

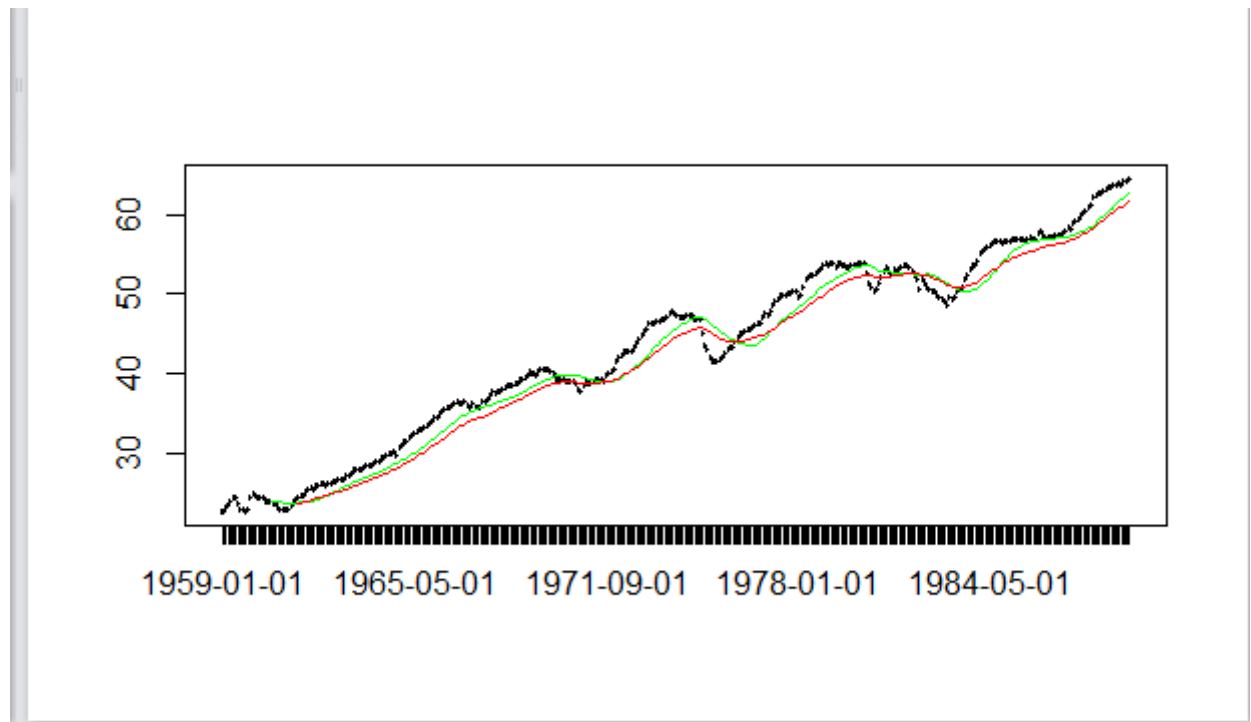
## Часть 1

(выполнил Юмашев А.)

перед запуском необходимо выполнить:

```
install.packages('tseries')
```

1) Построен график ряда и скользящих статистик SMA и ЕМА ("график ряда и скользящих статистик.png")



2) Выполнен тест Дики-Фуллера для исходного ряда:

