuje výraznou nejistotu v dlouhodobějších predikcích.

```
train_end_date <- as.Date("2009-01-01") + length(train_ts_daily) - 1
forecast_dates <- seq.Date(from = train_end_date + 1, by = "day", length.out = h)

pred_df <- data.frame(
  Datum = forecast_dates,
  Skutecnost = as.numeric(test_ts_daily),
  Predikce = as.numeric(final_forecast_sarima_mean),
  Lower = as.numeric(final_forecast_sarima_lower),
  Upper = as.numeric(final_forecast_sarima_upper)
)

ggplot(pred_df, aes(x = Datum)) +
  geom_line(aes(y = Skutecnost, color = "Skutečnost"), size = 1) +
  geom_line(aes(y = Predikce, color = "Predikce"), size = 1, linetype = "dashed") +
  geom_ribbon(aes(ymin = Lower, ymax = Upper), alpha = 0.2, fill = "orange") +
  labs(title = "Finální predikce SARIMA(3,1,4)(1,1,1)[7] + GJR-GARCH(1,1)",
        y = "Spotřeba elektřiny (ND)", x = "Datum") +
  scale_color_manual(values = c("Skutečnost" = "blue", "Predikce" = "orange")) +
  theme_minimal()
```



Model 3 ARFIMA(2,0.5,2) s GJR-GARCH(1,1):

Specifikace modelu

Pro modelování dlouhé paměti časové řady spotřeby elektřiny byl použit model ARFIMA, který umožňuje odhadnout stupeň dlouhé paměti dat. Výsledná hodnota parametru frakční diferenciace $d = 0.49959$ značí výraznou dlouhou paměť v časové řadě.

```
# ARFIMA model
arfima_model <- fracdiff(train_ts_daily, ar=2, ma=2, h=0.0015)

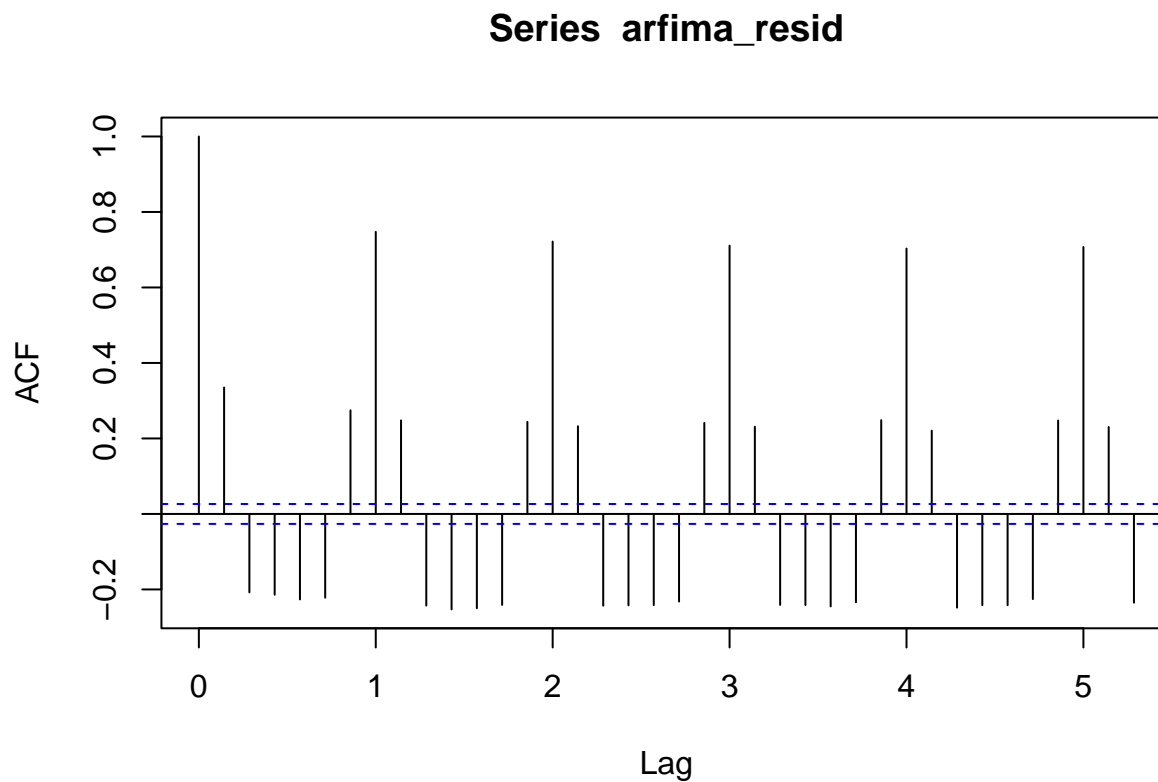
summary(arfima_model)

##
## Call:
##   fracdiff(x = train_ts_daily, ar = 2, ma = 2, h = 0.0015)
##
## Coefficients:
##   Estimate Std. Error z value Pr(>|z|)
## d  0.49959    0.02339   21.36  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## sigma[eps] = 2259.224
## [d.tol = 0.0001221, M = 100, h = 0.0015]
## Log likelihood: -5.039e+04 ==> AIC = 100789 [2 deg.freedom]
```

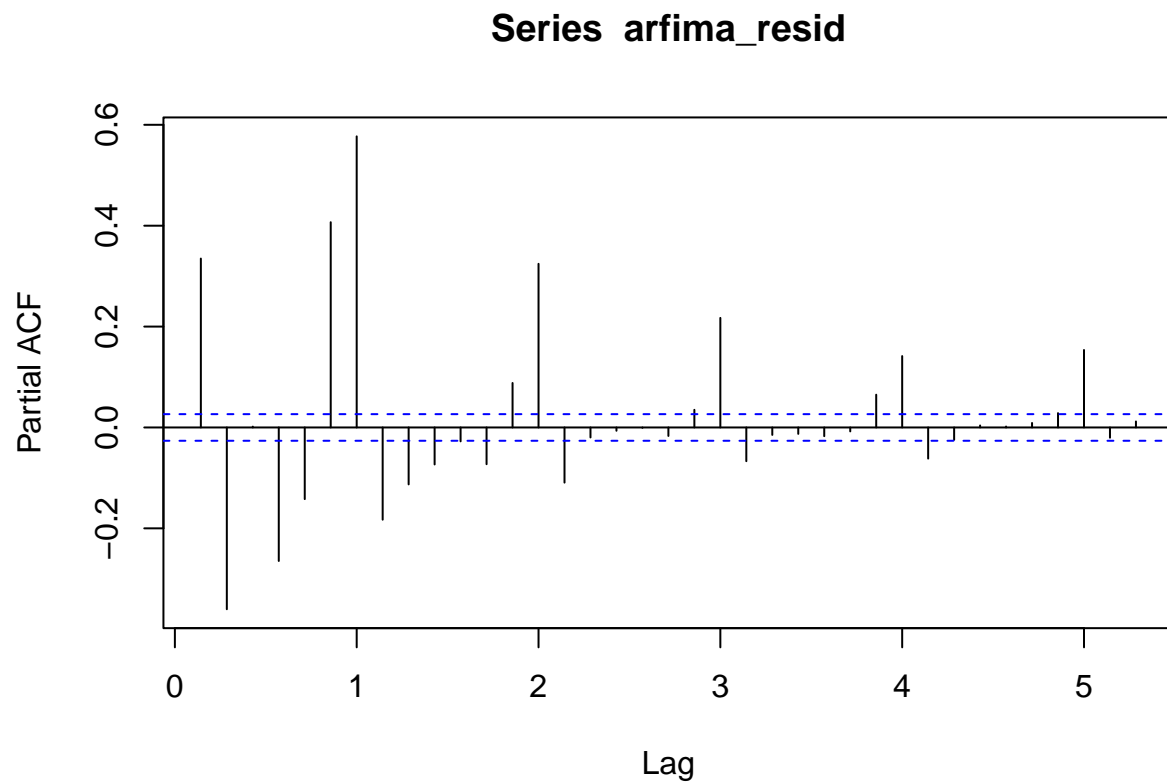
```
arfima_resid <- residuals(arfima_model)
```

