###### Міністерство освіти і науки України

###### Національний технічний університет України

###### “Київський політехнічний інститут”

###### Кафедра автоматизації проектування енергетичних процесів та систем

# **Лабораторна робота №1-5**

з дисципліни : “Еколого-економічна оптимізація виробництва”

на тему :

“Факторний аналіз впливу економічних чинників на викиди забруднюючих речовин в атмосферне повітря”

|  |
| --- |
| Виконала: |
| студентка ІІІ курсу, групи ТМ-21 |
| Підберезна Олена  Перевірила:  Караєва Н. В. |
|  |
|  |
|  |
|  |

Київ - 2015

**КОРЕЛЯЦІЙНИЙ АНАЛІЗ**

**Задачі:**

1. Сформувати таблицю з вхідними даними.
2. Побудувати кореляційну матрицю засобами Excel.
3. Розробити алгоритм розрахунку коефіцієнту кореляції.
4. Розробити відповідний програмний продукт.
5. Проаналізувати результати розрахунку.

Згідно даних Держкомстату України [1] сформована таблиця 1 із вхідними параметрами (показниками), які характеризують регіональний розподіл валового регіонального продукту (ВРП), викиди забруднюючих речовин атмосферне повітря(ЗР), обсяги реалізованої продукції(РП), дохід населення(ДН), експорт(Е) та імпорт(І) товарів за 2010 рік. Відповідні дані за 2011 рік наведено у таблиці 2, а дані за 2012 рік – таблиця 3.

Таблиця 1. Значення окремих показників еколого-економічного розвитку регіонів України за 2010 рік

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Обл.** | Індикатори | | | | | |
| ВРП(млн грн) | ЗР(млн кг) | РП(млн грн) | ДН(млн грн) | Е(млн дол) | І(млн дол) |
| Вінницька | 23589,00 | 185,1 | 36323,40 | 33602,00 | 566,30 | 344,70 |
| Волинська | 14429,00 | 57,2 | 37912,00 | 19137,00 | 434,80 | 571,50 |
| Дніпропетровська | 116136,00 | 1140,5 | 382203,60 | 88922,00 | 8021,90 | 5387,70 |
| Донецька | 128986,00 | 1589,9 | 499222,60 | 118223,00 | 13126,30 | 3067,60 |
| Житомирська | 18743,00 | 87,5 | 22820,00 | 26124,00 | 393,80 | 428,30 |
| Закарпатська | 15299,00 | 87,1 | 20237,40 | 20841,00 | 1156,60 | 1348,60 |
| Запорізька | 42736,00 | 326,1 | 106893,40 | 45779,00 | 3210,00 | 1363,00 |
| Івано-Франківська | 20446,00 | 224,9 | 30503,30 | 26504,00 | 507,60 | 652,00 |
| Київська | 44953,00 | 269 | 144435,60 | 42732,00 | 1301,90 | 3018,10 |
| Кіровоградська | 15749,00 | 72,2 | 26131,90 | 20213,00 | 336,80 | 187,50 |
| Луганська | 45541,00 | 599,2 | 104822,20 | 51338,00 | 3324,30 | 1119,00 |
| Львівська | 41655,00 | 246,3 | 98047,10 | 54838,00 | 974,40 | 2028,30 |
| Миколаївська | 24055,00 | 83,2 | 46809,10 | 26034,00 | 1597,10 | 866,10 |
| Одеська | 53878,00 | 181,2 | 132574,60 | 52924,00 | 1684,40 | 3520,30 |
| Полтавська | 44291,00 | 172,7 | 101680,50 | 34462,00 | 2219,70 | 1066,20 |
| Рівненська | 15882,00 | 56,2 | 20305,90 | 22362,00 | 396,60 | 438,10 |
| Сумська | 18333,00 | 88,9 | 25232,90 | 24918,00 | 746,50 | 469,90 |
| Тернопільська | 12726,00 | 63,9 | 17267,50 | 19587,00 | 131,40 | 209,80 |
| Харківська | 65293,00 | 281,4 | 147365,10 | 67102,00 | 1433,20 | 1834,10 |
| Херсонська | 15649,00 | 74,2 | 26734,20 | 20978,00 | 350,40 | 161,20 |
| Хмельницька | 18096,00 | 83,8 | 22369,10 | 26987,00 | 291,50 | 459,40 |
| Черкаська | 22354,00 | 138,6 | 45686,70 | 26194,00 | 672,70 | 378,40 |
| Чернівецька | 9892,00 | 44,6 | 10321,40 | 16114,00 | 104,00 | 113,10 |
| Чернігівська | 17008,00 | 96,7 | 22057,40 | 23179,00 | 324,70 | 380,70 |
| м.Київ | 196639,00 | 265,3 | 1163859,00 | 143903,00 | 7104,90 | 25941,90 |

Таблиця 2. Значення окремих показників еколого-економічного розвитку регіонів України за 2011 рік

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Обл.** | Індикатори | | | | | |
| ВРП(млн грн) | ЗР(млн кг) | РП(млн грн) | ДН(млн грн) | Е(млн дол) | І(млн дол) |
| Вінницька | 29099,00 | 169,9 | 41898,50 | 38990,00 | 661,70 | 384,56 |
| Волинська | 17637,00 | 52,8 | 37706,10 | 22584,00 | 646,35 | 1059,59 |
| Дніпропетровська | 140020,00 | 1157,9 | 460608,40 | 101868,00 | 10363,19 | 6717,16 |
| Донецька | 161021,00 | 1729,3 | 685930,10 | 135599,00 | 17197,63 | 4622,79 |
| Житомирська | 21928,00 | 86,8 | 27216,30 | 30069,00 | 515,46 | 460,14 |
| Закарпатська | 18054,00 | 89,4 | 26373,30 | 24446,00 | 1397,59 | 1997,93 |
| Запорізька | 49525,00 | 341,2 | 124321,50 | 52272,00 | 4151,23 | 2271,05 |
| Івано-Франківська | 26752,00 | 275,9 | 34627,80 | 31224,00 | 941,96 | 1068,28 |
| Київська | 59154,00 | 278,7 | 183799,30 | 48990,00 | 1696,20 | 4260,31 |
| Кіровоградська | 20041,00 | 73,9 | 32978,00 | 23443,00 | 443,39 | 226,85 |
| Луганська | 57202,00 | 553,5 | 127395,60 | 58619,00 | 6505,03 | 1743,25 |
| Львівська | 52103,00 | 256,6 | 122339,70 | 63602,00 | 1201,93 | 3202,35 |
| Миколаївська | 27633,00 | 89,9 | 49214,40 | 29800,00 | 1654,40 | 1041,48 |
| Одеська | 61499,00 | 173,8 | 137037,80 | 61435,00 | 1541,12 | 3145,41 |
| Полтавська | 52252,00 | 181,3 | 109562,90 | 39299,00 | 3356,32 | 1132,44 |
| Рівненська | 19302,00 | 62,5 | 25135,90 | 26144,00 | 543,11 | 430,97 |
| Сумська | 22907,00 | 88 | 33050,50 | 28347,00 | 1017,38 | 814,09 |
| Тернопільська | 16294,00 | 65,4 | 33644,30 | 22712,00 | 241,43 | 324,11 |
| Харківська | 76866,00 | 303 | 168111,10 | 77316,00 | 1845,83 | 2277,27 |
| Херсонська | 18448,00 | 74,9 | 27742,90 | 24096,00 | 339,15 | 238,54 |
| Хмельницька | 22843,00 | 83,5 | 28642,70 | 31336,00 | 393,70 | 554,66 |
| Черкаська | 27012,00 | 140,2 | 55116,00 | 29646,00 | 1058,43 | 508,90 |
| Чернівецька | 11969,00 | 45,4 | 12614,40 | 18682,00 | 136,57 | 162,97 |
| Чернігівська | 21165,00 | 98 | 27018,10 | 26277,00 | 418,67 | 562,49 |
| м.Київ | 223774,00 | 254,5 | 1297770,50 | 164057,00 | 9061,71 | 22775,44 |

Таблиця 3. Значення окремих показників еколого-економічного розвитку регіонів України за 2012 рік