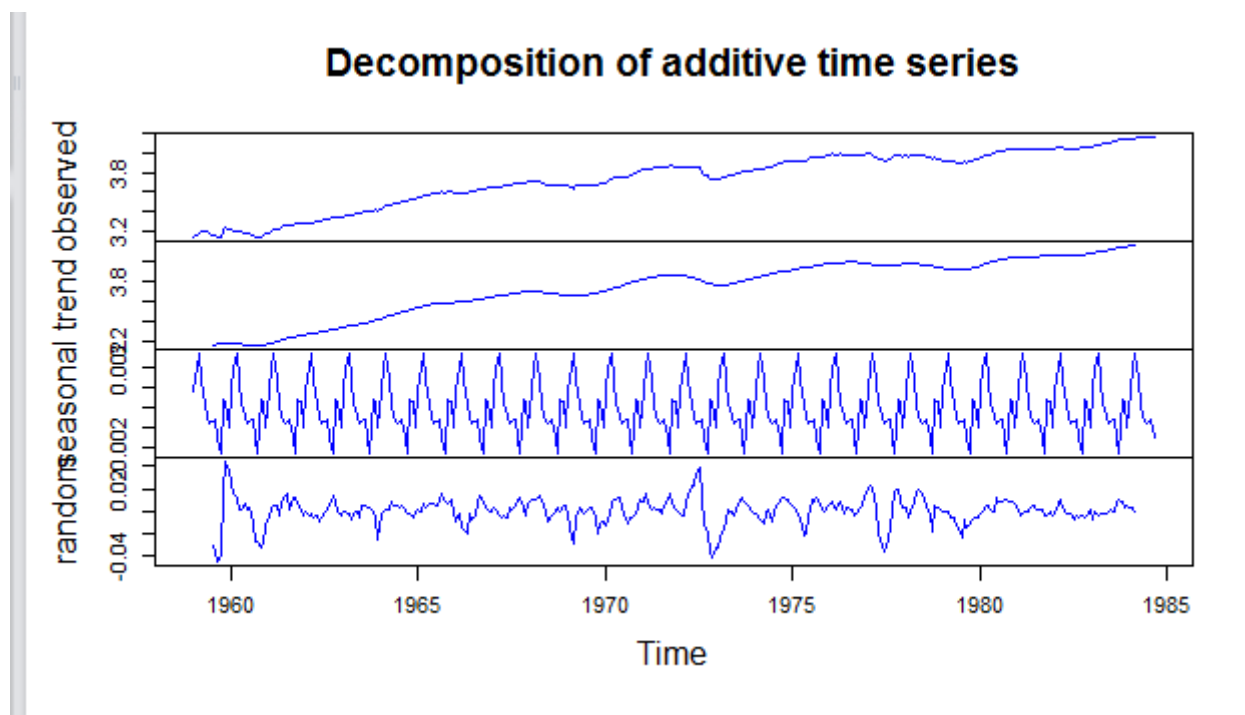
```
Augmented Dickey-Fuller Test  
data: sourcedata$value  
Dickey-Fuller = -3.2505, Lag order = 7, p-value = 0.07962  
alternative hypothesis: stationary
```

