Modelovanie časových radov z monitorovacieho systému Hawkular

Bc. Pavol Loffay[1]

25. novembra 2015

Abstrakt: Práca spracováva predikciu časových radov prevziatych z monitorovacieho systému Hawkular[a]. Tento systém dokáže monitorovať Java aplikácie, alebo fyzický hardvér na ktom je spustený. Z množiny pozorovaných metrík boli vybraté následujúce: vyťaženie Java hromady (heap), miesto na disku a počet voľných databázových spojení. Tieto časové rady som analyzoval a cieľom bolo zostaviť model, ktorý nejlepšie popisoval priebeh danej časovej rady. V práci som postupoval podľa Box – Jenkinsovej metodológie.

Kľúčové slová: Časová rada; ARIMA; Hawkular; ACF

JEL klasifikácia: C53

[a]Dostupné na http://www.hawkular.org

1 Úvod

Monitorovanie doležitých business aplikácii, ako sú napríklad bankové systémi alebo rôzne servery na ktorých bežia služby ktoré sú využívané 24/7 je veľmi dôležité. Vačšina monitorovacích systémov dokáže upozorniť administrátora na vyťaženie pri prekročení hraničnej hodnoty. V monitorovacom systéme Hawkular je možné zapnúť predikciu, takže administrátor bude upozornený vopred ak by náhodov malo dôjsť k prekročeniu spomenutej hraničnej hodnoty. Použíté modely časových rád v systéme Hawkular sú varianty exponenciálneho vyrovnania. Modely ARIMA nemohli byť použité z dôvodu nestacionarity dát a náročnosti na výpočet – systém je schopný analyzovať aj niekoľko tisíc model naraz.

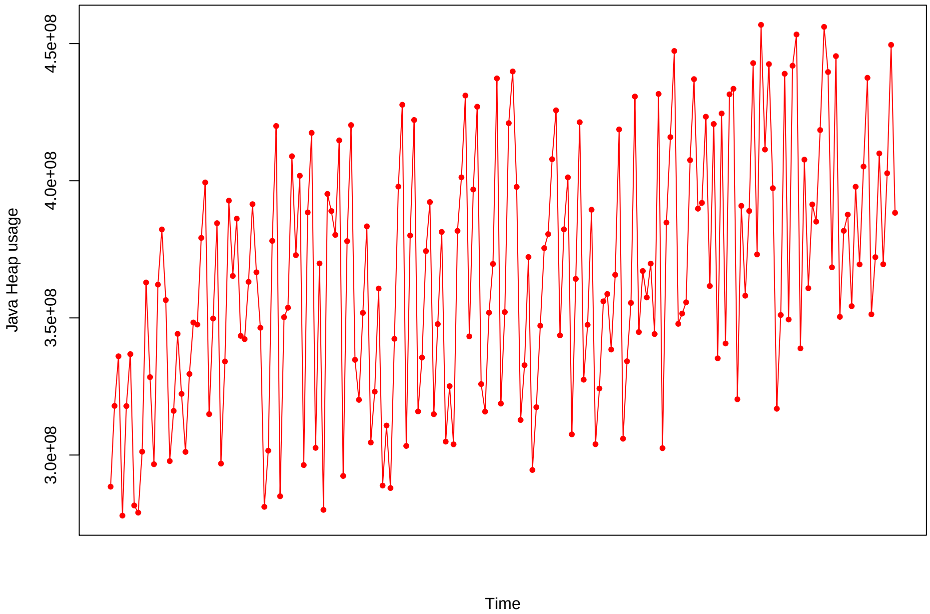
V tejto práci sa zameriam na konštrukciu optimálneho ARIMA modelu, ktorý následne porovnám s exponenciálnym vyrovnaním konkrétne Holtovou metódou s linárnym trendom.

2 Analýza Java Heap metriky

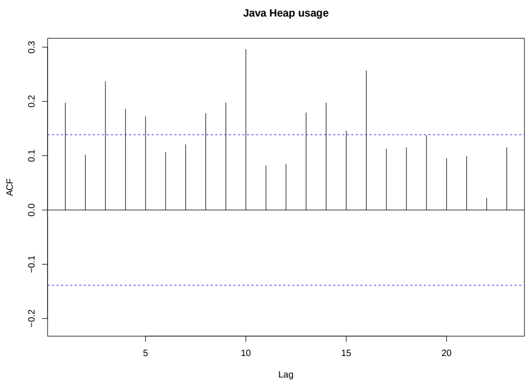
V tejto kapitole budeme analyzovať časovú radu, ktorá popisuje vyťaženosť Java heap-u v čase.

