Lecture 3

Testing Statistical Hypotheses

Data analysis is an important step in time series modeling and forecasting. It requires: obtaining and preparing the dataset, exploratory analysis of data series, performing statistical tests and the results' interpretation, the former being of major importance for all the other stages of analysis. Dealing with hydro-meteorological time series requires attention to data acquisition (measurement methodology and frequency), the series length (at least 50 years, in the case of studies concerning the climate change) and their completeness [90].

In this chapter, we focus on testing some statistical hypothesis and methods of detecting the long range dependence property in hydro-meteorological time series.

In what follows we denote the observed data by $(x_i)_{i=\overline{1,n}}$. They are realizations of a time series process, denoted by $(X_i)_{i=\overline{1,n}}$.

1. Normality tests

The Gaussian distribution is well-known, its properties being established for many years ago. It is a reference distribution to which one may report the experimental data to discover their properties. Testing the series normality is also important because many statistical methods rely on the hypothesis that the series are Gaussian.

The simplest way to check the null hypothesis (H_0: the series is normally distributed) against its alternative (H_1: the series is not normally distributed) is the use of graphical representations. One of them is the quantile - quantile plot (Q-Q plot) employed for deciding whether a univariate random sample comes from a given distribution *G*. The Q-Q plot is obtained by plotting quantiles of the sample against the theoretical quantiles of *G*. If the sample comes from the specified distribution (in our case, the Gaussian one) then the points are close to a straight line.

MINITAB, SPSS, R have the option to draw the Q-Q plot. We mention that R is freeware software.

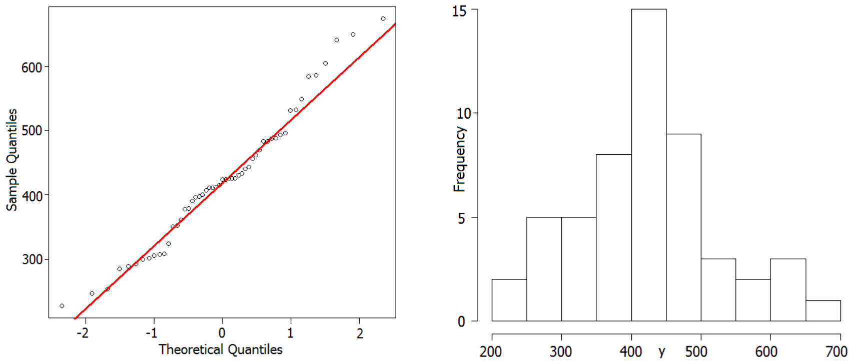
In the following we present the R code for obtaining the Q-Q plot of Constanta annual precipitation series (1961-2013).

```
data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",",
header=TRUE)
qqnorm(y); qqline(y, col = 2)
```

Looking at the chart (**Fig. 1**a.) we could not reject H_0. We couldn't also decide the opposite since the points from the upper part of the Q-Q plot are not very close to the line.

The same decision could be taken, looking at the histogram of the series (**Fig. 1**b.), which is a two-dimensional representation of the observed data against their frequency. The R-function used to create the histogram is:

```
hist(y, right=FALSE).
```



a. b.

Fig. 1. a. Q-Q plot and b. Histogram of Constanta annual precipitation series (1961-2013)

The decision to reject the normality hypothesis can be taken very easy for Constanta monthly series (1961-2013) because the plots significantly deviate from the straight line and the histogram is right - skewed (**Fig. 2**).

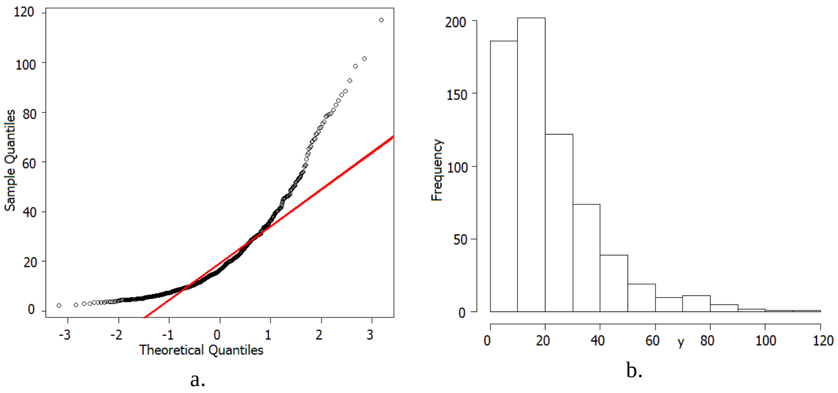


Fig. 2. a. Q-Q plot and b. histogram of Constanta monthly precipitation series (1961-2013)

Since the decision on data normality is still difficult for Constanta annual series, other methods, based on statistical tests, are more appropriate. Among the tests used for this aim we mention: Kolmogorov-Smirnov [63] [96], Jarque - Bera [57], Shapiro-Wilk [95], Lilliefors [67] (presented in detail in [7]), Anderson – Darling [2] and Cramer von Mises [98].

The Anderson - Darling test is used to determine if a dataset comes from a specified distribution (the normal one, in our case), so it is considered to be also a goodness of fit test. It compares the fit of an observed cumulative distribution function to the expected cumulative distribution function.

Considering the sample data $\{x_i\}_{i=\overline{1,n}}$, and $\{x_{(i)}\}_{i=\overline{1,n}}$ its values increasingly ordered, the Anderson - Darling statistic is defined by:

$$A^2 = -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)[\ln F(x_{(i)}) + \ln(1 - F(x_{(n-i+1)}))], \qquad (1.1)$$

where F is the cumulative distribution function of the specified distribution.

If Anderson-Darling is used as normality test, (1.1) becomes:

$$A^2 = -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)[\ln p_{(i)} + \ln(1 - p_{(n-i+10})],$$

where:

$$p_{(i)} = \Phi([x_{(i)} - \overline{x}]/s), \qquad (1.2)$$

Φ being the cumulative distribution function of the standard normal distribution, $\overline{x}$, the mean and s - the standard deviation of $\{x_i\}_{i=\overline{1,n}}$.

The Cramer-von Mises test for the composite hypothesis of normality is also based on the cumulative distribution function. The test statistic is:

$$W = \frac{1}{12n} + \sum_{i=1}^{n}\left(p_{(i)} - \frac{2i-1}{2n}\right),$$

where $p_{(i)}$ are given in (1.2).

The Pearson chi-square test for normality is applied to binned data so that the value of the statistic of the test depends on how the data was binned. Firstly, the data are standardized by subtracting the sample mean and dividing each value by the sample standard deviation. Then, the number of bins (k) is chosen by a formula (there is no optimal formula for this choice!), as, for example, $k = 1 + \log_2 n$, where n is the sample data.

The test statistic is:

$$\chi^2 = \sum_{i=1}^{k}(o_i - e_i)^2 / e_i,$$

where o_i is the observed frequency of the i^{th} bin, e_i is the expected frequency of the i^{th} bin, calculated by:

$$e_i = F(x_2) - F(x_1),$$

F is the cumulative distribution function of the distribution being tested, and x_1, x_2 are the limits of the i^{th} bin.

When the mean and variance are known, the test statistic is asymptotically χ^2 distributed with k-1 degrees of freedom. Therefore, the null hypothesis is rejected at a significance level α if the test statistic is greater than the quartile $\chi^2_{1-\alpha}(k-1)$.

Simulation studies showed that the normality tests have different powers. The relative powers of the discrete statistics tests of Kolmogorov–Smirnov, Cramér -von Mises, Anderson - Darling and Watson and of two test statistics for nominal data (chi-square and Kolmogorov - Smirnov) for an uniform null distribution against a selection of fully specified alternative distributions has been studied in [97]. The results show that the Pearson's chi-square and the nominal Kolmogorov-Smirnov are more powerful for the studied triangular, sharp and bimodal alternative distributions.

In the same idea, to compare the power of the Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests, the Monte Carlo procedure was employed in [87]. The results proved that the least powerful test is the Kolmogorov - Smirnov one, and the most powerful, Shapiro - Wilk, followed by Anderson - Darling. However, the Shapiro - Wilk test is biased if the sample size and the significance level are small (that is the power could be less than α). Due to its inferior power by comparison to the other tests, it is not advisable to use the Pearson chi-square test for checking the composite hypothesis of normality [76] [118].

In the following, the statistical tests are performed at a significance level of 0.05, if another level is not specified.

To perform normality tests using the R software, one has to install the packages fBasics [111] and nortest [118] and to write the following commands:

```
    library(fBasics)
    ksnormTest(y, title = NULL, description = NULL) # performs the Kolmogorov –Smirnov test
    jarqueberaTest(y, title = NULL, description = NULL) # performs the Jarque - Bera test
    shapiroTest(y, title = NULL, description = NULL) # performs the Shapiro - Wilk test
    library(nortest)
    lillieTest(y, title = NULL, description = NULL) # performs the Lilliefors test
    ad.test(y) # performs the Anderson - Darling test
    cvmTest(y, title = NULL, description = NULL) # performs the Cramer-von  Mises test
    pearson.test(y, n.classes = ceiling(2 * (n^(2/5))), adjust = TRUE)
    # performs the Pearson chi - square test, where:
        # n is the sample volume,
        # y is the vector containing the data values,
        # n.classes is the number of classes,
        # adjust - logical; if TRUE (default), the p-value is computed from a chi-square
#distribution with n.classes-3 degrees of freedom, otherwise from a chi-square distribution
# with n.classes-1 degrees of freedom.
```

Remarks. 1. *y* is a numeric vector containing a minimum of 3 and maximum of 5000 values. In our case, it contains 53, respectively 636 values.

2. For all tests, but Jarque - Bera, the R software provides the values of the test statistics and the p-values. For the Jarque - Bera test it returns $\chi^2(2)$ and the asymptotic p-value. For Cramer - von Mises and Anderson - Darling tests, the p -values are computed respectively from the statistics $Z = W(1+0.5/n)$ and $Z = A(1+0.75/n+2.25/n^2)$ [98]. The reader may also refer to [111] [118].

3. In the case of discrete distributions, the Kolmogorov - Smirnov and Cramer - von Misses tests can be performed in R using the functions ks.test() and cvm.test() from the package dgof [3]:

- ks.test() function supports one sample test for discrete null distributions. The second argument, *y*, can be an empirical cumulative distribution function (an R function with class "ecdf") or an object of class "stepfun" that specify a discrete distribution. For example:

```
     library(dgof)
    dgof::ks.test(c(1, 3), ecdf(c(2, 5)))
```

will return:

```
            One-sample Kolmogorov-Smirnov test
            data:  c(1, 3)
            D = 0, p-value = 1
            alternative hypothesis: two-sided
```

In the following, instead of writing "will return" before the results returned by R, we shall list only the results, after a white row and a Tab.

- The first two arguments of the function cvm.test() are the same as for ks.test(); the third one, type, specifies the variant of the Cramér-von Mises test used: W2 - is the default, U2 - for cyclical data, A2 - is the Anderson - Darling alternative.

We exemplify the application of this test on a sample of size 30, generated from the discrete uniform distribution on the natural numbers between 1 and 20:

```
x <- sample(1:20, 30, replace = TRUE)
cvm.test(x, ecdf(1:20))

        Cramer-von Mises - W2
        data:  x
        W2 = 0.0358, p-value = 0.947
        alternative hypothesis: Two.sided

cvm.test(x, ecdf(1:20), type= "A2")

        Cramer-von Mises - A2
        data:  x
        A2 = 0.2422, p-value = 0.9441
        alternative hypothesis: Two.sided

cvm.test(x, ecdf(1:20), type= "U2")

        Cramer-von Mises - U2
        data:  x
        U2 = 0.0275, p-value = 0.9386
        alternative hypothesis: Two.sided
```

The null hypothesis is rejected if the p-value is less than the significance level.

The results of some of the mentioned tests, applied to our data are given in **Table 1**. The normality hypothesis couldn't be rejected for the annual series, but it was rejected for the monthly one.

Table 1. Results of normality tests

Data		Jarque - Bera	Shapiro-Wilk	Lilliefors	Anderson-Darling	Cramer-von Mises
Annual	Stat.	1.3209	0.9722	0.0858	0.4672	0.073
	p-val	0.5166	0.2508	0.4284	0.2413	0.2503
Monthly	Stat.	765.6462	0.8324	0.1395	29.5533	0.998
	p-val	< 2.2e-16	2.2e-16	2.2e-16	2.2e-16	1

Minitab offers the possibility of choosing among three normality tests: Anderson - Darling, Kolmogorov - Smirnov (but in fact, it is the Lilliefors one) and Ryan - Joiner (analogous to Anderson - Darling). SPSS also offers two alternatives: the Lilliefors and Shapiro - Wilk tests.

2. Homoskedasticity tests

A sequence of random variables is said to be heteroskedastic if there are subsequences whose variances differ from the others. The heteroskedasticity is the opposite of homoscedasticity.

Many statistical tests have been proposed [9][16][48][78] for testing the homoscedasticity in the hypothesis of data normality, or in regression models [14] [16] [44] [45] [107] *etc*. The null and alternative hypotheses in these tests are respectively:

$$\mathrm{H}_0\text{: } \sigma_1^2 = \sigma_2^2 = \ldots = \sigma_k^2,$$

$$\mathrm{H}_1\text{: } \sigma_i^2 \neq \sigma_j^2 \text{, for at least one pair } (i, j),$$

where k is the number of groups and σ_i^2 is the variance of the i^{th} group.

The Bartlett test is the most used in applications that don’t involve different types of regressions [9]. Since it is sensitive to the data departures from normality, other tests have been developed; among them, the Levene one [65], whose statistics is [7]:

$$W = \frac{n-k}{k-1} \cdot \frac{\sum_{i=1}^{k} n_i (\overline{Z_{i\bullet}} - \overline{Z_{\bullet\bullet}})^2}{\sum_{i=1}^{k} \sum_{j=1}^{n_i} (Z_{ij} - \overline{Z_{i\bullet}})^2},$$

where n is the total number of observations, n_i is the number of observations in the group i, Z_{ij} is the absolute value of the deviation of a value (X_{ij}) in a group from the group’s mean ($\overline{X_{i\bullet}}$), $\overline{Z_{i\bullet}}$ is the average of the values from the group i, $\overline{Z_{\bullet\bullet}}$ is the overall mean of Z_{ij}.