Můžeme zde pozorovat výraznou cyklickou autokorelační strukturu. Bude složité odstranit tento vliv jen pomocí GARCH efektu, potřebovali bychom pomot ideálně provést sezónní otištění, ale mělo by toto za důsledek modelování trendu, nýbrž sezónnosti.

```
# Korelogramy reziduí ARIMA modelu  
acf(arfima_resid)
```



```
pacf(arfima_resid)
```



Diagnostika reziduí ARFIMA

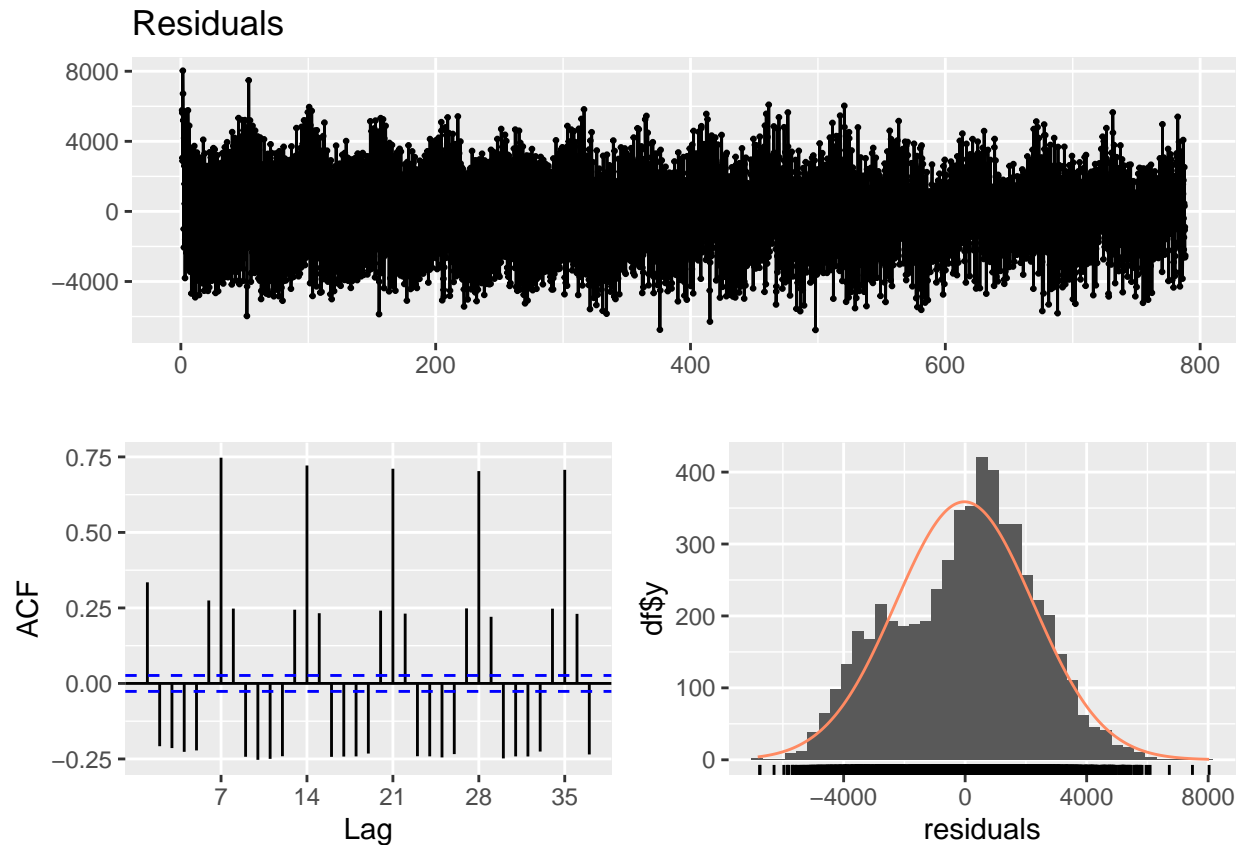
Diagnostika reziduí ukazuje silnou autokorelaci a heteroskedasticitu:

Ljung-Boxův test výrazně odmítá nulovou hypotézu o neautokorelovaných reziduích (p-value < 0.01).

ARCH LM test indikuje výraznou přítomnost ARCH efektů, tedy nestálosti v rozptylu.

Jarque-Bera test ukazuje odchylku od normality reziduí (p-value < 0.01).

```
checkresiduals(arfima_resid)
```



```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 10055, df = 14, p-value < 2.2e-16
##
## Model df: 0.   Total lags used: 14
Box.test(arfima_resid, lag=7, type="Ljung-Box")

##
##  Box-Ljung test
##
## data:  arfima_resid
## X-squared = 5164, df = 7, p-value < 2.2e-16
ArchTest(arfima_resid, lags=7)

##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  arfima_resid
## Chi-squared = 1581.8, df = 7, p-value < 2.2e-16
jarque.bera.test(arfima_resid)

##
##  Jarque Bera Test
```



```
##
## data:  arfima_resid
## X-squared = 80.958, df = 2, p-value < 2.2e-16
```

GJR-GARCH(1,1) model volatility

K zachycení podmíněné volatility reziduí byl zvolen GJR-GARCH(1,1) model s ARMA(2,2) v průměrové rovnici a s asymetrickou distribucí reziduí (sstd). Všechny parametry modelu byly statisticky významné. Z modelu vyplynulo:

Velmi vysoká persistence volatility (parametr $\beta_1 = 0.9949$).

Statisticky významný leverage efekt (parametr $\gamma_1 < 0$), který značí rozdílnou reakci volatility na pozitivní a negativní šoky

```
spec_arfima_garch <- ugarchspec(
  variance.model = list(model="gjrGARCH", garchOrder=c(1,1)),
  mean.model = list(armaOrder=c(2,2), include.mean=TRUE),
  distribution.model="sstd"
)

fit_garch_arfima <- ugarchfit(spec_arfima_garch, arfima_resid, solver='hybrid')
```

