Липецкий государственный технический университет

Кафедра прикладной математики

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ МАТЕМАТИЧЕСКИХ ИССЛЕДОВАНИЙ

Лекция 9

Анализ временных рядов

Составитель - Сысоев А.С., к.т.н., доцент

Липецк - 2020

Outline

9.1. Задачи анализа временных рядов

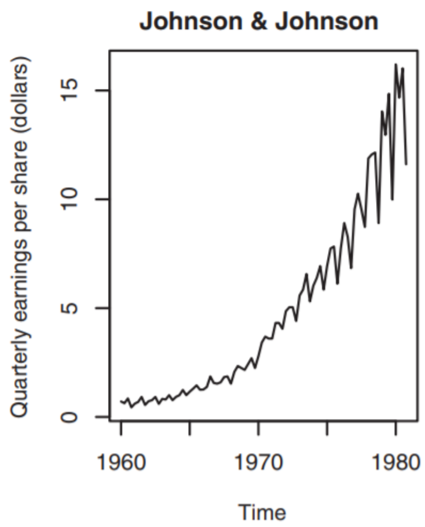
<u>Данные бывают:</u>

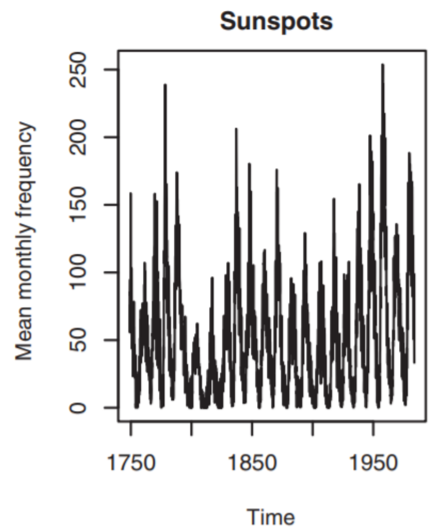
- Перекрестными – предполагается наблюдение за многими однотипными объектами с целью получения информации о процессе / объекте / системе.
- Долгосрочными (лонгитюдными) – изучается одна и та же группа объектов в течении долгого времени, за которое эти объекты успевают существенным образом поменять какие-либо свои значимые признаки.

Возможно выстроить результаты наблюдения таким образом, чтобы они образовывали временной ряд $Y_1, Y_2, Y_3, \ldots, Y_t, \ldots, Y_T$.

Анализ временных рядов строится для решения двух основных задач:

1. Исследовать то, что происходит (задача описания).
2. Сказать, что будет дальше (задача прогноза).





9.2. Создание объектов типа временной ряд в среде R

Необходимые составляющие объекта типа временной ряд:

- Непосредственно наблюдения,
- Время начала и конца рассматриваемого временного периода,
- Информация о периодичности (месячные данные, квартальные или годовые).

```
myseries <- ts(data, start=, end=, frequency=)
```

```
> sales <- c(18, 33, 41,  7, 34, 35, 24, 25, 24, 21, 25, 20,
             22, 31, 40, 29, 25, 21, 22, 54, 31, 25, 26, 35)
> tsales <- ts(sales, start=c(2003, 1), frequency=12)
> tsales
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2003  18  33  41   7  34  35  24  25  24  21  25  20
2004  22  31  40  29  25  21  22  54  31  25  26  35
> plot(tsales)
> start(tsales)
[1] 2003    1
> end(tsales)
[1] 2004   12
> frequency(tsales)
[1] 12
> tsales.subset <- window(tsales, start=c(2003, 5), end=c(2004, 6))
> tsales.subset
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2003                  34  35  24  25  24  21  25  20
2004  22  31  40  29  25  21
```

1 Creates a time-series object

2 Gets information about the object

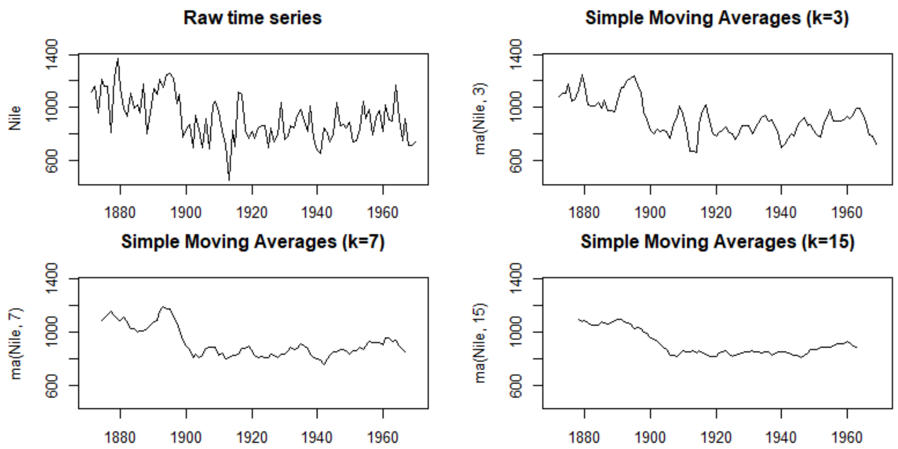
3 Subsets the object

9.3. Процедура скользящего среднего в среде R

Обычно временные ряды имеют значимую нерегулярную компоненту или ошибку.
Подход – использовать процедуру скользящего среднего.
Центрированное скользящее среднее (теряется (k – 1) / 2 значений)

$$S_t = (Y_{t-q} + \ldots + Y_t + \ldots + Y_{t+q}) \;/\; (2q + 1)$$

```
library(forecast)
opar <- par(no.readonly=TRUE)
par(mfrow=c(2,2))
ylim <- c(min(Nile), max(Nile))
plot(Nile, main="Raw time series")
plot(ma(Nile, 3), main="Simple Moving Averages (k=3)", ylim=ylim)
plot(ma(Nile, 7), main="Simple Moving Averages (k=7)", ylim=ylim)
plot(ma(Nile, 15), main="Simple Moving Averages (k=15)", ylim=ylim)
par(opar)
```

