Федеральное государственное бюджетное образовательное учреждение высшего образования

Московский авиационный институт

(национальный исследовательский университет)

Кафедра 804 «Теория вероятностей и компьютерное моделирование»

О т ч е т

по научно-исследовательской работе

направление подготовки 01.04.04 «Прикладная математика»

Студента Величутина А.И

Курс 3 группа 8О-304Б-16

Научный руководитель Платонов Е.Н.

____

Дата оценка

подпись

Москва 2019

1) Постановка задачи

Ознакомиться с языком программирования R, построить с помощью методов ARMA и ARIMA прогноз индекса RTS.

2) Модели ARMA / ARIMA

AR (авторегрессия) – модель временных рядов, в которой значения временного ряда в данный момент линейно зависят от предыдущих значений этого же ряда.

MA (скользящее среднее) – модель, в которой значение функции каждой точки равны среднему значению исходной функции за предыдущий период.

Объединив эти две модели получили ARMA (p, q), где p, q – целые числа, задающие порядок модели.

p – порядок регрессии, а q – порядок скользящего среднего

$$X_t = c + \varepsilon_t + \sum_{i=1}^{p} a_i X_{i-1} + \sum_{i=1}^{q} \beta_i \varepsilon_{i-1}$$

$a_1, \ldots, a_p$ – параметры модели (коэффициенты авторегрессии)

$\varepsilon_1, \ldots, \varepsilon_q$ – параметры модели скользящего среднего

c – постоянная

ε_t – белый шум

Таким образом, ARMA – линейная модель множественной регрессии, в которой в качестве объясняющих переменных выступают прошлые значения самой зависимой переменной, а в качестве регрессионного остатка – скользящие средние из элементов белого шума.

Метод применим только к стационарным рядам.

Модель ARIMA тоже самое, что и ARMA, но рассматривает приращения. Может работать с нестационарными рядами и прогнозировать их.

$$\Delta^d X_t = c + \varepsilon_t + \sum_{i=1}^{p} a_i \Delta^d X_{i-1} + \sum_{i=1}^{q} \beta_i \varepsilon_{i-1}$$

Δ^d – оператор разности временного ряда порядка d.

Модель ARIMA (p, d, q) означает, что разности временного ряда порядка d подчиняются модели ARIMA (p, q)

3) Решение задачи

Все данные RTSI были взяты с официального сайта finam.ru.

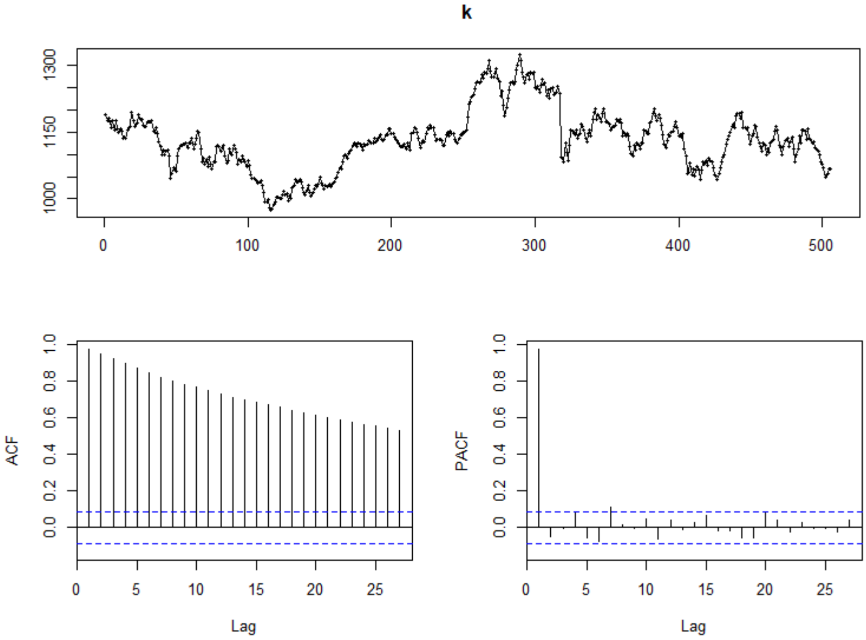
Рассмотрим пример с ежедневными данными за 2 года (с 1.01.2017 по 1.01.2019) при использовании стационарной модели ARMA (1,1).