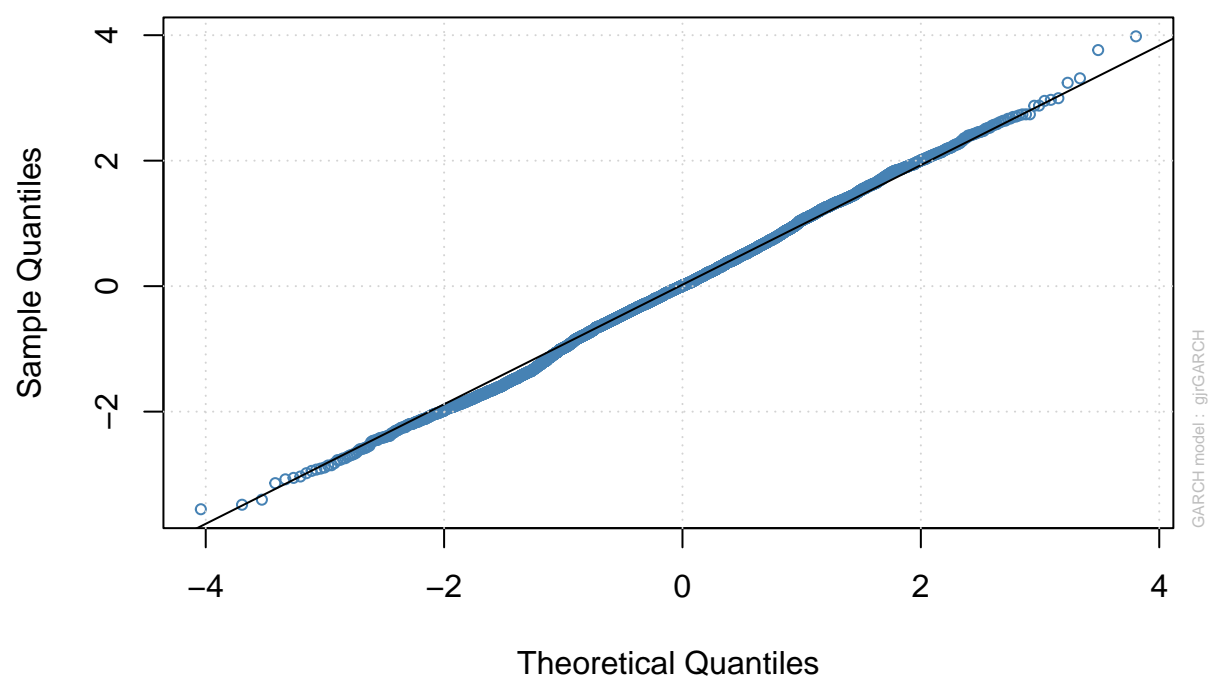
Diagnostika reziduí GJR-GARCH

Další diagnostika modelu potvrdila přetrvávající autokorelaci v standardizovaných reziduích a jejich druhých mocninách. Tato skutečnost ukazuje, že i přes použití sofistikovaného modelu zůstává v řadě určitá nevysvětlená struktura.

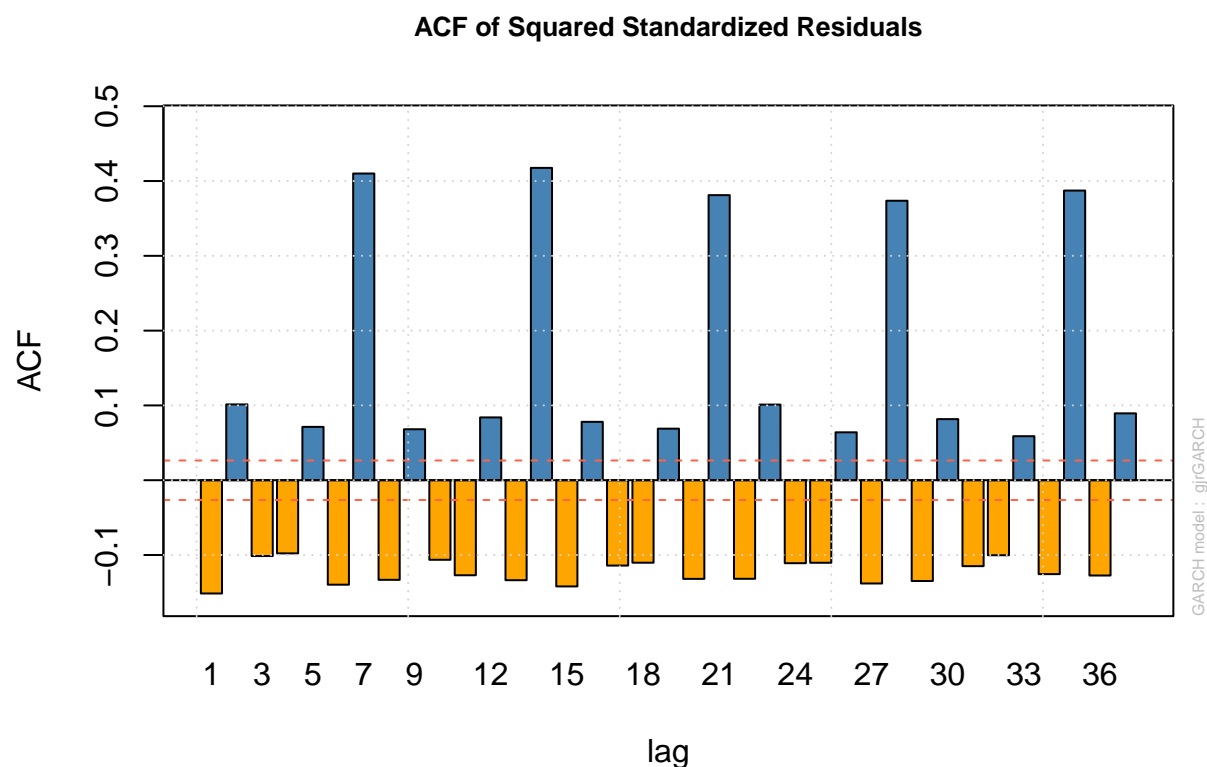
Q-Q plot standardizovaných reziduí ukazuje relativně dobré přizpůsobení zvolené distribuce.

```
plot(fit_garch_arfima, which = 9) # QQ plot standardizovaných reziduí
```

sstd – QQ Plot



```
plot(fit_garch_arfima, which = 11) # Standardizovaná ACF kvartátů reziduí
```



Predikční výkon modelu ARFIMA s GJR-GARCH byl nejméně uspokojivý ze všech testovaných modelů:

```
MSE <- mean((test_ts_daily - final_forecast_arfima_mean)^2)
MAE <- mean(abs(test_ts_daily - final_forecast_arfima_mean))
RMSE <- sqrt(MSE)

predikce_tab_arfima <- data.frame(
  MSE = round(MSE,2),
  MAE = round(MAE,2),
  RMSE = round(RMSE,2)
)

kable(predikce_tab_arfima, caption="Predikční chyby ARFIMA + GJR-GARCH")
```

