MASTER THESIS IN MATHEMATICS /APPLIED MATHEMATICS

Program realization of statistical test for normality in Java

by

Rafael Vides

Magisterarbete i matematik / tillämpad matematik
Master thesis in mathematics / applied mathematics

Date:
2005-06-20

Project name:
Program realization of statistical test for normality in Java

Author:
Rafael Vides

Supervisors:
Dmitrii Silvestrov and Anatoliy Malyarenko

 Examiner:
Dmitrii Silvestrov

Comprising:
20 points
Invictus

Out of the night, that covers me,
Black as the Pit from pole to pole,
I thank whatever gods may be
For my unconquerable soul.

In the fell clutch of circumstance
I have not winced nor cried aloud.
Under the bludgeonings of chance
My head is bloody, but unbowed.

Beyond this place of wrath and tears
Looms but the Horror of the shade,
And yet the menace of the years
Finds, and shall find, me unafraid.

It matters not how strait the gate,
How charged with punishments the scroll,
I am the master of my fate:
I am the captain of my soul.

By Thomas William Ernest Henley
(1849–1903).
Acknowledgments

This Master Thesis is dedicated

to my ancestor.

to my parents that I never knew.

to my aunt Blanquita with all the love that a son has, thanks for your upbringing.

to my lovely son and heir.

to my family around the world.

to Margit and her family, with love.

to my mentors Dmitrii Silvestrov and Anatoliy Malyarenko, thanks for your advices.
Abstract

Statistics can be used in many areas of problem solving. The main ones are significance and hypothesis testing to answer questions of comparison. The chi-square test is a statistical test that can be used to determine whether observed frequencies are significantly different from expected frequencies.

This Master Thesis contain a Java Application that reads real financial data and perform the chi square test if the data follows a normal distribution. The test is tailored specifically to the nature of the data structures involved.

I had followed, observed and written down day-to-day the development of the share price of six Swedish companies: Asea Brown Broveri, Ericsson, FöreningsSparbanken, Skandinavia Enskilda Banken, Svenska Kullager Fabrik, Volvo, the data used in this project was collected between the years 2003 and 2004 from the OMX-group web site in Sweden.

I would like to express my deepest gratitude to Bengt Jansson from the OMX-group who provides me with essential information in order to make a success in the realization of this project.
# Table of Content

1. Introduction ........................................................................................................... 1

2. Hypothesis Testing ................................................................................................. 1

3. Test of Goodness of Fit ......................................................................................... 1

4. Chi Square Test ...................................................................................................... 2
   4.1. Complete satisfied Hypothetical Distribution ........................................... 2
   4.2. Test of Normal Distribution ......................................................................... 2

5. Critical Values of the Chi Square Distribution ................................................. 5
   5.1. Approximation of Functions .......................................................................... 5
   5.2. Gorner’s scheme ............................................................................................ 7

6. Description of the Companies ............................................................................ 8
   6.1 Asea Brown Boveri ...................................................................................... 8
   6.2 Ericsson ......................................................................................................... 8
   6.3 FöreningsSparbanken .................................................................................. 8
   6.4 Skandinavia Enskilda Banken ...................................................................... 9
   6.5 Svenska Kullager Fabrik .............................................................................. 9
   6.6 Volvo ............................................................................................................. 9

7. Checking for Normality Application Program ................................................. 10
   7.1 Normality class ............................................................................................. 10
   7.2 Normality Filter class ................................................................................... 10
   7.3 File Open Listener class ................................................................................ 10
   7.4 File Exit Listener class .................................................................................. 10
   7.5 Logarithm return histogram ........................................................................ 10

8. User’s guide ........................................................................................................... 11

9. Working with Checking for Normality Graphic User Interface ................. 13

10. Mathematical results ......................................................................................... 17

11. Conclusion ............................................................................................................ 22

12. References ........................................................................................................... 23

13. Appendix ............................................................................................................. 24
1. Introduction

Chi-square test is a widely used statistical method for an extremely broad range of applications, which shares one common feature that is the availability of sufficient sample size for them to have approximately a chi-square distribution.

The chi-square is one of the most popular statistics because it is easy to calculate and interpret, its purpose is to determine whether the observed frequencies markedly differ from the frequencies that we would expect by chance.

Chi-square is a statistical test commonly used to compare observed data with data we would expect to obtain according to a specific hypothesis.

Chi-square distribution is used for sample sizes of 50 or more values. Due to sampling variation, the observed numbers may not equal the expected numbers. The Chi square test is used to compare how well the sample fits the expected values and provides a measure of discrepancy between the data and the model.

2. Hypothesis Testing

A major part of classical statistics is hypothesis testing and the objectives is to estimates the respective parameters and the test of hypothesis about their values about the unknown population parameters based on information contained in sample data.

Hypothesis tests are conducted in all fields in which theory can be tested against observation.

3. Test of Goodness of Fit

In general, the term goodness of fit is associated with the statistical testing of hypothetical models with data. The Chi Square distribution is a mathematical distribution that is used directly or indirectly in many tests of significance.

The chi square distribution has one parameter, its degrees of freedom (df). It has a positive skew, if one of its tails is longer than the other its observed distributions with positive skew are sometimes called skewed to the right whereas distributions with negative skew are called skewed to the left, the skew is less with more degrees of freedom.

The Chi Square test consists of three different types of analysis

1) Goodness of fit: The test for Goodness of fit determines if the sample under analysis was drawn from a population that follows some specified distribution.

2) Test for Homogeneity: The Test for Homogeneity answers the proposition that several populations are homogeneous with respect to some characteristic.

3) Test of Independence is for testing the null hypothesis that two criteria of classification, when applied to a population of subjects are independent. If they are not independent then there is an association between them.
4. Chi Square Test $\chi^2$

4.1 Completely satisfied hypothetical distribution.

Being a statistical test, Chi Square can be expressed as a formula

$$\chi^2 = \sum_{i=1}^{r} \left( \frac{\nu_i - n p_i}{n p_i} \right)^2 = \sum \frac{\nu_i^2}{n p_i} - n$$

where:

- $r$ is a finite number of parts $S_1, S_2, S_3, \cdots, S_r$
- $p_i$ is the corresponding values of the probability function, assuming that $p_i > 0$
- $\nu_i$ is the corresponding group frequencies in the sample belongs to the set $S_i$.
- $n$ is the total sample

Thus, $\chi^2$ is simple expressed in terms of the observed frequencies $\nu_i$ and the expected frequencies $n p_i$ for all $r$ groups.