library("forecast", lib.loc = .libPaths()[1]) #указываем, какую библиотеку будем использовать

mytable <- read.table(file = "C:/Rdata/RTS1.csv", sep = ',', header = TRUE)

k <- mytable$X.CLOSE #k – значения столбца X.CLOSE

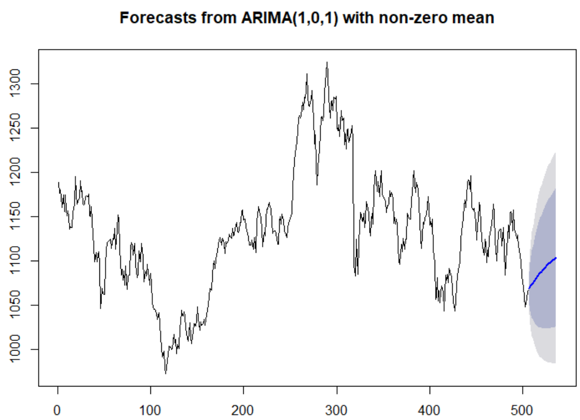
tsdisplay(k)

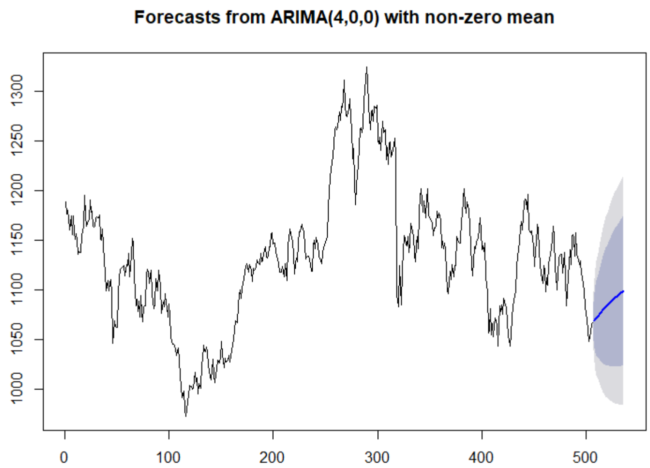


mod_ARMA1 <- Arima(k, order = c(1,0,1))

mod_ARMA2 <- Arima(k, order = c(4,0,0))

Сравним обе модели






```
> summary(mod_ARMA1)
Series: k
ARIMA(1,0,1) with non-zero mean

Coefficients:
         ar1     ma1       mean
      0.9723  0.0581  1131.8071
s.e.  0.0103  0.0448    23.7609

sigma^2 estimated as 224.2:  log likelihood=-2087.41
AIC=4182.82   AICc=4182.9   BIC=4199.72

Training set error measures:
                     ME     RMSE      MAE         MPE      MAPE      MASE       ACF1
Training set -0.1789196 14.93028 11.00144 -0.03333555 0.9707785 0.9945608 1.54944e-05
```

```
> summary(mod_ARMA2)
Series: k
ARIMA(4,0,0) with non-zero mean

Coefficients:
         ar1      ar2      ar3     ar4       mean
      1.0309  -0.0554  -0.0909  0.0916  1132.2121
s.e.  0.0442   0.0636   0.0636  0.0442    25.6532

sigma^2 estimated as 223.2:  log likelihood=-2085.27
AIC=4182.53   AICc=4182.7   BIC=4207.89

Training set error measures:
                     ME     RMSE     MAE         MPE      MAPE      MASE        ACF1
Training set -0.2145802 14.86673 11.0024 -0.03636641 0.9707414 0.9946477 0.005518938
```