Table 3: Predikční chyby ARFIMA + GJR-GARCH

MSE	MAE	RMSE
24926668	4328.76	4992.66

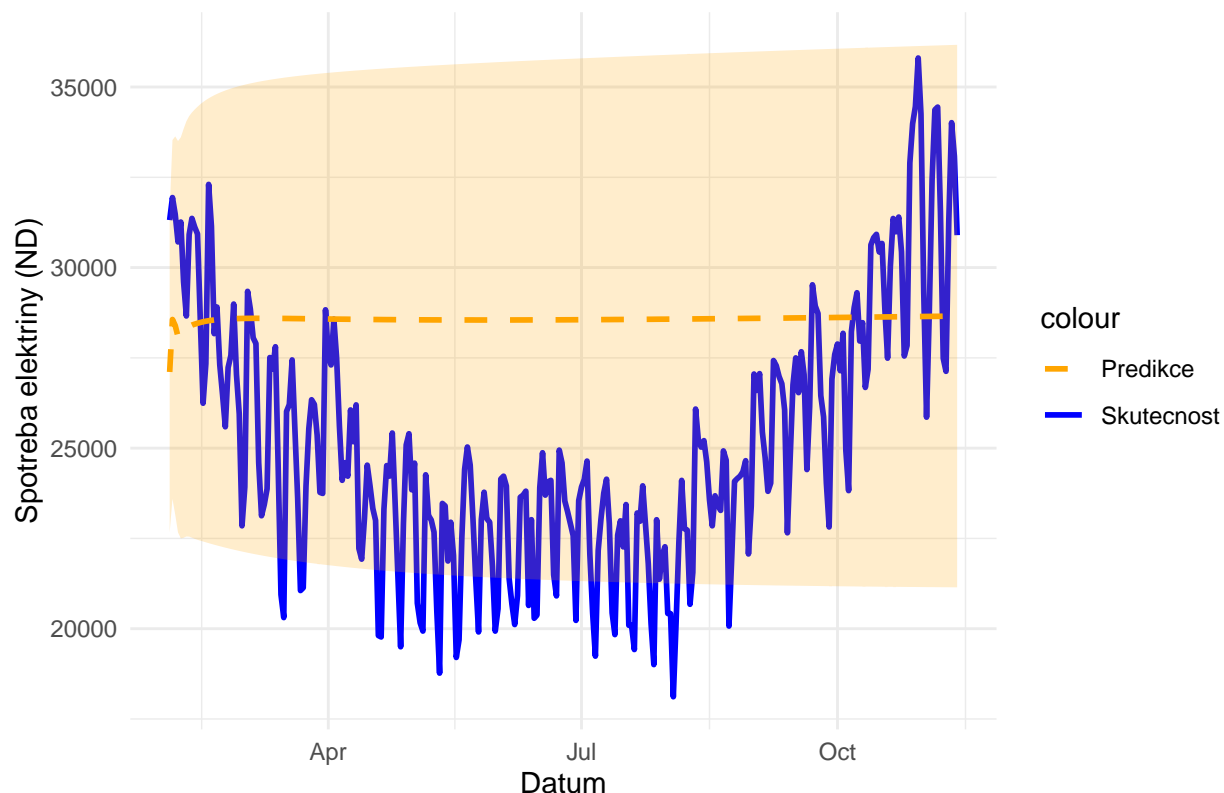
Predikce vykazuje vysokou nejistotu, což je patrné i z rozsahu intervalů spolehlivosti. Grafický vývoj ukazuje, že model systematicky nadhodnocuje predikce oproti skutečným hodnotám spotřeby elektřiny.

```
train_end_date <- as.Date("2009-01-01") + length(train_ts_daily) - 1
forecast_dates <- seq.Date(from = train_end_date + 1, by = "day", length.out = h)

pred_df <- data.frame(
  Datum = forecast_dates,
  Skutecnost = as.numeric(test_ts_daily),
  Predikce = as.numeric(final_forecast_arfima_mean),
  Lower = as.numeric(final_forecast_arfima_lower),
  Upper = as.numeric(final_forecast_arfima_upper)
)

ggplot(pred_df, aes(x = Datum)) +
  geom_line(aes(y = Skutecnost, color = "Skutečnost"), size = 1) +
  geom_line(aes(y = Predikce, color = "Predikce"), size = 1, linetype = "dashed") +
  geom_ribbon(aes(ymin = Lower, ymax = Upper), alpha = 0.2, fill = "orange") +
  labs(title = "Finální predikce ARFIMA(2,0.5,2) + GJR-GARCH(1,1) (včetně intervalů)",
    y = "Spotřeba elektřiny (ND)", x = "Datum") +
  scale_color_manual(values = c("Skutečnost" = "blue", "Predikce" = "orange")) +
  theme_minimal()
```

Finální predikce ARFIMA(2,0.5,2) + GJR-GARCH(1,1) (včetně intervalu)



Závěr k modelu ARFIMA + GJR-GARCH

Model ARFIMA(2,0.5,2) kombinovaný s GJR-GARCH(1,1) nedokázal přesvědčivě zachytit strukturu spotřeby elektřiny a volatility. I přes teoretickou atraktivitu modelu s dlouhou pamětí byla jeho predikční schopnost slabší než u modelů SARIMA. Výsledky naznačují, že je třeba upřednostnit jiné typy modelů, které lépe vystihnou sezonní strukturu a variabilitu dat.

4. Manažerské shrnutí analýzy časové řady spotřeby elektřiny:

V této analýze byly testovány tři různé modely na časové řadě spotřeby elektřiny ve Velké Británii. Srovnávány byly modely SARIMA(2,0,2)(0,1,1)[7] s GJR-GARCH(1,1), SARIMA(3,1,4)(1,1,1)[7] s GJR-GARCH(1,1) a ARFIMA(2,0.5,2) s GJR-GARCH(1,1). Hodnocení bylo založeno na splnění předpokladů, diagnostice reziduí a predikční kvalitě (MSE, MAE, RMSE).

Níže uvedená tabulka shrnuje výsledky jednotlivých modelů a jejich hodnocení:

```
summary_table <- data.frame(  
  Model = c("SARIMA(2,0,2)(0,1,1)[7]",  
            "SARIMA(3,1,4)(1,1,1)[7]",  
            "ARFIMA(2,0.5,2)"),  
  Naplnění_předpokladů = c("ANO, s výhradami (heteroskedasticita)",  
                           "ANO, s mírnými výhradami (slabá autokorelace)",  
                           "NE (silná autokorelace reziduí a ARCH efekty)"),  
  RMSE = c(5222, 4629, 4993),  
  Zdůvodnění = c("Vyšší predikční chyba a přetrvávající heteroskedasticita",  
                  "Nejnižší predikční chyba a dobré splnění předpokladů",  
                  "Nejhorší predikce a silné porušení předpokladů"),  
  Posouzení_rizik = c("Vysoká nejistota, nevhodné pro krátkodobé rozhodování",  
                      "Přijatelná přesnost, vhodné pro krátkodobé plánování",  
                      "Vysoké riziko, značná nejistota predikcí")  
)  
  
kable(summary_table, caption = "Manažerské shrnutí a hodnocení modelů")
```

Table 4: Manažerské shrnutí a hodnocení modelů

Model	Naplnění_předpokladů	RMSE	Zdůvodnění	Posouzení_rizik
SARIMA(2,0,2)(0,1,1)[7]	ANO, s výhradami (heteroskedasticita)	5222	Vyšší predikční chyba a přetrvávající heteroskedasticita	Vysoká nejistota, nevhodné pro krátkodobé rozhodování
SARIMA(3,1,4)(1,1,1)[7]	ANO, s mírnými výhradami (slabá autokorelace)	4629	Nejnižší predikční chyba a dobré splnění předpokladů	Přijatelná přesnost, vhodné pro krátkodobé plánování
ARFIMA(2,0.5,2)	NE (silná autokorelace reziduí a ARCH efekty)	4993	Nejhorší predikce a silné porušení předpokladů	Vysoké riziko, značná nejistota predikcí

Ekonomická a finanční interpretace výsledků:

Nejlepším modelem pro predikci spotřeby elektřiny se ukázal model SARIMA(3,1,4)(1,1,1)[7] s GJR-GARCH(1,1), který vykazoval nejnižší hodnoty chybových metrik (MSE, MAE, RMSE). Tento model dobře zachytil sezónní a volatilní charakter dat a poskytl rozumně přesné predikce, což ho činí vhodným nástrojem pro plánování v energetice a finanční řízení rizik spojených s nákupem elektřiny.

Model SARIMA(2,0,2)(0,1,1)[7] s GJR-GARCH(1,1) ukázal vyšší nejistotu predikcí a přetrvávající nestabilitu rozptylu, což jej činí méně vhodným pro krátkodobé ekonomické rozhodování, kde je klíčová přesnost a spolehlivost.

ARFIMA model vykazoval značnou dlouhou paměť, ale jeho praktická využitelnost byla výrazně omezena silnou autokorelací reziduí a vysokými chybami v predikcích. Proto není doporučen pro řízení energetických nákupů ani pro plánování produkce elektřiny.

5. Závěr

Analýza prokázala, že model SARIMA(3,1,4)(1,1,1)[7] kombinovaný s GJR-GARCH(1,1) poskytuje nejspolehlivější výsledky s ohledem na predikční schopnosti a splnění klíčových statistických předpokladů. Tento model efektivně zachytil jak sezónní vlivy, tak dynamické změny volatility v časové řadě spotřeby elektřiny. To umožňuje přesnější odhady budoucí spotřeby, což je klíčové zejména v odvětví energetiky, kde přesnost předpovědí přímo ovlivňuje efektivitu alokace zdrojů a snižování nákladů.