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Обл.** | Індикатори | | | | | |
| ВРП(млн грн) | ЗР(млн кг) | РП(млн грн) | ДН(млн грн) | Е(млн дол) | І(млн дол) |
| Вінницька | 33024,00 | 182,7 | 46188,40 | 44265,00 | 652,40 | 498,36 |
| Волинська | 20005,00 | 50,4 | 52334,60 | 25741,00 | 605,78 | 1015,14 |
| Дніпропетровська | 147970,00 | 1173,1 | 447705,40 | 118823,00 | 10129,72 | 6538,26 |
| Донецька | 170775,00 | 1714,7 | 664615,90 | 158003,00 | 14127,18 | 4168,47 |
| Житомирська | 24849,00 | 85,7 | 30183,80 | 34110,00 | 550,04 | 384,27 |
| Закарпатська | 21404,00 | 72,1 | 27533,70 | 28028,00 | 1385,09 | 2009,08 |
| Запорізька | 54828,00 | 316 | 124278,20 | 59191,00 | 4004,80 | 2013,43 |
| Івано-Франківська | 32286,00 | 249,1 | 38354,50 | 36186,00 | 821,54 | 983,75 |
| Київська | 69663,00 | 308,1 | 227722,60 | 55941,00 | 1982,97 | 4793,23 |
| Кіровоградська | 22056,00 | 73,8 | 44308,20 | 27129,00 | 630,61 | 284,87 |
| Луганська | 58767,00 | 529,5 | 117310,40 | 67025,00 | 4192,71 | 1989,59 |
| Львівська | 61962,00 | 253,9 | 131984,60 | 72828,00 | 1343,54 | 3373,48 |
| Миколаївська | 29205,00 | 87,1 | 52795,30 | 33907,00 | 2370,72 | 895,65 |
| Одеська | 64743,00 | 169 | 141059,40 | 70429,00 | 1784,64 | 4304,58 |
| Полтавська | 56580,00 | 178,9 | 118997,70 | 44835,00 | 3404,41 | 1322,74 |
| Рівненська | 21795,00 | 60,4 | 25530,50 | 29557,00 | 503,54 | 458,65 |
| Сумська | 24933,00 | 80,1 | 36761,80 | 32415,00 | 1122,09 | 678,82 |
| Тернопільська | 17957,00 | 65 | 48897,30 | 26102,00 | 236,82 | 336,85 |
| Харківська | 82223,00 | 319,4 | 179260,00 | 88352,00 | 2022,11 | 2741,04 |
| Херсонська | 19357,00 | 73,6 | 26885,80 | 27221,00 | 328,47 | 206,64 |
| Хмельницька | 26237,00 | 79,7 | 32905,90 | 35458,00 | 460,42 | 564,88 |
| Черкаська | 31265,00 | 146,4 | 63723,90 | 33484,00 | 1007,65 | 576,91 |
| Чернівецька | 13166,00 | 41,1 | 13290,50 | 21012,00 | 124,92 | 174,89 |
| Чернігівська | 23934,00 | 93,6 | 33293,90 | 29654,00 | 542,36 | 544,79 |
| м.Київ | 275685,00 | 259,2 | 1378512,30 | 194436,00 | 12770,31 | 26115,99 |

Блок-схема алгоритму розрахунку коефіцієнта кореляції наведена на рисунку 1.

Вхід

Сума Х

Сума Х\*Y

Сума Y\*Y

Середнє значення Y = Сума Y / n

Середнє значення X\*Y = Сума X\*Y / n

Дисперсія Х = Сума ХХ / n – (Середнє значення Х)2

Дисперсія Y = Сума YY / n – (Середнє значення Y)2

Середнє квадратичне У =

Вихід

Рис. 1. Блок-схема алгоритму розрахунку коефіцієнта кореляції

Згідно даних таблиць 1-3 побудуємо матрицю кореляції засобами Excel.

На рисунку 2 наведений скріншот з результатами побудови матриці кореляції за 2010 р.

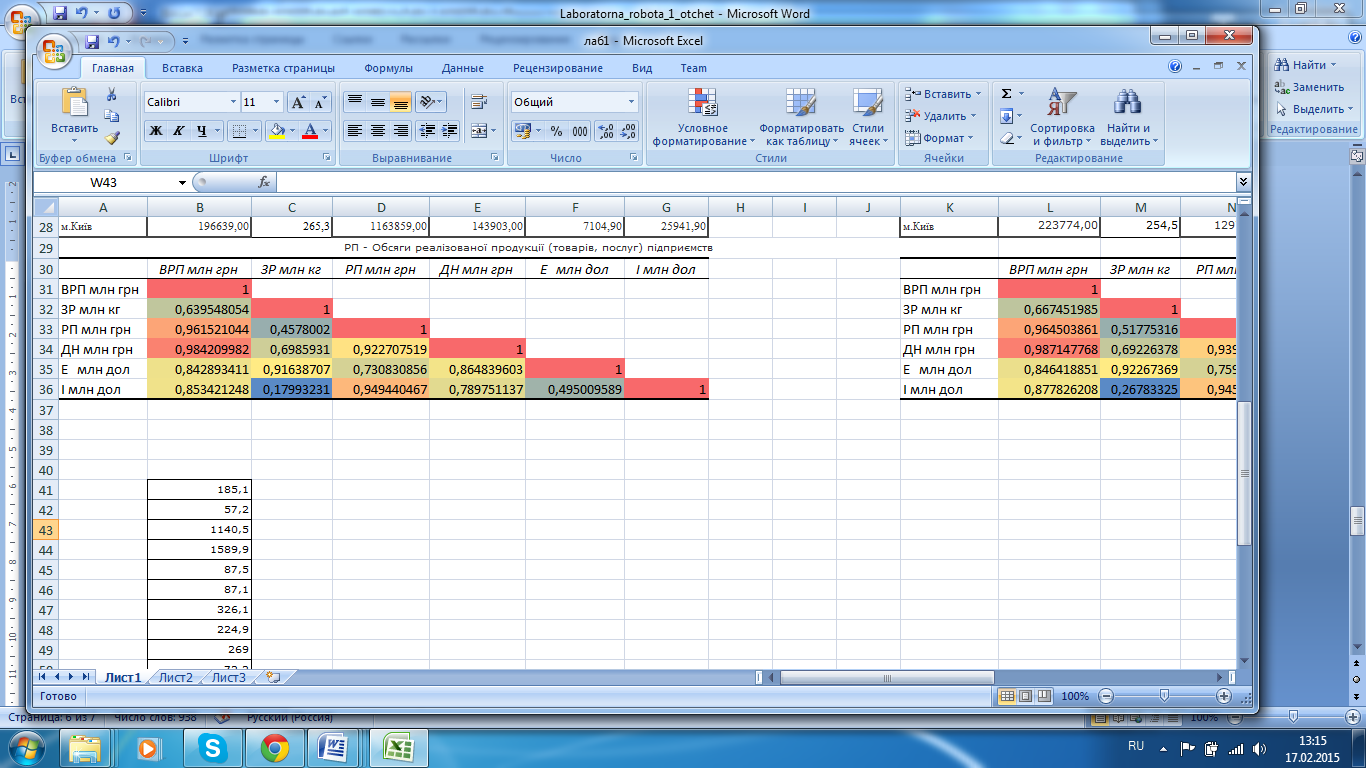


Рис. 2. Скріншоти з результатами побудови матриці кореляції за 2010 р.

Таблиця 5 показує аналіз значень коефіцієнтів кореляції впливу економічних факторів на викиди забруднюючих речовин атмосферне повітря відповідно до шкали Чедока [2].

Таблиця 5. Аналіз значень коефіцієнтів кореляції

|  |  |  |  |
| --- | --- | --- | --- |
| Діапазон | Сила зв’язку | фактори | показник |
| 0.9 – 1 | досить висока | Експорт | 0,916 |
| 0.5 – 0.7 | примітна | Дохід населення | 0,698 |
| Валовий регіональний продукт | 0,639 |
| 0.3 – 0.5 | помірна | Реалізована продукція | 0,457 |
| 0.1 – 0.3 | слабка | Імпорт | 0,179 |

Динаміка зміни значень коефіцієнтів кореляції за період 2010-2012 роки наведено на рисунку 3.

