

Пакеты прикладных программ, задание 1

Задание выполняло 3 человека:

- 1) Купина Марина (1, 2 часть задания)
- 2) Носков Глеб (3 часть задания)
- 3) Чикунов Максим (оформление отчёта и рисование графиков)

Инструкция по запуску:

Подключенные библиотеки: `library(tseries)`, `library(forecast)`, `library(MLmetrics)`

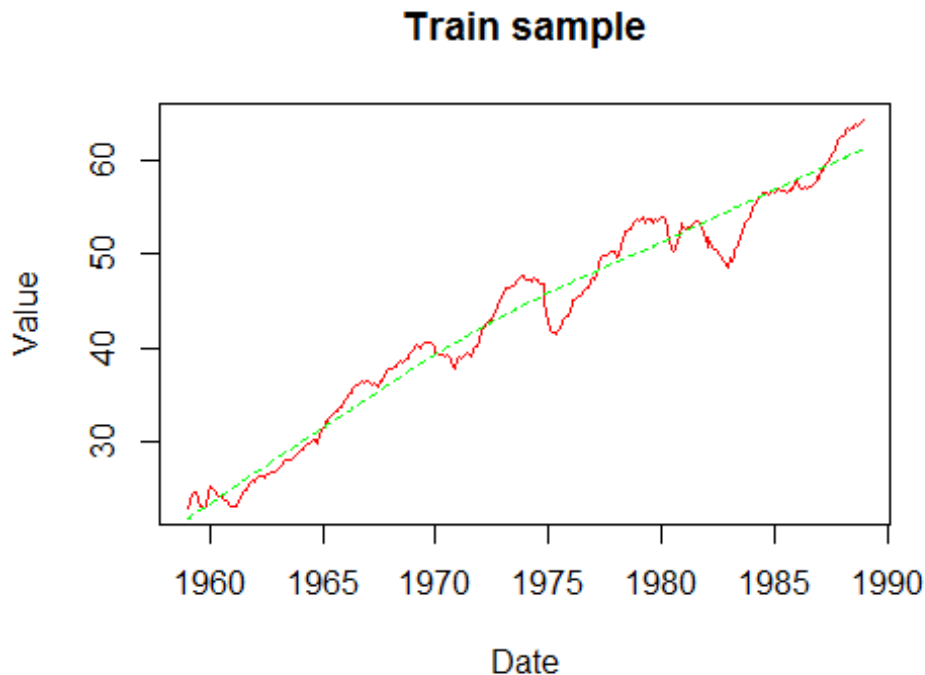
Для запуска необходимо:

- 1) Открыть файл `task1.rmd` через RStudio.
- 2) Указать пути до файлов `training.csv`, `testing.csv`.
- 3) Запустить программу.

Цели и подходы к решению задания:

- 1) Проверить является ли ряд стационарным в широком смысле.

Для выполнения этой части воспользуемся тестом Дики-Фуллера, который подтверждает или отвергает гипотезу о стационарности временного ряда. Считаем данные из файла `training.csv` и проведем тест Дики-Фуллера.



Тест Дики-Фуллера:

```
##  
## Augmented Dickey-Fuller Test  
##  
## data: train$Value  
## Dickey-Fuller = -3.2505, Lag order = 7, p-value = 0.07962  
## alternative hypothesis: stationary
```

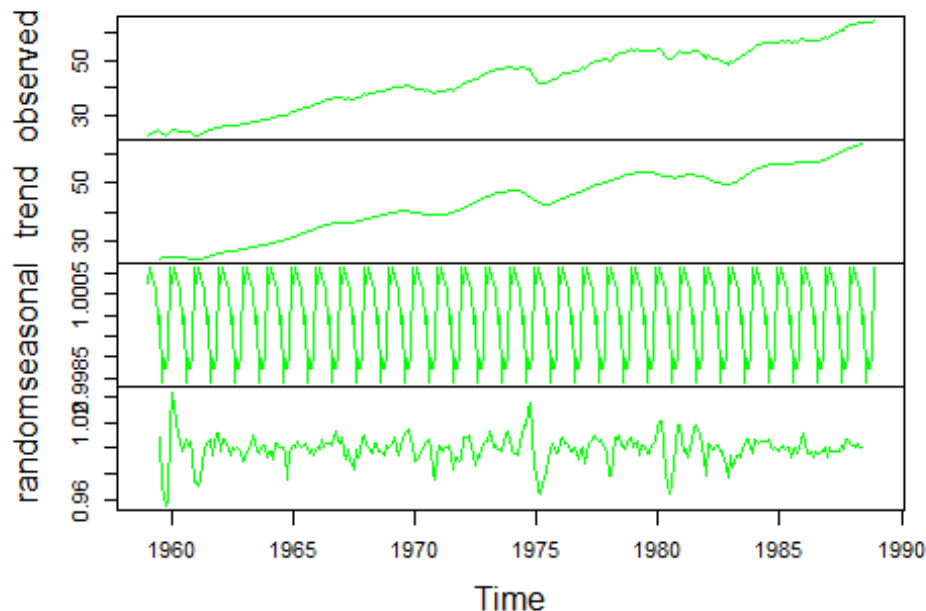
Значения p-value выше уровня значимости (5%), что не даёт нам возможности опровергнуть нулевую гипотезу о том, что ряд не стационарен. Однако наличие тренда подтверждает наше опасение: Ряд не стационарен(ниже).