The null hypothesis is rejected at the significance level α if the p-value is less than α .

In the robust version of the Levene test, proposed by Brown - Forsythe [16], the group mean is replaced the group median.

Analyzing the sensitivity of these to the samples’ lengths, O'Brien [80] introduced a correction factor. Hines [51] also proposed a method for improving the Brown - Forsythe test, which takes into account that the possible linear relationships (structural zeroes) of the investigated data were not previously investigated. Noguchi and Gel [79] introduced a new correction factor based on a combination of [51] and [80].

Levene’s test and its versions are implemented in R lawstat package [116].

```
library(lawstat) #loads the library lawstat
data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013_grupuri.csv", sep= "," , header=TRUE) #loads the file containing the monthly data
x<-data[,1]
y<-data[,2]
levene.test(x, y, location="mean") # performs Levene’s test

    data:  x
    Test Statistic = 3.0043, p-value = 0.05028

levene.test(x, y, location="median", correction.method="none") # performs the Brown – # Forsythe test

    data:  x
    Test Statistic = 2.314, p-value = 0.0997

levene.test(x, y, location="mean", correction.method="correction.factor") # performs  the Levene test with the correction factor [80].

    data:  x
    Test Statistic = 3.0135, p-value = 0.04982

levene.test(x, y, location="median", correction.method="zero.correction") # performs the # Brown-Forsythe test with modified structural zero removal method and correction factor #[80].

    data:  x
    Test Statistic = 2.3308, p-value = 0.09806.
```

Hartley's $F_{\max}$ [48] is a test for homogeneity of variances, used if all the samples have the same size and the data in each group are normally distributed. The $F_{\max}$ statistics is computed as the ratio of the largest and smallest variances of the groups. Tables for this test are due to David [27] and Nelson [77].

Cochran's test [23] was developed for testing the homogeneity of groups’ variances against the hypothesis that the highest variance is different from the others if all the k groups contain the same number of data. The test statistic is:

$$Q = \frac{\max_{1 \le i \le k} s_i^2}{\sum_{i=1}^{k} s_i^2},$$

where s_i^2 the variance of the i^{th} group.

The homogeneity hypothesis is rejected at the significance level, α , if Q > $Q_{\alpha;k,n-1}$, where $Q_{\alpha;\,k,n-1}$ is the critical value from the tables of Cochran test, function of the sample volume, *n*, the number of groups, *k*, and the significance level [7].

Other autocorrelation tests have been designed for testing special types of heteroskedasticity as: the Breush-Pagan test [14], that assumes the heteroskedasticity as a result of a linear combination of independent variables or the White test [107], which is based on the regression of the squared residual from a given model onto the initial regressors, their squared product and cross-products.

The Breusch - Pagan and White tests are implemented in R in lmtest package [117], and het.test package [115], respectively.

3. Autocorrelation tests

The autocorrelation (or serial correlation) refers to the correlation between the time series values with its previous/future values.

If (X_t) is a time series and $\gamma(h) = Cov(X_t, X_{t+h})$ is the *autocovariance function* of (X_t) at lag *h* ($h \in \mathbf{N}^*$), then *the autocorrelation function of* (X_t) *at lag h* is defined by $\rho(h) = \gamma(h)/\gamma(0)$, $h \in \mathbf{N}^*$.

The most used estimator of the autocorrelation function is *the empirical autocorrelation function* (ACF), defined by:

$$\hat{\rho}(h) = \frac{\sum_{t=1}^{n-h}(x_t - \overline{x})(x_{t+h} - \overline{x})}{\sum_{t=1}^{n}(x_t - \overline{x})^2},$$

where $(x_i)_{i=\overline{1,n}}$ are observed values of (X_t) and $\overline{x}$ is the average of $(x_i)_{i=\overline{1,n}}$.

The chart of ACF is called *correlogram*.

For taking the decision about the existence of the autocorrelation in a time series, the confidence interval at a selected confidence level (usually 0.95 or 0.99) is also computed, and the correlogram is analyzed. If all the values of the autocorrelation function are inside the empirical confidence interval, one can reject the autocorrelation hypothesis.

The Durbin - Watson test [33] [34] [35] is widely used for checking the null hypothesis of correlation's absence against the first order autocorrelation of resi -duals in the regression analysis. It is based on the hypothesis that the errors in the regression model are generated by an AR(1) process.

The test is inconclusive in the interval (d_l, d_u) , whose limits, d_l, d_u , are specified in the tables of the test, at the specified significance level. Also, it cannot be used for testing the residuals' autocorrelation when there are lagged endogenous variables among the exogenous variables. To overcome this drawback, Durbin [32] introduced the *h*- test, whose statistics is:

$$h = \left(1 - \frac{d}{2}\right)\sqrt{\frac{n}{1 - n\sigma^2}},$$

where σ^2 is the estimated variance of the coefficient corresponding to the lagged dependent variable, β, *n* is the sample size and

$$d = \sum_{t=2}^{n}(x_t - x_{t-1})^2 \Big/ \sum_{t=1}^{n} x_t^2$$

is the Durbin - Watson statistics.

The Durbin - Watson test is implemented in lmtest [117] package in R.

Other procedures, that will not be discussed here, as the Cochrane - Orcutt test [24], the balisage method of Hildreth - Lu [72] or generalized differencing, have also been proposed for dealing with serial correlation of errors in regression models.

The correlogram analysis is a good option for searching for a linear dependent structure in a time series; otherwise, different autocorrelation tests must be used. In the following, we shortly present the so-called *portmanteau tests*, designed for testing the null hypothesis that the residuals in a model form a white noise.

Box and Pierce [13] portmanteau test was built for testing the hypothesis that the residual in an ARMA(p, q) model is a white noise. It is based on the statistics:

$$Q_{BP} = n\sum_{k=1}^{h}\hat{\rho}_k^2,$$

where *n* is the sample size, $\hat{\rho}_k$ is the sample autocorrelation of order *k*, and *h* is the number of lags.

Since simulation studies showed low performances of this statistics, different improvements have been proposed. One of them is that of Ljung and Box [70], whose statistics is defined by:

$$Q_{LB} = n(n+2)\sum_{k=1}^{h}\frac{\hat{\rho}_k^2}{n-k}.$$

If $Q_{LB} > \chi^2_{1-\alpha,h-p-q}$, then the null hypothesis of residuals' independence is rejected, where $\chi^2_{1-\alpha,h-p-q}$ is the α - quantile of the chi - square distribution with $h-p-q$ degrees of freedom.

This test can be also used to test the hypothesis of independence for any time series, replacing $h-p-q$ by *h*.

Other portmanteau tests have been introduced by:

- Monti [75], whose statistics is:

$$Q_M = n(n+2)\sum_{k=1}^{h}\frac{\hat{\tau}_k^2}{n-k},$$

where $\hat{\tau}_k^2, 1 \le k \le h$ is the value of the partial autocorrelation function of residuals (see the next chapter for definition);

- Li and McLeod [66], whose statistics is:

$$Q_{ML} = n(n+2)\sum_{k=1}^{h}\frac{\hat{\rho}_{aa}^2(k)}{n-k},$$

where $\hat{u}_t$ are the estimated residuals in an ARMA model,

$$\hat{\rho}_{aa}(k) = \frac{\sum_{t=1}^{n-h}(\hat{u}_t^2 - \overline{\mu}^2)(\hat{u}_{t+h}^2 - \overline{\mu}^2)}{\sum_{t=1}^{n}(\hat{u}_t^2 - \overline{\mu}^2)^2},$$

and $\overline{\mu}$ is the mean of $\hat{u}_t^2$.

Q_M and Q_{ML} follow asymptotic chi - square distributions with *m* - *p* - *q* and *m* degrees of freedom, respectively, in the hypothesis that the fitted ARMA model is correct. For more details on these tests, see [4].

The Box-Pierce, Ljung-Box and Li and McLeod tests are implemented in portes package [120] in R.

In the following, we illustrate the use of a part of these tests on the monthly precipitation data and the residual from the ARMA(1,1) model.

Firstly the package must be installed, via 'Install Package'. Then the following instructions must be written for performing the Box - Pierce test on the data series:

```
    library(portes) #loads the library 'portes'
    data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",",
header=TRUE) # loads the file containing the monthly data
    x<-data
    y<-ts(x) # defines the time series y using the input data
    BoxPierce(y) # performs the Box - Pierce test on the data
```

```
        Lags   Statistic        df    p - value
          5     4.205184         5    0.5202686
         10     8.874683        10    0.5440365
         15    11.503326        15    0.7161672
         20    15.415612        20    0.7521529
         25    17.554509        25    0.8607075
         30    17.971275        30    0.9589969
```

```
    LjungBox(y) # performs the Ljung - Box test on the data
```

```
      Lags      Statistic       df    p - value
        5        4.67418         5    0.4569221
       10       10.34766        10    0.4105394
       15       13.93453        15    0.5305002
       20       20.21133        20    0.4447819
       25       24.04476        25    0.5167882
       30       24.96289        30    0.7268334
```

```
    fit <- arima(x, c(1, 0, 1)) ) # fits and ARMA(1,1) model to the data
    lags <- c(5, 10, 15, 20,35, 30)  # defines the lags
    res <- resid(fit) # determines the residual in the ARMA(1,1) model
    BoxPierce(res, lags, order = 2) # applies the Box-Pierce test on the residual of ARMA(1,1)
# model; specification of the order is required for computing the degrees of freedom of
# asymptotic chi - square distribution. Generally, it is equal to the number of estimated
# parameters in the model.
```

```
          Lags    Statistic      df       p - value
           5       4.385739       3        0.2227114
           10      7.820398       8        0.4512081
           15     10.363177      13        0.6640005
           20     13.708626      18        0.7478784
           25     14.973888      23        0.8955670
           30     15.408471      28        0.9737939
```

```
    LjungBox(res, lags, order = 2) # performs the Ljung - Box test on the residual of
# ARMA(1,1) model
```

```
          Lags    Statistic         df      p - value
          5         4.997150         3      0.1720060
          10        9.180725         8     0.3272805
          15      12. 647649        13      0.4753816
          20      17.974094         18      0.4573605
          25      20.271432         23      0.6254872
          30      21.236837         28      0.8153370
```

```
    LiMcLeod(res, lags, order = 0, SquaredQ=TRUE) # applies the Li - Mc Leod test on the
# residual of ARMA(1,1) model. Since the study object is issued from an ARMA model, the
# order' specification is not necessary, being automatically computed. If  SquaredQ = TRUE, # the statistics value in the Li - McLeod test is computed by squared values because it checks # for ARCH effects. If  SquaredQ = FALSE, the residual are used for the computation of the # test - statistics.
```

```
          Lags     Statistic        df        p - value
           5         4.300665        5        0.5069880
          10         9.692352       10        0.4678862
          15       14.995210        15        0.4517623
          20       19.148617         20       0.5121855
          25       24.844189         25       0.4711326
          30       30.578967         30       0.4363097
```

The tests' results reject the autocorrelation hypothesis of the data or residual in the fitted model.

Other statistical tests for serial independence are based on ranks, on empirical copulae, on divergent measures, spectral theory, information theory, *etc*. For the classification of these methods, the reader may refer to [29].

Practical needs of working with many random processes or with multivariate data series lead to the development of statistical independence tests based on multivariate extensions of Spearman's ρ - test [39] [42] [104]. They also imply the development of other nonparametric [6] [12] [28] [85] [88], parametric [99] or symbolic [74] techniques for testing the statistical independence.

In one of the following chapters we shall use the Szekely - Rizzo - Bakirov test (SRB) for independence [100]. We shortly present it in the following.

For any two multivariate random variables (random vectors) $X \in \mathbf{R}^p$ and $Y \in \mathbf{R}^q$ with finite expectations, the *distance covariance* $v^2(X,Y)$ and the *distance correlation* $R^2(X,Y)$ are defined in [100] as:

$$v^2(X,Y) = \| f_{XY} - f_X f_Y \|^2$$

$$R^2(X,Y) = \begin{cases} 0 & ; \nu^2(X)\nu^2(Y) = 0 \\ \dfrac{\nu^2(X,Y)}{\sqrt{\nu^2(X)\nu^2(Y)}} & ; \nu^2(X)\nu^2(Y) \neq 0 \end{cases}$$

where f_X, f_Y, f_{XY} are respectively the characteristic functions of *X*, *Y* and the joint characteristic function of (*X*, *Y*).

X and *Y* are statistical independent if:

$$f_{XY} = f_X\, f_Y\,,\ v^2(X,Y)=0 \text{ and } R^2(X,Y)=0.$$

Practically, when the analytical expressions of the two distributions are not known, the only way of knowing *X* and *Y* is by recording their samples of the same length *n*, registered as matrices $\mathbf{X}_n \in \mathbf{R}^{n\times p}$, $\mathbf{Y}_\mathrm{n} \in \mathbf{R}^{n\times q}$. When *n* is big, the sample matrices ($\mathbf{X}_\mathrm{n}$, $\mathbf{Y}_\mathrm{n}$) = $\{(X_k,\ Y_k)|k\in\overline{1,n}\} \in \mathbf{R}^{n\times p}\times\mathbf{R}^{n\times q}$ becomes representative for the variables (*X*, *Y*) that produce them.