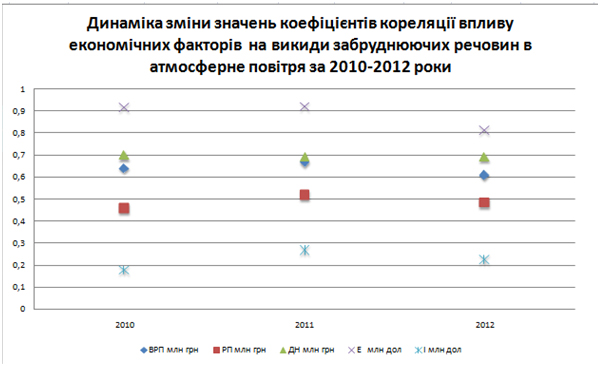
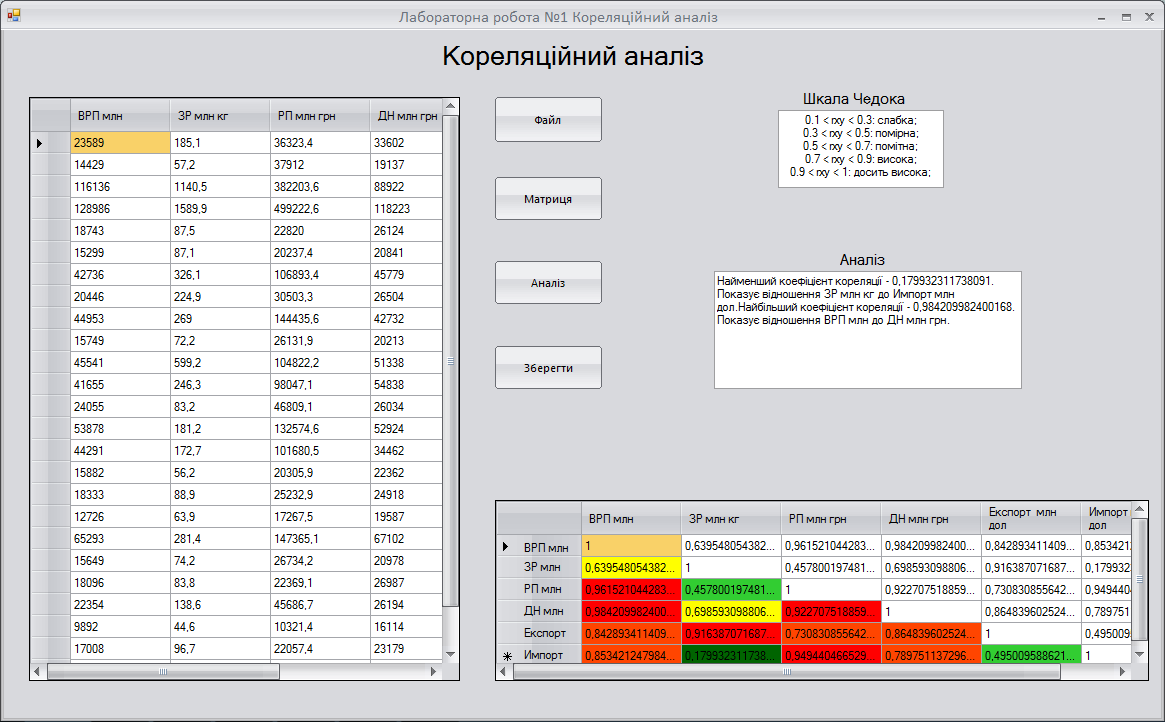


Рис. 3. Динаміка зміни значень коефіцієнтів кореляції за період 2010-2012 р.

У відповідності алгоритму розрахунку коефіцієнта кореляції розроблено відповідно програмний продукт на мові програмування С#. Приклад скріншоту з результатом розрахунку наведено на рисунку 4. Код програми наведений в Додатку 1.

Рис. 4. Скріншот результату роботи програми з даними за 2010 рік.

Співпадіння між результатами розрахунку засобами Excel та власним програмним продуктом, свідчать про коректність розроблення програмного продукту.

Висновки: у ході лабораторної роботи проводився кореляційний аналіз впливу економічних чинників на викиди забруднюючих речовин в атмосферне повітря. Найбільший вплив має показник експорту(коефіцієнт кореляції у 2010 році становить 0.916), найменший вплив має показник імпорту (коефіцієнт кореляції у 2010 році становить 0.179) .

Розроблений програмний продукт може використовуватися користувачами для дослідження факторних аналізів, коректність правильного обчислення даних підтверджено результатами порівняння оброблених даних програмою і обрахованих засобами Excel.

**МУЛЬТИКОЛІНЕАРНИЙ АНАЛІЗ**

**Задачі:**

1. Обчислити середні значення по кожному фактору.
2. Обчислити дисперсію по кожному фактору.
3. Сформувати нормалізовану матрицю.
4. Розрахувати кореляційну матрицю на основі нормалізованих даних.
5. Розрахувати детермінант кореляційної матриці.
6. Обчислити критерій .
7. Визначити обернену матрицю.
8. Обчислити F-критерій (Фішера).
9. Обчислити *t*-критерії Стьюдента.
10. Вибір показників для регресійної моделі.

Для обчислення середнього значення відповідного фактора необхідно за допомогою функції СРЗНАЧ розрахувати середнє значення відповідного діапазону. Для першого стовпчика необхідно виділити комірку В28 та ввести формулу «=СРЗНАЧ(B3:B27)», де B3:B27 – діапазон значень першого стовпчика. Виділивши комірку В27 і затиснувши ліву кнопку миші, перетягнути курсор вправо для вибору шести клітинок (відповідно до кількості факторів) і натиснути комбінацію клавіш «Ctrl»+«R» для копіювання формули.

Під кожною коміркою із середнім значенням необхідно розрахувати дисперсію. Для першого стовпчика вибираємо клітинку В29 і вводимо формулу «=ДИСПР(B3:B27)». Аналогічно копіюємо формулу в наступні п’ять правих комірок.

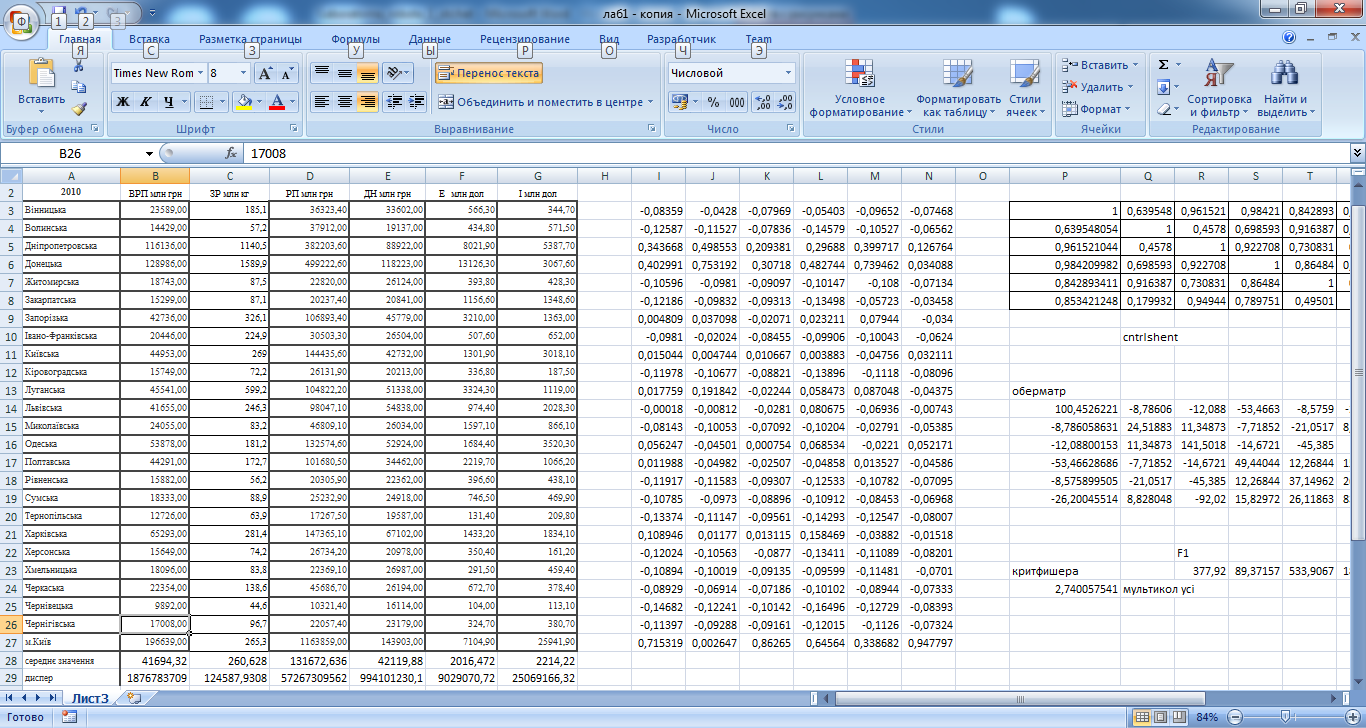


Рис. 5. Скріншот розрахунку середнього значення і дисперсії з даними за 2010 рік.