Graf časovej rady:

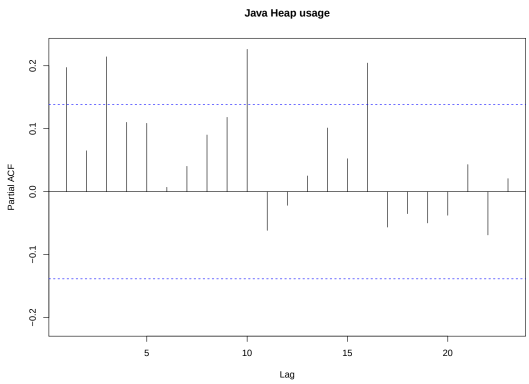
[1]Masarykova univerzita, Fakulta informatiky, obor: Service Science Management Engineering, p.loffay@mail.muni.cz



Obr. 1: Vyťaženosť Java Heap-u v čase.



(a) Autokorelačná funkcia.



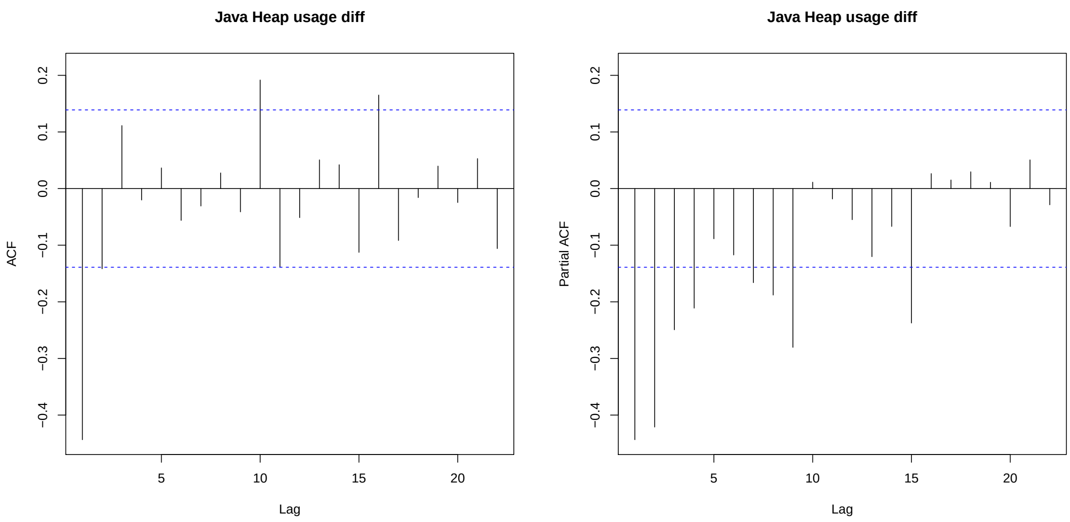
(b) Parciálna autokorelačná funkcia.

2.1 Identifikácia ARIMA modelu

Z obrázka 1 je možné usúdiť, že časová rada obsahuje mierny lineárny trend, čo by znamenalo že je nestacionárna. Následne si vykresíme autokorelačnú (zkrátene) a parciálknu autokorelačnú funkciu (zkrátene PACF).

Z obrázka 2a je vidieť že hodnoty ACF sú kladné ale avšak relatívne blízke nule, stým že významne neklesajú k nule. V grafe ACF 2a taktiež nie sú prítomné prítomné periodicky posunuté vysoké hodnoty, takže rada neobsahuje sezónnosť.

Keďže hodnotz ACF so zvyšujúcim spozdením neklesali k nule rozhodli sme sa urobiť ADF test na test stacionarity.



Obr. 3: ACF a PACF diferencovanej časovej rady.

```
> adfTest(as.numeric(b$avg), lags = 0, type="nc")
Title:
 Augmented Dickey-Fuller Test
Test Results:
  PARAMETER:
    Lag Order: 0
  STATISTIC:
    Dickey-Fuller: -0.976
  P VALUE:
    0.3042
```

Z vyšie uvedeného ADF testu môžeme vidieť, že nulovú hypotézu časová rada nie je stacionárna by sme na hladine významnosti 0.05 ne zametli. Takže naša časová rada je nestacionárna. Pre overenie skusíme KPSS test, ktorého nulová hypotéza je že časová rada je stacionárna.

```
> kpss.test(as.numeric(b$avg))
KPSS Test for Level Stationarity
data:  as.numeric(b$avg)
KPSS Level = 2.5916, Truncation lag parameter = 3, p-value = 0.01
```

Na hladine významnosti by sme nulovú hypotézu zamietli. Oba testy nám ukázali, že časová rada nie je stacinárna. Následne môžeme pomocou KPSS testu testovať, či je daná časová rada trend stacinárna:

```
> kpss.test(as.numeric(b$avg), null='Trend')
KPSS Test for Trend Stationarity
data:  as.numeric(b$avg)
KPSS Trend = 0.073128, Truncation lag parameter = 3, p-value = 0.1
> ndiffs(as.numeric(b$avg))
[1] 1
> diff = diff(as.numeric(b$avg))
```

Z výstupu je jasné, že rada je trend stacinárna takže nulovú hypotézu na hladine 0.05 nezamietame.