4.2 Test of Normal Distribution

In practice, samples are very often grouped into intervals of length $h$ with the mid-points $\xi_i = \xi_0 + ih$ where $i = 0, \pm 1, \pm 2, \cdots$. In such cases it is usual to assume that all samples values belonging to a certain interval fall in the mid-point of that interval.

We are then in reality sampling from a distribution of the discrete type, where the variable may take any value $\xi_i = \xi_0 + ih$ with the probability

$$p_i = \int_{\xi_i - \frac{1}{2}h}^{\xi_i + \frac{1}{2}h} f(x) \, dx$$

Let a sample on $n$ values $x_1, x_2, \cdots, x_n$ be grouped into $r$ classes, the $i$th containing $\nu_i$ observation situated in the interval $\left( \xi_i - \frac{1}{2}h, \xi_i + \frac{1}{2}h \right)$, where $\xi_i = \xi_1 + (i-1)h$.

I want to test the hypothesis that the sample has been drawn from some normal population, with unknown values of the parameters $m$ and $\sigma$. 

Rafael Vides
If the hypothesis is true, the probability $p_i$ corresponding to the $i:th$ class is
\[
p_i = \frac{1}{\sigma \sqrt{2\pi}} \int e^{-\frac{(x-m)^2}{2\sigma^2}} dx
\]
where: the integral is expressed over the $i:th$ class interval.

For the two extreme classes ($i = 1 and i = r$), the intervals should be $(-\infty, \xi_1 + \frac{1}{2} h)$ and $(\xi_r - \frac{1}{2} h, +\infty)$ respectively.

I have written for brevity
\[
g(x) = e^{-\frac{(x-m)^2}{2\sigma^2}}
\]
\[
\frac{\partial p_i}{\partial m} = \frac{1}{\sigma^3 \sqrt{2\pi}} \int (x-m) g(x) dx
\]
\[
\frac{\partial p_i}{\partial \sigma} = \frac{1}{\sigma^4 \sqrt{2\pi}} \int (x-m)^2 g(x) dx - \frac{p_i}{\sigma}
\]

The equation $\sum_{i=1}^{r} \frac{\nu_i}{p_i} \frac{\partial p_i}{\partial \sigma_j} = 0$, then give after some simple reductions, all integrals being extended over the respective class intervals specified above,
\[
m = \frac{1}{n} \sum_{i} \nu_i \frac{\int x g(x) dx}{\int g(x) dx}
\]
\[
\sigma^2 = \frac{1}{n} \sum_{i} \nu_i \frac{\int (x-m)^2 g(x) dx}{\int g(x) dx}
\]
We first assume that the grouping has been arranged such that the two extreme classes do not contain any observed values. We then have $V_1 = V_r = 0$.

For small values of $h$, an approximated solution may be obtained by replacing the functions under the integrals by their values in the mid-point $\xi_i$ of the corresponding class interval.

In this way we obtain estimates $m^*$ and $\sigma^*$ given by the expressions

$$m^* = \frac{1}{n} \sum_i v_i \xi_i$$

$$\sigma^{*2} = \frac{1}{n} \sum_i v_i \left( \xi_i - m^* \right)^2$$

Thus, the $m^*$ and $\sigma^{*2}$ are identical with the mean $\bar{x}$ and the variance $s^2$ of the grouped sample, calculated according to the usual rule $\bar{x} = \sum_{-\infty}^{\infty} p_i \xi_i$, $\sigma^2 = \int_{-\infty}^{\infty} (x - m)^2 f(x) dx$, that all sample values in a certain class are placed in the mid-point of the class interval.

In order to obtain a closer approximation, we may develop the functions under the integrals in Taylor’s series about the mid-point $\xi_i$. For small $h$, we then find by some calculation that the above formulae should be amended as follows:

$$m^* = \frac{1}{n} \sum_i v_i \xi_i + O\left(h^4\right)$$

$$\sigma^{*2} = \frac{1}{n} \sum_i v_i \left( \xi_i - m^* \right)^2 - \frac{h^2}{12} + O\left(h^4\right)$$

Even when $h$ is not very small, and when the extreme classes are not actually empty, but contain only a small part of the total sample, the same procedure will lead to a reasonable approximation.
5. Critical Values for the Chi Square distribution

In practical applications of various test of significance, the 5%, 1%, and 0.1% levels of significance are often used. Which level we adopt in a given case will depend on the particular circumstances of the case.

The conventional terminology is as follow the first one a value exceeding the 5% limit but not the 1% limit as almost significant, the second one a value between the 1% and 0.1% limits as significant and the third one a value exceeding the 0.1% limit as highly significant.

In the Java application program, you can find the critical values of the chi square distribution in the combo box labelled select confidence level, in this way you have the alternative to conduct your requests choosing between 99%, 95 %, and 90%.

The application program uses the critical values showing in the table 3 on page 559 from [1].

5.1 Approximation of Functions

Recall that

\[ F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{y^2}{2}} \, dy , \]

\[ F(x) = 1 - \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \sum_{j=1}^{5} a_j d^j , \]

where \( d = \frac{1}{1 + 0.3316419 x} \)

and \( a_1 = 0.31938153 \quad a_2 = -0.356563782 \)
\( a_3 = 1.781477937 \quad a_4 = -1 - 821255978 \)
\( a_5 = 1.330274429 \)

Recall that the value of the probability \( p_i = \frac{1}{\sigma \sqrt{2\pi}} \int e^{-\frac{(x-m)^2}{2\sigma^2}} \, dx . \)

Assume that \( y = \frac{x-m}{\sigma} \), \( x = y \sigma + m \) and \( dx = \sigma dy \).
Now, rewritten $p_i$, such as:

\[
p_i = \frac{1}{\sigma \sqrt{2\pi}} \int_{a}^{b} e^{-\frac{(x-m)^2}{2\sigma^2}} \, dx
\]

Substituting the values of $y = \frac{x-m}{\sigma}$, $x = y \sigma + m$, and $dx = \sigma dy$ in the equation of $p_i$.

We obtain

\[
p_i = \frac{1}{\sigma \sqrt{2\pi}} \int_{a-m}^{b-m} e^{-\frac{y^2}{2}} \sigma dy
\]

\[
p_i = \frac{1}{\sigma \sqrt{2\pi}} \int_{a-m}^{b-m} e^{-\frac{y^2}{2}} dy
\]

\[
p_i = \frac{1}{\sqrt{2\pi}} \int_{a-m}^{b-m} e^{-\frac{y^2}{2}} dy
\]

\[
p_i = F\left(\frac{b-m}{\sigma}\right) - F\left(\frac{a-m}{\sigma}\right), \text{ where}
\]

Rafael Vides

6
\[ F(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} \, dy \]

\[ F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{y^2}{2}} \, dy \]

The last formulae is useful in the approximation of function.