Нормалізована матриця має той самий розмір як і вихідна (25 регіонів × 6 факторів). Нормалізовану матрицю бажано розмістити на одному рівні з вихідною, наприклад, в діапазоні комірок І3:N27. Для розрахунку першого нормалізованого значення необхідно вибрати клітинку І3 та задати формулу «=(B3-B$28)/ КОРЕНЬ(27\*B$32)», де B$28 – зафіксоване середнє значення, B$29 – зафіксована дисперсія, 25 – кількість об’єктів.

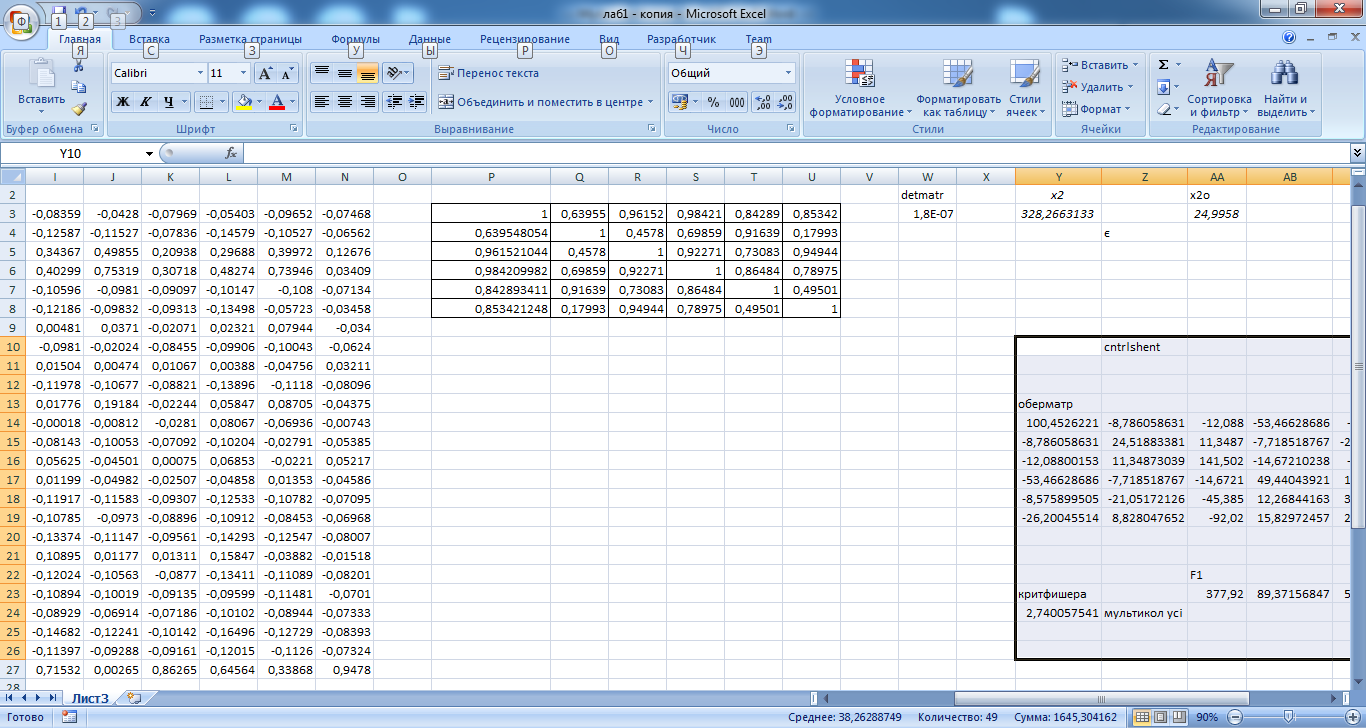


Рис. 6. Значення показників нормалізованої матриці за 2010 рік.

Для розрахунку кореляційної матриці на основі нормалізованих даних необхідно визначити квадратний діапазон комірок розміром 6 × 6 (відповідно до кількості факторів). Наприклад, кореляційна матриця буде зберігатися в комірках P3:U8. Далі виділяють комірку P3 і, затиснувши ліву клавішу миші, розширюють вибір до клітинки U8. В рядку формул задають вираз «=МУМНОЖ(ТРАНСП(I3:N27);I3:N27)»

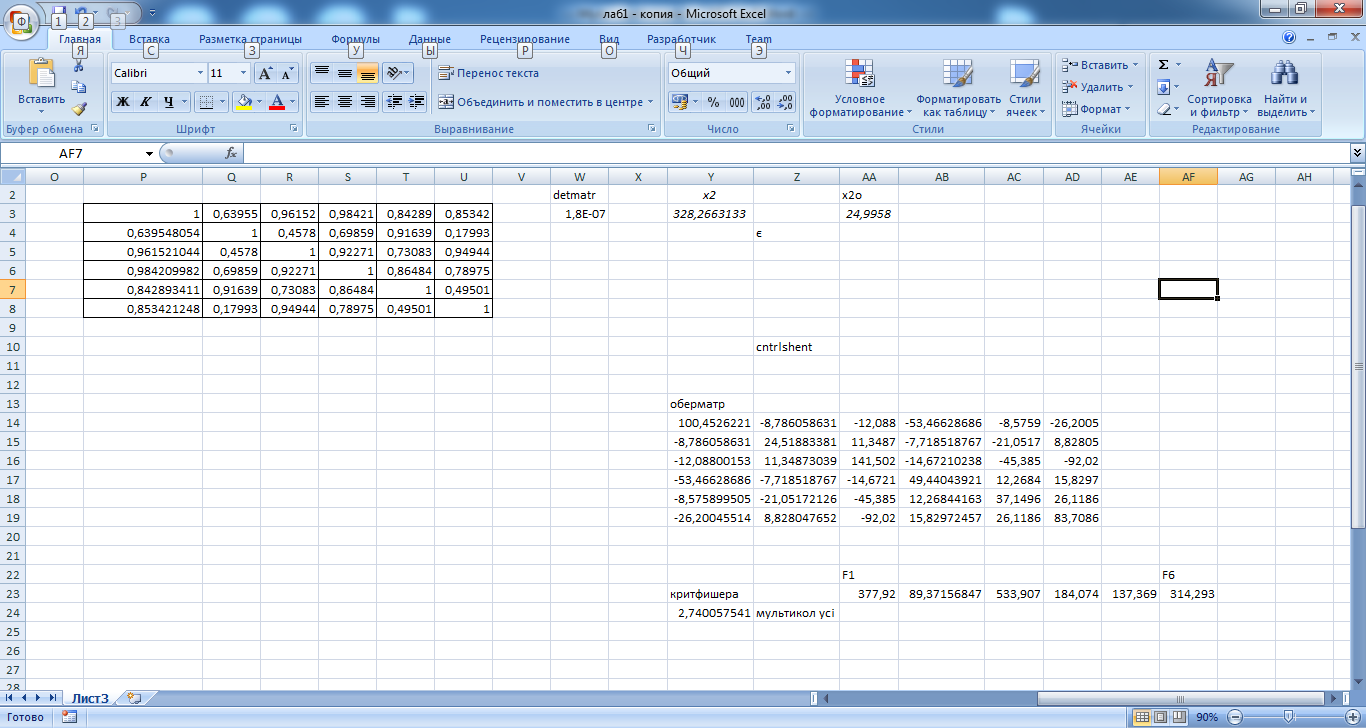


Рис. 7. Коефіцієнти кореляційної матриці.

Детермінант квадратної матриці в Excel визначається за допомогою функції МОПРЕД. Наприклад, виділимо комірку Р12 і введемо формулу «=МОПРЕД(P3:U8)», де P3:U8 – діапазон комірок кореляційної матриці.