Therefore, the *empirical distance covariance* (EDCov, $v_n^2(X,Y)$), and the *empirical distance correlation* (EDCor, $R_n^2(\mathbf{X}_n,\mathbf{Y}_n)$) are introduced [100] using only the experimental data:

$$R_n^2(\mathrm{X}_n,\mathrm{Y}_n)=\begin{cases}0 & ,\ v_n^2(\mathrm{X}_n)v_n^2(\mathrm{Y}_n)=0\\ \dfrac{v_n^2(\mathrm{X}_n,\mathrm{Y}_n)}{\sqrt{v_n^2(\mathrm{X}_n)v_n^2(\mathrm{Y}_n)}} & ,\ v_n^2(\mathrm{X}_n)v_n^2(\mathrm{Y}_n)\neq 0\end{cases},$$

respectively

$$v_n^2(\mathrm{X}_n,\mathrm{Y}_n)=\frac{1}{n^2}\sum_{k,l=1}^{n}A_{kl}B_{kl}\,,$$

where

$$\mathrm{v}_n^2(\mathrm{X}_n)=\mathrm{v}_n^2(\mathrm{X}_n,\mathrm{X}_n),\ \mathrm{v}_n^2(\mathrm{Y}_n)=\mathrm{v}_n^2(\mathrm{Y}_n,\mathrm{Y}_n)\,,$$

$$A_{kl}=a_{kl}-\overline{a}_{k\bullet}-\overline{a}_{\bullet l}+a_{\bullet\bullet}\,,\ B_{kl}=b_{kl}-\overline{b}_{k\bullet}-\overline{b}_{\bullet l}+b_{\bullet\bullet}\,,$$

$$a_{kl}=\left\|X_k-X_l\right\|_p\,,\ b_{kl}=\left\|Y_k-Y_l\right\|_q\,,$$

$$\overline{a}_{k\bullet}=\frac{1}{n}\sum_{l=1}^{n}a_{kl}\,,\ \overline{b}_{k\bullet}=\frac{1}{n}\sum_{l=1}^{n}b_{kl}\,,\ \overline{a}_{\bullet l}=\frac{1}{n}\sum_{k=1}^{n}a_{kl}\,,\ \overline{b}_{\bullet l}=\frac{1}{n}\sum_{k=1}^{n}b_{kl}\,,$$

$$a_{\bullet\bullet}=\frac{1}{n^2}\sum_{k,l=1}^{n}a_{kl}\,,\ b_{\bullet\bullet}=\frac{1}{n^2}\sum_{k,l=1}^{n}b_{kl}\,.$$

It was proved [100] that:

$$v_n^2(\mathrm{X}_n,\mathrm{Y}_n)=S_1^n+S_2^n-2S_3^n=\left\|f_{XY}^n-f_X^n f_Y^n\right\|^2,$$

where f_X^n, f_Y^n, f_{XY}^n are respectively the characteristic functions of $\mathbf{X}_n$, $\mathbf{Y}_n$ and the joint characteristic function of ($\mathbf{X}_n$, $\mathbf{Y}_n$). S_1^n, S_2^n, S_3^n are computed in terms of L_p, L_q norms related to $\mathbf{X}_n$ and $\mathbf{Y}_n$.

Also:

$$\lim_{n\to\infty} v_n^2(\mathrm{X}_n,\mathrm{Y}_n)=v^2(X,Y)\ \textit{a.s./a.e.};$$

$$\lim_{n\to\infty} R_n^2(\mathrm{X}_n,\mathrm{Y}_n)=R^2(X,Y)\ \textit{a.s./a.e.},$$

so the statistical independence of *X* and *Y* is proved if $\lim_{n\to\infty} v_n^2(\mathrm{X}_n,\mathrm{Y}_n)=0$ *a.e.*

Practically, applying SRB means the computation of $v_n^2(\mathbf{X}_n,\mathbf{Y}_n)$, followed by testing for EDCor convergence.

The implementation of the procedure described above has been done by Maria L. Rizzo and Gabor J. Szekely, in the R package energy [114] . We present here the results of its application to two matrices containing five annual precipitation series.

```
    data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Anuale_carte.csv", sep=",",
header=TRUE) # reads the data containing ten annual precipitation series
    library(energy) # loads the library
    x<- data[1:41,2:6] # defines the first matrix, containing the first data series
    y<-data[1:41,7:11] # defines the second matrix, containing the last five data series
    DCOR(x, y) # computes the empirical distance covariance EDCov, the empirical distance # correlation variance, distance variance of
x and distance variance of x

        $dCov
```

```
[1] 104.9466

$dCor
[1] 0.8806615

$dVarX
[1] 135.1349

$dVarY
[1] 105.0874
```

4. Outliers’ detection

An outlier (aberrant value) is a value that appears to deviate markedly from other members of the sample in which it occurs [8], or that differs so much from the other observations such as there are suspicions that it was produced by a different mechanism [49].

There are different methods for outliers’ detection, which can be classified, for example, as: statistical-based approaches (parametric - Gaussian model-based, regression model-based – or nonparametric – histogram-based, kernel-based), nearest neighbor-based, clustering-based (SVM-based, Bayesian network-based), classification-based and spectral decomposition - based (Principal Component Analysis - based) approaches [20] [108].

In outlier analysis, different domains of data require specific detection techniques. Temporal outlier analysis studies the aberrant values of the data across time. Given a time series, one can find particular elements as outliers (point outliers, generally referred as outliers), or subsequence outliers. In our study, we shall discuss only the first type of outliers.

The most popular univariate parametric methods rely on the assumption that data distribution is known, and the process that produces them is independent, identically distributed or Gaussian. Overviews of techniques for outliers’ detection have been provided by many authors [1] [20] [52]. Here we shortly present some statistical methods.

Dixon’s method [30], developed in 1950, is used for samples with small dimensions (up to 40) and is based on order statistics. Different ratios are defined for identification of the potential outliers, function of the number of presumed aberrant values. A value is considered to be an outlier when the corresponding statistics value is greater than the critical value of the test.

Grubbs’ method [46] is used for testing the null hypothesis that there is no outlier in the data series $(x_i)_{i=\overline{1,n}}$, against the alternative that there is at least an outlier, based on the statistics:

$$G = \max_{i=\overline{1,n}} \frac{\left|x_i - \overline{x}\right|}{s},$$

where $\overline{x}$ is the sample mean and s is the standard deviation of the $(x_i)_{i=\overline{1,n}}$.

If at a chosen significance level, α, *G* is higher than the critical value for Grubbs’ test, then the null hypothesis is rejected and the corresponding x_i can be accepted as an outlier.

Note that the test assumes data normality.

The Tietjen - Moore test is a generalized version of Grubbs’ method. It is designed for the detection of multiple outliers in a univariate data series which is approximately normally distributed, when the suspected number of outliers, *k*, is correctly specified. More precisely, the hypothesis that no outlier exists in the data series is tested against the alternative that there exist exactly *k* outliers. After sorting the sample in ascending order, different test statistics are defined, for the *k* largest points, the *k* smallest points or for testing the outliers’ existence in both tails. The values of the test statistics are in the interval [0, 1]. In the presence of outliers, the value is close to zero and in their absence, it is 1. The computation of the test critical region is done by simulation.

The main drawback of this test is that the number of outliers must be exactly specified. To surpass this inconvenience, *the generalized ESD test* (Extreme Studentized Deviate) [91] can be used. In this test, the hypothesis that there is no outlier in the data series is tested against that of the existence of up to *r* outliers. It is also supposed that the process generating the data series follows an approximately Gaussian distribution. After computing the statistics

$$R_i = \max_{i=\overline{1,n}} \frac{\left|x_i - \overline{x}\right|}{s},$$

the observation that maximizes $\left|x_i - \overline{x}\right|$ is removed and R_i is recomputed with the remained data. The process is repeated until *r* observations have been removed. Corresponding to each test statistics, the critical value is computed by:

$$\lambda_i = \frac{(n-i)t_{p,n-i-1}}{\sqrt{(n-i+1+t^2_{p,n-i-1})(n-i+1)}}, \; i=\overline{1,r},$$

where: $t_{p,n-i-1}$ is the 100p percentage point from the Student distribution with $n-i-1$ degrees of freedom and $p = 1 - \frac{\alpha}{n-i-1}$, α being the significance level.

The number of outliers is defined by:

$$N = \max\{i : R_i > \lambda_i\}.$$

This method is better than the Grubbs' one because it adjusts the critical values function of the outliers' number taken into account.

When the process is not Gaussian, a box-plot is suggestive for the outliers' identification.

A box plot is a graphical representation of the data dispersion, which draws the quartiles (first, third and the median), together with two fences. Any observation situated outside these fences is considered a potential outlier.

Tests for outliers' detection are implemented in R, in the packages: outliers (for univariate data series) [119], mvoutliers (for multivariate data series), tsoutliers (for time series, based on ARIMA models), and extremevalues.

For performing the Grubbs and Dixon tests for outliers' detection and for drawing the box plot for Constanta annual series, the following sequence of code is written in R:

```
   data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Anuale_carte.csv", sep=",",
header=TRUE) # reads the data containing ten annual precipitation series
   x<- data [,4] # reads the data containing Constanta annual precipitation series for the
# period 1965-2005
   library(outliers) # loads the library
   grubbs.test(x, type=10) # performs Grubbs' test for the detection one outlier, statistically
# different from the other values. If type = 11, the test is used for checking if the lowest and # highest values are outliers. If type = 20 the test is used to verify if data series contains
# two outliers on the same tail. In this case, the sample volume must be between 3 and 30.

         Grubbs test for one outlier
         data:  x
         G = 2.2971, U = 0.8648, p-value = 0.3694
         alternative hypothesis: highest value 674.8 is an outlier

   grubbs.test(x, type=11)

         Grubbs test for two opposite outliers
         data:  x
         G = 4.0857, U = 0.7879, p-value = 1
         alternative hypothesis: 227 and 674.8 are outliers

   y<-data [1:30,4] # reads the first 30 values in Constanta annual precipitation series
   grubbs.test(y, type=20)

         Grubbs test for two outliers
         data:  y
         U = 0.7103, p-value = 0.3181
         alternative hypothesis: highest values 548.7 , 584.3 are outliers

   dixon.test(y, type = 0, opposite = FALSE, two.sided = TRUE) #performs Dixon's test on y # type is a natural number that selects the test statistic, function of the sample volume. If
# it is zero, the selection is automatically performed. opposite is a parameter used for the
# selection of the extreme value (maximum or minimum).

         Dixon test for outliers
         data:  y
         Q = 0.157, p-value = 0.8301
         alternative hypothesis: highest value 584.3 is an outlier

   boxplot(x) # draws the box plot for the series x (Fig. 3a.)
boxplot(y) # draws the box plot for the series y (Fig. 3b)
```

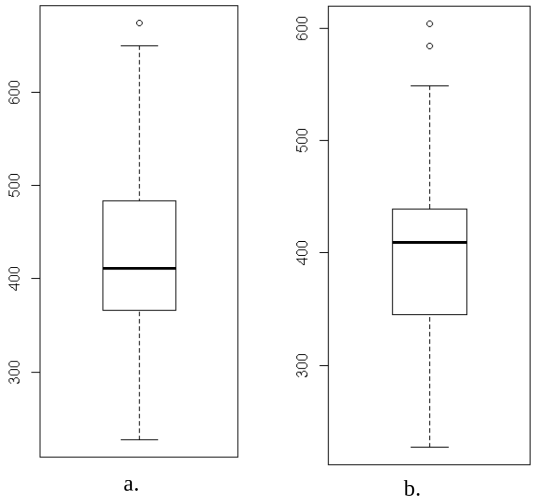


Fig. 3. Box plot of: a. Constanta annual precipitation series (1965-2005); b. Constanta annual precipitation series (1965-1994)

For performing the Tietjen-Moore and ESD tests we used the codes from [122] [123], adapted to the series *y* from the previous example and to *k* =2 outliers. The result of the first test is: the value of the test statistics is 0.7099646, and the critical value at 0.05 level of significance is 0.5477848. The result of the second test is:

```
No. Outliers   Test Stat.    Critical Val.
     1         2.174088      2.908473
     2         1.987073      2.892705
```

In both cases the test statistics are smaller than the critical values, so the hypothesis of outliers' presence can be rejected.

5. Change points' detection

It is known that meteorological time series are affected by more or less abrupt changing conditions in the environment. A breakpoint or a change point is defined to be the moment when the process generating the series changes.

The literature usually treats the changes in mean, but changes in variance, in frequency structure or in the system are also considered. We focus here on chan -ges in the mean.

Solutions to this problem have been proposed in [11][15][21][26]. The earliest studies considered the case of a sequence of independent identical and normally distributed variables for which the change point in the mean had to be detected [22]. Later, Bayesian [19] [86] and non - Bayesian solutions have been done for this problem in the case of dependent processes.

A different approach for the change points' finding appears in machine learning literature, where the problem is formulated as unsupervised clustering of a set of temporal observations, for the identification of homogeneous sequences with respect to a given distance measure (Minkowski or Dynamic Time Warping). Common methods for finding the right segmentation consider top - down, bottom - up or sliding window algorithms [7][68]. The reader may refer for an overview of these techniques to [61] [105].

Last period, the breakpoints' detection is done in hydrological applications by using CUSUM or segmentation procedures, as alternatives to the classical tests of Buishand, Lee and Heghinian, Pettitt.

CUSUMs are relatively simple tests based on the cumulative sums charts, which have the advantage to offer graphical interpretations of the results. Parame-tric CUSUMs are based on the comparison of the probability distribution functions before and after the changing moment, in the hypothesis of data indepen-dence. The nonparametric CUSUM test is a rank - based method that allows comparisons of successive observations with the median of the series for detecting a change in its mean after some observations. The test statistic is the cumulative sum of the *k* signs of the difference from the median. It is more robust to autocorrelation existence than the parametric one [47]. A version of it is implemented in Change Point Analyzer software.

Different authors used segmentation procedures. Liu [69] performed the break point analyzes for the segmentation of the seasonal runoff in many seasons, and his technique was validated by the Monte-Carlo method. Duggins [31] proposed an alternative method for the breakpoints detection, using transition matrices. Tsakalias and Koutsoyiannis [103] developed a heuristic algorithm, which emulates the exploratory data analysis of the human expert and which encodes a number of search strategies in a pattern directed computer program. Hubert segmentation procedure [53] [54] was generalized by Kehagias *et al.* [60]. They proposed a dynamical programming solution to the problem: divide a given time series into homogeneous segments so that the contiguous segments are heterogeneous. A new segmentation algorithm for long hydro-meteorological time series that combines the dynamic programming with the remaining cost concept has been proposed by Gedikli *et al.* [40] [41]. Also, a user-friendly program (segmenter) has been built to perform segmentation-by-constant and segmentation-by-linear-regression.