p-value > 0.05, ряд не является стационарным

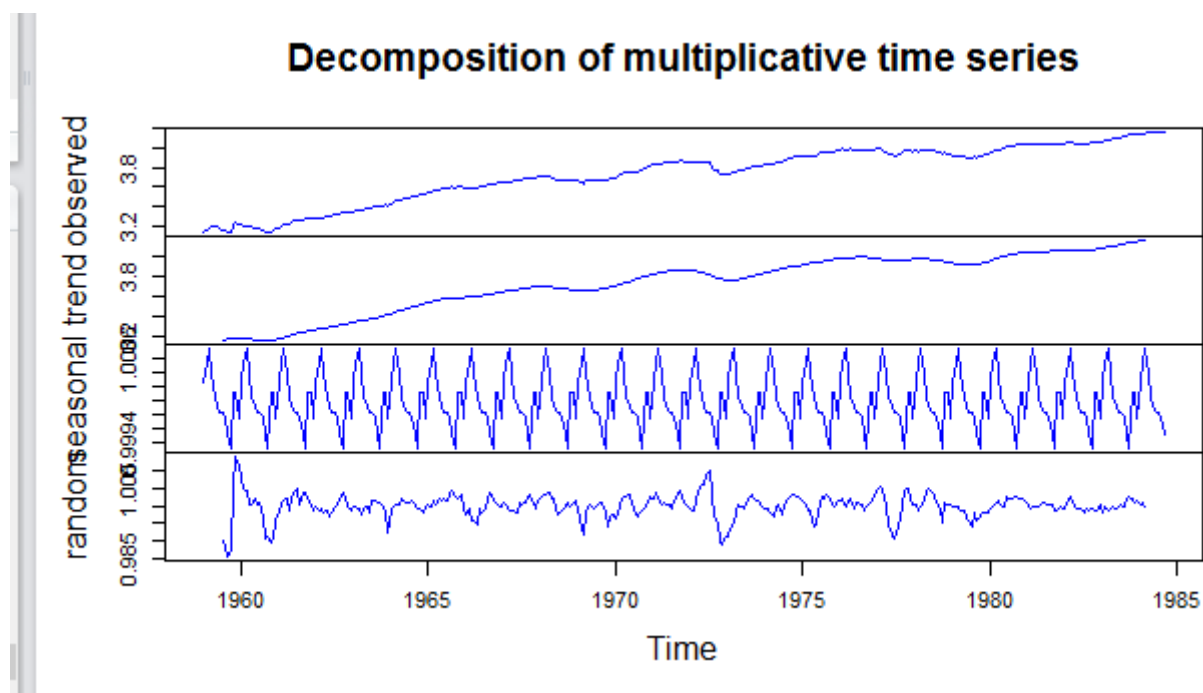
## Часть 2

(Выполнил Юмашев А.)

1) Выполнено разложение на тренд, сезонность, остаток в соответствии с аддитивной и мультипликативной моделями. Выполнена визуализация аддитивной модели:



И визуализация мультипликативной модели:



2) Проведены тесты Дики-Фуллера:

### **Аддитивная модель**

#### *а) Тренд*

```
Augmented Dickey-Fuller Test  
data: na.omit(decompadd$trend)  
Dickey-Fuller = -2.0206, Lag order = 7, p-value = 0.5681  
alternative hypothesis: stationary
```

Стационарность отсутствует, так как  $p\text{-value} > 0.05$ .

#### *б) Сезональность*

```
Augmented Dickey-Fuller Test  
data: na.omit(decompadd$seasonal)  
Dickey-Fuller = -20.81, Lag order = 7, p-value = 0.01  
alternative hypothesis: stationary  
  
warning message:  
In adf.test(na.omit(decompadd$seasonal), alternative = "stationary") :  
  p-value smaller than printed p-value
```

Ряд стационарен, так как  $p\text{-value} < 0.05$ .

#### *в) Остаток*

```
Augmented Dickey-Fuller Test  
data: na.omit(decompadd$random)  
Dickey-Fuller = -6.8575, Lag order = 7, p-value = 0.01  
alternative hypothesis: stationary  
  
warning message:  
In adf.test(na.omit(decompadd$random), alternative = "stationary") :  
  p-value smaller than printed p-value
```

Ряд стационарен, так как  $p\text{-value} < 0.05$ .

### **Мультипликативная модель**

#### *а) Тренд*

```
Augmented Dickey-Fuller Test  
data: na.omit(decompmult$trend)  
Dickey-Fuller = -2.0206, Lag order = 7, p-value = 0.5681  
alternative hypothesis: stationary
```

Стационарность отсутствует, так как  $p\text{-value} > 0.05$ .

### б) Сезонность

```
Augmented Dickey-Fuller Test
data: na.omit(decompmult$seasonal)
Dickey-Fuller = -26.26, Lag order = 7, p-value = 0.01
alternative hypothesis: stationary

warning message:
In adf.test(na.omit(decompmult$seasonal), alternative = "stationary") :
  p-value smaller than printed p-value
```

Ряд стационарен, так как  $p\text{-value} < 0.05$ .

### в) Остаток

```
Augmented Dickey-Fuller Test
data: na.omit(decompmult$random)
Dickey-Fuller = -7.0593, Lag order = 7, p-value = 0.01
alternative hypothesis: stationary

warning message:
In adf.test(na.omit(decompmult$random), alternative = "stationary") :
  p-value smaller than printed p-value
```

Ряд стационарен, так как  $p\text{-value} < 0.05$ .