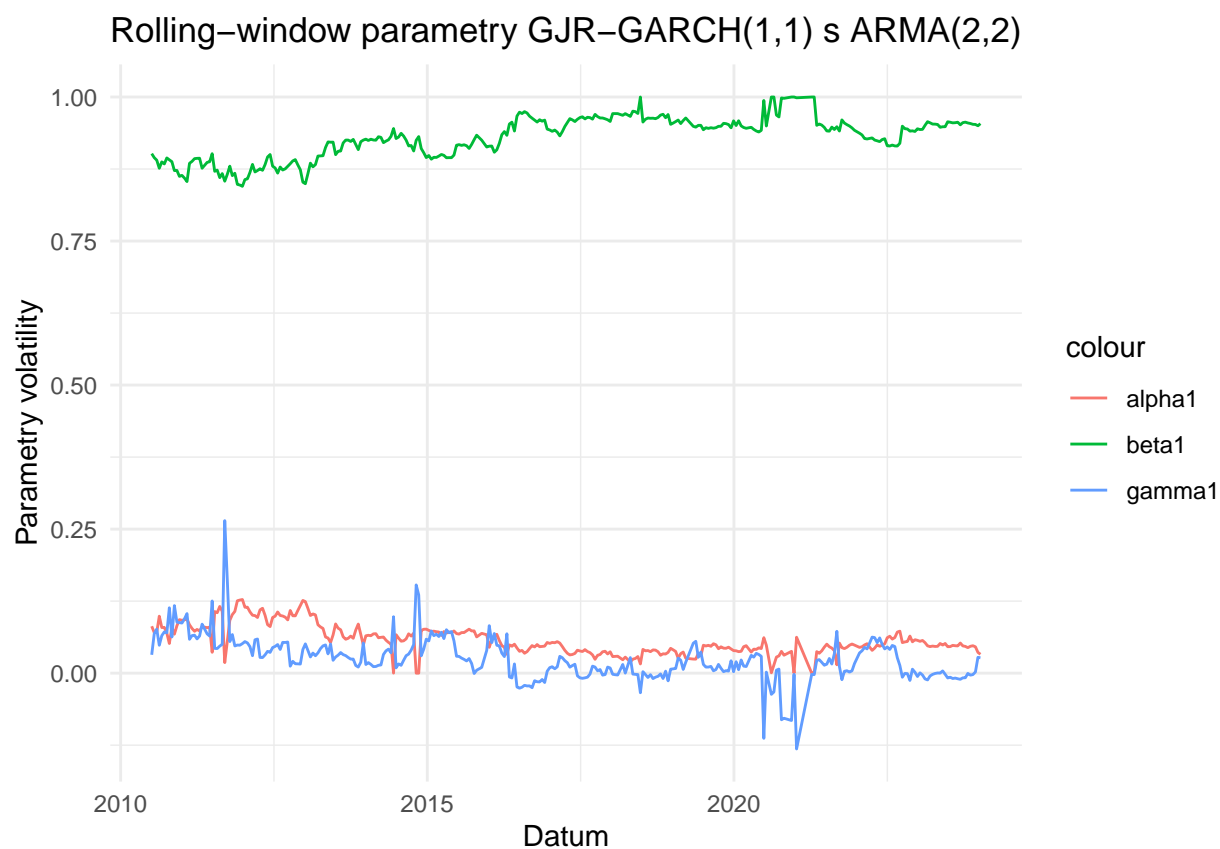
Z pohledu řízení rizik a finanční stability, je přesná předpověď spotřeby elektřiny důležitá pro minimalizaci neočekávaných nákladů souvisejících s nákupem energie na krátkodobých trzích. Model SARIMA(3,1,4)(1,1,1)[7] s GJR-GARCH(1,1) dokáže lépe předpovídat volatilitu trhu a připravit se na možné výkyvy v poptávce, což přináší ekonomickou výhodu ve formě snížených nákladů a lepšího plánování.

Naopak, použití alternativních modelů, jako jsou SARIMA(2,0,2)(0,1,1)[7] a ARFIMA(2,0.5,2), bylo spojeno s vyššími predikčními chybami a problematickými diagnostickými výsledky reziduí. To znamená, že spoléhání na tyto modely by vedlo k méně přesným predikcím, vyšší nejistotě a potenciálním finančním rizikům.

Proto jednoznačně doporučujeme pro praxi využívat model SARIMA(3,1,4)(1,1,1)[7] kombinovaný s modelem GJR-GARCH(1,1), neboť přináší robustní a ekonomicky racionální řešení pro krátkodobé plánování spotřeby elektřiny, efektivní řízení nákladů a snížení finančních rizik.

6. Příloha

```
# Kontrola výsledků
if(nrow(results_1) > 0){
  library(ggplot2)
  ggplot(results_1, aes(x=date)) +
    geom_line(aes(y=alpha1, color="alpha1")) +
    geom_line(aes(y=beta1, color="beta1")) +
    geom_line(aes(y=gamma1, color="gamma1")) +
    labs(title="Rolling-window parametry GJR-GARCH(1,1) s ARMA(2,2)",
         y="Parametry volatility",
         x="Datum") +
    theme_minimal()
}else{
  print("Model nekonvergoval v žádném rolling-window okně.")
}
```



Na grafu vidíme časový vývoj parametrů modelu volatility (GJR-GARCH):

1. α_1 (červená) - reakce volatility na nové informace (krátkodobé šoky),
2. β_1 (zelená) - přetrvávání volatility (dlouhodobá paměť),
3. γ_1 (modrá) - asymetrie volatility (citlivost na negativní šoky).

Ekonomická interpretace grafu:

β_1 vykazuje vysokou stabilitu blízkou hodnotě 1 během celého období. To naznačuje výraznou persistenci volatility v analyzované časové řadě, tedy že vliv minulých šoků na volatilitu je dlouhodobý a stabilní v čase.

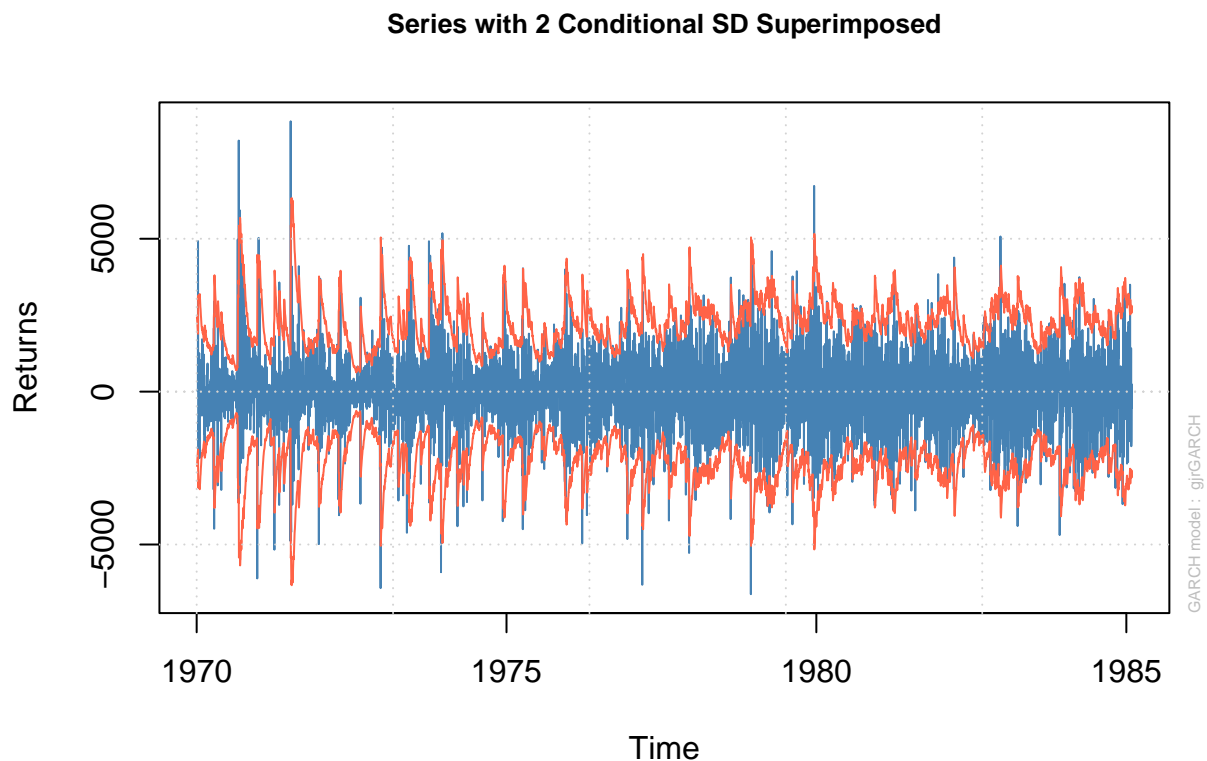
α_1 , reprezentující okamžitou reakci volatility na nové šoky, je velmi nízká a stabilní v průběhu času, což potvrzuje, že krátkodobé šoky mají relativně malý, ale konzistentní vliv na volatilitu.