5.2 Gorner’s scheme

In order to calculate the value of the probability, the use of Gorner’s scheme is appropriate in the computation of polynomials.

\[ p = A_n \, d^n + A_{n-1} d^{n-1} + \cdots + A_1 \, d + A_0 \]

Gorner’s scheme works as follow:

\[ p = A_n \quad \text{initial value.} \]

\[ p = A_n \, d + A_{n-1} \quad \text{after the first passage.} \]

\[ p = A_n \, d^2 + A_{n-1} d + \cdots + A_{n-2} \]

\[ \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \]

\[ p = A_n \, d^n + A_{n-1} d^{n-1} + \cdots + A_1 \, d + A_0 \]
6. Description of the Companies

6.1 Asea Brown Boveri ABB-Group

The history of ABB goes back to the late nineteenth century, and is a long and illustrious record of innovation and technological leadership in many industries.

ABB Corporation with their technology leadership in power automation technologies, global presence, application knowledge and local expertise, offers products, systems, solutions and services that allows customers to improve their operations while lowering environmental impact. In recent years, ABB has gone over from large-scale solutions to alternative energy and the advanced products and technologies in power and automation that constitute its Industrial IT offering.

6.2 Ericsson

Ericsson Company has been active worldwide since 1876, and today it’s is the world’s 10 largest mobile operators, which supply mobile systems in the world and some 40% of all mobile calls are made through their systems. The company supports all major standards for wireless communication driving the telecoms industry and providing total solutions from systems and applications to services and core technology for mobile handsets. Merge Sony Ericsson the company expands their supplier of complete mobile multi-media products.

Ericsson is today present in more than 140 countries, and its headquarters are located in Stockholm, Sweden.

6.3 FöreningsSparbanken

The bank's composite history dates all the way back to 1820, when Sweden's first savings bank was founded, in Gothenburg, on a European model. The savings bank idea rapidly took hold in Sweden, with a peak of 498 savings banks in 1928. After that, the savings banks began merging to become stronger.

FöreningsSparbanken was founded in 1997, by the merger of Föreningsbanken and Sparbanken Sverige.

Föreningsbanken had its origins in farming cooperative credit societies whose purpose was to satisfy Swedish agriculture's growing need of capital. Sweden's first farming cooperative credit society was founded in 1915 in Västerhaninge outside Stockholm.

Sparbanken Sverige was formed, in 1992, nearly 90 savings banks chose to continue being independent banks and to collaborate with Sparbanken Sverige instead.
6.4 Skandinavia Enskilda Banken SEB

The SEB Group is a North European financial banking group for companies, institutions and private individuals, with 670 branch offices around Sweden, Germany and the Baltic States.

SEB has more than 4 million customers, of whom 1.5 million are e-banking customers. The Group is represented in some 20 countries around the world and has a staff of about 18,000.

6.5 Svenska Kullager Fabrik SKF

SKF deliver products, solutions and services in the ball bearing and tightening branch.

SKF Corporation is divided in five divisions: Industrial, Automobile, Electrical, Aero Steel and Service.

SKF Concern is world leader in its branch having 40000 employees, which 4800 are in Sweden.

6.6 Volvo

Volvo Group is one of the world's leading suppliers of transport solutions for commercial use providing complete solutions for financing and service.

Volvo Group offers transport solutions to demanding customers around the world, their broad range of trucks, construction equipment and buses are built to work for customer, the concern also provide engines for leisure boats and workboats as well as diesel-powered generator sets.

In the aerospace industry the Volvo Group has broad operations including financing, leasing, insurance, action services, warranty, rentals, IT solutions and logistical operations.
7. Checking for Normality Application Program

The Program realization of statistical test for normality in Java is built in an easy way it allows the user to select different parameters from different combo boxes in order to perform statistics test called the Chi Square distribution.

Checking for Normality Application with a graphical user interface (GUI) include a selection of block of codes called methods, which have the responsibilities to do something when the user clicks on a radio button, calculate button otherwise select something from the combo boxes.

You can observe in the Checking for Normality Application that the main screen list visible objects such as Menu object, Frame object, Title object, Border object, Text area object in which each object has its own characteristics or state. The design for a software object is called a class, every class contain variables for data information and methods for manipulation of that data and information including the details how an object should appear and how it behaves.

The checking for normality application program contains packages using an import statement at the top of the class. These import statement allow easy access of the predefine classes in those packages. Import statements give you a shortcut to class names in other packages.

The classes listed below counting in the checking for normality application are:

7.1 Normality Class

Public class Normality extends Java frame and implements Item Listener and Action Listener. In this class, you can read all the variables and methods counting in the Checking for Normality Application program.

7.2 Normality Filter Class

Public class Normality Filter extends file filter, this class is responsible for the popup window in which appears the name of the company that the user had selected.

7.3 File Open Listener Class

Public class File Open Listener implements Action Listener and implement an action event, this class is responsible for the open button and menu item which give you access to the normality file which contain the *.nrm suffix.

7.4 File Exit Listener Class

Public class File Exit Listener implements Action Listener and performs an action event, in this case the exit application is responsible for close the application program.

7.5 Logarithm return histogram

The histogram of the logarithm return is made with the help of two packages found in the java archives, the first one is Jfreechart-1.0.0.pre.jar and the second is Jcommon-1.0.0-pre2.jar, these two packages have a special format in which the java system can found all the necessary classes to make the histogram. You can found these packages at jfreechart.org home page[3].
8. User’s guide

In order to facilitate the use of the Java application program, it is important to keep in mind some concepts. The graphic user interface is designed in an easy way, and it is divided in two panels:

**The first panel to the left, which contains:**

The radio button panel which contain two radio buttons labelled 1 year and 3 months, the first one can perform the test using one year data, and the second one execute quarterly test.

The year panel that indicates which year you had selected in this case the application program contains only the collection of information including the year 2003 and the year 2004.

The left confidence panel show you the level of significance that you had selected.

The last part is the result panel in this part you can read the result of the chi square test, if the hypothesis of normality is true or false.
The second panel to the right, which contains:

The logarithmic return panel in this appears the histogram of the chi square test, by inspection you can appreciate if the result follows a normal curve.