Whatever the procedure for the change point detection is, the null hypothesis (H_0) is that the series has no break point, and its alternative (H_1) is that the series has at least a breakpoint.

Changepoints tests are implemented in different software. We discuss here only the capabilities of Khronostat and R.

Using Khronostat one can perform the change point detection by four different methods: Buishand [17][18], Lee and Heghinian [64], Pettitt [84] and Hubert [53][54]. A short description of these tests is done in [7]. We mention that the Buishand and Lee & Heghinian tests are based on the hypothesis of series' normality. The nonparametric test of Pettitt can be employed even if the series distribution is unknown, in the independence hypothesis. Among them, only the segmentation procedure of Hubert detects multiple breaks and the moments of their apparition, based on the Scheffé test [93].

The results of these tests for Constanta annual series (1961-2013), at 0.95 confidence level, are: Buishand doesn't reject the null hypothesis (**Fig. 4**), Pettit, Lee&Heghinian (**Fig. 5**) and Hubert segmentation procedure reject it. The last three tests indicate 1994 as a change point.

The mDP algorithm, implemented in segmenter, based on the Scheffé test for the breakpoints selection, provides the same breakpoint - 1994 (corresponding to 34 on the abscissa) (**Fig. 6**).

There are many packages in R software that perform the break points detection, as: bcp, ecp, changepoint, cpm, BFAST, SLC, strucchange, wbs etc. In the following we present the capabilities of the first four packages.

The bcp package [37] provides an implementation of the Bayesian change point procedure of Barry and Hartigan [10] for univariate time series. It employs the Markov Chain Monte Carlo method and provides for each time moment the posterior probability of a change and the posterior mean. The procedure assumes that the observations are independent, normally distri -buted, with the same variance, but the independence hypothesis could be weakened.

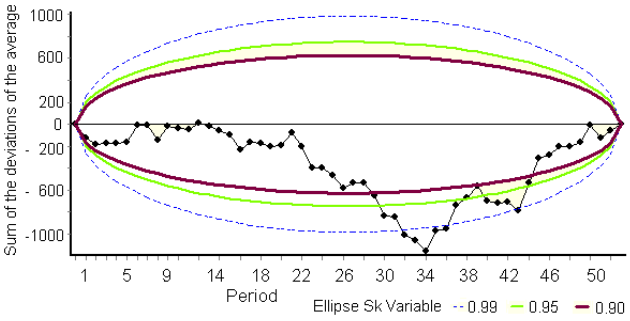


Fig. 4. Bois' ellipse associated with the Buishand test

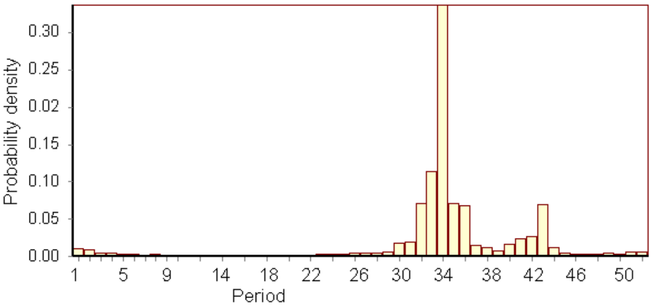


Fig. 5. A posteriori probability density of a break time position

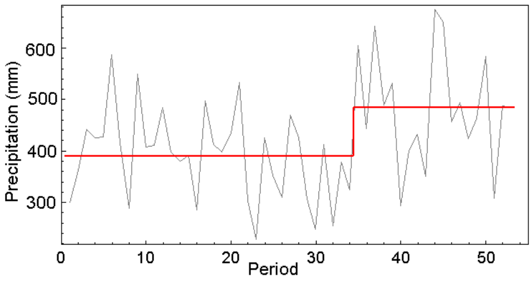


Fig. 6. Break point selection by mDP algorithm with constant regression, implemented in segmenter software

We present here the code for the change point detection for annual precipitation series registered at Constanta, using the bcp package.

```
    library(bcp)
    data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",", header=TRUE)
    X<-data[,1]
    Z<-as.vector(X)
    bcp.0 <- bcp(Z)
    plot.bcp(bcp.0) # produces Posterior Means: location in the sequence versus the posterior # means over the iterations; Posterior
Probability of a Change:  location in the sequence
# versus the relative frequency of iterations which resulted in a change point – Fig. 7
    bcp(Z, w0 = 0.2, p0 = 0.2, burnin = 50, mcmc = 500, return.mcmc = FALSE)
# returns the Bayesian change point summary (probability of a change in mean and posterior # means
    fitted.bcp(bcp.0)  # returns fitted values extracted from the bcp object
    residuals.bcp(bcp.0)  # returns residual extracted from the bcp object
```

In **Fig. 7** the upper part presents the posterior means calculated for each point in the data series, and the lower part, the posterior probability that the point is a change one. The posterior probability of changes is not significant, excepting for the years 1994 and 2003 (corresponding to 34 and 43 on the abscissa).

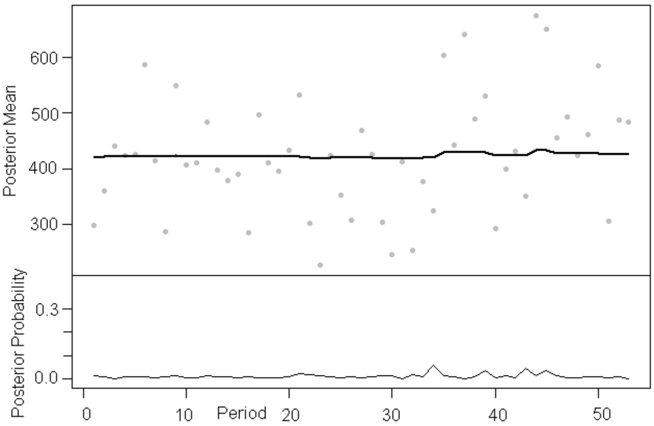


Fig. 7. bcp: posterior means and posterior probabilities of changes for Constanta annual series (1961 – 2013)

The ecp package [113] is designed to determine (multiple) distributional change points and their locations in univariate and multivariate time series, while making the assumptions that the observations are independent over time, and there is $\alpha \in (0, 2]$ such that the absolute moment of order α of the distribution of the series exists.

Divisive or agglomerative algorithms are the base of hierarchical estimation.

Divisive estimation sequentially identifies change points using a bisection algorithm. The agglomerative algorithm estimates change point locations looking for optimal segmentation. Both approaches detect changes in the data distribution.

```
    library(ecp)
    data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv", sep=",", header=TRUE)
    W<-data[,1]
    Z1<-matrix(W, ncol=1)
    Z1
    output1 <- e.divisive(Z1, R = 499, alpha = 1)
    output1

        $k.hat
        [1] 1

        $order.found
        [1]   1 637

        $estimates
        [1]   1 637

        $considered.last
        [1] 406

        $p.values
        [1] 0.146

        $permutations
        [1] 499

        $cluster
          [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
         [39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
         [77] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [115] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [153] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [191] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [229] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [267] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [305] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [343] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [381] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [419] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [457] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [495] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [533] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [571] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
        [609] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

changepoint is a complete R packages, which provides the implementation of different methods for the detection of changes in mean and variance. Here we present only some procedures treating the change points in mean.

'binseg.mean.cusum' is an approximate method that uses the Binary Segmentation [94] to calculate the number of breakpoints and their position for the cumulative sums test statistic. The first argument of the function is a vector that contains the series values; the second one is the maximum number of breakpoints (fixed by the user) and the third one, the value of the penalty function. This algorithm can be utilized without restrictive assumptions on the dataset distribution.

For example:

```
library(changepoint)
data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",", header=TRUE)
x<-data[,1]
binseg.mean.cusum(x, Q=2, pen=0.8)  # looks for two change change points


        $cps
        [,1]          [,2]
[1,] 34.00000      21.00000
[2,]  21.81962     18.64178

        $op.cpts  # gives the optimal change point locations for the penalty supplied,
        [1] 2       #  that is 34, in our case.

        $pen
        [1] 0.8
```

If data is normally distributed, 'binseg.mean.cusum( )' may be replaced by the function 'binseg.mean.norm' with the same arguments.

Another possibility is the use of 'cpt.mean( )', with the arguments:

- data - contains the data within which we look for the break point;
- penalty - with the possibilities of choice of None, SIC, BIC, AIC, Hannan -Quinn, Asymptotic, Manual;
- pen.value - a numeric value of penalty, when choosing "Manual", or a text that gives the formula to use, as: n - sample volume, null - null likelihood, alt - alternative likelihood, tau - proposed change point, diffparam - difference in number of alternative and null parameters;

- method - ‘AMOC’ (for single changepoint), ‘BinSeg’ [94], ‘SegNeigh’ [5] or ‘PELT’ [62] (for multiple change points);
- Q - maximum number of change points when ‘BinSeg’ is selected or maximum number of segments when ‘SegNeigh’ is used;
- test.stat - the test statistic or the data distribution: ‘Normal’ or ‘CUSUM’;
- class - if TRUE, then an object of class ‘cpt’ is returned;
- param.estimates - if TRUE and class is also TRUE, then parameter estimates are returned. Otherwise, no parameter estimate is returned.

Some examples are presented in the following:

```
library(changepoint)
data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",", header=TRUE)
x<-data[,1]
cpt.mean(x, penalty="SIC", method="AMOC", class=FALSE)

        cpt p      value
         34          1
# only a change point was detected in the data series and this is the 34-th value

ans=cpt.mean(x, penalty="Asymptotic", pen.value=0.01, method="AMOC")

        "cpttype":
        [1] "mean"

        Slot "method":
        [1] "AMOC"

        Slot "test.stat":
        [1] "Normal"

        Slot "pen.type":
        [1] "Asymptotic"

        Slot "pen.value":
        [1] 27.97932

        Slot "cpts":
        cpt
        34  53

        Slot "ncpts.max":
        [1] 1

        Slot "param.est":
        $mean
        [1] 389.5324 484.4105


 ans = cpt.mean(x, penalty="Manual", pen.value=0.8, method="AMOC", test.stat=
"CUSUM")
 ans

        An object of class "cpt"
        Slot "data.set":
        Time Series:
        Start = 1
        End = 53
        Frequency = 1
         [1] 299.2 361.0 440.9 424.2 426.1 586.2 415.1 288.4 548.6 407.4 410.9 483.8
        [13] 397.3 379.1 390.1 285.0 496.6 411.4 396.4 433.9 532.4 301.9 227.0 424.9
        [25] 352.3 308.8 469.3 425.6 305.3 246.6 412.3 253.8 378.2 324.1 604.3 443.3
        [37] 641.2 488.8 531.1 292.5 400.4 430.6 350.2 674.6 649.9 456.2 493.7 423.9
        [49] 461.7 583.8 307.0 487.6 483.0

        Slot "cpttype":
        [1] "mean"
```

```
Slot "method":
[1] "AMOC"

Slot "test.stat":
[1] "CUSUM"

Slot "pen.type":
[1] "Manual"

Slot "pen.value":
[1] 0.8

Slot "cpts":
cpt
35  53  # change points detected: the 35th and the 53rd values in the series

Slot "ncpts.max":
[1] 1

Slot "param.est":
$mean
[1] 395.6686     477.7500  # the values of the series in the change points.
```

```
ans = cpt.mean(x, penalty="AIC", method="SegNeigh", test.stat="Normal")
print(ans) # print details of the methods and a summary of results
```

```
summary(.)  :
Changepoint type        : Change in mean
Method of analysis      : SegNeigh
Test Statistic          : Normal
Type of penalty         : AIC with value 2
Maximum no. of cpts   : 5
Changepoints location  : 34, 39, 43, 45
```

```
plot(ans) # plots the data and the change point, as in Fig. 8.
```

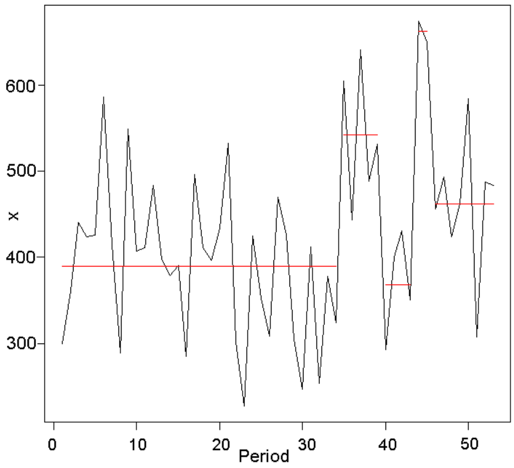


Fig. 8. Output of ‘plot(ans)’ command

Other options offered by this package are: ‘single.mean.cusum.calc’ ‘multiple.mean.cusum’, ‘multiple.mean.norm’ etc. The first one calculates the CUSUM test statistic for all possible break points’ locations and provides only the most probable one. Even if it can be used if no assumption on the data distribution is done, there is no test implemented to confirm that the change point detected is a true break point, so the test must be utilized with cautions. For example:

```
library(changepoint)
data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",", header=TRUE)
x<-data[,1]
single.mean.cusum.calc(x,extrainf=TRUE)
```

```
    cpt test        statistic
    35.00000      21.81962
```

‘multiple.mean.cusum.calc’ has the arguments:

- data - contains the data within which we look for the break point;
- mul.method – ‘BinSeg’ or ‘SegNeigh’
- penalty - with the possibilities of choice of: None, SIC, BIC, AIC, Hannan -Quinn, Asymptotic, Manual;
- pen.value - a numeric value of penalty, when choosing “Manual”, or a text that gives the formula to use, as: n - sample volume, null - null likelihood, alt - alternative likelihood, tau - proposed change point, diffparam - difference in number of alternative and null parameters;
- Q - maximum number of change points when ‘BinSeg’ is selected or maximum number of segments when ‘SegNeigh’ is used;
- class - if TRUE, then an object of class ‘cpt’ is returned;
- param.estimates - if TRUE and class is also TRUE, then parameter estimates are returned. Otherwise, no parameter estimate is returned.

For example:

```
library(changepoint)
data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",", header=TRUE)
x<-data[,1]
multiple.mean.cusum(x, mul.method="SegNeigh", penalty="AIC", pen.value= null, Q=5, class=FALSE, param.estimates=TRUE)
```

```
    $cps

                          [,1]   [,2]   [,3]   [,4]   [,5]
                  [,1]     0      0      0      0      0
                  [,2]    34      0      0      0      0
                  [,3]    34     21      0      0      0
                  [,4]    43     39     34      0      0
                  [,5]    45     43     39     34      0

    $op.cpts # gives the optimal change point locations for the penalty supplied
    [1] 4

    $pen # provides the penalty used for finding the optimal change points number
    [1] 2
```

Another complex package is cpm [109]. This framework is an approach to Phase II process monitoring (also known as sequential change detection) for performing breakpoints detection, using parametric and nonparametric procedures, on univariate data streams.

In Batch detection (also known as Phase I) the researcher has to look for the change points, using all the available observations in a data set with a fixed length. In Sequential detection, the observation is processed at the moment of its apparition and the decision concerning the break occurrence is based only on the previously received data. When a change point is detected, the change detector is restarted from the following observation in the sequence, allowing the detection of multiple break points.

The procedures implemented in cpm permit single or multiple change points detection in mean or variance, in Gaussian or non-Gaussian series, using the statistics: Student, Bartlett, GLR, Exponential, GLR Adjusted, Exponential Adjusted, FET, Mann-Whitney, Mood, Lepage, Kolmogorov-Smirnov and Cramer-von-Mises. The first three statistics are used for detection of changes in a Gaussian sequence, respectively in mean, in variance, in mean and variance. The fourth and fifth test statistics are used to determine changes of the parameter of an exponentially distributed data series; the sixth one performs the Fisher test for changes in the parameter of a Bernoulli distributed sequence. The last five tests can be used if the process generating the data is not Gaussian. They are respectively [92]:

- the Mann-Whitney test, for detecting the position of the shifts in a stream;
- the Mood test, for detecting the shifts in a stream;
- the Lepage test, for detecting the position and/or the shifts in a stream;
- the Kolmogorov-Smirnov and Cramer-von-Mises tests, for detection of arbitrary changes in a stream.

In Phase I, ‘detectChangePointBatch’ function is used for testing the hypothesis of the existence of a single change point in a series of observations, and estimating its location. For example, for Constanta annual precipitation series the application of Mann – Whitney methods is done by written the following commands:

```
data<- read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",", header=TRUE)
x <- data[,1]
resultsKS <- detectChangePointBatch(x, cpmType = "Kolmogorov-Smirnov", alpha = 0.05) # detects the change points by the Kolmogorov-Smirnov method
plot(x, type='l')
if (resultsKS$changeDetected) {
abline(v = resultsKS$changePoint, lty=2)
}
# plots the data series and marks the moment of the break with a vertical line (Fig. 9)
resultsKS
```

```
$changePoint
[1] 34

$changeDetected
[1] TRUE

$alpha
[1] 0.05

$threshold
[1] 0.9966167
```

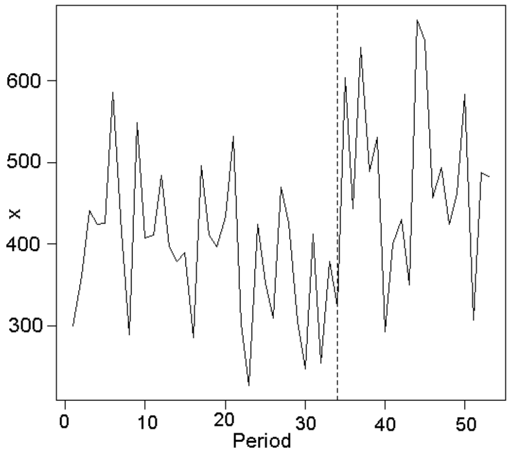


Fig. 9. Breakpoint detected in Constanta annual series (1961-2013) by the Mann – Witney method (Phase I)

‘detectChangePoint’ allows performing Phase II analysis. For example, for Constanta monthly series (1961 – 2013), we have the following commands:

```
data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv", sep=",", header=TRUE)
x <- data[,1]
resultsLepage <- detectChangePoint(x, cpmType = "Lepage", ARL0=500)
# detects the change points by the Mood method;
# ALRO ∈ {370, 500, 600, 700, ... , 1000, 2000, ... , 10000, 20000, 50000}
plot(x, type='l')
if (resultsLepage$changeDetected) { abline(v = resultsLepage$detectionTime, lty=2)}
resultsLepage
```

```
$changePoint #  change point estimated
[1] 435

$detectionTime # the moment at which the change was detected
[1] 440

$changeDetected
[1] TRUE
```

In the case when the stream of observations contains many change points, the ‘processStream’ function can be used for their detection because it allows processing the observations one-by-one. For example:

```
    data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv", sep=",", header=TRUE)
    x <- data[,1]
    res <- processStream(x, cpmType="Mann-Whitney", ARL0=500, startup=20)
#  startup represents the number of observations after which the monitoring  begins
    res

        $changePoints #the estimated change points
        [1] 435 474

        $detectionTimes
        [1] 440 481

    plot(x,type='l')
    abline(v=res$detectionTimes) #draws vertical lines with the abscissa equal to the
# moments when the change points were detected
    abline(v=res$changePoints, lty=2) #draws vertical lines with the abscissa equal to the
# estimated moments of changes – Fig. 10.
```

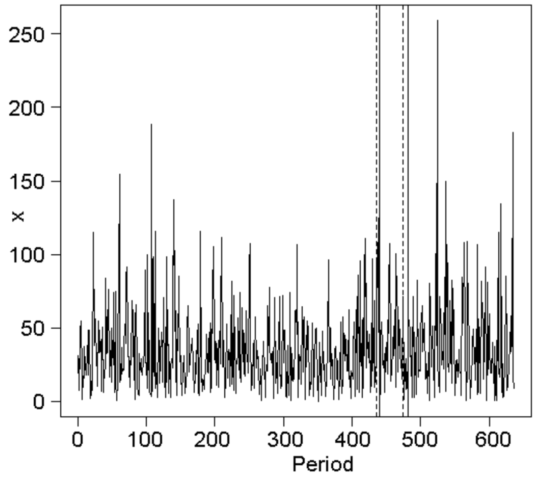


Fig. 10. Changes in mean for Constanta monthly series. The two solid black lines indicate the moments when the changes were detected, and the dotted lines indicate the estimated change point locations

6. Testing for Long Range Dependence property

By definition, a time series has the long range dependence (LRD) property if the series $\sum_{h=-\infty}^{\infty}\rho(h)$ is divergent.

This definition is equivalent to the following behavior of the series' autocova-riance: $\gamma(k) \sim k^{-\alpha}L(k)$, where $0<\alpha<1$ and $L(k)$ is a function with a slow variation at infinity.

LRD manifests in the time domain as a high level of correlation between points situated at big distances in time and, in the frequency domain, as a significant le-vel of power at frequencies near zero [36].

A measure for long term memory (and fractality) of a time series is the Hurst exponent, H [73]. Its values range in the interval (0, 1). The value of H in the interval $(0, 0.5)$ indicates an anti - persistent behavior of a time series, which is an up value is more likely followed by a down value, and vice versa. H = 0.5 indicates a random series. $H \in (0.5, 1)$ corresponds to a persistent series, that is the next value has more likely the same direction as the current one.

Different methods have been developed to carry out the LRD analysis: R/S [55] and Lo's modified R/S method [71], Aggregated Variance Method [102], Absolute values of aggregated series [102], Ratio of Variance of Residuals [102], Periodogram [43], Higuchi Method [50], Detrended Fluctuation Analysis - DFA [82] [83], Whittle's approximate MLE and Local Whittle estimators [89], methods based on wavelets etc. The reader might also refer to [102].

We shortly describe some of them here. For this aim we denote by $(x_k)_{k\in\overline{1,n}}$ the studied data series.

In the rescaled analysis (R/S method), the data series is divided in d sub-series of the same length. For each sub-series, the mean and standard deviation are computed, the data are normalized by subtracting the sub-series average, the cumulative sub-series is created by adding up the normalized values, the range is determined as difference between its maximum and minimum, and the rescaled range is computed dividing the sub-series range to its standard deviation. Finally, the average of the rescaled ranges (R/S) of all subseries is determined.

The procedure is repeated for different *d.*

Hurst coefficient is determined as the slope of the fitted line of log(R/S) on log(*d*).

Different corrections have been proposed by Lo, Andrews, Wang, etc. but we shall not discuss them here.

The function 'hurstexp' from the package pracma [121] in R calculates the Hurst exponent using the R/S method and some corrections of it, based on the article [106]. Here is an example:

```
data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv", sep=",", header=TRUE)
x <- data[,1]
library(pracma)
hurstexp(x, d = 50, display = TRUE) # d is the smallest box size - default is 50;
```

```
    Simple R/S Hurst estimation:             0.582716
    Corrected R over S Hurst exponent:  0.6345952
    Empirical Hurst exponent:                0.5824732
    Corrected empirical Hurst exponent:   0.5368507
    Theoretical Hurst exponent:              0.5444537
```

The rescale analysis (R/S) can also be performed using 'rsFit' function from fArma package [110] of R. For example:

```
data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv", sep=",", header=TRUE)
x <- data[,1]
library(fArma)
rsFit(x, levels = 50, minnpts = 3, cut.off = 10^c(0.7, 2.5), doplot = FALSE, trace = FALSE, title = NULL, description = NULL)
```

```
    Title:
    Hurst Exponent from R/S Method

    Call:
     rsFit(x = x, levels = 50, minnpts = 3, cut.off = 10^c(0.7, 2.5),  doplot = FALSE,  trace = FALSE, title = NULL, description =
  NULL)

    Method:
    R/S Method

    Hurst Exponent:
       H          beta
  0.5785334 0.5785334

    Hurst Exponent Diagnostic:
    Estimate      Std.Err      t-value        Pr(>|t|)
  X  0.5785334 0.02859643 20.23096  2.835059e-23

    Parameter Settings:
    n      levels  minnpts cut.off1 cut.off2
    636     50        3            5         316
```

In *Aggregated variance method*, after dividing the original data series into *d* subseries of the same length, the aggregated series is formed by the averages of the values in the sub-series, and the sample variance is computed. For different values of *d*, the sample variance is plotted against *d* on a log-log scale and a least squares line to the points of the plot is drawn, whose slope, β, is used for computing the Hurst coefficient. For fGN and ARIMA processes, $-1< \beta = 2H - 2 <0$.

In *Absolute values of the aggregated series method*, the aggregated series is formed as in Aggregated variance method, then the *n*-th absolute value of the resulted sub-series are computed instead of their variances. For different values of *d*, the *n*-th absolute value is plotted against *d* on a log-log scale and a least squares line to the points of the plot is drawn, whose slope, β, is used for computing the Hurst coefficient.

To apply the *Differenced variance method,* one has to pass through the stages:

- Produce the aggregated series of order *m*, for various *m*;
- Find the sample variance for each *m*;
- Plot the log(variance) as a function of log(*m*);
- If there are suspicions about the non-stationarity existence, difference the variances, as a function of *m*;
- Plot the results on a log–log plot.

If there are shifts in the mean or a slowly declining trend is present, then the plot from the third step should be in an exponential form, and that from the last one, in a linear form, with the slope $\beta = 2H - 2$.

The use of this technique is recommended to distinguish the cases: (a) H is near 0.5 and there are jumps in the mean; (b) H is significantly larger than 0.5, and there is a non-zero trend [101].

The *Higuchi's method* is based on dividing a given data series into sub-series, defining the length of the curve for an interval and detecting its fractal dimension. More precisely, the steps are:

- From the data series $(x_i)_{i=\overline{1,n}}$, built the sub-series (X_k^m) containing $(x_m, x_{m+k}, x_{m+2k}, \dots, x_{m+[(n-m)/k]\cdot k})$, where [ ] denotes the integer part;
- Define the length of X_k^m as:

$$L_m(k) = \frac{1}{k}\left\{ \sum_{i=1}^{\left[\frac{n-m}{k}\right]} \left| x_{m+ik} - x_{m+(i-1)k} \right| \cdot \frac{n-1}{\left[\frac{n-m}{k}\right]\cdot k} \right\},$$

- Define the length of the curve for the time interval k, $\langle L(k) \rangle$, as the mean value over k sets of $L_m(k)$.

If $\langle L(k) \rangle \sim k^{-D}$, the curve is fractal with the dimension D.

'hurstBlock' from fractal package [112] of R is a function used for estimating the Hurst exponent of a long memory time series, by choosing one of the methods specified in its argument. All implemented procedures work on the brute series, not on its spectrum.

The arguments of "hurstBlock" are:

- x - a vector containing the values of a time series, uniformly-sampled.
- fit - a function giving the linear regression method used for fitting the statistics (on a log-log scale). It can be: 'lm', 'lmsreg', and 'ltsreg', the default being 'lm'.
- method - a character string that indicates the method used for estimating the Hurst coefficient. The default is: "aggabs" (Absolute Values of the Aggregated Series), but one can also choose:
  - "aggvar" - Aggregated Variance Method [102]
  - "diffvar" - Differenced Variance Method [102]
  - "higuchi" - Higuchi's Method [50]
- scale.max - the maximum block size used in partitioning the data series. The default is 'length(x)'.
- scale.min - the minimum block size used in partitioning the data series. The default is 8.
- scale.ratio - ratio of successive scales used in partitioning the data series. The default is 2.
- weight - a function of a variable (x) used for weighting the resulting statistics (x) for each scale during the linear regression. It is supported when fit = lm. The default is 'function(x) rep(1,length(x))'