γ_1 , která zachycuje asymetrické efekty (tzv. leverage efekt), kolísá kolem nuly, což znamená, že asymetrický efekt negativních a pozitivních šoků na volatilitu je během celého období minimální až zanedbatelný.

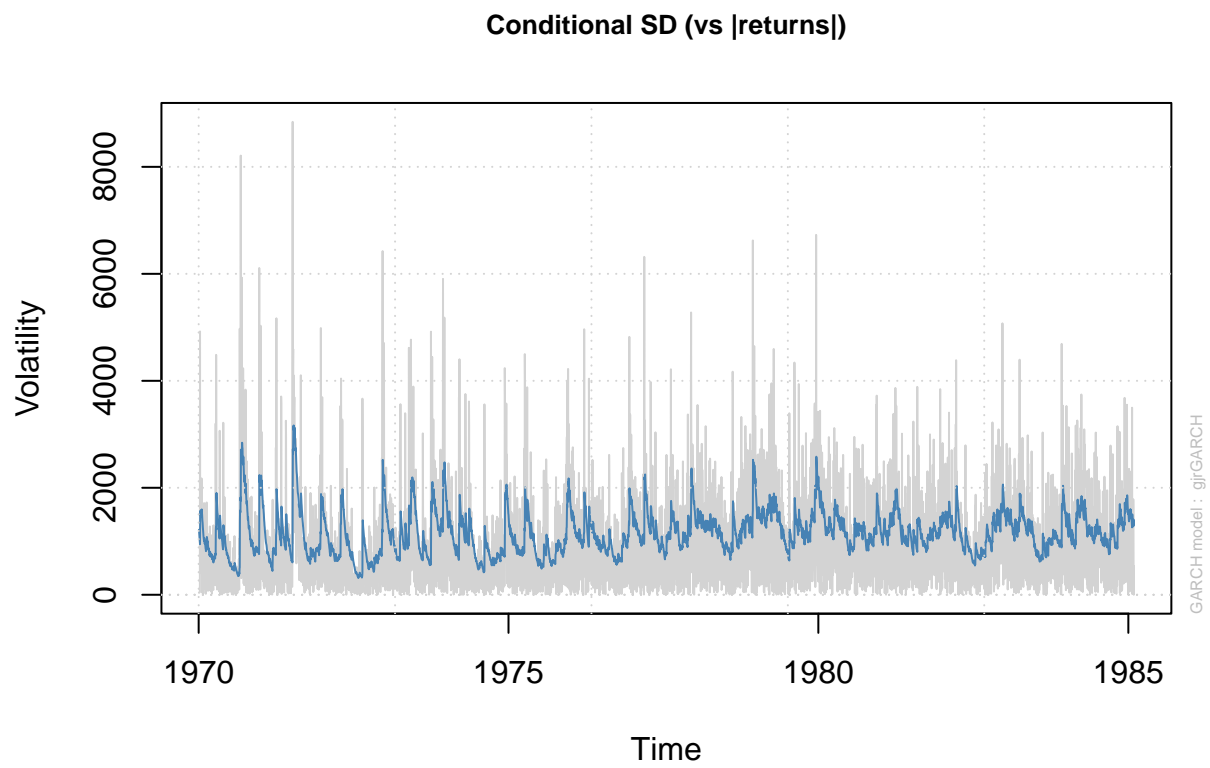
Závěrem lze říci, že rolling-window analýza potvrdila, že parametry volatility modelu GJR-GARCH(1,1) jsou velmi stabilní v čase. Vysoká stabilita parametru β_1 zdůrazňuje dlouhodobou setrvačnost volatility, což má praktické důsledky například při řízení rizik v ekonomickém a finančním kontextu, kde lze očekávat konzistentní chování volatility v budoucnosti.

Dodatečné grafy pro GARCH v Modelu 1:

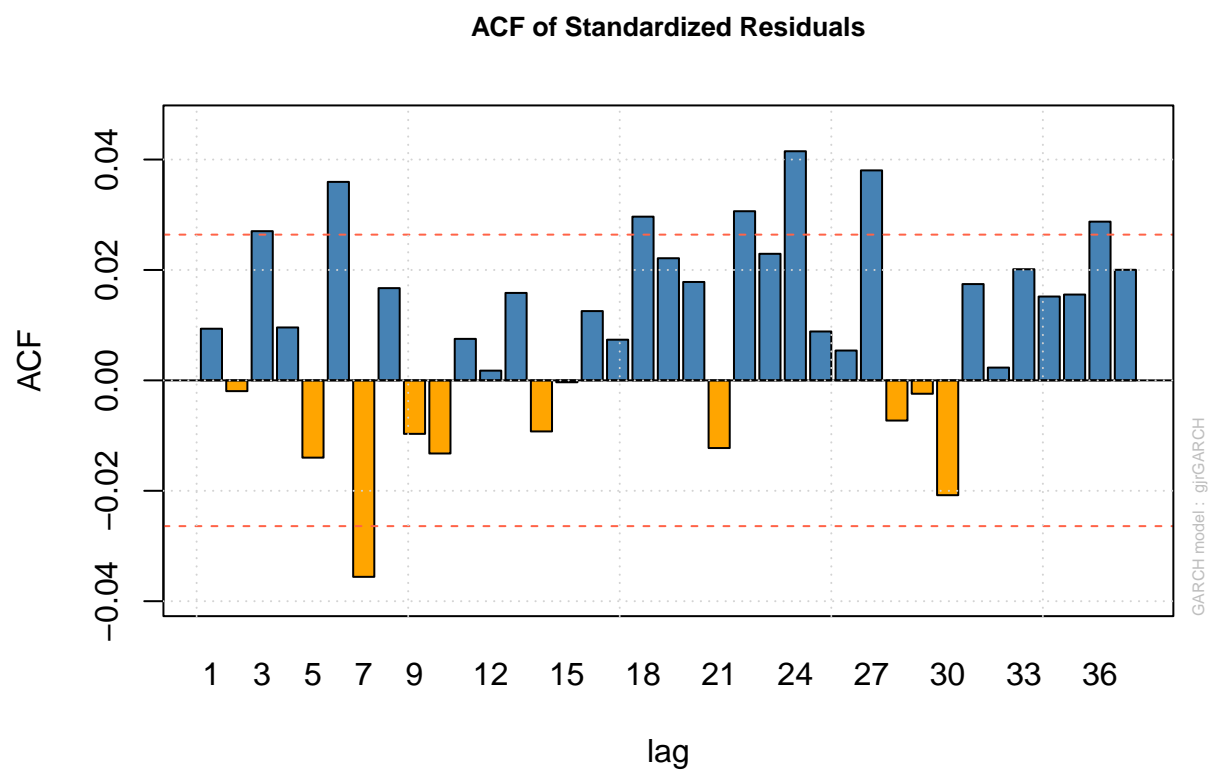
```
plot(fit_garch_updated, which=1) # Podmíněná volatilita
```




```
plot(fit_garch_updated, which=3)
```