The start panel contains the calculate button that is enable when you select the company that you want to test.

The right confidence panel is designed in a combo box in which you can select the level of significance you want to apply to your test, the possibility is to choose between three different confidence intervals the first is 99 percent, the second is 95 percent, the third is 90 percent.

The price panel combo box gives you the possibility to select between open prices, high prices, low prices and close prices.

The right company panel contains a combo box in which it appears the name of the six Swedish companies, the first one is Asea Brown Boveri, the second one is Ericsson, the third is Förenings Sparbanken, the fourth one is Skandinavian Enskilda Bank, the fifth one is Svenska Kullager Fabrik, and the sixth one and last is Volvo.
9. Working with the Normality Graphic User Interface

In order to perform the Chi Square test you must select from the combo boxes the name of the company, the price that you want to open, the year that you want to study, finally you can decide your confidence level. Click on File then select Open and another popup window appears, which show you the normality file of the company selected, click on Open button and the program runs and gives you the answer.

Checking for normality user interface has the possibility to start to run the application when you click on the open icon and follow the instructions written above.

The Application program checking for normality gives the user the possibility to perform the test yearly selecting the 1-year radio button or quarterly clicking on the 3 months radio button, in the last alternative the user can select the month to start the initiation of the Chi square test.
You can observe that in the open popup window appears the normality file, which contain the name of the company that you had already chose, you must select then and click on the open button.

The following open window contains the name of the company Asea Brown Boveri ABB.

The following window contains another name of the second company including in this project.
Now, after selecting the name of the company in the open popup window, the user can observe that the calculate button is enable and in order to perform the Chi-square test the only that the user need to do is to click on it, and the program runs.

The application program works, and after a few seconds the user can analyze the answer in the logarithmic return panel, in the same way the user can read in the result panel if the data of the company follows an hypothesis about their normal distribution.
The following graphic user interface shows you the result of the company Asea Brown Boveri, the test is performed using the financial data of the year 2003.

Important remark, if you want to test another company, please start to use the Checking for Normality application in the same manner that it is explained in the user’s guide.
10. Mathematical Results

Java Application program Checking for Normality has the possibility to make some calculation when the program is executing in RealJ. In order to have some information, I have include some tables with values.

The notation including in the tables is as follow:

$r$ means the groups into which our sample values have been arranged for tabulation purposes.

$\chi^2$ Chi Square value.

The names of the companies have been abbreviated to:

ABB Asea Brown Boveri,
ERIC Ericsson,
FSPA FöreningsSparbanken,
SEB Skandinavia Enskilda Bank,
SKF Svenska Kullager Fabrik,
VOLV Volvo.

The first result is performed using the financial data of the year 2003, the selecting price is closing price, and the confidence level is 99%.

<table>
<thead>
<tr>
<th>$r$</th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>92.14</td>
<td>113.57</td>
<td>120.14</td>
<td>92.50</td>
<td>71.29</td>
<td></td>
</tr>
</tbody>
</table>

The normality hypothesis is False False False False False False False

The first result is performed using the financial data of the year 2004, the selecting price is closing price, and the confidence level is 99%.

<table>
<thead>
<tr>
<th>$r$</th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>105.88</td>
<td>135.59</td>
<td>142.83</td>
<td>74.80</td>
<td>71.78</td>
<td></td>
</tr>
</tbody>
</table>

The normality hypothesis is False False False False False False False
Another way to execute the Chi Square test Checking for Normality is quarterly, using the data for the financial year 2003, selected price is the closing price, with a 99% level of confidence.

Quarterly performed test start with the name of the month and finish four months ahead.

The result is shown in the following table:

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>January</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$r$</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>$\chi^2$</td>
<td>14.35</td>
<td>15.15</td>
<td>15.37</td>
<td>17.51</td>
<td>14.70</td>
<td>16.22</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>February</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$r$</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>$\chi^2$</td>
<td>24.88</td>
<td>26.21</td>
<td>27.11</td>
<td>11.20</td>
<td>5.41</td>
<td>12.24</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td><strong>TRUE</strong></td>
<td>False</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>March</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$r$</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>$\chi^2$</td>
<td>24.87</td>
<td>25.20</td>
<td>17.97</td>
<td>9.37</td>
<td>4.59</td>
<td>14.39</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td><strong>TRUE</strong></td>
<td>False</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>April</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$r$</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$\chi^2$</td>
<td>27.70</td>
<td>29.85</td>
<td>35.68</td>
<td>8.82</td>
<td>5.59</td>
<td>3.09</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td><strong>TRUE</strong></td>
<td><strong>TRUE</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>May</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$r$</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>$\chi^2$</td>
<td>14.33</td>
<td>21.05</td>
<td>14.20</td>
<td>14.59</td>
<td>14.03</td>
<td>24.94</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>June</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$r$</td>
<td>3</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>$\chi^2$</td>
<td>16.70</td>
<td>21.55</td>
<td>98.27</td>
<td>17.78</td>
<td>15.16</td>
<td>21.46</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>Month</td>
<td>ABB</td>
<td>ERIC</td>
<td>FSPA</td>
<td>SEB</td>
<td>SKF</td>
<td>VOLV</td>
</tr>
<tr>
<td>----------</td>
<td>-----</td>
<td>------</td>
<td>------</td>
<td>-----</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>July</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td>16.54</td>
<td>19.62</td>
<td>4.34</td>
<td>16.35</td>
<td>16.15</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td><strong>TRUE</strong></td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>August</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td>19.29</td>
<td>20.90</td>
<td>3.86</td>
<td>21.92</td>
<td>14.89</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td><strong>TRUE</strong></td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>September</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td>17.09</td>
<td>16.96</td>
<td>25.90</td>
<td>23.86</td>
<td>13.95</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>October</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td>13.68</td>
<td>16.07</td>
<td>24.66</td>
<td>26.32</td>
<td>12.05</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>
Another way to execute the Chi Square test Checking for Normality is quarterly, using the data for the financial year 2004, selected price is the closing price, with a 99% level of confidence.

Quarterly performed test start with the name of the month and finish four months ahead.