```
data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv", sep=",", header=TRUE)
x <- data[,1]
library(fractal)
# calculates the Hurst coefficient of x using two techniques
y<-hurstBlock(x, method="higuchi", scale.min=8, scale.max=636, scale.ratio=2, weight=function(x) rep(1,length(x)), fit=lm)
y

     H estimate             : 0.9949413
     Domain                 : Time
     Statistic              : higuchi
     Length of series       : 636
     Block overlap fraction: 0
     Scale ratio            : 2

     Scale        8.0     16.0    32.00    64.00   128.00   256.000   512.000
     higuchi  2798.2   1401.1   702.65   351.19   173.58    84.716    43.431

# plots the results
plot(u, key=FALSE)
mtext(paste("higuchi", round(as.numeric(u),3), sep=", H="), line = 0.5, adj=1)
```

The same methods are implemented in fArma package from R.

The sequence of commands for performing them, in the case of Constanta monthly series, and the results are:

```
data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv", sep=",", header=TRUE)
```

```
x <- data[,1]
aggvarFit(x, levels = 50, minnpts = 3, cut.off = 10^c(0.7, 2.5), doplot = TRUE, trace = FALSE, title = NULL, description = NULL)

        Method:
        Aggregated Variance Method
        Hurst Exponent:
                 H            beta
          0.6088583    -0.7822834

        Hurst Exponent  Diagnostic:
            Estimate        Std.Err       t-value       Pr(>|t|)
        X 0.6088583     0.03364099   18.0987     4.414577e-23

absvalFit(x, levels = 50, minnpts = 3, cut.off = 10^c(0.7, 2.5), moment = 1, doplot = TRUE, trace = FALSE, title = NULL, description = NULL)

        Method:
        Absolute Moment - No. 1
        Hurst Exponent:
                 H          beta
          0.6749373     -0.3250627

        Hurst Exponent Diagnostic:
            Estimate       Std.Err         t -value       Pr(>|t|)
        X 0.6749373    0.03679841   18.34148    2.520645e-23

diffvarFit(x, levels = 50, minnpts = 3, cut.off = 10^c(0.7, 2.5), doplot = TRUE, trace = FALSE, title = NULL, description = NULL)

        Method:
        Differenced Aggregated Variance
        Hurst Exponent:
                 H          beta
          0.7531249   -0.4937501

        Hurst Exponent Diagnostic:
             Estimate      Std.Err         t-value        Pr(>|t|)
         X 0.7531249   0.1412358     5.332396    2.052936e-05

        Parameter Settings:

          n     levels  minnpts   cut.off1   cut.off2
        636     50        3           5          316

higuchiFit(x, levels = 50, minnpts = 2, cut.off = 10^c(0.7, 2.5), doplot = TRUE, trace = FALSE, title = NULL, description = NULL)

        Method:
        Higuchi Method
        Hurst Exponent:
               H            beta
          0.9578126    -1.0421874

        Hurst Exponent Diagnostic:
             Estimate      Std.Err         t-value        Pr(>|t|)
         X 0.9578126   0.02840518   33.71965    4.447243e-35
```

The arguments of the previous functions are:

- x - the series to be analyzed;
- level - the number of blocks from which the statistics values are computed;
- minnpts - the minimum number of points or block size used;
- cut.off - a vector containing the lower and upper cut of points. The default is c(0.7, 2.5);
- doplot - a logical value; if TRUE, a plot is displayed;
- trace - a logical value, by default FALSE. It indicates if the process is traced;
- title - a character string allowing for a title of the project;
- description - a character string allowing for a description.

The results depend on the cut.off and are plotted in **Fig. 11.**

To perform *Detrended Fluctuation Analysis* [82][83], the data series is firstly integrated and divided into sub-series of the same length, *m*, for which polynomial trends are fitted. The integrated series is detrended by subtracting from each sub-series the corresponding polynomial trend, and then the root mean-squared (*F*(*m*)) of the new series is computed. These steps are repeated for different lengths of the sub-series, for providing a linear relationship tween $F(m)$ and *m*.

'DFA' function from fractal package is employed for performing the Detrended Fluctuation Analysis. It is useful for analyzing the long-memory in data series whose spectral density function has the form $S(f) \sim f^{\alpha}$ at low frequencies, $f \in (0, 0.5)$ being the normalized frequency variable and $\alpha < -1$, the scaling exponent. If $\alpha > -1$, then cumulative summations of the data series must be performed for increasing the scaling exponent (each cumulative summation decreases the exponent by 2). The user may also use the differencing operation prior to the DFA analysis.

'DFA' function has the following arguments:

- x - a vector containing the values of a time series, uniformly-sampled;
- detrend - a character string that denotes the detrending type used on each sub-series. It can be a polynomial (for example, for the polynomial of first order the type is "poly1"), "bridge" or "none";

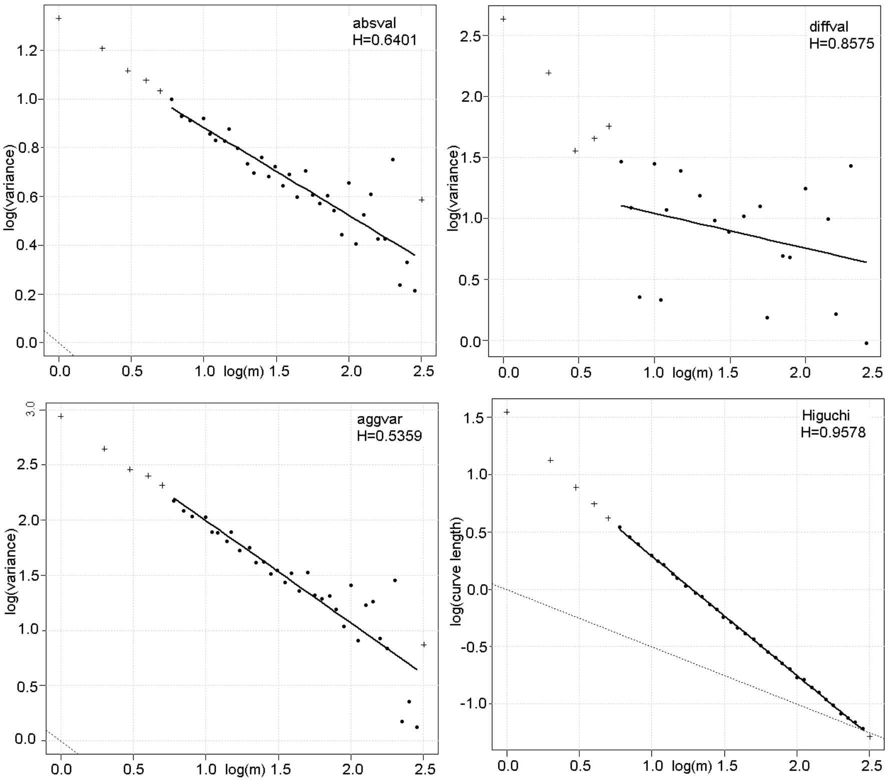


Fig. 11. Hurst coefficient computed by Absolute values, Aggregated variance, Differenced variance and Higuchi methods for Constanta monthly series

- overlap - the overlap of blocks in partitioning the time data expressed as a fraction in [0,1); the default is 0;
- scale.max - the maximum block size used in partitioning the series; the default is trunc(length(x)/2);
- scale.min - the minimum block size used in partitioning the data; the default is 2(k+1) for polynomial detrending (k is the degree of the polynomial) and it is min{4, length(x)/4} for the other detrending techniques;
- scale.ratio - the ratio of successive scales; the default is 2;
- sum.order - the number of differences or cumulative summations performed on the brute series prior to apply DFA. The default is 0, sum.order > 0 for cumulative summations, sum.order = p < 0 if a difference of p order is performed;
- verbose - a logical value, indicating if or not the detrending model and processing progress information is displayed (when it is TRUE/FALSE); the default is: FALSE.

```
data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv",sep=",", header=TRUE)
x <- data[,1]
library(fractal)
DFA.x <- DFA(x, detrend="poly0", sum.order=0)
eda.plot(DFA.x) #plots the charts with the results
```

```
print(DFA.x) #prints the results -
```

Fig. 12, *the left hand side*

```
Detrended fluctuation analysis for x
------------------------------------
H estimate                  : 0.06846433
Domain                      : Time
Statistic                   : RMSE
Length of series            : 636
Block detrending model : x ~ 1
Block overlap fraction      : 0
Scale ratio                 : 2

Scale     2.000   4.000   8.000 16.000 32.000 64.000 128.000 256.000
RMSE  20.799 24.992 27.149 28.263 28.205 28.529    26.682    26.832
```

```
DFA.x <- DFA(x, detrend="poly0", sum.order=1)
print(DFA.x)
```

```
Detrended fluctuation analysis for x
------------------------------------
H estimate                  : 0.968804
Domain                      : Time
Statistic                   : RMSE
Length of series            : 636
Block detrending model: x ~ 1
Block overlap fraction      : 0
Scale ratio                 : 2
Preprocessing               : 1st order cumulative summation

Scale      2.000  4.000  8.000    16.00    32.00    64.00   128.0    256.0
RMSE  22.879 45.425 87.928 169.43 339.58 652.68 1293.6 2538.7
```

```
eda.plot(DFA.x) #prints the results -
```

Fig. 12, *the right hand side*

The function ‘perFit’ from fArma package may be employed for computing the Hurst exponent using the *periodogram method* [43], that is based on the estimation of spectral density of a time series (which is the periodogram, in the case of finite variance).

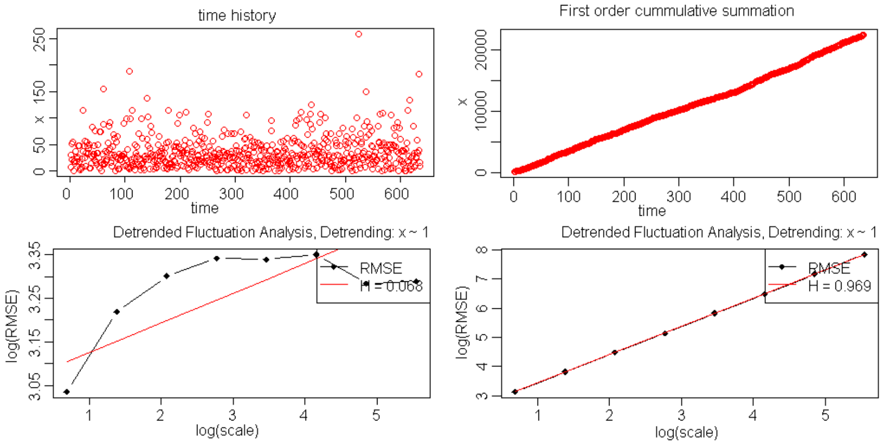


Fig. 12. The plot of DFA: the left-hand side, for the brute series, and the right-hand side, for the first order cumulative summation, applied to the brute series. The slope of the line which best fits a plot of log(RMSE) versus log(scale) is the scaling exponent

A series with long range dependence property has a spectral density with a power law behavior in frequency in the neighborhood of zero; therefore, for computational purposes only the lowest 10% of the frequencies are used. More precisely, the estimation of spectral density is determined by the least squares regression:

$$\ln(I_x(\omega_j)) = \alpha - d\ln[4\sin^2(\omega_j/2)] + e_j ,$$

where $I_x(\omega_j)$ is the sample periodogram at the *j*-th Fourier frequency $\omega_j = 2\pi j/T$ (*j* = 1, … , [T/2]), e_j is the residual, that must be independent identically distribu-ted, with a variance of $\pi^2/6$.

Therefore d in the previous equation is estimated by:

$$\hat{d}_{GPH} = \frac{\sum_{j=1}^{m}(Y_j - \overline{Y})\ln\{I(\omega_j\})}{\sum_{j=1}^{m}(Y_j - \overline{Y})^2} \text{ , } 0 < m < n,$$

where

$$Y_j = -\ln[4\sin^2(\lambda_j/2)] \text{ and } \overline{Y} = \tfrac{1}{m}\sum_{j=1}^{m} Y_j \text{ [43].}$$

Plotting the periodogram versus frequency, in log-log scale and fitting a straight line by the least squares method, its slope will be 1- 2*H*.

In the implementation of ‘perFit’, selecting the argument method = "per", one can vary the cut off and plot *H* versus the cut off to determine where the curve flattened, in order to estimate *H*. Alternatively selecting method = "cumper", the cumulative periodogram is used and the slope of the log-log fit is 2-2*H*.

For example:

```
data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_lunar_1961_2013.csv",sep=",", header=TRUE)
x <- data[,1]
library(fArma)
perFit(x, cut.off = 0.1, method = "per",  doplot = FALSE, title = NULL, description = NULL)

      Title:
      Hurst Exponent from Periodogram Method

      Call:
      perFit(x = x, cut.off = 0.1, method = "per", doplot = TRUE, title = NULL,
description = NULL)

      Method:
      Periodogram Method

      Hurst Exponent:
      H           beta
      0.5654380   -0.1308759

      Hurst Exponent Diagnostic:
      Estimate   Std.Err      t-value    Pr(>|t|)
    X 0.565438   0.08496654   6.654831   9.675144e-09