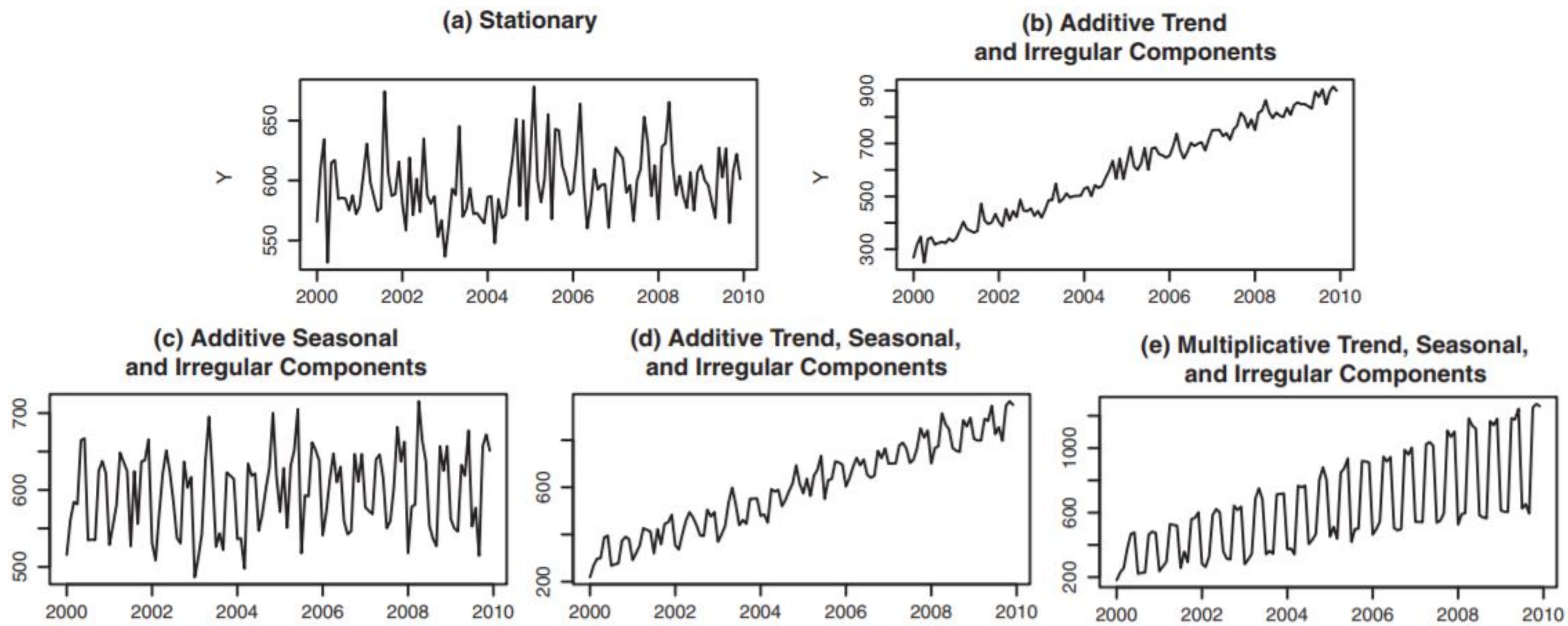


9.4. Сезонная декомпозиция в среде R

Временной ряд, у которого присутствует сезонная составляющая (месячная или квартальная периодичность), можно разложить на тренд, сезонную компоненту и ошибку.

Декомпозиция может быть представлена аддитивной или мультипликативной моделями.

$$Y_t = Trend_t + Seasonal_t + Irregular_t \qquad Y_t = Trend_t * Seasonal_t * Irregular_t$$



9.4. Сезонная декомпозиция в среде R

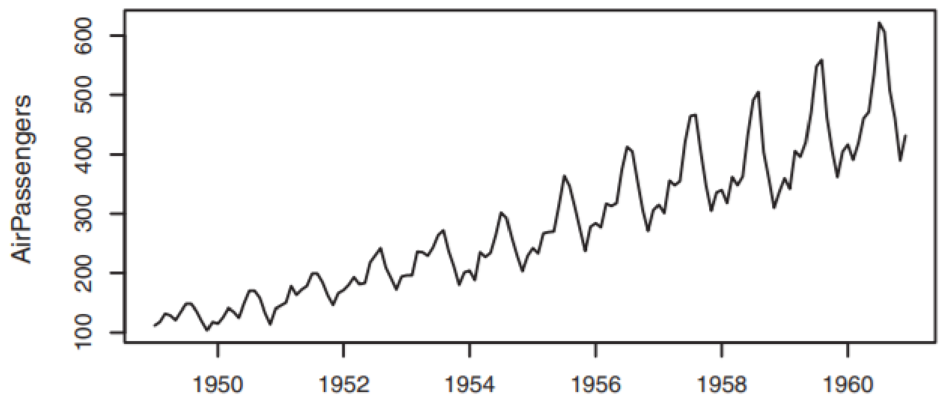
Популярный метод декомпозиции в R – loess (Local regrESSion).