## Часть 3 (выполнил Козлов Д.)

### Определение порядка интегрированности

Используем тест Дики-Фуллера для разности первого порядка:

```
dataint <- diff(sourcedata$Value, differences=1)
```

```
adf.test(dataint)
```

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

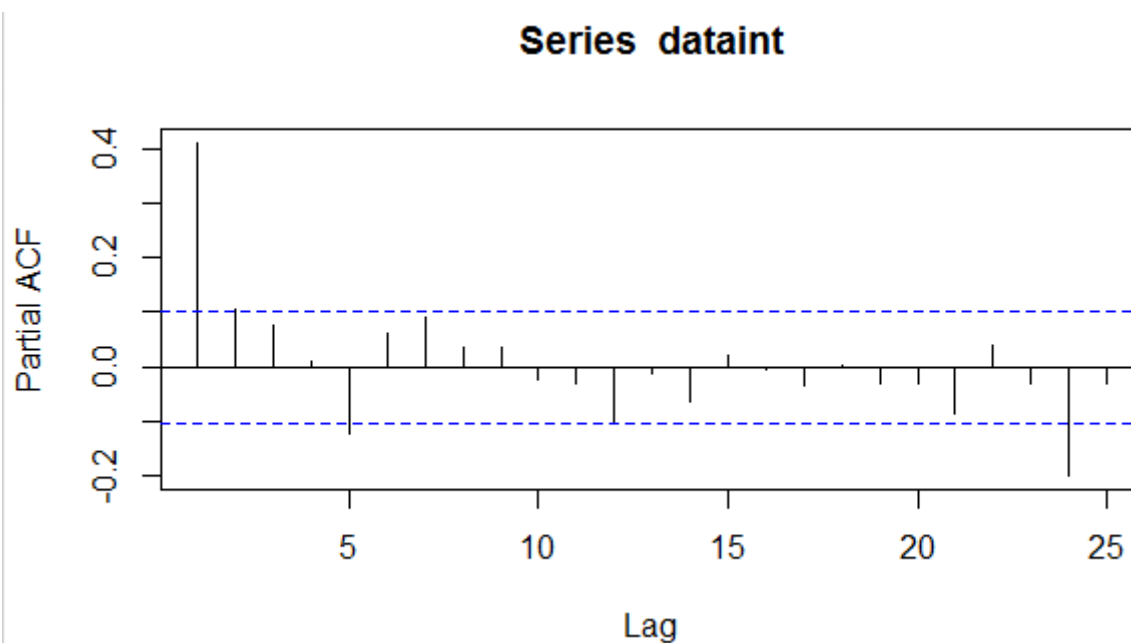
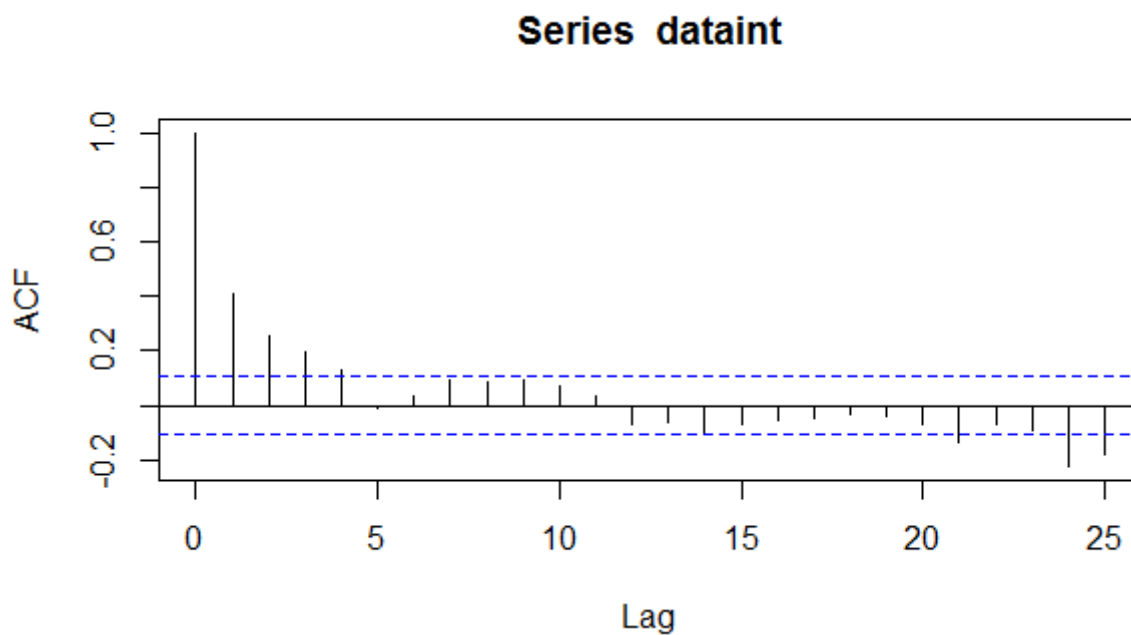
warning message:
In adf.test(dataint) : p-value smaller than printed p-value
```

Так как  $p\text{-value} < 0.05$ , то ряд разностей первого порядка стационарен. Следовательно, исходный ряд интегрирован с показателем 1.

### Применение модели ARIMA

Выберем параметры для модели ARIMA. Так как порядок интегрированности равен 1, то положим в модели Arima(a,b,c) параметр  $b=1$ .

Построим графики функции автокорреляции и частичной автокорреляции, пользуясь функциями `acf` и `pacf`.



Выберем параметры  $a=1$ ,  $c=0$  и  $a=0$ ,  $c=5$ .

Проведем исследования моделей и исследование достоверности прогноза:

```
Model1 <- Arima(sourcedata$Value, order=c(1,1,0))
```

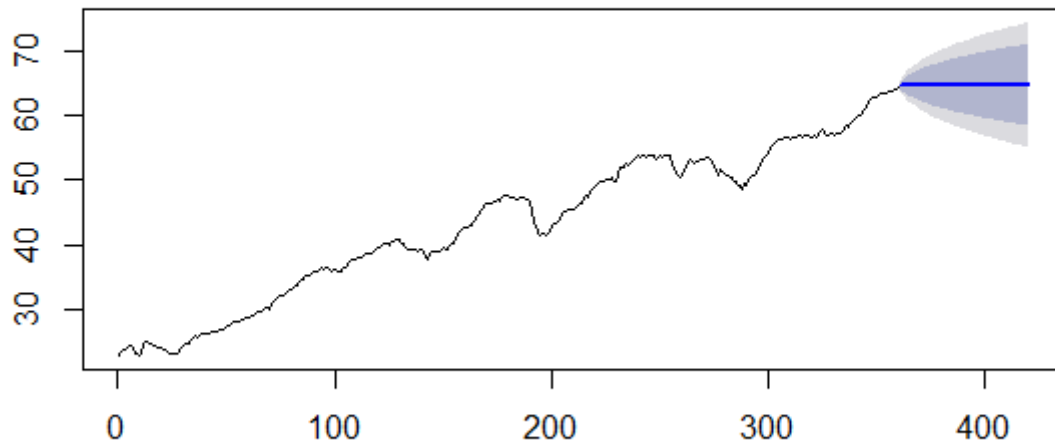
```
Model1forec <- forecast.Arima(Model1, h=60)
```

```
plot.forecast(Model1forec)
```

```
R2_Score(Model1forec$mean, testingdata$Value)
```

```
[1] -0.1347184
```

### Forecasts from ARIMA(1,1,0)



```
Model2 <- Arima(sourcedata$Value, order=c(0, 1, 5))
```

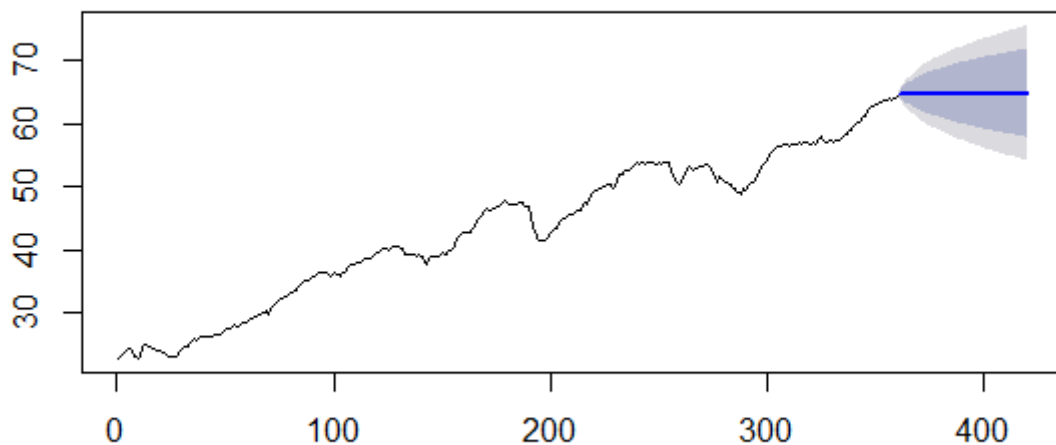
```
Model2forec <- forecast.Arima(Model2, h=60)
```