Видим, что средняя абсолютная ошибка прогноза (MAE)

mod_ARMA1 < mod_ARMA2 ~ (11.0014 < 11.0024)

а среднеквадратичное отклонение модели (RMSE)

mod_ARMA1 > mod_ARMA2 ~ (14.93028 > 14.86673)

Штрафной критерий Акайки (AIC)

mod_ARMA1 > mod_ARMA2 ~ (4182.82 > 4182.53)

Исходя из полученных данных можем утверждать, что модель mod_ARMA2 лучше.

Перейдём к ARIMA

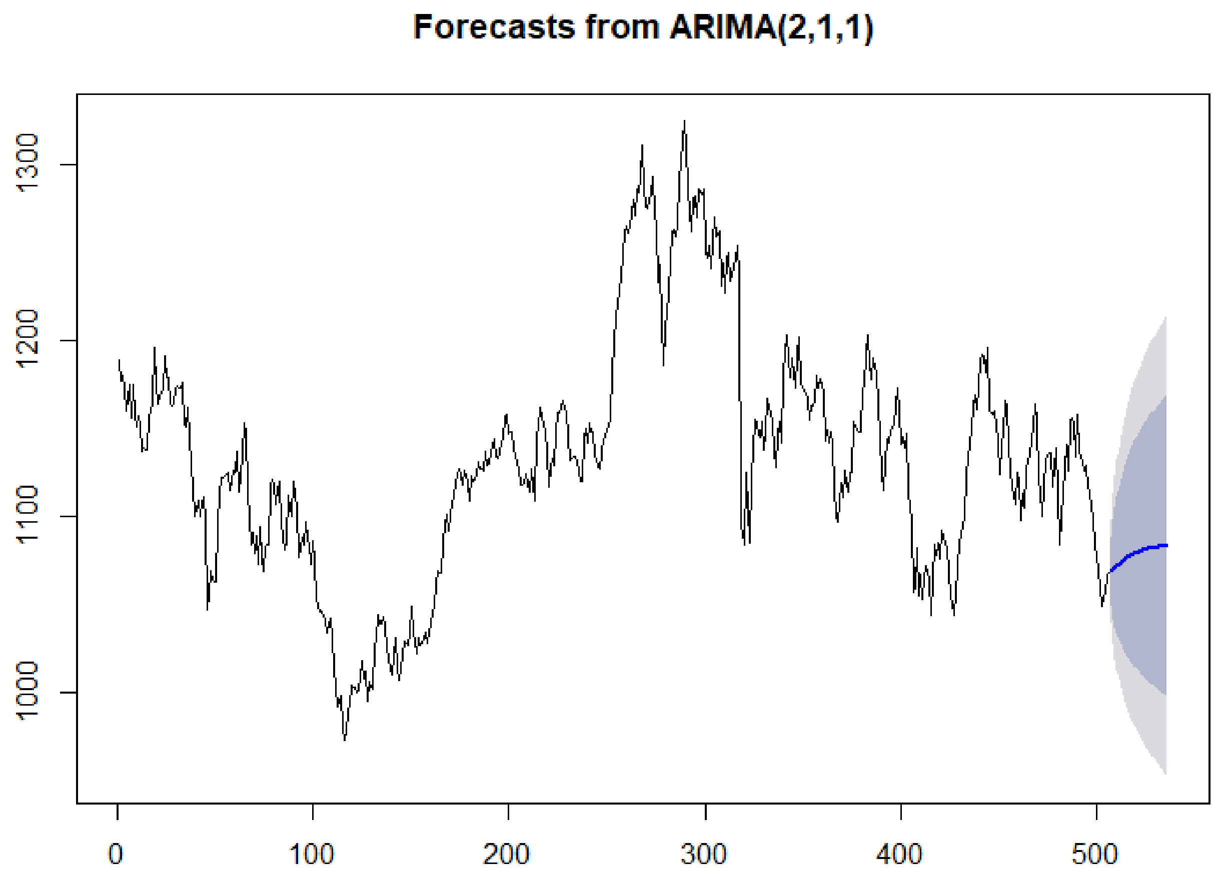
mod_ARIMA1 <- Arima(k, order = c(2,1,1))