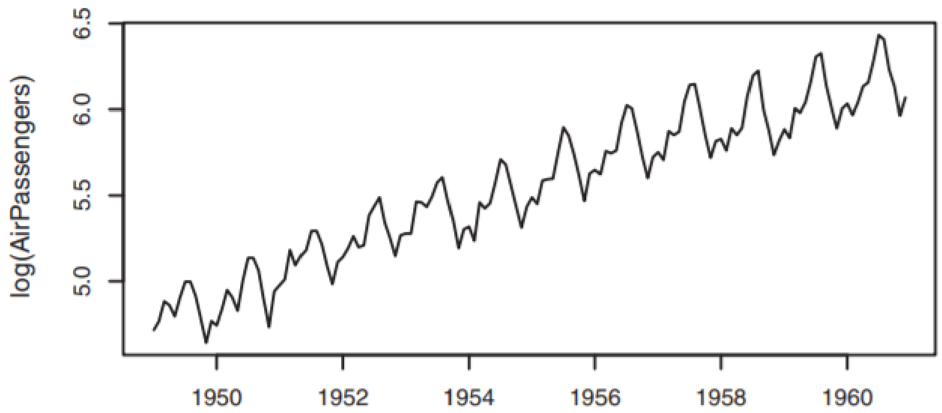
```
stl(ts, s.window=, t.window=)
```

Такой подход применяется только к аддитивным моделям. В случае мультипликативных моделей используется линеаризация:

$$\log(Y_t) = \log(Trend_t * Seasonal_t * Irregular_t) =$$

$$= \log(Trend_t) + \log(Seasonal_t) + \log(Irregular_t)$$





9.4. Сезонная декомпозиция в среде R

```
> plot(AirPassengers)
> lAirPassengers <- log(AirPassengers)
> plot(lAirPassengers, ylab="log(AirPassengers)")
```

❶ **Plots the time series**

```
> fit <- stl(lAirPassengers, s.window="period")
> plot(fit)
```

