***Задача 4.  
Вариант 8***

***Условие:***  
Основные характеристики доходности финансового актива — средняя доходность (математическое ожидание доходности) и дисперсия. Вам предлагается сравнить эти характеристики для двух ценных бумаг. Данные содержатся в файле «данные к задаче 4.ods» и образуют ряд связанных пар — наблюдения за недельными логарифмическими доходностями ценных бумаг.   
В столбцах x3 и y3 представлены доходности для варианта 3 и т.п.  
Наша задача — проверить гипотезы о равенстве средних доходностей и дисперсий по данным вашего варианта против двусторонних альтернатив.  
***Доходности:***

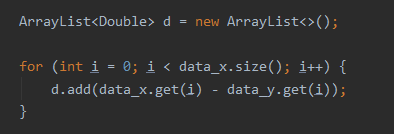
|  |  |
| --- | --- |
| x8 | y8 |
| 0,000409 | 0,001332 |
| -0,00057 | -0,00168 |
| -0,00072 | 0,003033 |
| 0,001241 | 0,000525 |
| 0,000201 | 0,002142 |
| 0,001716 | 0,000671 |
| 0,000604 | 0,001886 |
| 0,001605 | 0,002484 |
| 0,001529 | 0,001964 |
| 0,000787 | 0,00044 |
| 0,000138 | 0,001501 |
| 0,001181 | 9,58E-05 |
| 0,000518 | 0,000091 |
| 0,002096 | 0,002058 |
| 0,000344 | 0,00114 |
| -0,00066 | 0,000792 |
| 0,00134 | 0,000782 |
| 0,000409 | 0,000593 |
| 0,000293 | 0,000561 |
| 0,000855 | 0,001054 |
| 0,002179 | 0,002043 |
| 0,001666 | -0,00063 |
| 7,62E-05 | 0,002665 |
| -2,5E-05 | 0,001533 |
| -0,0004 | 0,002889 |
| -0,00083 | 0,003444 |
| 0,003185 | 0,003259 |
| 0,000246 | -0,00021 |
| 0,000732 | 0,000383 |
| 9,37E-05 | 0,001044 |
| 0,000316 | 0,000797 |
| 0,000927 | 0,000751 |
| 0,002475 | 0,003527 |
| 0,001954 | 0,000556 |
| 0,001391 | 0,000869 |
| -0,00106 | 0,002936 |
| 0,001538 | 0,000675 |
| 0,000433 | 0,001931 |
| 0,000665 | 8,44E-05 |
| 0,001349 | 0,001371 |
| 0,000255 | 0,000966 |
| 0,000137 | 1,11E-05 |
| 0,001444 | 0,000565 |
| 0,001348 | -0,00021 |
| 0,000138 | 0,001033 |
| 0,000141 | 0,000627 |
| 0,00032 | -0,00018 |
| -0,00035 | 0,002178 |
| 0,000611 | 0,000429 |
| -0,00029 | 0,000425 |
| 0,000952 | 0,001404 |
| 0,001514 | 0,001678 |
| -0,00148 | 4,43E-05 |
| 0,000774 | 0,000691 |
| 0,000731 | 0,00077 |
| 0,001604 | 0,000896 |
| 0,002237 | -0,00058 |
| -0,00089 | 0,001573 |
| 0,002336 | -0,00069 |
| -0,00073 | 0,000602 |
| 0,001473 | 0,002273 |
| -0,0002 | 0,002169 |
| -0,00042 | 0,002604 |
| 0,000356 | 0,000875 |

***Для выполнения домашнего задания использовался язык Java и Stream API для упрощения вычислений.***

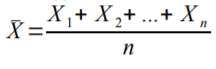
***Часть А:*** Проверка гипотезы о равенстве средних.

***А1.*** Рассчитайте разности доходностей для каждой недели: di=Xi−Yi. Найдите среднюю разность и её выборочное стандартное отклонение .

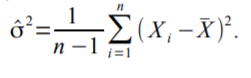
Для того, чтобы рассчитать разности доходностей для каждой недели, в цикле были обработаны известные данные:



После чего была рассчитана средняя разность по формуле



И её выборочное стандартное отклонение, из которого был взят корень.