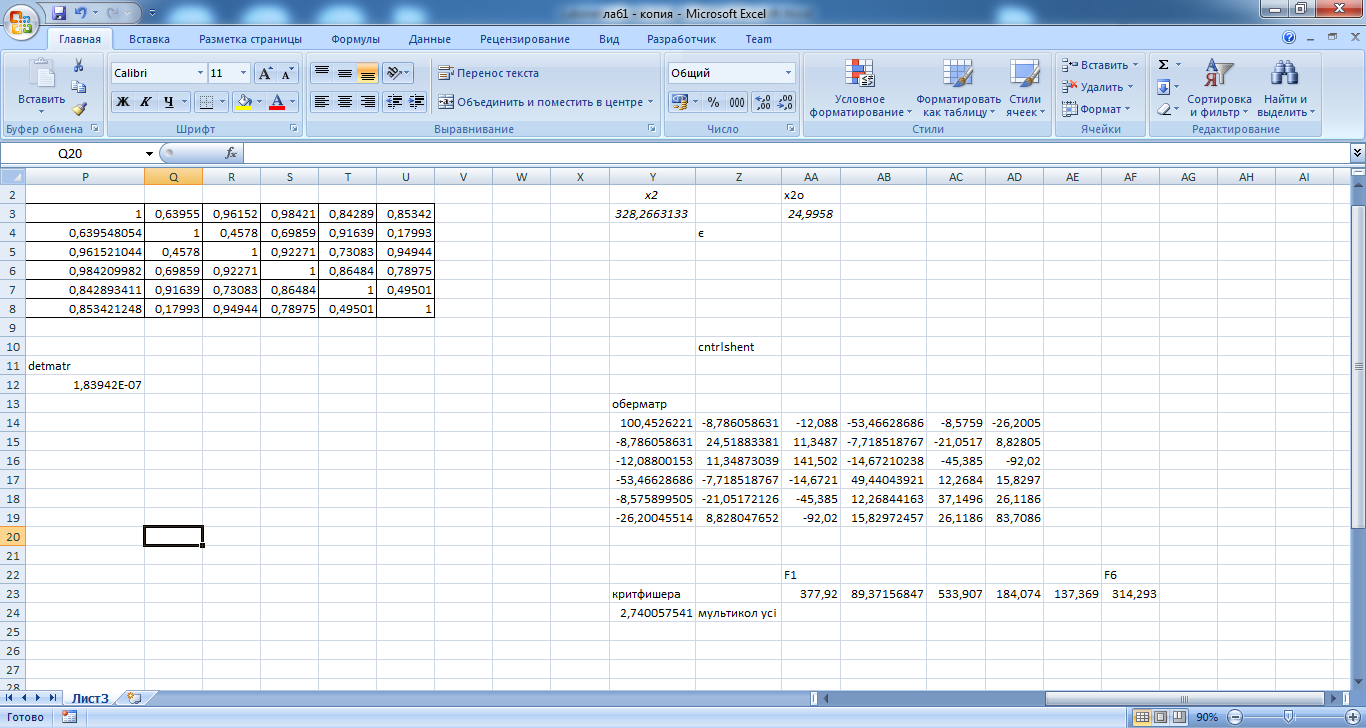


Рис. 8. Детермінант кореляційної матриці.

Для розрахунку фактичного значення -критерію у вибрану комірку, наприклад R12, вводять формулу «=-(25-1-(2\*6+5)/6)\*LN(P12)», де 25 – кількість об’єктів, 2\*6 – подвоєна кількість факторів, Р12 – значення детермінанту кореляційної матриці.

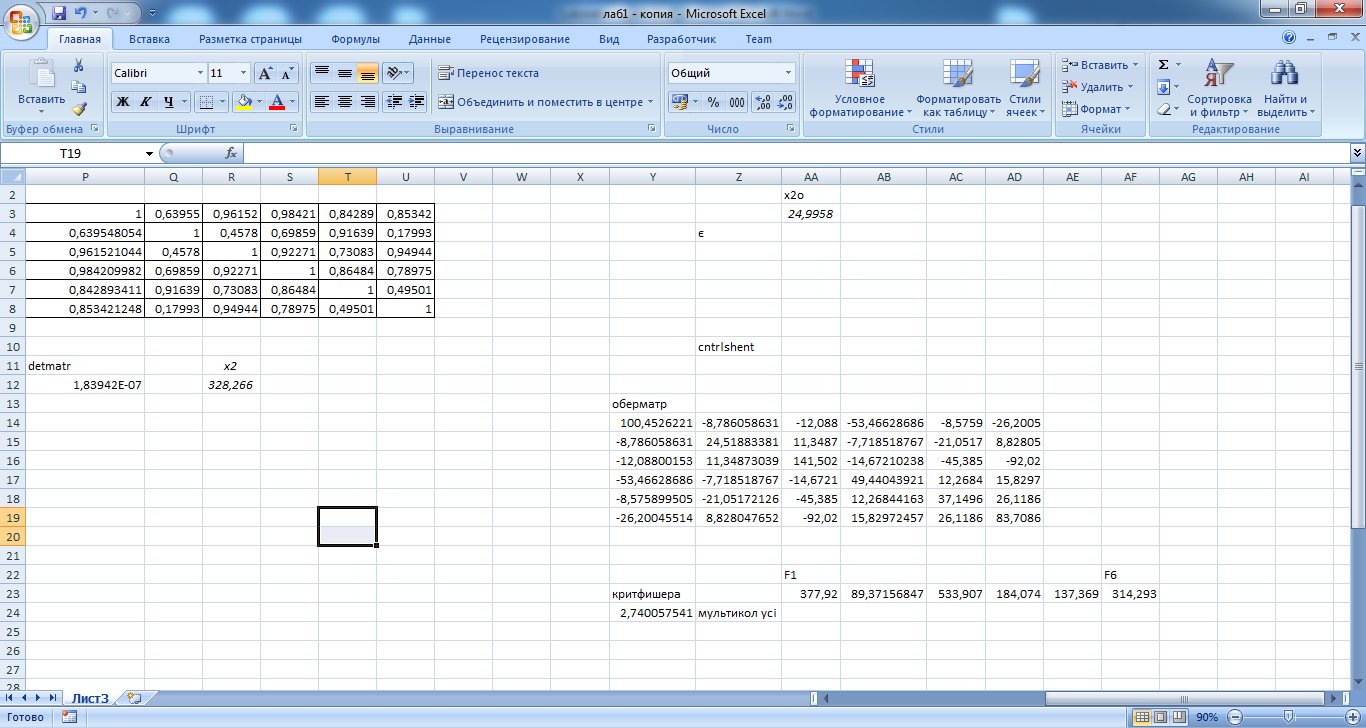


Рис. 9. Фактичне значення -критерію.

Для визначення мультиколінеарності по всім факторам розраховане значення необхідно порівняти із табличним при визначеному ступені вільності і заданому рівні значимості. Табличне значення розрахуємо до комірки T12 за допомогою формули «=ХИ2.ОБР(0,05;6\*(6-1)/2)».