❷ **Decomposes the time series**

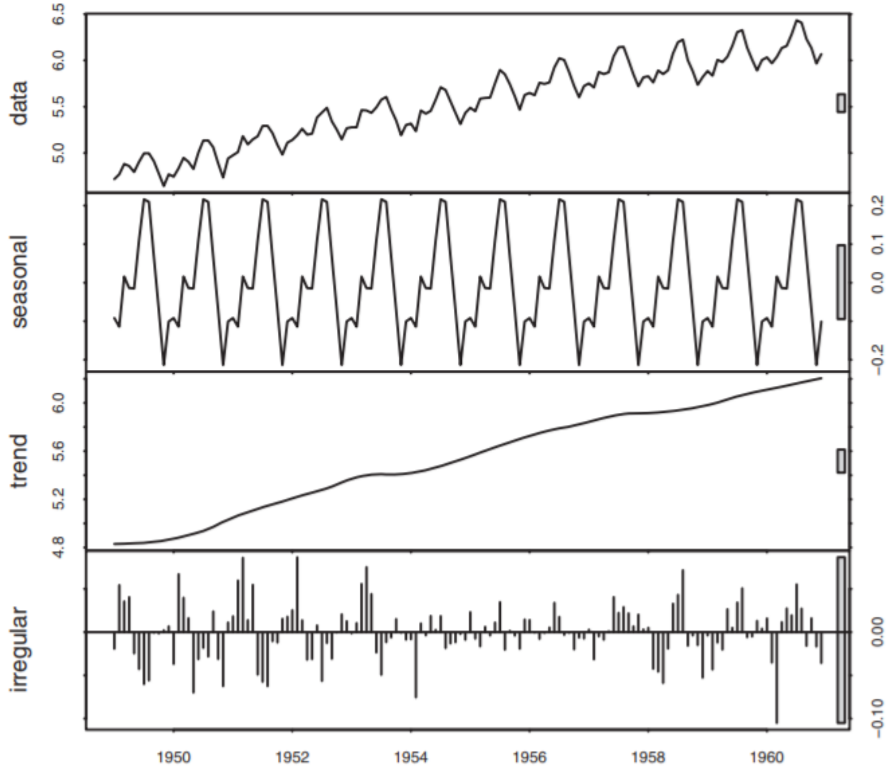
```
> fit$time.series
```

❸ **Components for each observation**

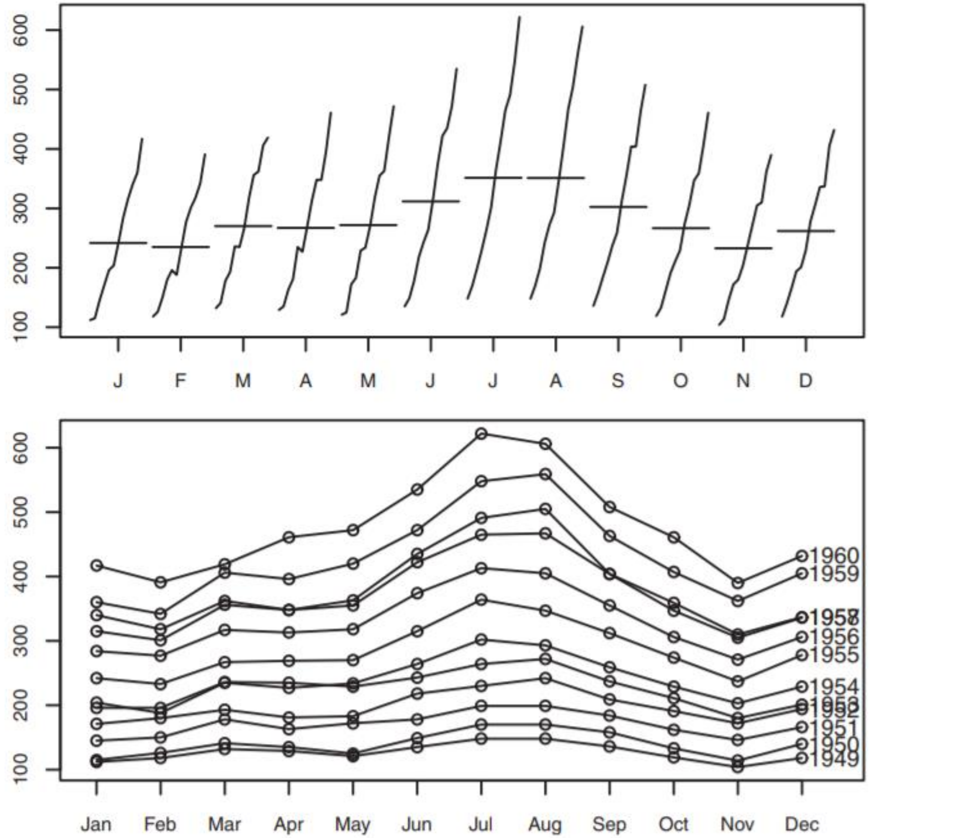
```
         seasonal trend  remainder
Jan 1949 -0.09164 4.829 -0.0192494
Feb 1949 -0.11403 4.830  0.0543448
Mar 1949  0.01587 4.831  0.0355884
Apr 1949 -0.01403 4.833  0.0404633
May 1949 -0.01502 4.835 -0.0245905
Jun 1949  0.10979 4.838 -0.0426814
Jul 1949  0.21640 4.841 -0.0601152
Aug 1949  0.20961 4.843 -0.0558625
Sep 1949  0.06747 4.846 -0.0008274
Oct 1949 -0.07025 4.851 -0.0015113
Nov 1949 -0.21353 4.856  0.0021631

Dec 1949 -0.10064 4.865  0.0067347
... output omitted ...
```

```
> exp(fit$time.series)

         seasonal trend remainder
Jan 1949   0.9124 125.1    0.9809
Feb 1949   0.8922 125.3    1.0558
Mar 1949   1.0160 125.4    1.0362
Apr 1949   0.9861 125.6    1.0413
May 1949   0.9851 125.9    0.9757
Jun 1949   1.1160 126.2    0.9582
Jul 1949   1.2416 126.6    0.9417
Aug 1949   1.2332 126.9    0.9457
Sep 1949   1.0698 127.2    0.9992
Oct 1949   0.9322 127.9    0.9985
Nov 1949   0.8077 128.5    1.0022
Dec 1949   0.9043 129.6    1.0068
... output omitted ...
```

9.4. Сезонная декомпозиция в среде R




```
par(mfrow=c(2,1))
library(forecast)
monthplot(AirPassengers, xlab="",  ylab="")
seasonplot(AirPassengers, year.labels="TRUE", main="")
```



9.5. Экспоненциальные модели прогнозирования временных рядов в среде R

Экспоненциальные модели, используемые для прогнозирования временных рядов:

- single exponential model (используется для рядов с постоянным уровнем и ошибкой, но без тренда и сезонной составляющей);
- double exponential model (Holt exponential smoothing) для рядов с постоянным уровнем и трендом.
- triple exponential model (Holt-Winters exponential smoothing) для рядов с постоянным уровнем, трендом и сезонной составляющей.

Функция `ets()` из пакета `forecast`:

```
ets(ts, model="ZZZ")
```

Доступные опции:

- A – аддитивные модели,
- M – мультипликативные модели,
- N – не указано,
- Z – автоматический выбор.