      Parameter Settings:
      n     cut.off
      636   10
```

Karagiannis *et al*. implemented some of the presented methods in a user - friendly software, called Selfis [59]. As a confirmatory analysis, after the LRD study, they propose the use of bucket shuffling, a procedure introduced in [38]. It decouples the short-term and the long-range correlations by shuffling sub-series of a given series, followed by the investigation of autocorrelation function [58], in five stages:

(i) The data series (x_i) is divided into *k* buckets of the same length, *b*;

(ii) A home is attached to each value, x_i. It is defined by the bucket with the number $H(i) = [i/b]$, where [ ] denotes the integer part;

(iii) The in-buckets are built by the couples (x_i, x_j) such that $H(i) = H(j)$;

(iv) The out-buckets pairs are built by the couples of values for which $H(i) \neq H(j)$. For such pairs, the corresponding offset is $|H(i) - H(j)|$;

(v) A randomization is performed. It can be of three types: external, internal or two - level. In external randomization the buckets' order is randomized, but their content is preserved. In internal randomization the buckets' order is preserved, but their content is randomized. In two levels randomization, the buckets are divided in atoms of the same size, followed by an external randomization of the blocks of atoms of which bucket.

(vi) The autocorrelation function of the randomized series is analyzed for deciding about the existence of LRD property, based on the fact that the autocorrelation function of the internally-randomized series preserves the same characteristics as those of the initial data series [38][60].

Therefore, if the study series has LRD, the ACF of the internally-randomized series should show the same slowly-decreasing behavior as the original one.

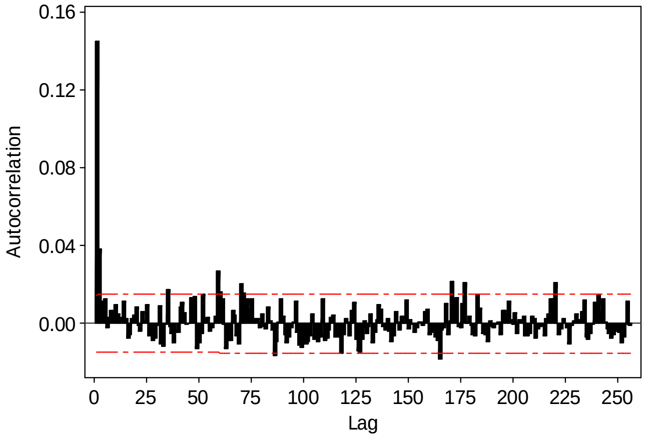
For Constanta daily series record from the period 1961- 2009, Selfis has been used for the estimation of Hurst's coefficients (H) by different methods, before and after internal shuffling.

The results are presented in **Table 2**, where 'Correl coef (%)' represents the correlation coefficients given as percentages and 'C.I (95%)' represents the corresponding confidence interval, at the confidence level of 95%. We remark that there is no significant difference between the Hurst coefficients before and after shuffling, so the series has not LRD.

Table 2. Results of LRD analysis running Selfis software, for Constanta daily series before and after internal shuffling

Method	Initial series		Shuffled series	
	H	Correl coef (%)/ C.I (95%)	H	Correl coef (%)/ C.I (95%)
Aggregated Variance	0.527	97.37	0.536	96.22
Absolute moments	0.472	94.13	0.480	93.46
R/S	0.585	99.88	0.568	99.87
Variance of residuals	0.721	96.49	0.672	98.67
Whittle	0.585	[0.575; 0.595]	0.505	[0.496; 0.515]

The autocorrelograms' analysis (**Fig. 13**) confirms this assertion.



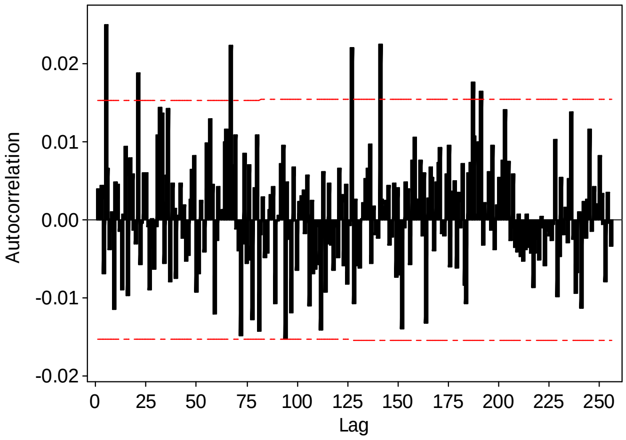


Fig. 13. Correlograms of Constanta daily series (1961 - 2009) before and after the internal shuffling

7. Goodness of fit tests

After data series modeling, one wants to check if the two data samples (registered and estimated by the model) come from the same distribution. Note that, in practice, the common distribution is, usually, unknown. Therefore, one wants to test the null hypothesis:

(H_0) The two samples come from a common distribution, e.g. $\{F(z) = G(z), \forall z\}$,

against the two - sided alternative:

(H_1): The two samples do not come from a common distribution, e.g. $\{\exists z: F(z) \neq G(z)\}$,

or against one of the one - sided alternatives:

(H_1): $\{\exists z: F(z) < G(z)\}$, $\{\exists z: F(z) > G(z)\}$,

where F and G are the continuous distribution functions.

The non-parametric test used for this purpose is called *the Kolmogorov –Smirnov test for two samples* and it does not rely on the normality assumption [25].

The implementation of this test is based on the computation of empirical cumulative distribution function (ecdf), and is done in R by using the function ecdf(x). To plot it, one can use the R command plot.ecdf (x).

It is possible to compare two independent samples, *x*, *y* by plotting their ecdf on the same chart, using the same scale. For example, if one wants to compare the samples formed respectively by the 20 annual precipitation values registered at Constanta and the next 20 annual precipitation values, he uses the commands:

```
   data<-read.csv("D:\\Lucrari_2.12.14\\2015_Carte\\Cta_annual_1961_2013.csv", sep=",", header=TRUE)
   x<-data[1:20,1] #builds the first sample, containing the first 20 values of annual series
   x

        [1]  299.2 361.0 440.9 424.2 426.1 586.2 415.1 288.4 548.6 407.4 410.9 483.8
        [13] 397.3 379.1 390.1 285.0 496.6 411.4 396.4 433.9

   y<-data[21:53,1] #builds the second sample, containing the last 33 values of annual
# series
   y

        [1]  532.4 301.9 227.0 424.9 352.3 308.8 469.3 425.6 305.3 246.6 412.3 253.8
        [13] 378.2 324.1 604.3 443.3 641.2 488.8 531.1 292.5 400.4 430.6 350.2 674.6
        [25] 649.9 456.2 493.7 423.9 461.7 583.8 307.0 487.6 483.0

plot.ecdf(x,y, pch= "*")  # plots ecdf(x) using the scale of ecdf(y)  - Fig. 14.

ks.test(x,y, alternative=c("two.sided", exact=NULL)) #performs the two-sample Kolmogorov-# Smirnov test. NULL means that the exact p-value is computed.