summary(mod_ARIMA1)

```
> summary(mod_ARIMA1)
Series: k
ARIMA(2,1,1)

Coefficients:
         ar1      ar2      ma1
      0.9929  -0.0666  -0.9532
s.e.  0.0712   0.0457   0.0569

sigma^2 estimated as 226:  log likelihood=-2083.79
AIC=4175.58   AICc=4175.66   BIC=4192.47

Training set error measures:
                     ME     RMSE      MAE         MPE      MAPE      MASE          ACF1
Training set -0.3159008 14.97404 11.00938 -0.03967434 0.9723299 0.9952791 -0.0009809079
```

prognoz3 <- forecast(mod_ARIMA1, h = 30)

plot(prognoz3)



Рассмотрим функцию auto.arima

Возвращает лучшую модель ARIMA в соответствии со значением AIC, AICc или BIC. Функция выполняет поиск по возможной модели в пределах предоставленных ограничений порядка.

timeseries <- ts(k, start = c(2017,1,1), frequency = 252)

timeseries

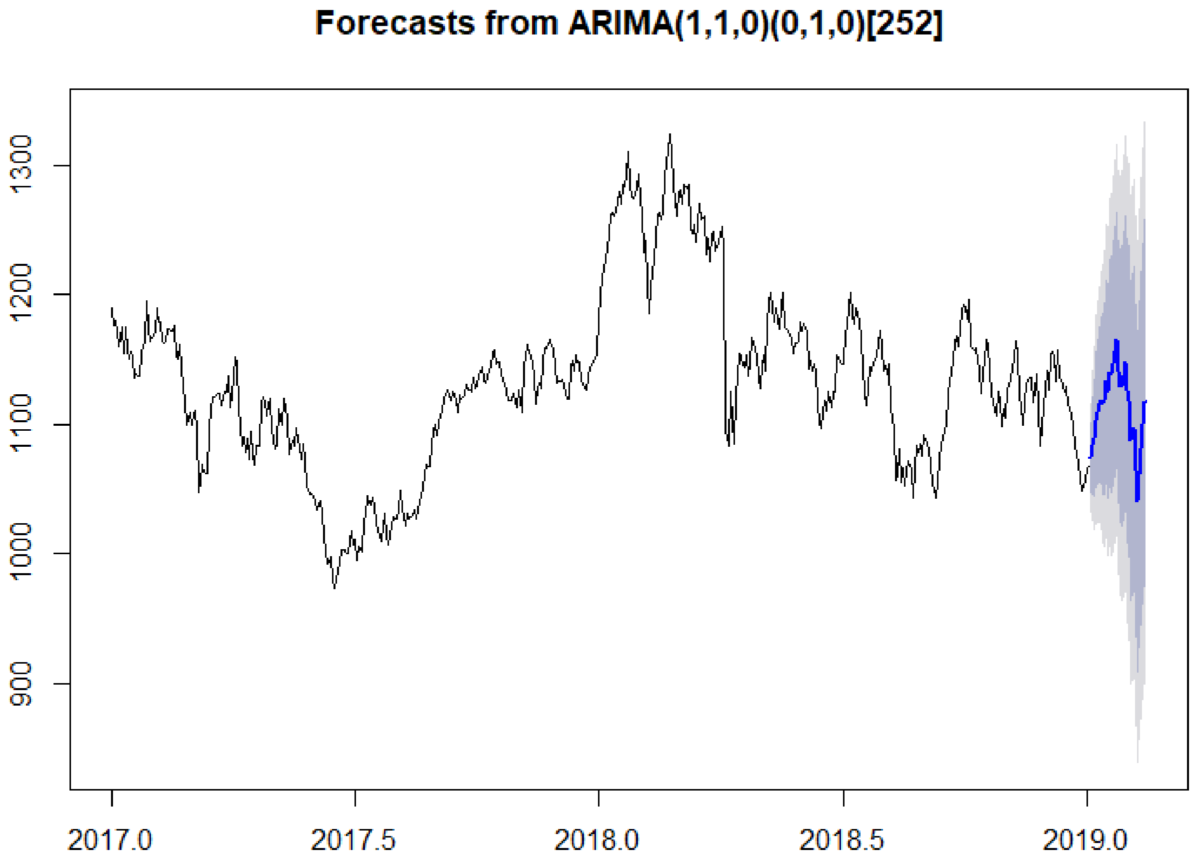
plot.ts(timeseries)