Type	Parameters fit	Functions
simple	level	ets(ts, model="ANN") ses(ts)
double	level, slope	ets(ts, model="AAN") holt(ts)
triple	level, slope, seasonal	ets(ts, model="AAA") hw(ts)

9.5. Экспоненциальные модели прогнозирования временных рядов в среде R

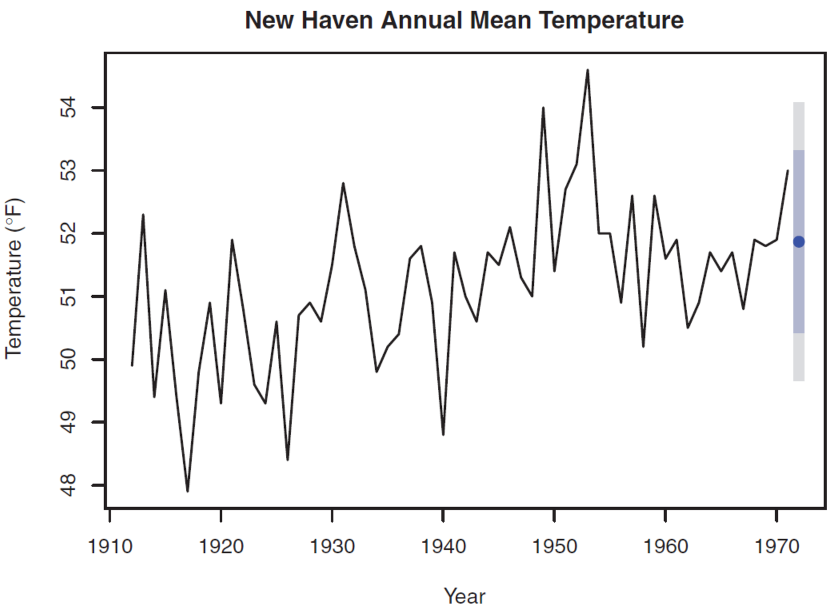
Простое экспоненциальное скользящее:

$$Y_t = \text{level} + \text{irregular}_t$$

Прогнозное значение для уровня $\mathbf{Y_{t+1}}$

$$Y_{t+1} = c_0 Y_t + c_1 Y_{t-1} + c_2 Y_{t-2} + c_2 Y_{t-2} + \ldots$$

$$c_i = \alpha(1-\alpha)^i,\ i = 0,\ 1,\ 2,\ \ldots \text{ and } 0 \le \alpha \le 1.$$




```
> library(forecast)
> fit <- ets(nhtemp, model="ANN")        ❶ Fits the model
> fit
ETS(A,N,N)
Call:
 ets(y = nhtemp, model = "ANN")
  Smoothing parameters:
    alpha = 0.182
  Initial states:
    l = 50.2759
   sigma:  1.126
   AIC  AICc   BIC
 263.9 264.1 268.1              ❷ l-step ahead forecast
 > forecast(fit, 1)
      Point Forecast Lo 80 Hi 80 Lo 95 Hi 95
 1972          51.87 50.43 53.31 49.66 54.08
 > plot(forecast(fit, 1), xlab="Year",
     ylab=expression(paste("Temperature (", degree*F,")",)),
     main="New Haven Annual Mean Temperature")
 > accuracy(fit)
                    ❸ Prints accuracy measures
                ME  RMSE    MAE    MPE  MAPE   MASE
 Training set 0.146 1.126 0.8951 0.2419 1.749 0.9228
```

`accuracy()`

Measure	Abbreviation	Definition
Mean error	ME	mean(e_t)
Root mean squared error	RMSE	sqrt(mean(e_t^2))
Mean absolute error	MAE	mean($\vert e_t \vert$)
Mean percentage error	MPE	mean(100 * e_t / Y_t)
Mean absolute percentage error	MAPE	mean($\vert$ 100 * e_t / Y_t $\vert$)
Mean absolute scaled error	MASE	mean($\vert q_t \vert$) where $q_t = e_t$ / (1/(T-1) * sum($\vert y_t - y_{t-1} \vert$)), T is the number of observations, and the sum goes from t=2 to t=T

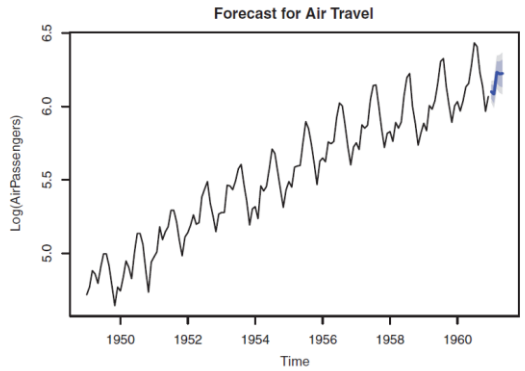
9.5. Экспоненциальные модели прогнозирования временных рядов в среде R

Экспоненциальное скользящее Хольта:

$$Y_t = \text{level} + \text{slope}^*t + \text{irregular}_t$$

Экспоненциальное скользящее Хольта-Винтерса:

$$Y_t = \text{level} + \text{slope}^*t + s_t + \text{irregular}_t$$




```
> library(forecast)
> fit <- ets(log(AirPassengers), model="AAA")
> fit
ETS(A,A,A)
Call:
 ets(y = log(AirPassengers), model = "AAA")
  Smoothing parameters:
    alpha = 0.8528
    beta  = 4e-04
    gamma = 0.0121
  Initial states:
    l = 4.8362
    b = 0.0097
    s=-0.1137 -0.2251 -0.0756 0.0623 0.2079 0.2222
           0.1235 -0.009 0 0.0203 -0.1203 -0.0925
  sigma:  0.0367
   AIC    AICc    BIC
-204.1 -199.8 -156.5
>accuracy(fit)
                     ME     RMSE     MAE       MPE   MAPE    MASE
Training set -0.0003695 0.03672 0.02835 -0.007882 0.5206 0.07532
```