- 2) Разложить временной ряд на тренд, сезонность, остаток в соответствии с аддитивной, мультипликативной моделями. Визуализировать их, оценить стационарность полученных рядов, сделать выводы.

Для разложения исходного ряда будем использовать функцию `decompose()`. О не стационарности исходного ряда может говорить наличие тренда, что подтверждает полученные нами результаты теста Дики Фуллера. Проверим стационарность тренда, сезонности и остатка.

Мультипликативная модель:

Decomposition of multiplicative time series



```
## Warning in adf.test(dc$seasonal[!is.na(dc$seasonal)], alternative =
## c("stationary")): p-value smaller than printed p-value

##
## Augmented Dickey-Fuller Test
##
## data: dc$seasonal[!is.na(dc$seasonal)]
## Dickey-Fuller = -37.814, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary

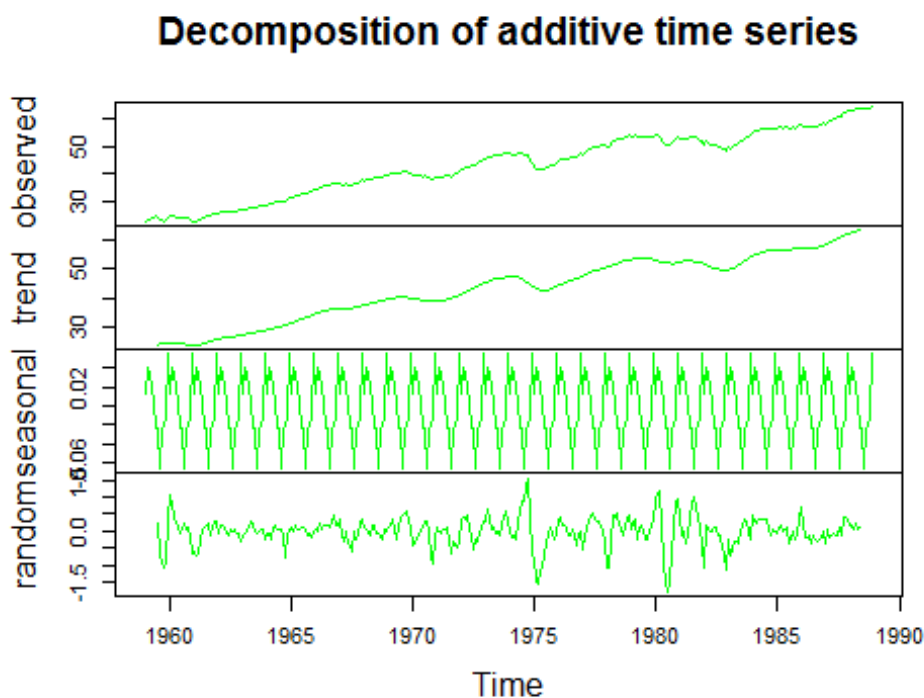
##
## Augmented Dickey-Fuller Test
##
## data: dc$trend[!is.na(dc$trend)]
## Dickey-Fuller = -2.853, Lag order = 7, p-value = 0.2169
## alternative hypothesis: stationary

## Warning in adf.test(dc$random[!is.na(dc$random)], alternative =
## c("stationary")): p-value smaller than printed p-value

##
## Augmented Dickey-Fuller Test
##
## data: dc$random[!is.na(dc$random)]
## Dickey-Fuller = -7.2542, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

Из полученных значений p-value мы можем утверждать, что сезонная компонента – стационарна, тренд - не стационарный, остаток – стационарен.

Аддитивная модель:



```
## Warning in adf.test(adddc$seasonal[!is.na(adddc$seasonal)], alternative =  
## c("stationary")): p-value smaller than printed p-value  
  
##  
## Augmented Dickey-Fuller Test  
##  
## data: adddc$seasonal[!is.na(adddc$seasonal)]  
## Dickey-Fuller = -63.636, Lag order = 7, p-value = 0.01  
## alternative hypothesis: stationary  
  