```
plot.forecast(Model2forec)
```

```
R2_Score(Model2forec$mean, testingdata$Value)
```

```
[1] -0.1030925
```

### Forecasts from ARIMA(0,1,5)



Сравним модели по критерию Акаике:

```
AIC(Model1, Model2)
```

Model1 2 263.2410

Model2 6 255.3061

Выбираем модель с меньшим коэффициентом Акаике, то есть Arima(1,1,0)

## **Дополнение. Код проекта на R**

### **Часть 1**

```
library("TTR")
```

```
#считываем данные
```

```
sourcedata <- read.csv(file="C:/Users/Lenovo/Desktop/training.csv", header=TRUE, sep=",")
```

```
plot(sourcedata[[1]], sourcedata[[2]])
```

```
#скользящие статистики
```

```
smastatistics <- TTR::SMA(x = sourcedata$Value, 20)
```

```
lines(x = sourcedata[[1]], y = smastatistics, col = 'Green')
```

```
emastatistics <- TTR::EMA(x = sourcedata$Value, 30)
```

```
lines(x = sourcedata[[1]], y = emastatistics, col = 'Red')
```

```
#тест Дики-Фуллера
```

```
adf.test(sourcedata$Value, alternative="stationary")
```

### **Часть 2**

```
tsseries <- ts(sourcedata$Value, frequency = 14, start = c(1959, 1))
```

```
#аддитивная модель
```

```
decompadd <- decompose(log(tsseries), type = c("additive"))
```

```
plot(decompadd, col = 'Blue')
```

```
adf.test(na.omit(decompadd$trend), alternative="stationary")
```

```
adf.test(na.omit(decompadd$seasonal), alternative="stationary")
```

```
adf.test(na.omit(decompadd$random), alternative="stationary")
```

```
#мультипликативная модель
```

```
decompmult = decompose(log(tsseries), type = c("multi"))
```

```
plot(decompmult, col = 'Blue')
```

```
adf.test(na.omit(decompmult$trend), alternative="stationary")
```

```
adf.test(na.omit(decompmult$seasonal), alternative="stationary")
```

```
adf.test(na.omit(decompmult$random), alternative="stationary")
```

### Часть 3

```
library("TTR")
```

```
library("forecast")
```

```
library(forecast)
```

```
library(MLmetrics)
```

```
sourcedata <- read.csv(file="C:/Users/Lenovo/Desktop/training.csv", header=TRUE, sep=",")
```

```
testingdata <- read.csv("C:/Users/Lenovo/Desktop/testing.csv", sep = ",")
```

```
# Тест Дики-Фуллера
```

```
dataint <- diff(sourcedata$Value, differences=1)
```

```
adf.test(dataint)
```

```
acf(x=dataint)
```

```
pacf(x=dataint)
```



```
# Применение модели Arima
```

```
Model1 <- Arima(sourcedata$Value, order=c(1,1,0))
```

```
Model1forec <- forecast.Arima(Model1, h=60)
```

```
plot.forecast(Model1forec)
```

```
R2_Score(Model1forec$mean, testingdata$Value)
```

```
Model2 <- Arima(sourcedata$Value, order=c(0, 1, 5))
```

```
Model2forec <- forecast.Arima(Model2, h=60)
```

```
plot.forecast(Model2forec)
```

```
R2_Score(Model2forec$mean, testingdata$Value)
```

```
AIC(Model1, Model2)
```