mod_AUTOARIMA <- auto.arima(timeseries)

summary(mod_AUTOARIMA)

```
> summary(mod_AUTOARIMA)
Series: timeseries
ARIMA(1,1,0)(0,1,0)[252]

Coefficients:
          ar1
      -0.0385
s.e.   0.0636

sigma^2 estimated as 456:  log likelihood=-1136.61
AIC=2277.22   AICc=2277.27   BIC=2284.29

Training set error measures:
                     ME     RMSE     MAE         MPE      MAPE      MASE         ACF1
Training set -0.3226232 15.06955 8.20536 -0.03656718 0.708704 0.1063611 -0.008911279
```

prognozAUTOARIMA <- forecast(mod_AUTOARIMA, h = 30) plot(prognozAUTOARIMA)



Можем заметить, что добавился анализ сезонности.

В нашем случае сезонность является год, frequency = 252 т.к. не на все дни есть значение RTSI.

Сравнив все модели, наилучшим оказался mod_AUTOARIMA с анализом сезонности.

4) Листинг программы

library("forecast", lib.loc = .libPaths()[1]) #указываем, какую библиотеку будем использовать

mytable <- read.table(file = "C:/Rdata/RTS1.csv", sep = ',', header = TRUE)

#head(mytable)

k <- mytable$X.CLOSE

#head(k) #проверка что данные записались в k

tsdisplay(k)

mod_ARMA1 <- Arima(k, order = c(1,0,1))

summary(mod_ARMA1)

prognoz1 <- forecast(mod_ARMA1, h = 30)

plot(prognoz1)

```
mod_ARMA2 <- Arima(k, order = c(4,0,0))
summary(mod_ARMA2)
prognoz2 <- forecast(mod_ARMA2, h = 30) #Прогнозирование Временного Ряда
plot(prognoz2)

timeseries <- ts(k, start = c(2017,1,1), frequency = 252)
timeseries
plot.ts(timeseries)
mod_AUTOARIMA <- auto.arima(timeseries)
summary(mod_AUTOARIMA)
prognozAUTOARIMA <- forecast(mod_AUTOARIMA, h = 30)
plot(prognozAUTOARIMA)

mod_ARIMA1 <- Arima(k, order = c(2,1,1))
summary(mod_ARIMA1)
prognoz3 <- forecast(mod_ARIMA1, h = 30)
plot(prognoz3)
```

5) Вывод

Ознакомился с временными рядами, моделями прогнозирования и их оцениванием. Получил навыки работы с языком программирования R.

6) Источники

[https://r-analytics.blogspot.com/2011/07/r_24.html]

[https://www.finam.ru/profile/mirovye-indeksy/rts/export/]

[https://ru.coursera.org/lecture/trendy-klassifikatsii/2-2-arma-i-arima-GcA4R]

[https://sites.google.com/site/econometricsacademy/econometrics-models/time-series-arima-models]

[https://www.youtube.com/watch?v=RzvxjtTBGzs&list=PLu5flfwrnSD5d02G9YJcDv30Fp5_70-sI&index=90]