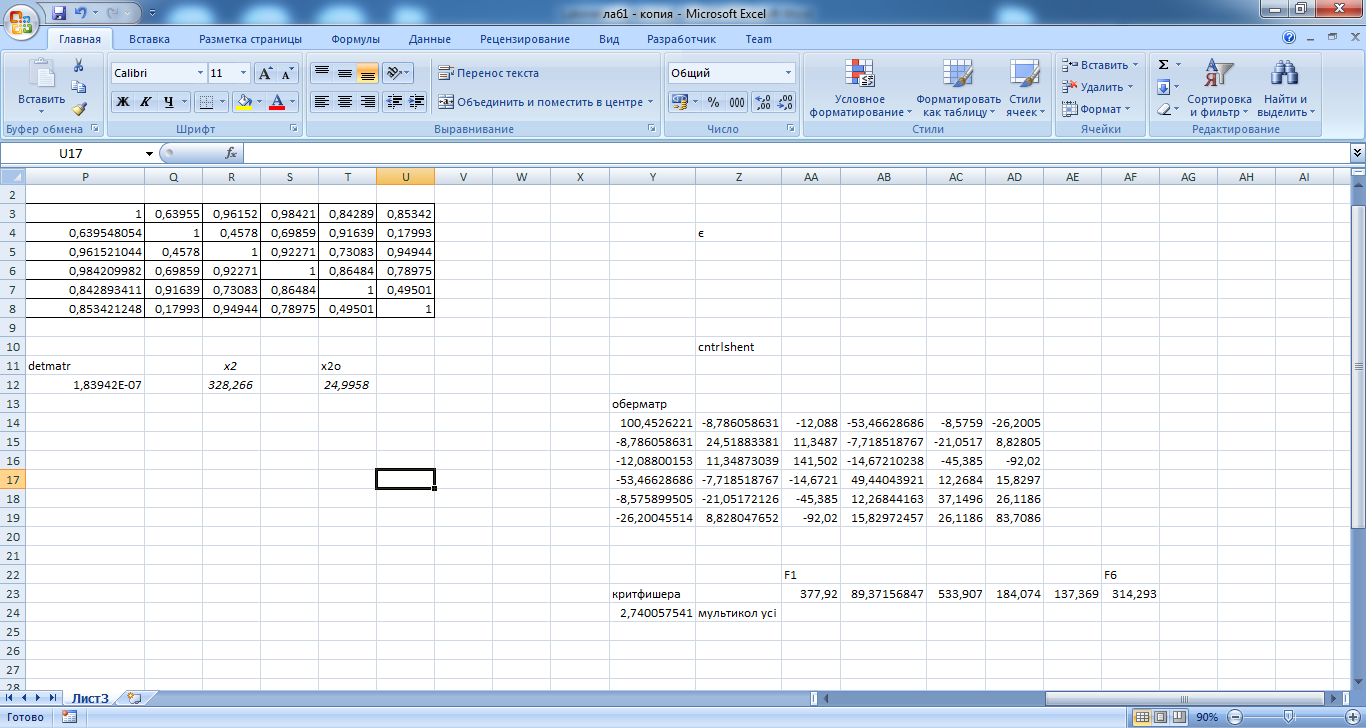


Рис. 10. Табличне значення -критерію.

Порівнюючи фактичне значення (комірка R12)із табличним значенням (комірка T12) -критерію, приходимо до висновку, що між показниками існує мультиколінеарність.

Для визначення оберненої матриці до відповідної матриці кореляції нормалізованого набору даних необхідно виділити квадратний діапазон комірок, наприклад, P14:U19. В рядку формул ввести вираз «=МОБР(P3:U8)» і одночасно натиснути комбінацію клавіш «Ctrl»+«Shift»+«Enter».

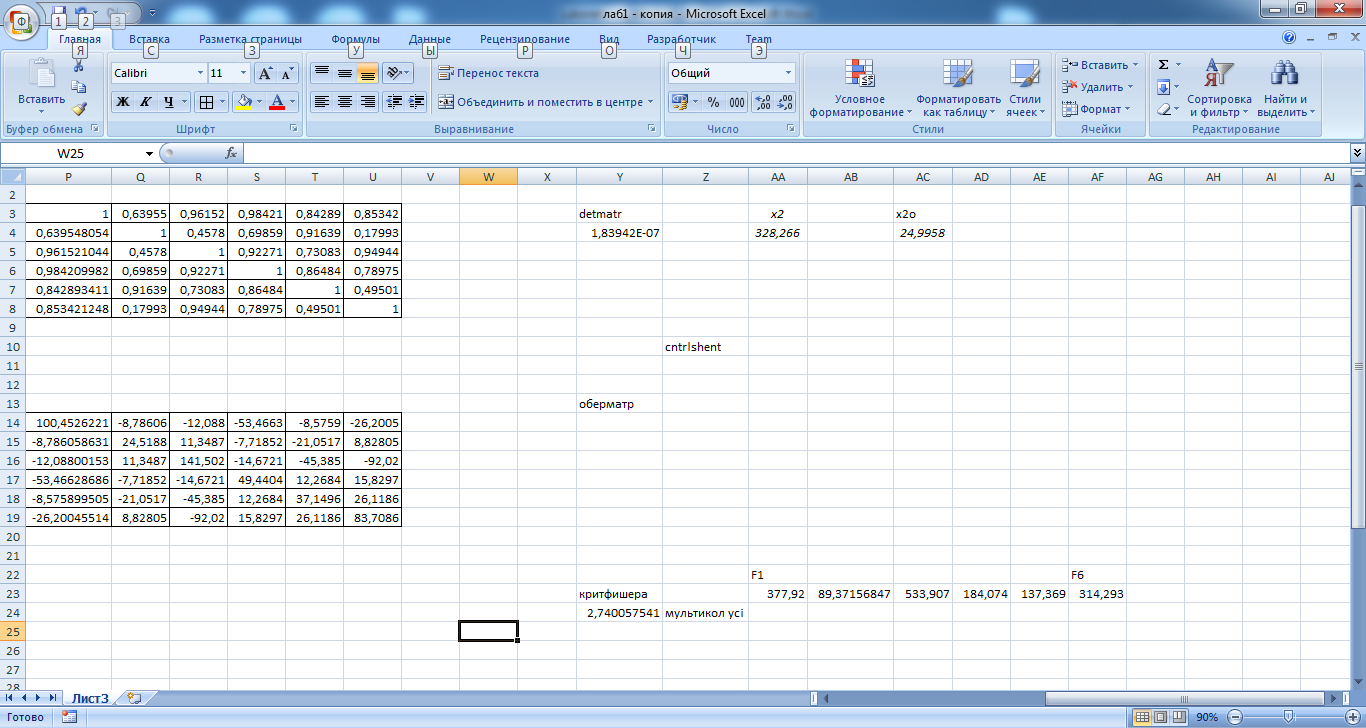


Рис. 11. Обернена матриця до відповідної матриці кореляції нормалізованого набору даних.

Фактичні значення F-критерію будемо розраховувати за формулою 1.



Формула 1. Розрахунок значень F-критерію.

Результати зберігаємо в масив комірок P23:U23. При цьому всі значення розраховуються окремо, оскільки в розрахунках необхідно використовувати діагональні елементи оберненої матриці. Перше значення для комірки P23 обчислюється за допомогою виразу «=(E14-1)\*(25-6)/(6-1)».

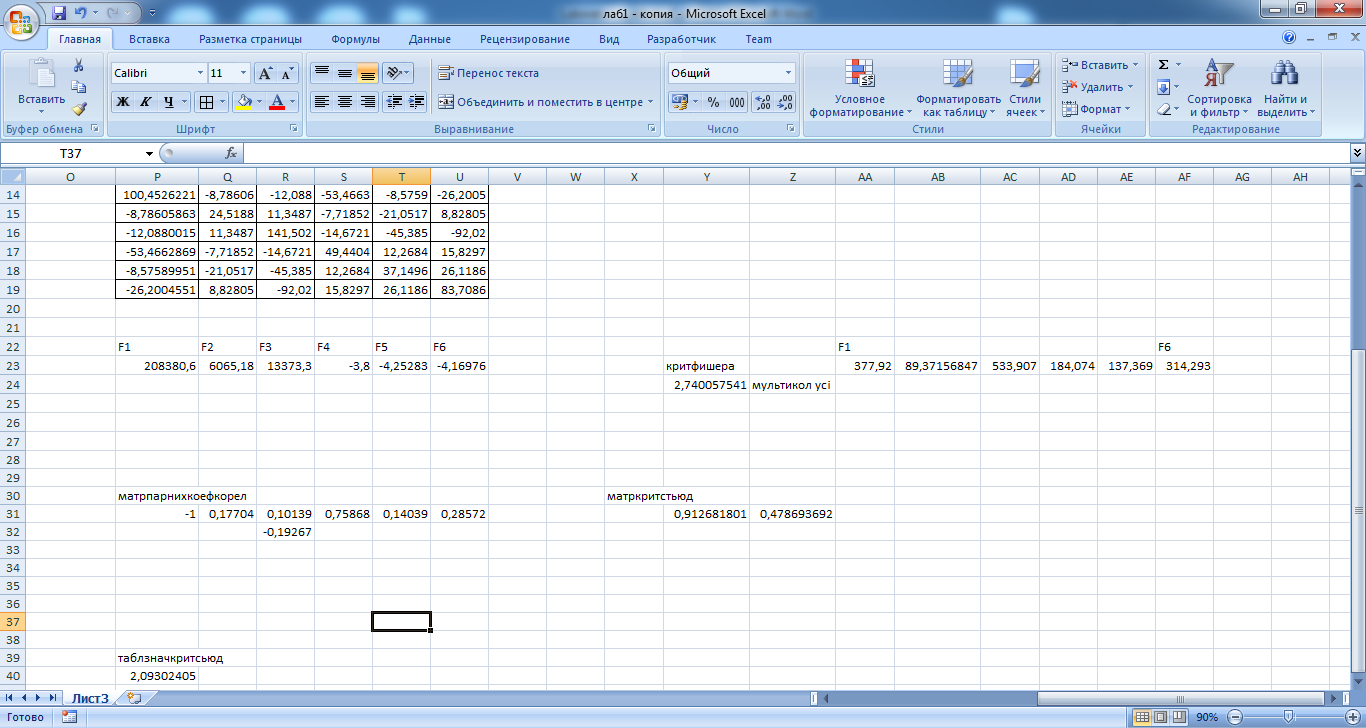


Рис. 12. Значення F-критеріїв.