Ďalej pokračujeme vykreslením ACF a PAC. Z obrázka parciálnej autokorelačnej funkcie **??** možme usúdiť, že do úvahy by spadal model *AR(15)* alebo *MA(1)*. Keďže ACF je možné obmedziť krivkou U, tak je lepšie vybrať model *MA(1)* [2].

Alternatívny spôsob voľby modelu je pomocou informačných kritérii. Tento spôsob je vhodný pre plne automatizované spracovanie [2] napríklad v ekonometrických softvéroch. K identifikácia modelu ARMA(p, q) sa priktupuje ako k minimalizácii funkcie 2.1

$$(\hat{p}, \hat{q}) = arg\ \min_{(k,l)} A(k,l) \tag{2.1}$$

A je vhodné kritérium pre ktorého konštrukciu musíme odhadnúť model ARMA(k,l). Pri minimalizácii postupujeme postupne inkrementujeme obidva parametre k, l. V tejto práci sme zvolili Akaikeho informačné kritérium:

$$A(k,l) = AIC(k,l) = ln\hat{\sigma}^2_{k,l} + \frac{2(k+l+1)}{n} \tag{2.2}$$

Z rovnice 2.2 je zrejmé, že kritérium penalizuje veľké rády k a l. $\hat{\sigma}^2_{k,l}$ je smerodajná odchylka reziduí modelu. Poďme si vypísať niekoľko kandidátov ARIMA modelov pomocu príkazu auto.arima().

```
> auto.arima(as.numeric(df$avg), approximation=FALSE, trace=TRUE, ic='aic', allowdrift=FALSE)
 ARIMA(2,1,2)                    : 7556.055
 ARIMA(0,1,0)                    : 7696.058
 ARIMA(1,1,0)                    : 7651.407
 ARIMA(0,1,1)                    : 7557.045
 ARIMA(1,1,2)                    : 7560.654
 ARIMA(3,1,2)                    : 7557.714
 ARIMA(2,1,1)                    : 7553.937
 ARIMA(1,1,1)                    : 7557.937
 ARIMA(3,1,1)                    : 7555.879
 ARIMA(2,1,0)                    : 7613.902

 Best model: ARIMA(2,1,1)
Series: as.numeric(df$avg)
ARIMA(2,1,1)
Coefficients:
          ar1      ar2      ma1
      -0.0985  -0.1774  -0.9441
s.e.   0.0716   0.0716   0.0199
```

Ako je vidieť funkcia zvolila model ARIMA(2,1,1) ktorého AIC kritérium bolo najnižšie. Odhadnutý model môžeme zapísať v tvare:

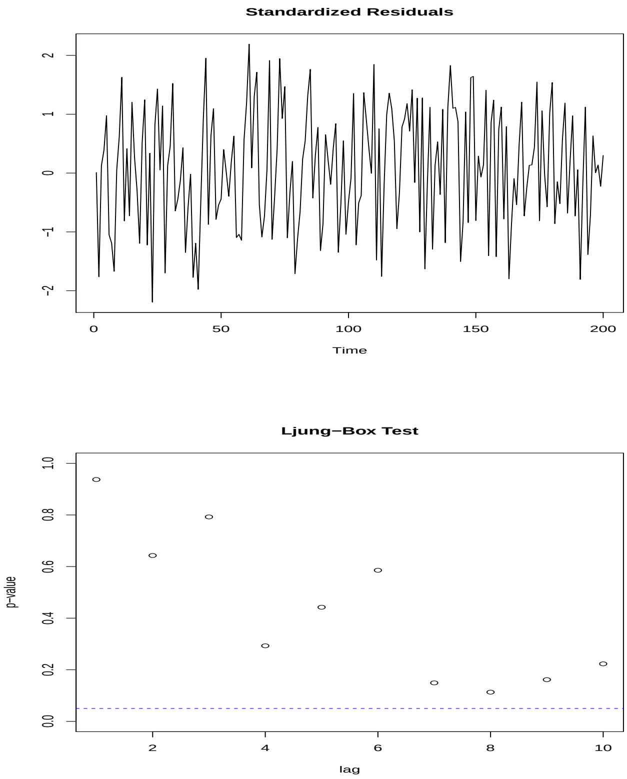
$$Y_t = -0.985Y_{t-1} - 0.1774Y_{t-2} - 0.9441\varepsilon_{t-1} + \varepsilon_t \tag{2.3}$$

Takto sme ukázali, že je možné konštrulovať ARIMA model aj analytickým spôsobom. Chcel by som avšak poznanenať, že voľbu modelu je lepšie nechať na overený štatistický softvér.