❶ **Smoothing parameters**

```
> pred <- forecast(fit, 5)
> pred
         Point Forecast Lo 80 Hi 80 Lo 95 Hi 95
Jan 1961          6.101 6.054 6.148 6.029 6.173
Feb 1961          6.084 6.022 6.146 5.989 6.179
Mar 1961          6.233 6.159 6.307 6.120 6.346
Apr 1961          6.222 6.138 6.306 6.093 6.350
May 1961          6.225 6.131 6.318 6.082 6.367
```

❷ **Future forecasts**

```
> plot(pred, main="Forecast for Air Travel",
    ylab="Log(AirPassengers)", xlab="Time")
> pred$mean <- exp(pred$mean)
> pred$lower <- exp(pred$lower)
> pred$upper <- exp(pred$upper)
> p <- cbind(pred$mean, pred$lower, pred$upper)
> dimnames(p)[[2]] <- c("mean", "Lo 80", "Lo 95", "Hi 80", "Hi 95")
> p
          mean Lo 80 Lo 95 Hi 80 Hi 95
Jan 1961 446.3 425.8 415.3 467.8 479.6
Feb 1961 438.8 412.5 399.2 466.8 482.3
Mar 1961 509.2 473.0 454.9 548.2 570.0
Apr 1961 503.6 463.0 442.9 547.7 572.6
May 1961 505.0 460.1 437.9 554.3 582.3
```

❸ **Makes forecasts in the original scale**

9.6. Модели ARIMA в среде R

Lag	1869	1870	1871	1872	1873	1874	1875	...
0			1120	1160	963	1210	1160	...
1		1120	1160	963	1210	1160	1160	...
2	1120	1160	963	1210	1160	1160	813	...

Лаг – сдвиг влево временного ряда на заданное число наблюдений (`lag(ts,k)`)

Автокорреляция – оценивает силу связи, которой связаны уровни ряда с некоторым лагом.

График автокорреляционной функции `Acf(ts)`

Частная автокорреляция

Стационарность, разность

Дополненная процедура Дикки-Фуллера - `adf.test()` из пакета `tseries`

9.6. Модели ARIMA в среде R

Авторегрессионная модель порядка p

$$AR(p){:}Y_t = \mu + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \ldots + \beta_p Y_{t-p} + \varepsilon_t$$

Модель скользящего среднего порядка q

$$MA(q){:}Y_t = \mu - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} \ldots - \theta_q \varepsilon_{t-q} + \varepsilon_t$$

$\mathrm{ARMA}(p, q)$

$$Y_t = \mu + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \ldots + \beta_p Y_{t-p} - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} \ldots - \theta_q \varepsilon_{t-q} + \varepsilon_t$$

$\mathrm{ARIMA}(p, d, q)$ Находятся разности d раз. Прогнозные значения получены на основании p текущих значений и q предыдущих ошибок. Прогнозные значения «интегрированы».