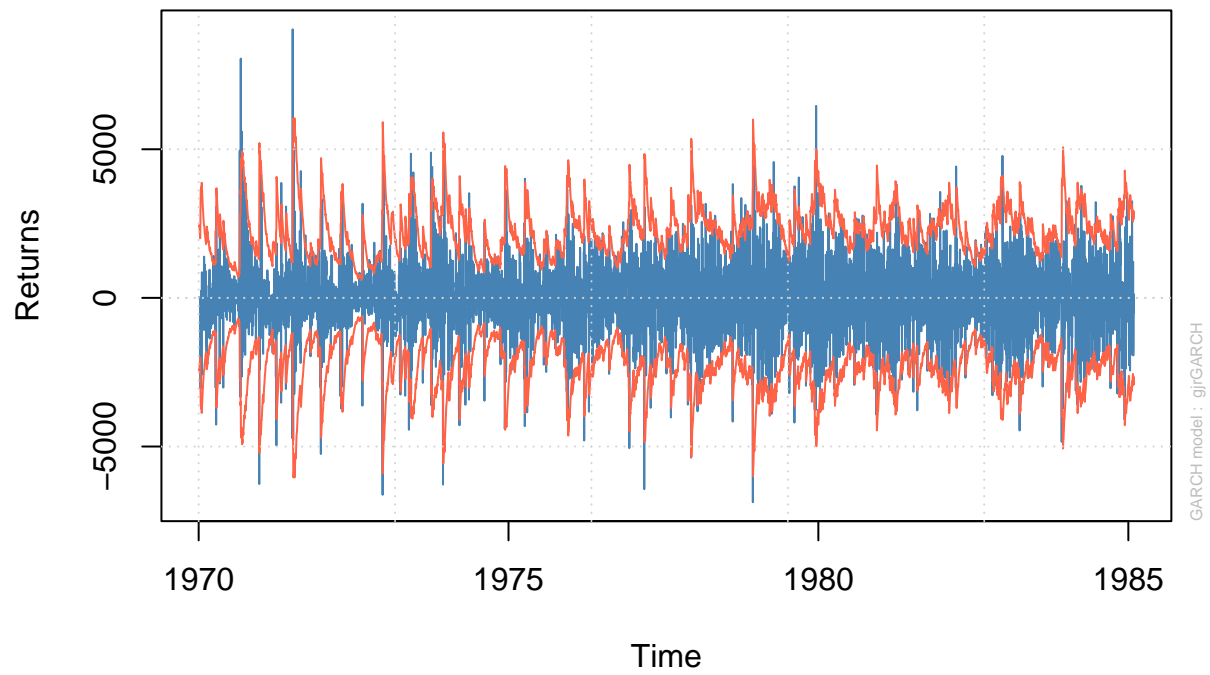
```
plot(fit_garch_updated, which = 10)
```



Dodatečné grafy pro GARCH v Modelu 2:

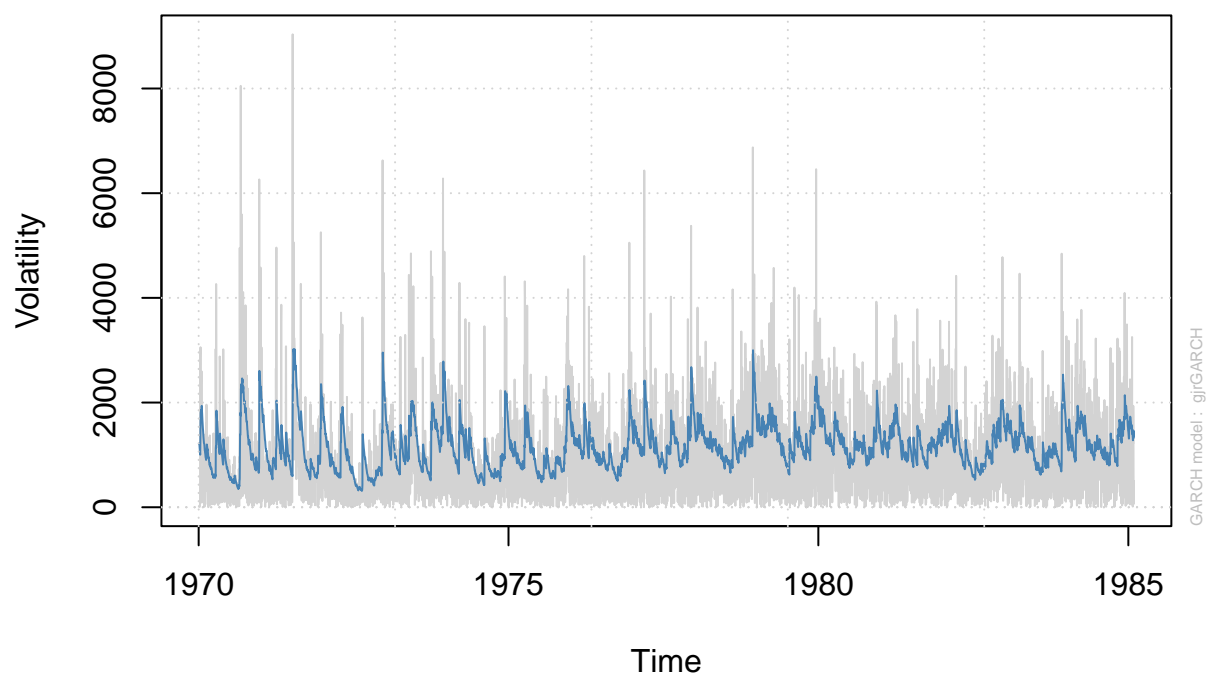
```
plot(fit_garch_sarima, which=1) # Podmíněná volatilita
```

Series with 2 Conditional SD Superimposed

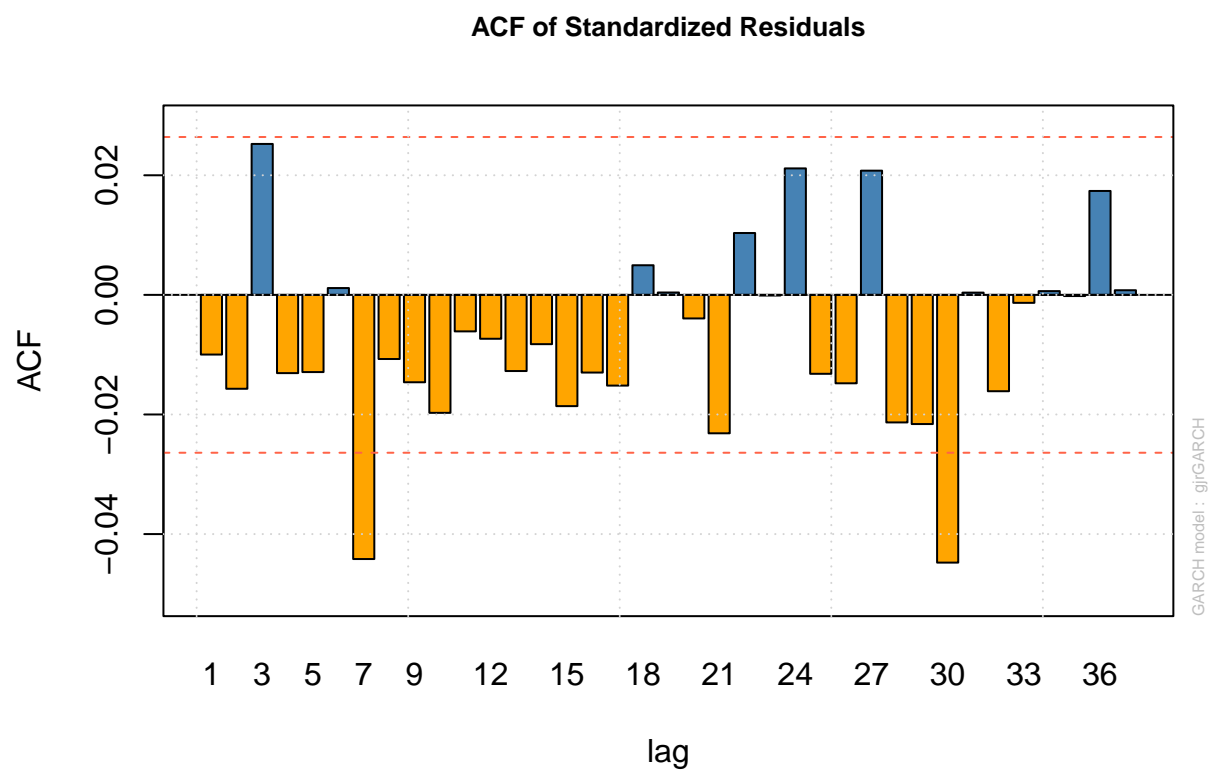


```
plot(fit_garch_sarima, which=3)
```