The result is shown in the following table:

<table>
<thead>
<tr>
<th>Month</th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td>January</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>16.56</td>
<td>16.30</td>
<td>17.16</td>
<td>7.37</td>
<td>16.30</td>
<td>18.50</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>February</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>28.94</td>
<td>19.17</td>
<td>10.57</td>
<td>14.61</td>
<td>14.59</td>
<td>17.88</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>March</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>27.80</td>
<td>18.63</td>
<td>9.06</td>
<td>5.98</td>
<td>14.24</td>
<td>17.24</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>TRUE</td>
<td>False</td>
</tr>
<tr>
<td>April</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>35.39</td>
<td>18.87</td>
<td>7.67</td>
<td>11.40</td>
<td>14.16</td>
<td>4.74</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>TRUE</td>
</tr>
<tr>
<td>May</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>18.66</td>
<td>16.75</td>
<td>9.95</td>
<td>14.82</td>
<td>19.42</td>
<td>15.86</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>June</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$\chi^2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>17.01</td>
<td>18.77</td>
<td>24.99</td>
<td>18.78</td>
<td>17.43</td>
<td>17.60</td>
</tr>
<tr>
<td></td>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>
### July

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r )</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>( \chi^2 )</td>
<td>18.16</td>
<td>17.33</td>
<td>25.33</td>
<td>20.08</td>
<td>18.05</td>
<td>16.66</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>

### August

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r )</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>( \chi^2 )</td>
<td>16.41</td>
<td>19.86</td>
<td>26.42</td>
<td>26.12</td>
<td>17.40</td>
<td>17.33</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>

### September

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r )</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>( \chi^2 )</td>
<td>14.73</td>
<td>16.83</td>
<td>27.13</td>
<td>28.05</td>
<td>19.58</td>
<td>18.07</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>

### October

<table>
<thead>
<tr>
<th></th>
<th>ABB</th>
<th>ERIC</th>
<th>FSPA</th>
<th>SEB</th>
<th>SKF</th>
<th>VOLV</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r )</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>( \chi^2 )</td>
<td>17.50</td>
<td>17.50</td>
<td>23.44</td>
<td>21.81</td>
<td>22.24</td>
<td>21.03</td>
</tr>
<tr>
<td>the normality hypothesis is</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>
Conclusion

Chi-square tests enable us to compare observed and expected frequencies objectively, since it is not always possible to tell just by looking at them whether they are different enough to be considered statistically significant which in this case implies that the differences are not due to chance alone, but instead may be indicative of other processes at work.

Chi Square is used to determine if there is reason to reject the statistical hypothesis that the frequencies in a random sample are as expected when the items are from a normal distribution.

In this project the Chi Square test for yearly financial data fails in both years. The result in the quarterly intervals is as follow: for the year 2003 six cases were true, and for the year 2004 two cases were true, that it means that the data ensue a normal curve.
References


Normality class

/**
 * File: Normality.java
 * Description: The Normality class is an application program that reads real
 * financial data and perform the statistics test called Chi Square and gives
 * you the answer if the data follows or not a normal distribution.
 * 
 * Author: Rafael Vides
 * 
 * e-Mail : rvs01001@student.mdh.se
 */

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import org.jfree.data.statistics.*;
import org.jfree.chart.*;
import org.jfree.chart.plot.*;
import org.jfree.util.*;
public class Normality extends JFrame implements ItemListener,
        ActionListener
{
    private final int WIDTH = 800;
    private final int HEIGHT = 700;

    private JPanel mainPanel = null;
    private JPanel leftCompanyPanel = null;
    private JPanel centralPanel = null;
    private JPanel leftPanel = null;
    private JPanel rightPanel = null;
    private JPanel theoreticalChartPanel = null;
    private JPanel historicalChartPanel = null;
    private JPanel resultPanel = null;
    private JPanel leftConfidencePanel = null;
    private ChartPanel logarithmicPanel = null;
    private JPanel rightConfidencePanel = null;
    private JPanel yearPanel = null;
    private JPanel rightCompanyPanel = null;
    private JPanel startPanel = null;
    private JPanel pricePanel = null;

    private JTextField confidenceField = null;

    private JTextArea results = null;

    private BoxLayout centralPanelLayout = null;
    private BoxLayout leftPanelLayout = null;
    private BoxLayout rightPanelLayout = null;

    private String companyString = "Asea Brown Boveri.st";
    private String yearString = "Selected year: ";
    private String confidenceString = "confidence level";
    private String startString = "Calculate";
private String[] confidence = {"99%", "95%", "90%"};
private String[] years = {"2003", "2004"};
private String selectYearString = "  Select year";
private String selectCompanyString = "  Select company 's share";
private String selectPriceType = "  Select price type";
private String[] priceTypes = {"Open", "High", "Low", "Close"};
private String shortCompany0 = "ABB";
private String shortCompany1 = "Ericsson-b";
private String shortCompany2 = "Fspa-a";
private String shortCompany3 = "SEB-a";
private String shortCompany4 = "SKF-b";
private String shortCompany5 = "VOLV-b";
private String[] interval = {"1 year", "3 months");

private JLabel upCompanyLabel = null;
priate JLabel leftYearLabel = null;
private JLabel leftConfidenceLabel = null;
private JLabel rightConfidenceLabel = null;
private JLabel rightCompanyLabel = null;
private JLabel rightYearLabel = null;
private JLabel selectPriceLabel = null;
private JButton startButton = null;
private JComboBox confidenceBox = null;
private JComboBox yearBox = null;
private JComboBox companyBox = null;
private JComboBox priceBox = null;
private JComboBox monthBox = null;
private ButtonGroup intervalGroup = null;
private JRadioButton[] intervalButtons = null;

private JMenuBar menuBar = null;
private JMenu file = null;
private KeyStroke ksFileOpen = null;
private ImageIcon iiFileOpen = null;
private JMenuItem itemFileOpen = null;
private JSeparator itemSeparator = null;
private KeyStroke ksFileExit = null;
private ImageIcon iiFileExit = null;
private JMenuItem itemFileExit = null;

private Container contentPane = null;

private JToolBar toolbar = null;
private JButton fileOpenButton = null;
private JButton fileExitButton = null;

private FileExitListener fileExitListener = null;
private FileOpenListener fileOpenListener = null;

private double confidenceLevel = 0.99;
private int integerLevel = 0;
private int numDays = 0;
private double[] data = null;
private double[] logReturns = null;
private int R = 0;
private int month = 0;

private double[] xi = null;
private int[] nu = null;
private double m = 0;
private double sigma2 = 0;
private double[] p = null;
private double chi2 = 0;

private final double MAGIC = 10.0;

private boolean yearly = true;