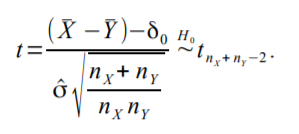


В итоге получен следующий ответ:



***А2.*** Рассчитайте значение t-статистики для гипотезы о равенстве средних в связанных парах.

Для вычисления t-статистики была использована формула

******

И получен следующий результат: -2.5111603374041436

***А3.*** Выпишите критическое значение для уровня значимости 5%

Критическое значение (1,9789706) было взято из таблицы (<http://old.exponenta.ru/educat/referat/XIkonkurs/student5/tabt-st.pdf>)

***А4.*** Сформулируйте вывод. Противоречат ли данные гипотезе о равенстве средних доходностей?

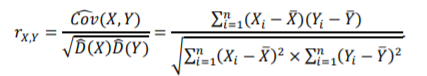
******

Да (модуль больше критического значения).

***А5***. Рассчитайте p-значение.

p-value было рассчитано через онлайн калькулятор = 0.013304.

***Часть Б.*** Проверка гипотезы о равенстве дисперсий.  
***Б1.*** Рассчитайте выборочный коэффициент корреляции для X−Y и X+Y  
Выборочная корреляция рассчитывалась как коэффициент корреляции Пирсона (выборочный аналог обычного коэффициента корреляции)



Его значение:

   
***Б2.*** Рассчитайте значение t-статистики для гипотезы о равенстве дисперсий в связанных парах.

t-статистика была высчитана при помощи формулы, приведённой в тексте задания.

  
***Б3.*** Выпишите критическое значение для уровня значимости 5%.

Критическое значение (1,9789706) было взято из таблицы (<http://old.exponenta.ru/educat/referat/XIkonkurs/student5/tabt-st.pdf>)

***Б4.*** Сформулируйте вывод. Противоречат ли данные гипотезе о равенстве дисперсий?



Нет. Модуль t-статистики меньше критического значения.  
***Б5.*** Рассчитайте p-значение.  
p-value было рассчитано через онлайн калькулятор = 0.013304.

Текст программы:

package matstat;  
  
import java.util.ArrayList;  
import java.util.Arrays;  
  
public class Main {  
  
 static double getAvr(ArrayList<Double> data) {  
 return data.stream().mapToDouble(z -> z).sum() / data.size();  
 }  
  
 static double get\_disperssion(ArrayList<Double> data, double avr) {  
 return Math.*sqrt*(data.stream().mapToDouble(z -> Math.*pow*(z - avr, 2)).sum() / (data.size() - 1));  
 }  
  
 public static void partA(ArrayList<Double> data\_x, ArrayList<Double> data\_y) {  
  
 ArrayList<Double> d = new ArrayList<>();  
  
 for (int i = 0; i < data\_x.size(); i++) {  
 d.add(data\_x.get(i) - data\_y.get(i));  
 }  
  
 double avr\_d = d.stream().mapToDouble(z -> z).sum() / d.size();  
 double sigma\_s\_kryschechkoi = *get\_disperssion*(d, avr\_d);  
  
 System.*out*.println("average d=" + avr\_d);  
 System.*out*.println("Скорректированная дисперсия:" + sigma\_s\_kryschechkoi);  
  
 //t-statistics  
 //A2  
 int x\_size = data\_x.size();  
 int y\_size = data\_y.size();  
  
 double x\_avr = *getAvr*(data\_x);  
 double y\_avr = *getAvr*(data\_y);  
  
 double x\_d = data\_x.stream().mapToDouble(z -> Math.*pow*(z - x\_avr, 2)).sum();  
 double y\_d = data\_y.stream().mapToDouble(z -> Math.*pow*(z - y\_avr, 2)).sum();  
  
 double sigma = Math.*sqrt*((x\_d + y\_d) / (x\_size + y\_size - 2));  
  
 double sqrt = Math.*sqrt*((x\_size + y\_size) \* 1.0 / (x\_size \* y\_size));  
  
 double t\_stat = (x\_avr - y\_avr) / (sigma \* sqrt);  
 System.*out*.println(" " + t\_stat);  
  
 //A3 t(126, 0,05)=1,9789706  
 //http://old.exponenta.ru/educat/referat/XIkonkurs/student5/tabt-st.pdf  
  