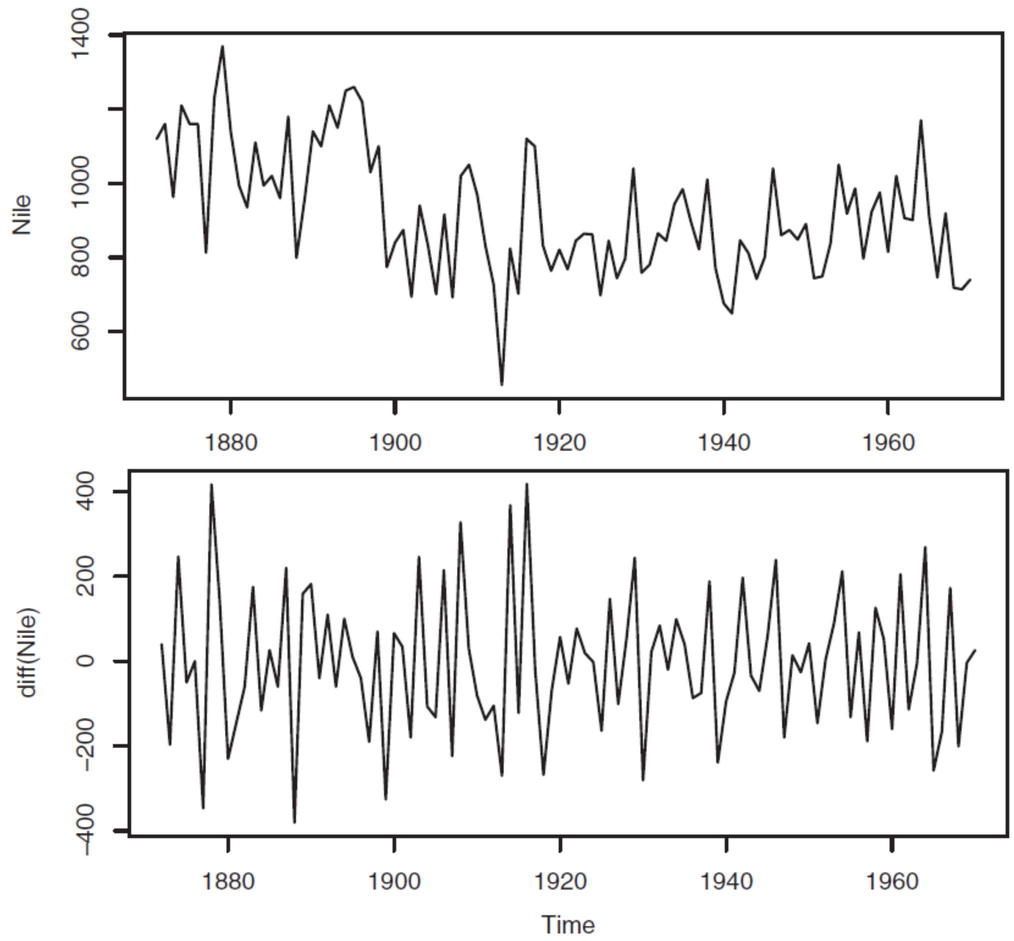
Применение модели ARIMA

1. Убедиться, что временной ряд стационарен.
2. Идентифицировать модель (выбрать p и q).
3. Подогнать модель.
4. Оценить модель.
5. Сделать прогноз.

9.6. Модели ARIMA в среде R

ПРОВЕРКА СТАЦИОНАРНОСТИ РЯДА

```
> library(forecast)
> library(tseries)
> plot(Nile)
> ndiffs(Nile)

[1] 1

> dNile <- diff(Nile)
> plot(dNile)
> adf.test(dNile)

    Augmented Dickey-Fuller Test

data:  dNile
Dickey-Fuller = -6.5924, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary
```

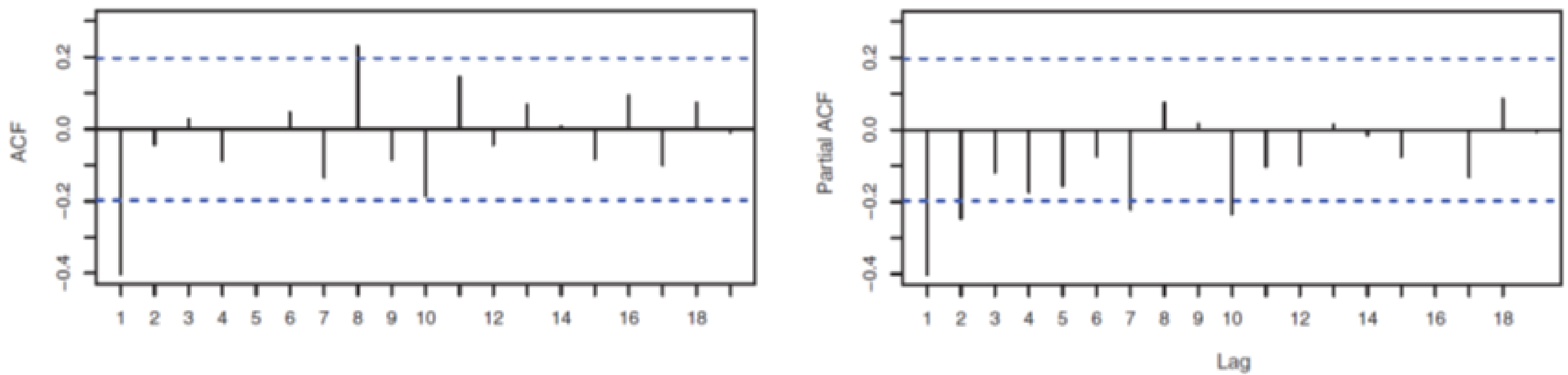


9.6. Модели ARIMA в среде R

ИДЕНТИФИКАЦИЯ МОДЕЛИ

Model	ACF	PACF
ARIMA(p, d, 0)	Trails off to zero	Zero after lag p
ARIMA(0, d, q)	Zero after lag q	Trails off to zero
ARIMA(p, d, q)	Trails off to zero	Trails off to zero

```
Acf(dNile)
Pacf(dNile)
```