public Normality()
{
    setSize(WIDTH,HEIGHT);
    setTitle("Checking for normality");
    menuBar = new JMenuBar();
    setJMenuBar(menuBar);
    file = new JMenu("File");
    file.setToolTipText("Data reading");
    file.setMnemonic('F');
}
ksFileOpen = KeyStroke.getKeyStroke(KeyEvent.VK_O,
                          Event.CTRL_MASK);
iiFileOpen = new ImageIcon("FileOpen.jpg");
itemFileOpen = new JMenuItem("Open...", iiFileOpen);
itemFileOpen.setToolTipText("Opens data file");
itemFileOpen.setAccelerator(ksFileOpen);
itemFileOpen.setMnemonic(KeyEvent.VK_O);
file.add(itemFileOpen);

itemSeparator = new JSeparator();
file.add(itemSeparator);

ksFileExit = KeyStroke.getKeyStroke(KeyEvent.VK_X,
                         Event.ALT_MASK);
iiFileExit = new ImageIcon("FileExit.jpg");
itemFileExit = new JMenuItem("Exit", iiFileExit);
itemFileExit.setToolTipText("Terminates the program");
itemFileExit.setAccelerator(ksFileExit);
itemFileExit.setMnemonic(KeyEvent.VK_X);
file.add(itemFileExit);

menuBar.add(file);

toolbar = new JToolBar();

fileOpenButton = new JButton(iiFileOpen);
fileOpenButton.setToolTipText(itemFileOpen.getToolTipText());
fileOpenButton.addActionListener( this);
toolbar.add(fileOpenButton);

fileExitButton = new JButton(iiFileExit);
fileExitButton.setToolTipText(itemFileExit.getToolTipText());
fileExitButton.addActionListener( this);
toolbar.add(fileExitButton);

contentPane = getContentPane();
contentPane.setLayout( new BorderLayout());
contentPane.add(toolbar, BorderLayout.NORTH);

mainPanel = new JPanel(new BorderLayout());
mainPanel.setBorder(BorderFactory.createLineBorder(
                        Color.RED, 1));
contentPane.add(mainPanel, BorderLayout.CENTER);

leftCompanyPanel = new JPanel();
leftCompanyPanel.setBorder(BorderFactory.createLineBorder(
                          Color.BLACK, 1));
upCompanyLabel = new JLabel(companyString);
leftCompanyPanel.add(upCompanyLabel);
mainPanel.add(leftCompanyPanel, BorderLayout.NORTH);

centralPanel = new JPanel();
centralPanelLayout = new BoxLayout(centralPanel, BoxLayout.X_AXIS);
centralPanel.setLayout(centralPanelLayout);
mainPanel.add(centralPanel, BorderLayout.CENTER);

leftPanel = new JPanel();
leftPanelLayout = new BoxLayout(leftPanel, BoxLayout.Y_AXIS);
leftPanel.setLayout(leftPanelLayout);
leftPanel.setBorder(BorderFactory.createLineBorder(
                           Color.BLACK, 1));
centralPanel.add(leftPanel);
rightPanel = new JPanel();
rightPanelLayout = new BoxLayout(rightPanel, BoxLayout.Y_AXIS);
rightPanel.setLayout(rightPanelLayout);
rightPanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
centralPanel.add(rightPanel);

theoreticalChartPanel = new JPanel();
theoreticalChartPanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
leftPanel.add(theoreticalChartPanel);

intervalGroup = new ButtonGroup();
intervalButtons = new JRadioButton[2];
for (int i=0; i<2; i++) {
    intervalButtons[i] = new JRadioButton(interval[i]);
    intervalGroup.add(intervalButtons[i]);
    theoreticalChartPanel.add(intervalButtons[i]);
    intervalButtons[i].addActionListener(this);
}
intervalButtons[0].setSelected(true);

monthBox = new JComboBox(months);
monthBox.addItemListener(this);
theoreticalChartPanel.add(monthBox);
monthBox.setEnabled(false);

historicalChartPanel = new JPanel(new BorderLayout());
historicalChartPanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
leftPanel.add(historicalChartPanel);

leftYearLabel = new JLabel(yearString);
leftYearLabel.setText(yearString + "2003");
leftPanel.add(leftYearLabel);

resultPanel = new JPanel(new BorderLayout());
resultPanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
leftPanel.add(resultPanel);

leftConfidencePanel = new JPanel();
resultPanel.add(leftConfidencePanel, BorderLayout.NORTH);

leftConfidenceLabel = new JLabel(confidenceString);
leftConfidencePanel.add(leftConfidenceLabel);

confidenceField = new JTextField(5);
confidenceField.setEditable(false);
confidenceField.setText("99% ");
leftConfidencePanel.add(confidenceField);

results = new JTextArea();
results.setEditable(false);
resultPanel.add(results, BorderLayout.CENTER);

logarithmicPanel = new ChartPanel(null);
logarithmicPanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
rightPanel.add(logarithmicPanel);
startPanel = new JPanel();
startPanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
rightPanel.add(startPanel);

startButton = new JButton(startString);
startButton.setEnabled(false);
startButton.addActionListener(this);
startPanel.add(startButton);

rightConfidencePanel = new JPanel(new GridLayout(1,2));
rightConfidencePanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
rightPanel.add(rightConfidencePanel);

rightConfidenceLabel = new JLabel("Select "+confidenceString);
rightConfidencePanel.add(rightConfidenceLabel);

confidenceBox = new JComboBox(confidence);
confidenceBox.addItemListener(this);
rightConfidencePanel.add(confidenceBox);

yearPanel = new JPanel(new GridLayout(1,2));
yearPanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
rightPanel.add(yearPanel);

pricePanel = new JPanel(new GridLayout(1,2));
pricePanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
rightPanel.add(pricePanel);

rightCompanyPanel = new JPanel(new GridLayout(1,2));
rightCompanyPanel.setBorder(BorderFactory.createLineBorder(Color.BLACK, 1));
rightPanel.add(rightCompanyPanel);

rightYearLabel = new JLabel(selectYearString);
yearPanel.add(rightYearLabel);

selectPriceLabel = new JLabel(selectPriceType);
pricePanel.add(selectPriceLabel);

rightCompanyLabel = new JLabel(selectCompanyString);
rightCompanyPanel.add(rightCompanyLabel);

yearBox = new JComboBox(years);
yearBox.addItemListener(this);
yearPanel.add(yearBox);

priceBox = new JComboBox(priceTypes);
priceBox.addItemListener(this);
pricePanel.add(priceBox);

companyBox = new JComboBox(companies);
companyBox.addItemListener(this);
rightCompanyPanel.add(companyBox);

fileExitListener = new FileExitListener();
fileOpenListener = new FileOpenListener(this);
itemFileExit.addActionListener(fileExitListener);
itemFileOpen.addActionListener(fileOpenListener);

setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

public void itemStateChanged(ItemEvent e){
    Object source = e.getSource();
    int index = 0;
    String content = null;

    if (source == confidenceBox){
        index = confidenceBox.getSelectedIndex();
        switch (index) {
            case 0: confidenceLevel = 0.99;
                integerLevel = 0;
                break;
            case 1: confidenceLevel = 0.95;
                integerLevel = 1;
                break;
            case 2: confidenceLevel = 0.9;
                integerLevel = 2;
                break;
        }
        content = (String)confidenceBox.getSelectedItem();
        confidenceField.setText(content);
        return;
    }

    if (source == yearBox){
        index = yearBox.getSelectedIndex();
        switch (index){
            case 0: leftYearLabel.setText(yearString + "2003");
                break;
            case 1: leftYearLabel.setText(yearString + "2004");
                break;
        }
        return;
    }

    if (source == companyBox){
        index = companyBox.getSelectedIndex();
        switch (index){
            case 0: selectCompanyString = "Asea Brown Boveri.st";
                break;
            case 1: selectCompanyString = "Ericsson-B.st";
                break;
            case 2: selectCompanyString = "FöreningsSparbanken-a.st";
                break;
            case 3: selectCompanyString = "Skandinavia Enskilda Bank-a.st";
                break;
            case 4: selectCompanyString = "Svenska Kullager Fabrik-b.st";
                break;
            case 5: selectCompanyString = "Volvo-b.st";
                break;
        }
        content = (String)companyBox.getSelectedItem();
        upCompanyLabel.setText(content);
        return;
    }
}
if (source == priceBox) {
    index = priceBox.getSelectedIndex();
    switch (index) {
        case 0: selectPriceType = "Open"; break;
        case 1: selectPriceType = "High"; break;
        case 2: selectPriceType = "Low"; break;
        case 3: selectPriceType = "Close"; break;
    }
    return;
}

if (source == monthBox) {
    month = monthBox.getSelectedIndex();
    if (month <= 9) {
        leftYearLabel.setText(yearString + "2003");
    }
    else {
        leftYearLabel.setText(yearString + "2004");
    }
    startButton.setEnabled(false);
    return;
}

private double normal(double x) {
    boolean flag = false;
    double z = x;
    if (x < 0) {
        flag = true;
        z = -x;
    }
    double d = 1.0/(1.0+0.3316419*z);
    final double[] A = { 0,
                        0.31938153,
                        -0.356563782,
                        1.781477937,
                        -1.821255978,
                        1.330274429
                        };

    double p = A[5];
    for (int k=4; k >=0; k--) {
        p = p*d+A[k];
    }
    double y = 1 - Math.exp(-z*z/2)*p/Math.sqrt(2*Math.PI);
    if (flag) {
        y = 1 - y;
    }
    return y;
}

public void actionPerformed(ActionEvent e) {
    Object source = e.getSource();
    if (source == intervalButtons[0]) {

yearly = true;
monthBox.setEnabled(false);
yearBox.setEnabled(true);
return;
}

if (source == intervalButtons[1]) {
yearly = false;
monthBox.setEnabled(true);
yearBox.setEnabled(false);
startButton.setEnabled(false);
return;
}

if (source == fileExitButton) {
dispose();
System.exit(0);
}

if (source == fileOpenButton) {
openFile();
return;
}

if (source == startButton) {
logReturns = new double[numDays-1];
for (int i=0; i<numDays-1; i++) {
    logReturns[i] = Math.log(data[i+1]/data[i]);
}

double[] temp = new double[numDays];
temp[0]=0;
for (int i=1; i<numDays; i++) {
    temp[i] = logReturns[i-1];
}
for (int i=2; i<numDays; i++) {
    double x = temp[i];
    int l = 1;
    int r = i;
    while (l<r) {
        int m = (l+r)/2;
        if (temp[m]<=x) {
            l = m+1;
        } else {
            r = m;
        }
    }
    for (int j=i; j>=r+1; j--) {
        temp[j] = temp[j-1];
    }
    temp[r] = x;
}
for (int i=0; i<numDays-1; i++) {
    logReturns[i] = temp[i+1];
}

m = 0;
for (int i=0; i<numDays-1; i++) {
    m += logReturns[i];
}
m /= (numDays-1);
sigma2 = 0;
for (int i=0; i<numDays-1; i++) {
    sigma2 += (logReturns[i]-m)*(logReturns[i]-m);
}
sigma2 /= (numDays-2);

int xiIndex = -1;
int logIndex = -1;

while (true) {
    xiIndex++;
    int n = 0;
    double np = 0;
    while (np < MAGIC) {
        logIndex++;
        n++;
        temp[xiIndex] = (logReturns[logIndex]+logReturns[logIndex+1])/2;
        if (xiIndex == 0) {
            np += n*normal((temp[xiIndex]-m)/Math.sqrt(sigma2));
        } else {
            np += n*(normal((temp[xiIndex]-m)/Math.sqrt(sigma2)) - normal((temp[xiIndex-1]-m)/Math.sqrt(sigma2)));
        }
    }
    np = (numDays-1-logIndex) * (1-normal((temp[xiIndex]-m)/Math.sqrt(sigma2)));
    if (np >= 2*MAGIC) {
        continue;
    } else {
        break;
    }
}

R = xiIndex +1;
System.out.println("R= " + R);
xi = new double[R];
nu = new int[R];

for (int i=0; i<R; i++) {
    xi[i] = temp[i];
}

for (int i=0; i<R; i++) {
    nu[i] = 0;
    if (i == 0) {
        int j = 0;
        double x = logReturns[j];
        while (x <= xi[i]) {
            nu[i]++;
            j++;
            x = logReturns[j];
        }
    }
if (i == R-1) {
    int j = numDays-2;
    double x = logReturns[j];
    while (x > xi[i]) {
        nu[i]++;
        j--;
        x = logReturns[j];
    }
}

if (i>0 && i<R-1) {
    int j = 0;
    double x = logReturns[j];
    while (x <= xi[i-1]) {
        j++;
        x = logReturns[j];
    }
    while (x < xi[i]) {
        nu[i]++;
        j++;
        x = logReturns[j];
    }
}