Критичне (табличне) значення *F*-критерію розраховується за допомогою статистичної функції «=FРАСПОБР(0,05;(6-1);(25-6))».

Таблиця 6. Аналіз значень критеріїв Фішера.

|  |  |  |
| --- | --- | --- |
| Фактичне значення | Знак | Табличне значення |
| 377,92 | > | 2,740057541 |
| 89,37156847 | > |
| 533,9067 | > |
| 184,0737 | > |
| 137,3686 | > |
| 314,2927 | > |

Отже, усі фактори є мультиколінеарними.

Частинні коефіцієнти кореляції розраховуються на основі значень елементів оберненої матриці за формулою 2. Значення *t*-критеріїв Стьюдента розраховується на основі частинних коефіцієнтів кореляції за формулою 3. На цьому етапі тестування мультиколінеарності має сенс лише для пари різних показників. Тому пропускають розрахунки діагональних елементів (формула не може бути обчислена). Табличного значення *t*-критерію Стьюдента в Excel обчислюється за допомогою статистичної формули СТЬЮДЕНТ.ОБР.



Формула 2. Розрахунок частинних коефіцієнтів кореляції.



Формула 3. Розрахунок *t*-критеріїв.

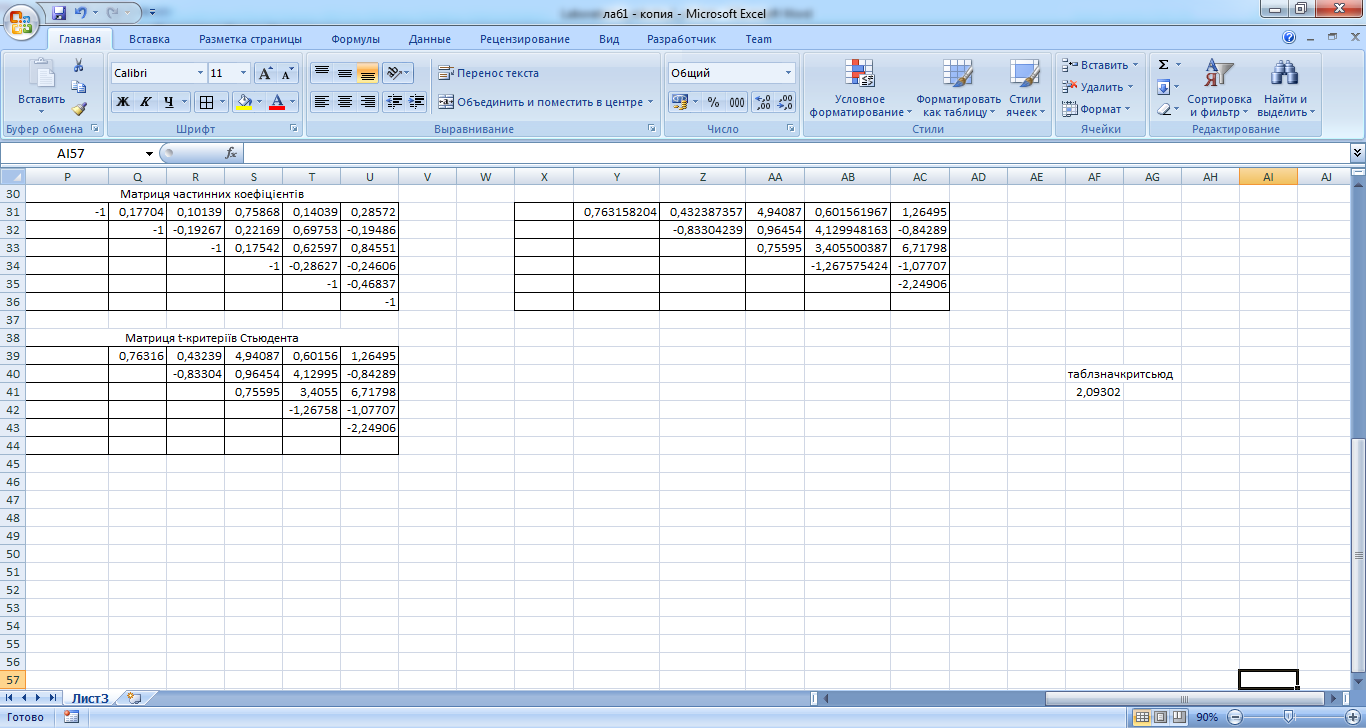


Рис. 13. Матриця частинних коефіцієнтів та матриця критеріїв Стьюдента.

Таблиця 7. Аналіз значень критеріїв Стьюдента.

|  |  |  |  |
| --- | --- | --- | --- |
| Фактори | Фактичне значення | Знак | Табличне значення |
| ДН і ВРП | 4,940 | > | 2,093 |
| Е і ЗР | 4,129 | > |
| Е і РП | 3,405 | > |
| І і РП | 6,717 | > |

У відповідності алгоритму проведення мультиколінеарного аналізу розроблено відповідно програмний продукт на мові програмування С#. Приклад скріншоту з результатом розрахунку наведено на рисунку 14. Код програми наведений в Додатку 2.

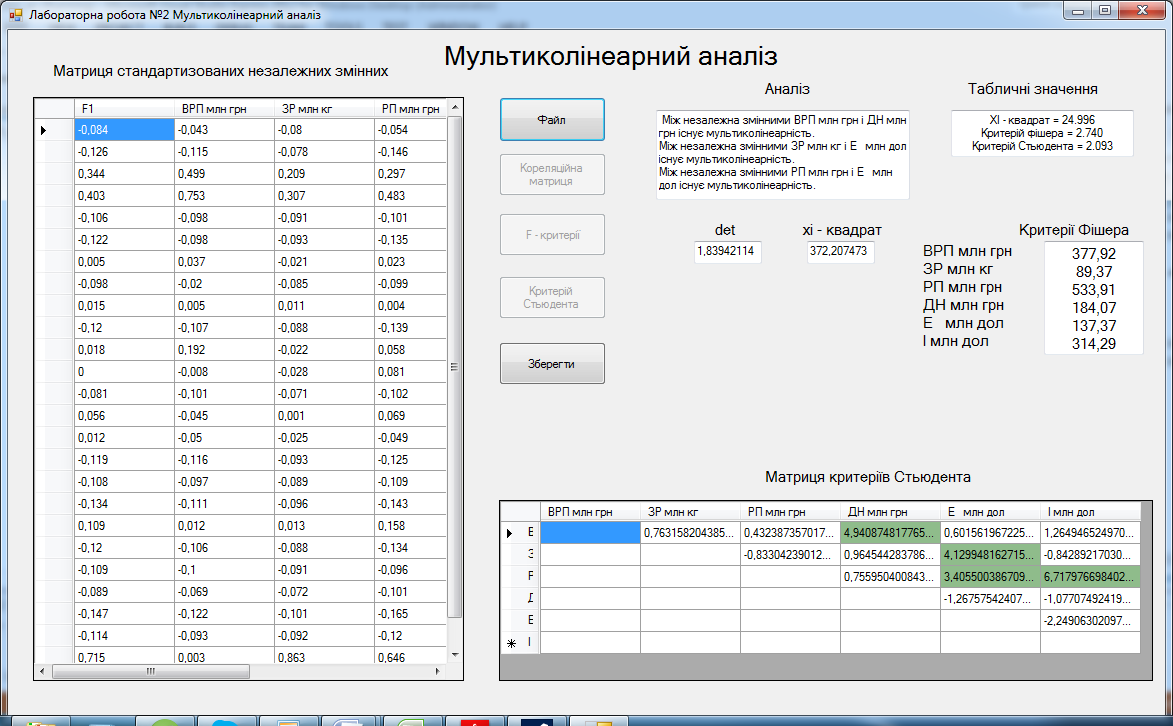


Рис. 14. Результат виконання програми.