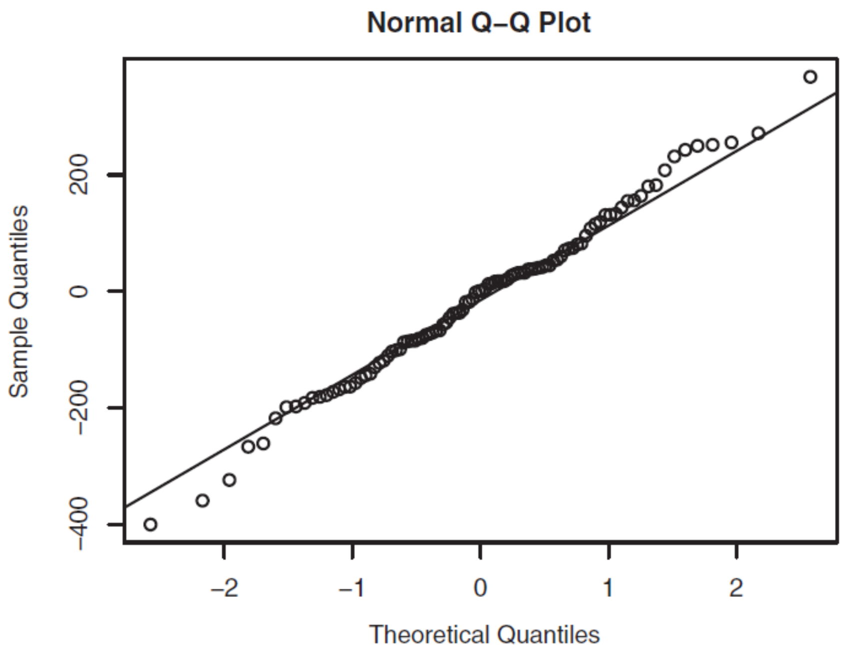


ПОДГОНКА МОДЕЛИ

```
> library(forecast)
> fit <- arima(Nile, order=c(0,1,1))
> fit

Series: Nile
ARIMA(0,1,1)

Coefficients:
          ma1
      -0.7329
s.e.   0.1143

sigma^2 estimated as 20600:  log likelihood=-632.55
AIC=1269.09   AICc=1269.22   BIC=1274.28

> accuracy(fit)

                 ME  RMSE   MAE    MPE  MAPE   MASE
Training set -11.94 142.8 112.2 -3.575 12.94 0.8089
```

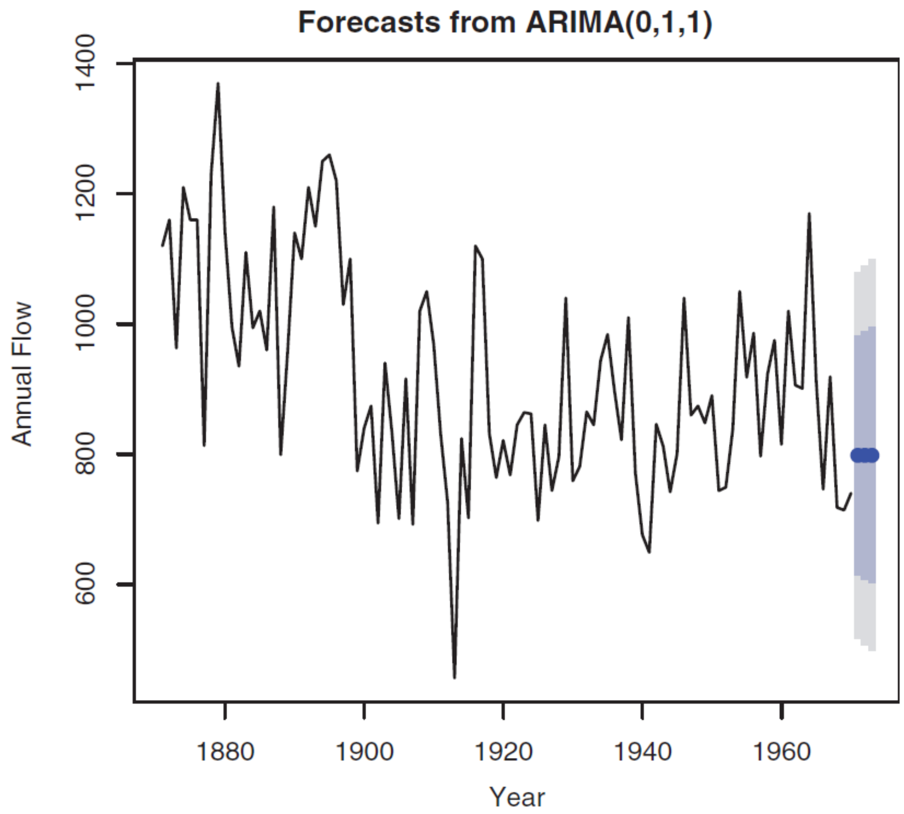
ОЦЕНКА МОДЕЛИ

```
> qqnorm(fit$residuals)
> qqline(fit$residuals)
> Box.test(fit$residuals, type="Ljung-Box")

    Box-Ljung test

data:  fit$residuals
X-squared = 1.3711, df = 1, p-value = 0.2416
```



ПРОГНОЗИРОВАНИЕ

```
> forecast(fit, 3)

     Point Forecast    Lo 80     Hi 80    Lo 95    Hi 95
1971       798.3673 614.4307  982.3040 517.0605 1079.674
1972       798.3673 607.9845  988.7502 507.2019 1089.533
1973       798.3673 601.7495  994.9851 497.6663 1099.068

> plot(forecast(fit, 3), xlab="Year", ylab="Annual Flow")
```



9.6. Модели ARIMA в среде R

Автоматическое прогнозирование модели ARIMA

```
> library(forecast)
> fit <- auto.arima(sunspots)
> fit
Series: sunspots
ARIMA(2,1,2)

Coefficients:
       ar1     ar2    ma1    ma2
      1.35  -0.396  -1.77  0.810
s.e.  0.03   0.029   0.02  0.019

sigma^2 estimated as 243:  log likelihood=-11746
AIC=23501   AICc=23501   BIC=23531

> forecast(fit, 3)


         Point Forecast       Lo 80    Hi 80     Lo 95    Hi 95
Jan 1984       40.437722  20.4412613 60.43418   9.855774 71.01967
Feb 1984       41.352897  18.2795867 64.42621   6.065314 76.64048
Mar 1984       39.796425  15.2537785 64.33907   2.261686 77.33116

> accuracy(fit)
                   ME RMSE   MAE MPE MAPE MASE
Training set -0.02673 15.6 11.03 NaN  Inf 0.32
```

Литература

Robert I. Kabacoff R in Action, Second Edition. Data analysis and graphics with R