##  
## Augmented Dickey-Fuller Test  
##  
## data: adddc$trend[!is.na(adddc$trend)]  
## Dickey-Fuller = -2.853, Lag order = 7, p-value = 0.2169  
## alternative hypothesis: stationary  
  
## Warning in adf.test(adddc$random[!is.na(adddc$random)], alternative =  
## c("stationary")): p-value smaller than printed p-value  
  
##  
## Augmented Dickey-Fuller Test  
##  
## data: adddc$random[!is.na(adddc$random)]  
## Dickey-Fuller = -6.835, Lag order = 7, p-value = 0.01  
## alternative hypothesis: stationary
```

Для аддитивной модели имеем аналогичные результаты.

- 3) Проверить, является ли временной ряд интегрированным порядка k . Если является, применить к нему модель ARIMA, подобрав необходимые параметры с помощью функции автокорреляции и функции частичной автокорреляции. Выбор параметров обосновать. Отобразить несколько моделей. Предсказать значения для тестовой выборки. Визуализировать их, посчитать r^2 score для каждой из моделей.

Интегрируемость будет проверять по определению: Интегрированный временной ряд - нестационарный временной ряд, разности некоторого порядка от которого являются стационарным временным рядом.

Проверим, является ли дифференцированный ряд стационарным:

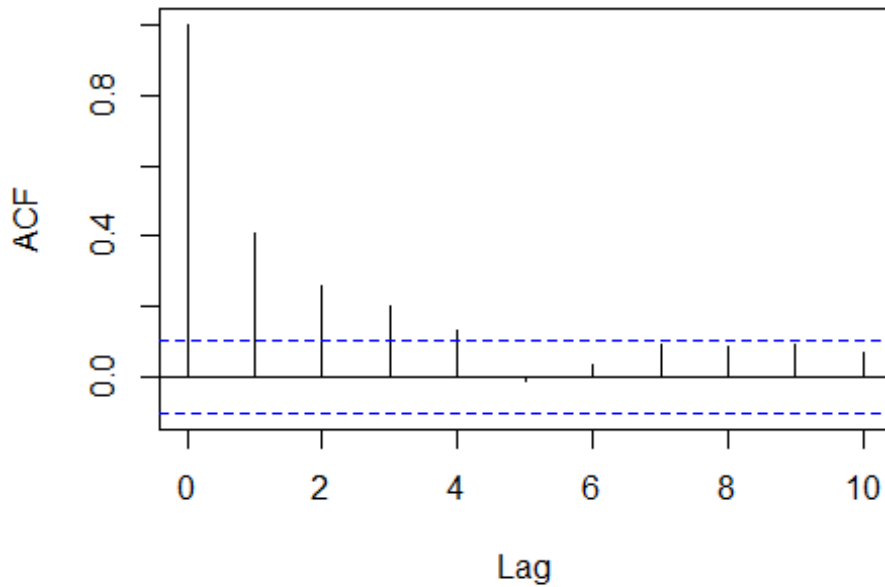
```
## Warning in adf.test(Valuesdiff1, alternative = c("stationary")): p-value
## smaller than printed p-value

##
## Augmented Dickey-Fuller Test
##
## data: Valuesdiff1
## Dickey-Fuller = -5.0661, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

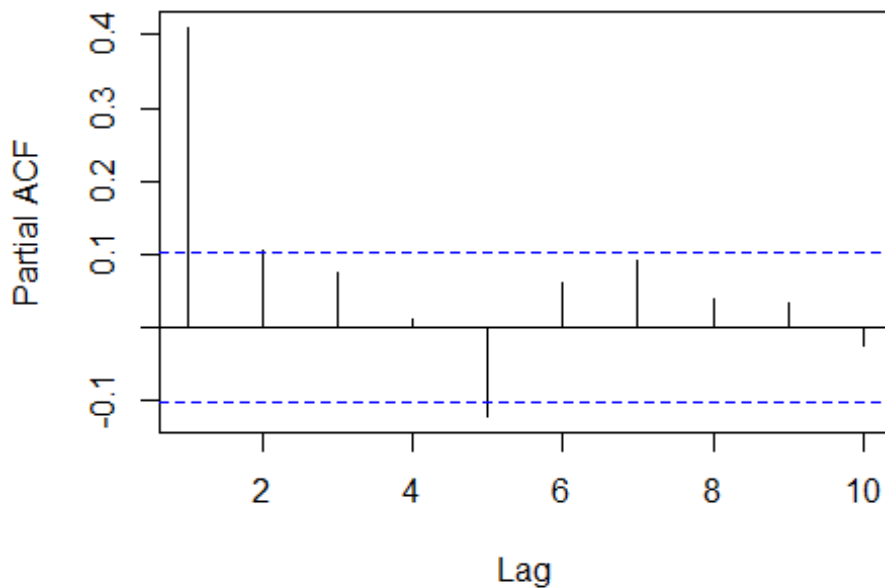
Получаем, что ряд соответствующих разностей - стационарный => исходный ряд интегрируем. Значит временной ряд является интегрированным порядка $d=1$.