Conditional SD (vs |returns|)



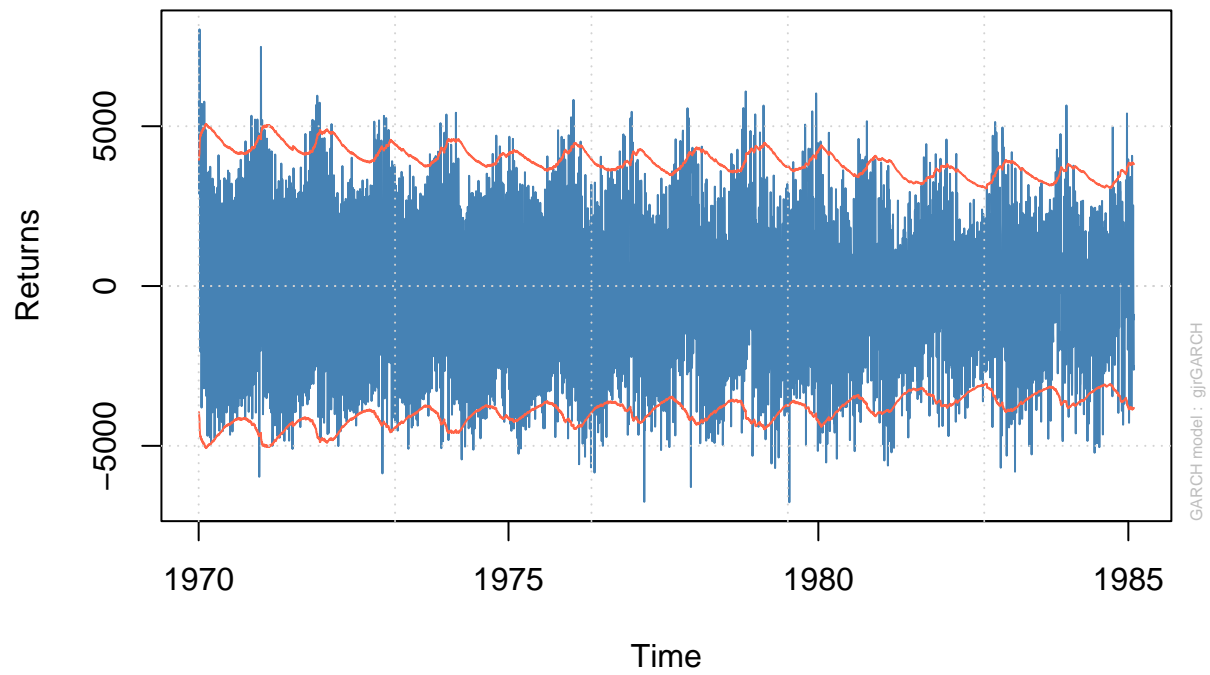
```
plot(fit_garch_sarima, which = 10) # Standardizovaná ACF reziduí
```



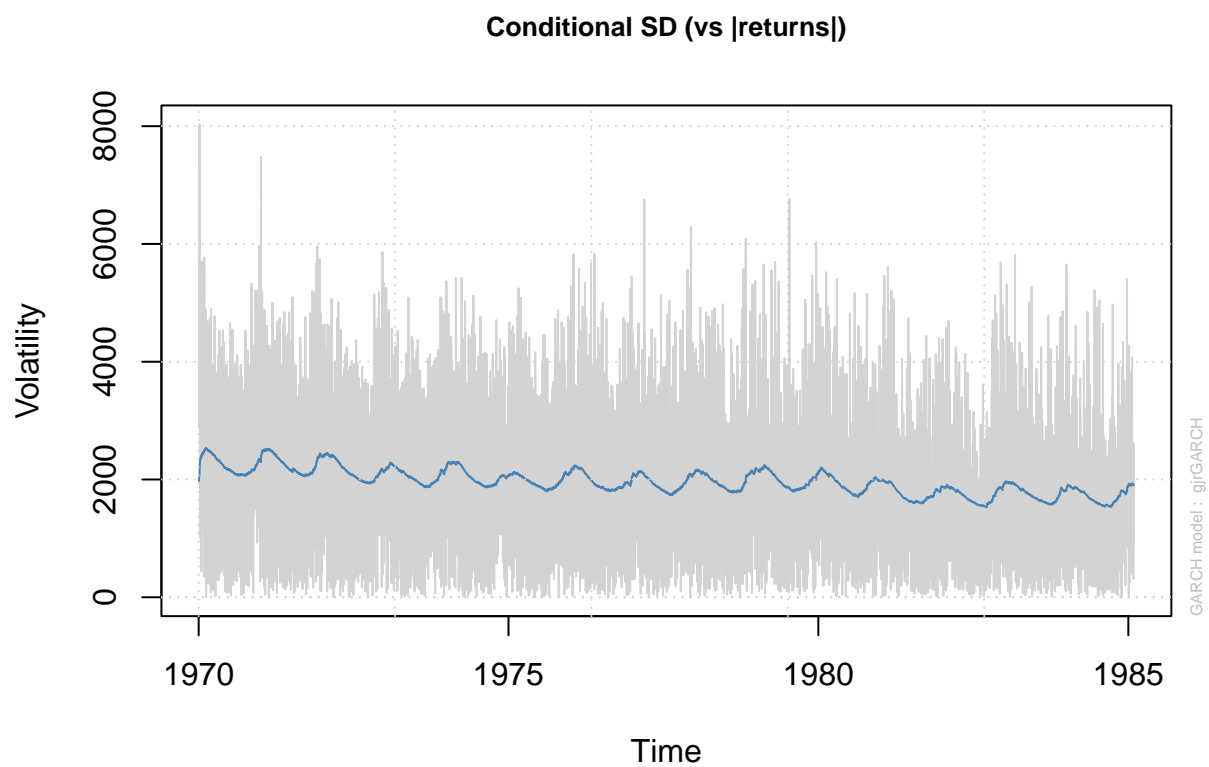
Dodatečné grafy pro GARCH v Modelu 3:

```
plot(fit_garch_arfima, which=1) # Podmíněná volatilita
```

Series with 2 Conditional SD Superimposed



```
plot(fit_garch_arfima, which=3)
```



```
plot(fit_garch_arfima, which = 10) # Standardizovaná ACF reziduí
```