 //A4  
 System.*out*.println("t statistics:" + t\_stat + " " + ((Math.*abs*(t\_stat) > 1.9789706) ? "rejected (module is more than 1,9789706)" : "accepted (is less or equals to 1,9789706)"));  
 }  
  
  
 public static void main(String[] args) {  
 String x = "0,0004086\t-0,000574\t-0,0007217\t0,0012408\t0,0002007\t0,0017155\t0,0006044\t0,0016047\t0,0015286\t0,0007868\t0,0001381\t0,0011805\t0,000518\t0,0020961\t0,0003439\t-0,000656\t0,0013404\t0,000409\t0,0002933\t0,0008552\t0,002179\t0,0016658\t0,0000762\t-0,0000251\t-0,0003999\t-0,0008344\t0,0031851\t0,0002456\t0,0007316\t0,0000937\t0,0003158\t0,0009268\t0,0024746\t0,0019535\t0,0013906\t-0,0010556\t0,0015383\t0,0004326\t0,0006653\t0,0013492\t0,0002549\t0,0001369\t0,0014441\t0,0013478\t0,000138\t0,0001407\t0,0003199\t-0,0003453\t0,0006113\t-0,00029\t0,0009523\t0,0015136\t-0,0014778\t0,0007741\t0,0007308\t0,0016039\t0,0022365\t-0,0008889\t0,0023363\t-0,0007337\t0,0014734\t-0,0002044\t-0,0004228\t0,000356\n";  
 String y = "0,0013324\t-0,0016763\t0,0030333\t0,0005248\t0,0021415\t0,0006712\t0,0018863\t0,0024842\t0,001964\t0,0004396\t0,0015009\t0,0000958\t0,000091\t0,0020582\t0,0011396\t0,0007923\t0,0007823\t0,0005934\t0,000561\t0,001054\t0,0020426\t-0,0006337\t0,0026646\t0,0015331\t0,0028891\t0,0034444\t0,0032585\t-0,0002147\t0,0003833\t0,0010444\t0,0007967\t0,0007514\t0,0035265\t0,0005562\t0,0008686\t0,0029362\t0,000675\t0,0019311\t0,0000844\t0,0013705\t0,0009659\t0,0000111\t0,0005648\t-0,0002123\t0,001033\t0,0006271\t-0,0001848\t0,0021777\t0,0004286\t0,000425\t0,0014037\t0,0016781\t0,0000443\t0,0006911\t0,0007701\t0,0008956\t-0,0005799\t0,0015734\t-0,0006948\t0,0006016\t0,0022734\t0,0021693\t0,0026041\t0,0008749\n";  
 x = x.replace(',', '.');  
 y = y.replace(',', '.');  
  
 ArrayList<Double> data\_y = new ArrayList<>();  
 ArrayList<Double> data\_x = new ArrayList<>();  
  
 Arrays.*stream*(x.split("\t")).mapToDouble(Double::*parseDouble*).forEach(data\_x::add);  
 Arrays.*stream*(y.split("\t")).mapToDouble(Double::*parseDouble*).forEach(data\_y::add);  
  
 System.*out*.println("~ ~ ~ A ~ ~ ~");  
 *partA*(data\_x, data\_y);  
 System.*out*.println("~ ~ ~ B ~ ~ ~");  
 *partB*(data\_x, data\_y);  
 }  
  
 public static void partB(ArrayList<Double> data\_x, ArrayList<Double> data\_y) {  
  
 //B1  
 double x\_avr = *getAvr*(data\_x);  
 double y\_avr = *getAvr*(data\_y);  
  
 double up = *getUp*(data\_x, data\_y, x\_avr, y\_avr);  
 double down = *getDown*(data\_x, data\_y, x\_avr, y\_avr);  
  
 double corr = up / down;  
 System.*out*.println("Corr= " + corr);  
  
 //B2  
 double t\_stat = corr \* Math.*sqrt*(data\_x.size() + data\_y.size() - 2) / Math.*sqrt*(1 - Math.*pow*(corr, 2));  
 System.*out*.println("t statistics: " + t\_stat);  
  
 //B3 t(126, 0,05)=1,9789706  
  
 //B4  
 System.*out*.println("t statistics:" + t\_stat + " " + ((Math.*abs*(t\_stat) > 1.9789706) ? "rejected (module is more than 1,9789706)" : "accepted (is less or equals to 1,9789706)"));  
 }  
  
 static double getUp(ArrayList<Double> data\_x, ArrayList<Double> data\_y, double x\_avr, double y\_avr) {  
 double res = 0;  
  
 for (int i = 0; i < data\_x.size(); i++) {  
 res += (data\_x.get(i) - x\_avr) \* (data\_y.get(i) - y\_avr);  
 }  
  
 return res;  
 }  
  
 static double getDown(ArrayList<Double> data\_x, ArrayList<Double> data\_y, double x\_avr, double y\_avr) {  
 double x\_d = data\_x.stream().mapToDouble(z -> Math.*pow*(z - x\_avr, 2)).sum();  
 double y\_d = data\_y.stream().mapToDouble(z -> Math.*pow*(z - y\_avr, 2)).sum();  
  
 return Math.*sqrt*(x\_d \* y\_d);  
 }  
}