Po úspšnom odhade rádu modelu je by som chcel zmieniť ako sa počítajú jednotlivé koeficienty. Pre AR model platí, že sa dajú vypočítať pomocou OLS alebo Yule – Walkerových rovníc [1]. Výpočet kofeicientov MA modelu je zložitejší a je možný pomocou rekurzívnej Levison-Durbin metódy. Odhadom presných parametrov modelu sa v tejto práci ďalej zaoberať.

Na záver sa pozrieme na rezíduá odhadnutého modelu. Ak sme postupovali správne rezíduá by mali pripomínať biely šum a nemali by byť medzi sebou korelované (inakšie by sme ich vedeli modelovať). Toto tvrdenie si overíme Ljung – Box testom, ktorého nulová hypotéza hovorí o tom, že sú dáta nezávislé distribuované. Z grafu **??** môžeme prehlásiť, že nulovú hypotézu na hladine významnosti 0.05 nezamietame.

Na následujúcich grafoch si vykresíme rezíduá, ich autokorelačnú funkciu a p-hodnoty pre rôzdne opozdenia Ljung – Box testu.



Obr. 4: Rezíduá odhadnutého ARIMA modelu.

3 Záver

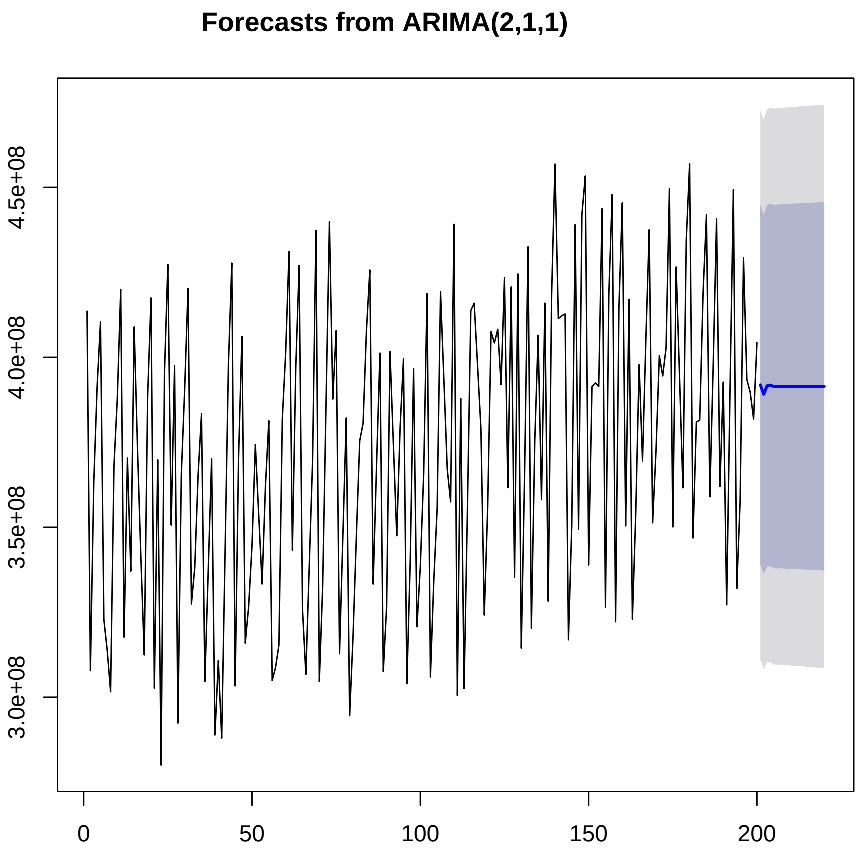
Poďakovanie

Na záver by som chcel poďakovať Ing. Danielovi Němcovi, Ph.D. za návrh na vypracovanie tejto témi a za veľmi príjemné a užitočné konzultácie. Ďalej by som chcel poďakovať Ester Železňákovej za gramatickú korektúru textu.

Literatúra

[1] Brockwell, P.; Davis, R.: *Time Series: Theory and Methods*. Praha: Springer-Verlag New York, 2009, ISBN 978-0-387-97429-3.

[2] Tomáš, C.: *Finanční ekonometrie*. Ekopress, 2008, ISBN 978-80-86929-43-9.



Obr. 5: Predikcia na 20 krokov do predu.

A Prílohy

- Skript v jazyku R
- Zdrojový text tejto správy v LaTeX – e