        Two-sample Kolmogorov-Smirnov test
        data:  x and y
        D = 0.2545, p-value = 0.3301
        alternative hypothesis: two-sided.
```

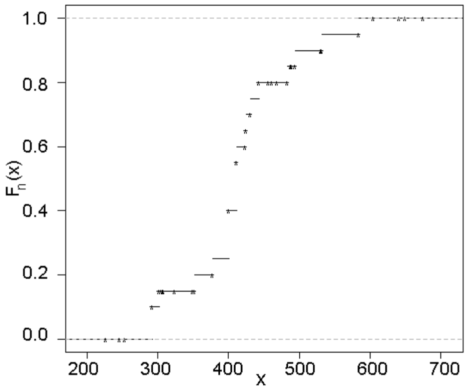


Fig. 14. Plot of ecdf(x) using the scale of ecdf(y)

So, the null hypothesis cannot be rejected.

If one of the distributions is known the Anderson - Darling and Cramer - von Mises tests could be successfully used.

References

[1] Aggarwal, C. C., *Outlier Analysis*, Springer, New York, NY, USA, 2013.

[2] Anderson, T. W., Darling, D. A., Asymptotic theory of certain “goodness of fit” criteria based on stochastic processes, *Annals of Mathematical Statistics*, **23**, 1952, pp. 193 - 212.

[3] Arnold, B.T., Emerson, J.W., Nonparametric Goodness-of-Fit Tests for Discrete Null Distributions, *The R Journal*, **3**/2, Dec. 2011, pp. 34 - 49.

[4] Arranz, M., Portmanteau Test Statistics in Time Series, http://www.tol-project.org/

[5] Auger, I. E., Lawrence, C. E., Algorithms for the Optimal Identification of Segment Neighborhoods, *Bulletin of Mathematical Biology*, **51**(1), 1989, pp. 39 - 54.

[6] Bakirov, N. K., Rizzo, M. L., Székely, G. J., A multivariate nonparametric test for independence, *Journal of Multivariate Analysis*, **97**(8), 2006, pp. 1742 - 1756.

[7] Barbulescu, A., Maftei, C., Bautu, E., *Modeling the hydro-meteorological time series. Applications to Dobrudja region*, LAP LAMBERT Academic Publishing GmbH & Co., Saarbrucken, Deutschland, 2010.

[8] Barnett, V., Lewis, T., *Outliers in Statistical Data*, John Wiley Sons, New York, 1994.

[9] Bartlett, M. S., Properties of sufficiency and statistical tests, *Proceedings of the Royal Statistical Society*, Series A, **160**, 1937, pp. 268 - 282.

[10] Barry, D., Hartigan, J. A., A Bayesian Analysis for Change Point Problems, *Journal of The American Statistical Association*, **88**, 1993, pp. 309 - 319.

[11] Basseville, M., Nikiforov, I., *Detection of abrupt changes: theory and application*, Prentice Hall, Englewood Cliffs, New York, 1993.

[12] Beran, R. Bilodeau, M., Lafaye de Micheaux, P. , Nonparametric tests of independence between random vectors, *Journal of Multivariate Analysis*, **98(**9), 2007, pp. 1805 - 1824.

[13] Box, G. E. P., Pierce, D. A., Distribution of residual correlations in autoregressive-integrated moving average time series models, *Journal of the American Statistical Association*, **65**, 1970, 1509 - 1526.

[14] Breusch, T. S., Pagan, A. R., A Simple Test for Heteroscedasticity and Random Coefficient Variation, *Econometrica*, **47**(5), 1979, pp. 1287 - 1294.

[15] Brodsky, B., Darkhovsky, B., *Nonparametric Methods in Change - Point Problems*, Springer – Verlag, New York, 1993.

[16] Brown, M. B., Forsythe, A. B., Robust tests for equality of variances, *Journal of the American Statistical Association*, **69,** 1974, pp. 364 - 367.

[17] Buishand, T.A., Some methods for testing the homogeneity of rainfall records, *Journal of Hydrology*, **58**, 1982, pp. 11 - 27.

[18] Buishand, T. A., Tests for detecting a shift in the mean of hydrological time series, *Journal of Hydrology*, **63**, 1984, pp. 51 - 69.

[19] Carlin, B.P., Gelfand, A.E., Smith, A.F.M., Hierarchical Bayesian analysis of change-point problems, *Applied statistics*, **41**, 1992, pp. 389 - 405.

[20] Chandola, V., Banerjee, A. Kumar, V., Anomaly detection: A survey, *ACM Computing Surveys*, **41**(3), pp. 15:1 - 15:58, 2009.

[21] Chen, J., Gupta, A., *Parametric Statistical Change Point Analysis*, Birkhauser Verlag, 2000.

[22] Chernoff, H., Zacks. S., Estimating the current mean of a normal distribution which is subjected to change in time, *Annals of Mathematical Statistics*, **35,** 1964, pp. 999 - 1018.

[23] Cochran, W.G., The distribution of the largest of a set of estimated variances as a fraction of their total, *Annals of Human Genetics* (*London*), **11**(1), Jan. 1941, pp. 47 - 52.

[24] Cochrane, D., Orcutt, G. H., Application of Least Squares Regression to Relationships Containing Auto-Correlated Error Terms, *Journal of the American Statistical Association*, **44** (245), 1949, pp. 32 - 61.

[25] Conover, W. J., *Practical Nonparametric Statistics*, John Wiley & Sons, New York, 1971.

[26] Csorgo, M., Horvath, L., *Limit theorems in change-point analysis*, John Wiley & Sons, New York, 1997.

[27] David, H. A., Upper 5 and 1% points of the maximum F-ratio, *Biometrika*, **38**, 1952, pp. 422 - 424.

[28] Delgado, M. A., Testing the serial independence using the sample distribution function, *Journal of time series analysis*, **17**, 1996, pp. 271 - 285.

[29] Diks, C., Nonparametric tests for independence, http://www1.fee.uva.nl/cendef/upload/6/ ecss_diks_r1.pdf

[30] Dixon, W.J., Analysis of extreme values, *Annals of Mathematical Statistics*, **21**(2), 1950, pp. 488 - 506.

[31] Duggins, J., Williams, M., Kim, D-Y., Smith, E., Change point detection in SPI transition probabilities, *Journal of Hydrology*, **388**(3–4), 2010, pp. 456 - 463.

[32] Durbin, J., Testing for Serial Correlation in Least Squares Regression When Some of the Regressors Are Lagged Dependent Variables, *Econometrica*, **38**, 1970, pp. 410 - 421.

[33] Durbin, J., Watson, G. S., Testing for Serial Correlation in Least Squares Regression I, *Biometrika*, **37**, 1950, pp. 409 - 429.

[34] Durbin, J., Watson, G. S., Testing for Serial Correlation in Least Squares Regression II, *Biometrika*, **38**, 1951, pp. 159 - 178.

[35] Durbin, J., Watson, G. S., Testing for Serial Correlation in Least Squares Regression III, *Biometrika*, **71**, 1971, pp. 1 - 19.

[36] Embrechts, M., Maejima, P., *Self - similar processes*, Princeton, University Press, Princeton and Oxford, 2002

[37] Erdman, C., Emerson, J. W., bcp: An R Package for Performing a Bayesian Analysis of Change Point Problems, *Journal of Statistical Software*, **23**(3), 2007, pp. 1 - 13.

[38] Erramilli A., Narayan, O., Willinger, W., Experimental queueing analysis with long - range dependent packet traffic, IEEE/ACM *Transactions on Networking*, **4**(2), 1996, pp. 209 - 223.

[39] García, J. E., González-López, V. A., Independence tests for continuous random variables based on the longest increasing subsequence, *Journal of Multivariate Analysis*, **127**, 2013, pp. 126 - 146.

[40] Gedikli, A., Aksoy, H., Unal, N. E., AUG - Segmenter: a user-friendly tool for segmentation of long time series, *Journal of Hydroinformatics*, **12**(3), 2010, pp. 318 - 328.

[41] Gedikli, A., Aksoy, H., Unal, N. E., Kehagias, A., Modified dynamic programming approach for offline segmentation of long hydrometeorological time series, *Stochastic Environmental Research and Risk Assessment*, **24**(5), 2010, pp. 547 - 557.

[42] Genest, C., Nešlehová, J. G. Rémillard, B., On the estimation of Spearman's rho and related tests of independence for possibly discontinuous multivariate data, *Journal of Multivariate Analysis*, **117**, 2013, pp. 214 - 228.

[43] Geweke, J., Porter-Hudak, S., The estimation and application of long memory time series models, *Journal of Time Series Analysis*, **4**, 1983, pp. 221 - 238.

[44] Glejser, H., A new test for heteroscedasticity, *Journal of the American Statistical Association*, **64**(325), 1969, pp. 316 - 323.

[45] Goldfeld, S. M., Quandt, R. E., Some Tests for Homoscedasticity, *Journal of the American Statistical Association*, **60**(310), 1965, pp. 539 - 547.

[46] Grubbs F. E., Procedures for detecting outlying Observations in Samples, *Technometrics*, **11**(1), 1969, pp. 1-21.

[47] Hackl, P., Maderbacher, M., On the Robustness of the Rank-Based CUSUM Chart against Autocorrelation, 1999, http://epub.wu.ac.at/1764/1/document.pdf

[48] Hartley, H. O., The maximum F-ratio as a short cut test for heterogeneity of variance, *Biometrika*, **37**, 1950, pp. 308 - 312.

[49] Hawkins, D. M., *Identification of Outliers*, Chapman and Hall, London, 1980.

[50] Higuchi, T., Approach to an irregular time series on the basis of the fractal theory, *Physica D Nonlinear Phenomena*, **31**, 1988, pp. 277 - 283.

[51] Hines, W. G. S., Hines, R. J. O., Increased power with modified forms of the Levene (med) test for heterogeneity of variance, *Biometrics*, **56(2)**, 2000, pp. 451 - 454.

[52] Hodge, V. J., Austin, J., A survey of outlier detection methodologies, *Artificial Intelligence Reviews*, **22**(2), 2004, pp. 85 - 126.

[53] Hubert, P., The segmentation procedure as a tool for discrete modeling of hydrometeorogical regimes, *Stochastic Environmental Research and Risk Assessment*, **14**, 2000, pp. 297 304.

[54] Hubert, P., Carbonnel, J. P., Chaouche, A., Segmentation des séries hydrométéorologiques. Application à des séries de précipitations et de débits de l' Afrique de l'Ouest, *Journal of Hydrology*, **110**, 1989, pp. 349 - 367.

[55] Hurst, H. E., Long-term storage of reservoirs: an experimental study, *Transactions of the American Society of Civil Engineers*, **116**, 1951, pp. 770 - 799.

[56] James, N. A., Matteson, D. S., ecp: An R Package for Nonparametric Multiple Change Point Analysis of Multivariate Data, *Journal of Statistical Software*, **62**(7), pp. 1 - 25, http://www.jstatsoft.org/v62/i07/paper

[57] Jarque, C. M., Bera, A. K., Efficient tests for normality, homoscedasticity and serial independence of regression residuals, *Economics Letters*, **6** (3), 1980, pp. 255 - 259.

[58] Karagiannis, T., Faloutsos, M, SELFIS: A Tool For Self - Similarity and Long - Range Dependence Analysis, 2002, http://alumni.cs.ucr.edu/~tkarag/papers/kdd02.pdf

[59] Karagiannis, T., Faloutsos, M., Molle, M., A user-friendly self-similarity analysis tool, *ACM SIGCOMM Computer Communication Review*, Special Section on Tools and Technologies for Networking Research and Education **33**(3), 2003, pp. 81 - 93.

[60] Kehagias, A., Nidelkou, E., Petridis, V., A dynamic programming segmentation procedure for hydrological and environmental time series, *Stochastic Environmental Research and Risk Assessment*, **20**(1-2), 2006, pp. 77 - 94.

[61] Keogh, E., Kasetty, E., 2003. On the need for time series data mining benchmarks: A survey and empirical demonstration, *Data Mining and Knowledge Discovery*, **7**(4), pp. 349 - 371.

[62] Killick, R., Fearnhead, P., Eckley, I. A., Optimal detection of changepoints with a linear computational cost, *Journal of Americal Statistical Association*, **107**(500), 2012, pp. 1590 - 1598.

[63] Kolmogorov, A., Sulla determinazione empirica di una legge di distribuzione, *Giornale dell'Istituto Italiano degli Attuari*, **4**, 1933, pp. 83 - 91.

[64] Lee, A. F. S., Heghinian, S. M., A shift of mean level in a sequence of independent normal random-variables – Bayesian-approach, *Technometrics*, **19**, 1997, pp. 503 – 506.

[65] Levene, H., Robust tests for equality of variances. In: Olkin, I., Hotelling, H. et *al.* (eds.), Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling, Stanford University Press, 1960, pp. 278 - 292.

[66] Li, W., McLeod, A., Distribution of the residual autocorrelations in multivariate ARMA time series models, *Journal of the Royal Statistical Society*, *Series B*, **43**, 1981, pp. 231 - 239.

[67] Lilliefors, H., On the Kolmogorov–Smirnov test for normality with mean and variance unknown, *Journal of the American Statistical Association*, **62**, 1967, pp. 399 - 402.

[68] Lin, J., Keogh, E., Lonardi, S., Chiu, B., A symbolic representation of time series, with implications for streaming algorithms, In: DMKD '03: Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, 2003, pp. 2 - 11.

[69] Liu, P., Guo, S., Xiong, L., Chen, L., Flood season segmentation based on the probability change-point analysis technique, *Hydrological Sciences Journal*, **55**(4), 2010, pp. 540 - 554.

[70] Ljung, G. M., Box, G. E. P., On a measure of lack of fit in time series models, *Biometrika*, **65**, 1978, pp. 297 - 303.

[71] Lo, A. W., Long term memory in stock market prices, *Econometrica*, **59**, 1991, pp. 1279 - 1313

[72] Maddala, G. S., Lahiri, K., *Introduction to Econometrics*, Wiley, Chichester, 2009.

[73] Mandelbrot, B. B., Wallis, J. R., Robustness of the rescaled range R/S in the measurement of noncyclic long-run statistical dependence, *Water Resources*, **5**, 1969, pp. 967 - 988.

[74] Matilla - Garcıa, M., Rodriguez, J. M., Marin, M. R., A symbolic test for testing indepen - dence between time series, *Journal of Time Series Analysis*, **31**, 2010, pp. 76 - 85.

[75] Monti, A. C., A Proposal for Residual Autocorrelation Test in Linear Models, *Biometrika*, **81**, 1994, pp 776 -780.

[76] Moore, D. S., Tests of the chi-squared type. In: D'Agostino, R.B., Stephens, M.A. (eds.), Goodness-of-Fit Techniques, Marcel Dekker, New York, 1986.

[77] Nelson, L. S., Upper 10%, 5% and 1% points of the maximum F-ratio, *Journal of Quality Technology*, **19**(3), 1987, pp. 165 - 167.

[78] Neyman, J., Pearson, E. S., On the use and interpretation of certain test criteria, *Biometrika*, **20**, 1928, pp. 175 - 240.

[79] Noguchi, K., Gel, Y. R., Combination of Levene-type tests and a finite-intersection method for testing equality of variances against ordered alternatives. Working paper, Department of Statistics and Actuarial Science, University of Waterloo, 2009.

[80] O'Brien, R. G., Robust techniques for testing heterogeneity of variance effects in factorial designs, *Psychometrika*, **43**, 1978, pp. 327 - 344.

[81] Pearson, K., On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling, *Philosophical Magazine*, **5**, 1900, pp. 157 - 175.

[82] Peng, C., Buldyrev, S., Havlin, S., Simons, M., Stanley, H., Goldberger, A., Mosaic organization of DNA nucleotides, *Physical Reviews E*, **49**(2), 1994, pp. 1685 - 1689.

[83] Peng, C. K., Mietus, J., Hausdorff, J. M., Havlin, S., Stanley, H. E., Goldberger, Al., Long-range anti-correlations and non-Gaussian behavior of the heartbeat, *Physical Review Letter*, **70**, 1993, pp. 1343 - 1346.

[84] Pettitt, A. N., A non - parametric approach to the change - point problem, *Applied Statistics*, **28**(2), 1979, pp. 126 - 135.

[85] Pinkse, J., A consistent nonparametric test for serial independence, *Journal of Econome -trics*, **84**, pp. 205 – 231.

[86] Ray, B., Tsay, R., Bayesian methods for change-point detection in long - range dependent processes, *Journal of Time Series Analysis*, **23**, 2002, pp. 687 - 705.

[87] Razali, N. M., Wah, B.Y., Tests for Normality: Comparison of Powers, Tests for Normality: Comparison of Powers, *Journal of Statistical Modeling*, **2**(1), 2011, pp. 2 1 - 33.

[88] Robinson, P. M., Consistent nonparametric entropy-based testing, *The Reviews of Economic Studies*, **58**, 1991, pp. 437 - 453.

[89] Robinson, P. M., Gaussian semiparametric estimation of long-range dependence, *Annals of Statistics*, **23**, 1995, 1630 - 1661.

[90] Robson, A., Analysis guidelines. In Detecting trend and other changes in hydrological data, Kundzewicz, Z. W., Robsson, A. (eds.), WCDMP – 45, WMO/TD-No. 1013, 2000, pp. 11 -14.

[91] Rosner, B., Percentage Points for a Generalized ESD Many - Outlier Procedure, *Technome -trics*, **25**(2), 1983, pp. 165 - 172.

[92] Ross, G.J., Parametric and Nonparametric Sequential Change Detection in R: The cpm package, www.gordonjross.co.uk/cpm.pdf.

[93] Scheffé, H., *The analysis of variance*, Wiley, New York, 1959.

[94] Scott, A. J., Knott, M., A Cluster Analysis Method for Grouping Means in the Analysis of Variance, *Biometrics*, **30**(3), 1974, pp. 507 - 512.

[95] Shapiro, S. S., Wilk, M. B., An Analysis of Variance Test for Normality (Complete Samples), *Biometrika,* **52**, no. 3/4, 1965, pp. 591 - 611.

[96] Smirnov, N., Table for estimating the goodness of fit of empirical distributions, *Annals of Mathematical Statistics*, **19**, 1948, pp. 279 - 281.

[97] Steele, M., Chaseling, J., Goodness-of-Fit Tests Powers of Discrete Goodness-of-Fit Test Statistics for a Uniform Null Against a Selection of Alternative Distributions, *Communications in Statistics - Simulation and Computation*, **35,** 2006, pp. 1067 - 1075.

[98] Stephens, M. A., Tests based on EDF statistics. In: D'Agostino, R.B. and Stephens, M.A. (eds), Goodness-of-Fit Techniques, Marcel Dekker, New York, 1986.

[99] Szekely, G. J., Rizzo, M. L., The distance correlation t-test of independence in high dimension, *Journal of Multivariate Analysis*, **117**, 2013, pp. 193 - 213.

[100] Szekely, G. J., Rizzo, M. L., Bakirov, N. K., Measuring and testing dependence by correlation of distances, *The Annals of Statistics*, **35**(6), 2007, pp. 2769 - 2794.

[101] Taqqu, M. S., Teverovsky, V., Testing for long-range dependence in the presence of shifting means or a slowly declining trend, using a variance-type estimator, *Journal of Time Series Analysis*, **18**(3), 1997, pp. 279 – 304.

[102] Taqqu, M. S., Teverovsky, V., Willinger, W., Estimators for Long-Range Dependence: an Empirical Study, *Fractals*, **3**(4), 1995, pp.785 - 798.

[103] Tsakalias, G., Koutsoyiannis, D., A comprehensive system for the exploration and analysis of hydrological data, *Water Resources Management*, **13**(4), 1999, pp. 269 - 302.

[104] Wang, G., Zou, C., Wang, Z., Necessary test for complete independence in high dimensions using rank-correlations, *Journal of Multivariate Analysis*, **121**, 2013, pp. 224 - 232.

[105] Warrenliao, T., Clustering of time series data – a survey, *Pattern Recognition*, **38**(11), 2005, pp. 1857 - 1874.

[106] Weron, R., Estimating long range dependence: finite sample properties and confidence intervals, *Physica A*, **312**, 2002, pp. 285 - 299.

[107] White, H., A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity, *Econometrica*, **48**(4), 1980, pp. 817- 838.

[108] Zhang, Y., Meratnia, N., Havinga, P., Outlier Detection Techniques for Wireless Sensor Networks: A Survey, *IEEE Communications Surveys & Tutorials*, **12**(2), Second Quarter, 2010, pp. 1 - 12.

[109] http://cran.r-project.org/web/packages/cpm/cpm.pdf

[110] http://cran.r-project.org/web/packages/fArma/fArma.pdf

[111] http://cran.r-project.org/web/packages/fBasics/fBasics.pdf

[112] http://cran.r-project.org/web/packages/fractal/fractal.pdf

[113] http://cran.r-project.org/web/packages/ecp/ecp.pdf

[114] http://cran.r-project.org/web/packages/energy/energy.pdf

[115] http://cran.r-project.org/web/packages/het.test/het.test.pdf

[116] http://www.inside-r.org/packages/cran/lawstat/docs/levene.test

[117] http://cran.r-project.org/web/packages/lmtest/lmtest.pdf

[118] http://cran.r-project.org/web/packages/nortest/nortest.pdf

[119] http://cran.r-project.org/web/packages/outliers/outliers.pdf

[120] http://cran.r-project.org/web/packages/portes/portes.pdf

[121] http://cran.r-project.org/web/packages/pracma/pracma.pdf

[122] http://www.itl.nist.gov/div898/handbook/eda/section3/eda35h2.r

[123] http://www.itl.nist.gov/div898/handbook/eda/section3/eda35h3.r