Для моделирования будем использовать модель ARIMA, в качестве параметра d которой можем указать значение $d = 1$ (так как ряд - интегрированный). Чтобы построить модель нам нужно определить ещё 2 дополнительных параметра: (p, q) . Для их определения нам необходимо изучить автокорреляционную (ACF) и частично автокорреляционную (PACF) функции для ряда первых разностей.

Series Valuesdiff1



Series Valuesdiff1



В коррелограмме ACF находим минимальный лаг, который значимо отличается от нуля. Это значение = 5. Поэтому в качестве параметра q можем взять коэффициент $q = 4$.

По аналогичной логике определяем возможное значение p по коррелограмме PACF. Из графика не понятно, попадает ли значение лага 2 в границы значимости, поэтому выведем значения функции.

```
##
## Partial autocorrelations of series 'Valuesdiff1', by lag
##
##      1      2      3      4      5      6      7      8      9     10
## 0.410 0.105 0.076 0.011 -0.121 0.062 0.091 0.038 0.035 -0.024
```

Лаг=2 лежит вне границ, поэтому можем положить коэффициент $p=2$.

Поэтому в качестве моделей мы можем рассмотреть 3 основных: ARIMA(p,1,0)
ARIMA(0,1,q) ARIMA(p,1,q)

Дополнительно для самопроверки используем функцию `auto.arima()`, которая определяет коэффициенты для модели с наименьшим модифицированным критерием Акаике(AICc).

Построим несколько моделей с разными коэффициентами и отберем наилучшую модель по информационному критерию Акаике.

```
## Series: train$Value
## ARIMA(0,1,4) with drift
##
## Coefficients:
##      ma1      ma2      ma3      ma4      drift
##    0.3703 0.2371 0.2090 0.1858 0.1172
## s.e. 0.0519 0.0551 0.0567 0.0530 0.0352
##
## sigma^2 estimated as 0.1135: log likelihood=-116.5
## AIC=245.01 AICc=245.24 BIC=268.31
```

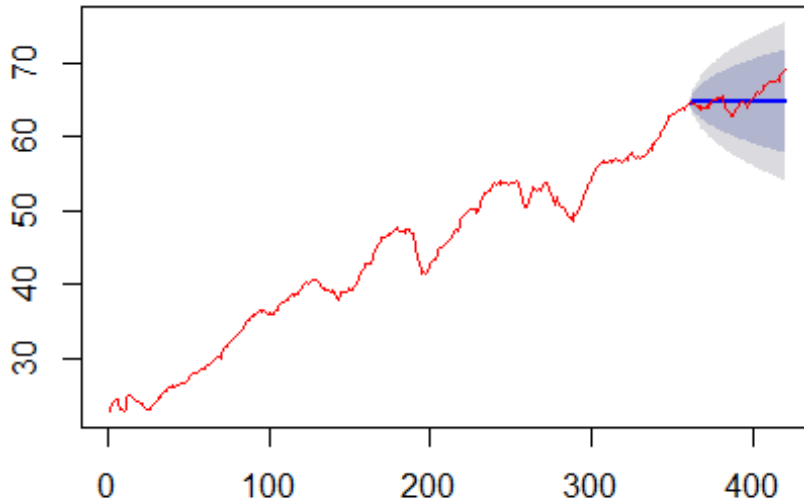
Построим модели ARIMA1(2,1,0), ARIMA2(0,1,4), ARIMA3(1,1,1), ARIMA4(2,1,4), определим для них `r2_score` и ту, у которой наименьший критерий Акаике.

```
##      df      AIC
## arima1  3 258.5099
## arima2  5 253.3399
## arima3  3 255.6413
## arima4  7 257.3009
```

Наименьшему критерию соответствуем модель ARIMA(0,1,4)

Построим графики временного ряда training и test и добавим к ним предсказанные значения

Forecasts from ARIMA(0,1,4)



Вывод: В ходе проверки выяснилось, что все модели, коэффициенты в которых подобраны разумным образом, предсказывают значения примерно одинаково. Однако модель ARIMA предсказывает значения для будущего не совсем удачно, поэтому имеет смысл использовать более мощные математические инструменты для прогнозирования временных рядов.