m = 0;
for (int i=0; i<R; i++) {
    m += nu[i]*xi[i];
}
m /= (numDays-1);
sigma2 = 0;
for (int i=0; i<R; i++) {
    sigma2 += nu[i]*(xi[i]-m)*(xi[i]-m);
}
sigma2 /= (numDays-1);
p = new double[R];
p[0] = normal(((xi[0]+xi[1])/2-m)/Math.sqrt(sigma2));
p[R-1] = 1.0-normal(((xi[R-1]+xi[R-2])/2-m)/Math.sqrt(sigma2));
for (int j=1; j<R-1; j++) {
    p[j] = normal(((xi[j]+xi[j+1])/2-m)/Math.sqrt(sigma2)) -
          normal(((xi[j-1]+xi[j])/2-m)/Math.sqrt(sigma2));
}

chi2 = 0;
for (int i=0; i<R; i++) {
    double s = nu[i] - (numDays-1)*p[i];
    s = s*s;
    s /= (numDays-1);
    s /= p[i];
    chi2 += s;
}
System.out.println("chi2="+chi2);

double theoreticalValue = 0;
switch (integerLevel) {
    case 0: theoreticalValue = CHI21[R-2];
            break;
    case 1: theoreticalValue = CHI25[R-2];
            break;
    case 2: theoreticalValue = CHI210[R-2];
            break;
}
results.setText(null);
if (chi2 <= theoreticalValue) {
    results.append("The normality hypothesis is true");
}
else {
    results.append("The normality hypothesis is false");
}

HistogramDataset dataset = new HistogramDataset();
dataset.addSeries("log returns",logReturns,3*R);

JFreeChart chart = ChartFactory.createHistogram ( "",
                                              "",
                                              dataset,
                                              PlotOrientation.VERTICAL,
                                              true,
                                              true,
                                              false);
logarithmicPanel.setChart(chart);
logarithmicPanel.setVisible(true);

double[] historicalData = new double[numDays];
for (int i=0; i<numDays; i++) {
    historicalData[i] = data[i];
}

public void openFile () {
    JFileChooser chooser = new JFileChooser();
    String part1 = null;
    String search = null;
    String sub2 = null;
    int emptyLines = 0;
    boolean countEmptyLines = false;
    switch (companyBox.getSelectedIndex()) {
    case 0: part1 = shortCompany0;
            break;
    case 1: part1 = shortCompany1;
            break;
    case 2: part1 = shortCompany2;
            break;
    case 3: part1 = shortCompany3;
            break;
    case 4: part1 = shortCompany4;
            break;
    case 5: part1 = shortCompany5;
            break;
    }
    String part2 = null;
    switch (yearBox.getSelectedIndex()) {
    case 0 : part2 = "2003";}
break;
case 1 : part2 = "2004";
break;
}
chooser.setFileFilter(new NormalityFilter(part1+part2+".nrm"));
int state = chooser.showOpenDialog(null);
File file = chooser.getSelectedFile();
if (state == JFileChooser.APPROVE_OPTION && file != null) {
    try {
        DataInputStream in = new DataInputStream(
            new FileInputStream(file));
        String input = null;
        numDays = 0;
        data = new double[300];
        if (!yearly) {
            switch (month) {
            case 0:
                case 10: search = "jan";
                    break;
            case 1:
                case 11: search = "feb";
                    break;
            case 2:
                case 12: search = "mar";
                    break;
            case 3:
                case 13: search = "apr";
                    break;
            case 4:
                case 14: search = "may";
                    break;
            case 5:
                case 15: search = "jun";
                    break;
            case 6:
                case 16: search = "jul";
                    break;
            case 7:
                case 17: search = "aug";
                    break;
            case 8:
                case 18: search = "sep";
                    break;
            case 9:
                case 19: search = "oct";
                    break;
            }
        }
        while ((input = in.readLine()) != null) {
            if (input.length() == 0 && countEmptyLines) {
                emptyLines++;
            }
            if (input.length() > 0) {
                if (Character.isDigit(input.charAt(0))) {
                    if (!yearly) {
                        sub2 = input.substring(3,6);
                        if (sub2.equals(search)) {
                            countEmptyLines = true;
                        }
                    } else {
                        if (countEmptyLines) {
                            break;
                        }
                    }
                }
            }
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}
if (emptyLines == 3) {
    break;
}
else {
    continue;
}

int index = priceBox.getSelectedIndex();
String sub = input.substring(6*index+10, 6*index+15);
data[numDays] = Double.parseDouble(sub);
numDays++;
}
}
in.close();
startButton.setEnabled(true);
}

}catch(IOException exception) {
    JOptionPane.showMessageDialog(null,
    "Error",
    "File can nor be read",
    JOptionPane.ERROR_MESSAGE);
}

}

public static void main(String args[]) {
    Normality f = new Normality();
f.show();
}

NormalityFilter class

/**
 * File: NormalityFilter.java
 *
 * Description: The NormalityFilter class is responsible for the popup window
 * in which appears the name of the company that the user had selected
 *
 * Author: Rafael Vides
 *
 * e-Mail : rvs01001@student.mdh.se
 *
 */

import javax.swing.filechooser.*;

public class NormalityFilter extends FileFilter {
    private String name = null;

    public NormalityFilter(String name) {
        super();
        this.name = name;
    }
}
public boolean accept(java.io.File f) {
    boolean accept = f.isDirectory();
    if (!accept) {
        String fileName = f.getName();
        accept = name.equals(fileName);
    }
    return accept;
}

public String getDescription() {
    return "Normality files(*.nrm)";
}

### FileOpenListener class

```java
/**
 * File: FileOpenListener.java
 *
 * Description: File Open Listener is responsible for the Open and Menu items,
 * which give you access to the normality files, the files of the companies in
 * with the suffix *.nrm
 *
 * Author: Rafael Vides
 *
 * e-Mail: rvs01001@student.mdh.se
 *
 */

import javax.swing.*;
import javax.swing.event.*;
import java.awt.event.*;

public class FileOpenListener implements ActionListener {
    Normality frame;

    public FileOpenListener(Normality frame) {
        this.frame = frame;
    }

    public void actionPerformed(ActionEvent ae) {
        frame.openFile();
    }
}
```
FileExitsListener class

/**
 * File: FileExitListener.java
 * 
 * Description: The File Exit Listener class is responsible for close the
 * application program.
 * 
 * Author: Rafael Vides
 * 
 * e-Mail: rvs01001@student.mdh.se
 * 
 */

import javax.swing.*;
import javax.swing.event.*;
import java.awt.event.*;

public class FileExitListener implements ActionListener
{
    public FileExitListener()
    {
    }

    public void actionPerformed(ActionEvent ae)
    {
        System.exit(0);
    }
}