Співпадіння між результатами розрахунку засобами Excel та власним програмним продуктом, свідчать про коректність розроблення програмного продукту.

Висновок: на основі порівняння фактичних значень *t*-критеріїв з критичним значенням критерію Стьюдента робимо висновок, що між парами факторів \*\*\*\* існує мультиколінеарність ,оскільки відповідні фактичні значення більші за критичну величину. Для виключення із складу регресійної моделі обираємо показник \*\*\*, оскільки \*\*\*. Таким чином, до складу регресійної моделі будуть входити наступні показники.

Додаток 1

public partial class Form1 : Form

{

double min = 0;

double max = 0;

int mini = 0, minj = 0;

int maxi = 0, maxj = 0;

private void button3\_Click(object sender, EventArgs e)

{

inti = 0;

int j = 0;

double[,] matrs;

matrs = new double[dataGridView1.RowCount, dataGridView1.ColumnCount];

for ( i = 0; i< dataGridView1.RowCount; i++)

{

for ( j = 0; j < dataGridView1.ColumnCount; j++)

{

matrs[i, j] = Convert.ToDouble(dataGridView1.Rows[i].Cells[j].Value) ;

}

}

double[,] x;

double[,] y;

x = new double[matrs.GetLength(0), 1];

y = new double[matrs.GetLength(0), 1];

int k = 0;

int l = 0;

intsumRow = matrs.GetLength(0) - 1;

doublesumx = 0;

doublesumy = 0;

doublesumxy = 0;

doublesumxx = 0;

doublesumyy = 0;

doubleavrx = 0;

doubleavry = 0;

doubleavrxy = 0;

doubledispx = 0;

doubledispy = 0;

doubleavrsqrx = 0;

doubleavrsqry = 0;

doublekoefkorel = 0;

dataGridView2.RowCount = matrs.GetLength(1);

dataGridView2.ColumnCount = matrs.GetLength(1);

for (l = 0; l <matrs.GetLength(1); l++)

{

for (k = 0; k <matrs.GetLength(1); k++)

{

sumx = 0;

sumy = 0;

sumxy = 0;

sumxx = 0;

sumyy = 0;

avrx = 0;

avry = 0;

avrxy = 0;

dispx = 0;

dispy = 0;

avrsqrx = 0;

avrsqry = 0;

koefkorel = 0;

for (i = 0; i<sumRow; i++)

{

for (j = 0; j < 1; j++)

{

x[i, j] = matrs[i, l];

sumx = sumx + x[i, j];

y[i, j] = matrs[i, k];

sumy = sumy + y[i, j];

sumxy = sumxy + x[i, j] \* y[i, j];

sumxx = sumxx + x[i, j] \* x[i, j];

sumyy = sumyy + y[i, j] \* y[i, j];

}

avrx = sumx / sumRow;

avry = sumy / sumRow;

avrxy = sumxy / sumRow;

dispx = sumxx / sumRow - avrx \* avrx;

dispy = sumyy / sumRow - avry \* avry;

avrsqrx = Math.Sqrt(dispx);

avrsqry = Math.Sqrt(dispy);

koefkorel = (avrxy - avrx \* avry) / (avrsqrx \* avrsqry);

}

dataGridView2.Rows[l].Cells[k].Value = koefkorel;

}

dataGridView2.Columns[l].HeaderText = dataGridView1.Columns[l].HeaderText;

dataGridView2.Rows[l].HeaderCell.Value = dataGridView1.Columns[l].HeaderText;

}

double[,] rez;

rez = new double[dataGridView2.RowCount, dataGridView2.ColumnCount];

for (i = 0; i< dataGridView2.RowCount; i++)

{

for (j = 0; j < dataGridView2.ColumnCount; j++)

{

rez[i, j] = Convert.ToDouble(dataGridView2.Rows[i].Cells[j].Value);

}

}

min = rez[1, 0];

max = rez[1, 0];

for ( i = 0; i<rez.GetLength(0); i++)

{

for ( j = 0; j <rez.GetLength(1); j++)

{

if (i> j)

{

if (min >rez[i, j])

{

min = rez[i, j];

mini = i;

minj = j;

}

if (max <rez[i, j])

{

max = rez[i, j];

maxi = i;

maxj = j;

}

}

}

}

}

}

}

Додаток 2

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.OleDb;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Threading;

using Excel = Microsoft.Office.Interop.Excel;

namespace Lab1\_ec

{

public partial class Form1 : Form

{

double[,] rez;

int sumRow = 0;

/\* Microsoft.Office.Interop.Excel.Application xl = new Microsoft.Office.Interop.Excel.Application();

Microsoft.Office.Interop.Excel.Worksheet wa = xl.ActiveSheet;

Microsoft.Office.Interop.Excel.WorksheetFunction wsf = xl.WorksheetFunction;

double f1 = wsf.F\_Dist(0.05,27,5,false);

double f2 = wsf.ChiInv(0.05, 2);\*/

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

OpenFileDialog dialog = new OpenFileDialog();

dialog.Filter = "Файл Excel|\*.XLSX;\*.XLS"; ;

if (dialog.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

OleDbConnection connection = new OleDbConnection(string.Format("Provider=Microsoft.ACE.OLEDB.12.0;Data Source={0};Extended Properties=\"Excel 12.0 Xml;HDR=YES\";", dialog.FileName));

OleDbDataAdapter adapter = new OleDbDataAdapter("SELECT \* FROM [Лист1$]", connection);

DataTable table = new DataTable();

adapter.Fill(table);

dataGridView1.DataSource = table;

}

}

private void button2\_Click(object sender, EventArgs e)

{

var save = new SaveFileDialog

{

AddExtension = true,

Filter = @"Файл Excel|\*.XLS",

FilterIndex = 2,

RestoreDirectory = true

};

if (save.ShowDialog() != DialogResult.OK) return;

var sw = new StreamWriter(save.FileName, true, Encoding.UTF8);

foreach (DataGridViewRow row in dataGridView2.Rows)

if (!row.IsNewRow)

{

var first = true;

foreach (DataGridViewCell cell in row.Cells)

{

if (!first) sw.Write(";");

sw.Write(cell.Value.ToString());

first = false;

}

sw.WriteLine();

}

sw.Close();

}

private void button3\_Click(object sender, EventArgs e)

{

int i = 0;

int j = 0;

double[,] matrs;

matrs = new double[dataGridView1.RowCount, dataGridView1.ColumnCount-1];

for ( i = 0; i < matrs.GetLength(0); i++)

{

for ( j = 0; j < matrs.GetLength(1); j++)

{

matrs[i, j] = Convert.ToDouble(dataGridView1.Rows[i].Cells[j+1].Value) ;

}

}

double[] avr;

double[] sumkv;

double[] disp;

double[,] matrs2;

matrs2 = new double[dataGridView1.RowCount, dataGridView1.ColumnCount];

sumRow = matrs.GetLength(0) - 1;

avr = new double[matrs.GetLength(1)];

sumkv = new double[matrs.GetLength(1)];

disp = new double[matrs.GetLength(1)];

double summ = 0;

// Середнє

for ( j = 0; j < matrs.GetLength(1); j++)

{

summ = 0;

for ( i = 0; i < matrs.GetLength(0); i++)

{

summ += matrs[i, j];

}

avr[j] = summ / sumRow;

}

// Дисперсія

for (j = 0; j < matrs.GetLength(1); j++)

{

summ = 0;

for (i = 0; i < matrs.GetLength(0)-1; i++)

{

summ += (matrs[i, j] - avr[j]) \* (matrs[i, j] - avr[j]);

}

disp[j] = summ / sumRow;

}

for (j = 0; j < matrs.GetLength(1); j++)

Console.WriteLine(" di ={0}",disp[j]);

// Матриця кореляції

label9.Text = "Матриця стандартизованих незалежних змінних"; ;

for ( i= 0; i < matrs.GetLength(0)-1; i++)

{

for (j = 0; j < matrs.GetLength(1) ; j++)

{

matrs2[i, j] = (matrs[i, j] - avr[j]) / Math.Sqrt(disp[j] \* sumRow);

dataGridView1.Rows[i].Cells[j].Value = Math.Round(matrs2[i, j]\*1000)/1000.0;

}

}

label8.Text = "Кореляційна матриця";

int k = 0;

int l = 0;

double sumxy = 0;

double avrxy = 0;

double koefkorel = 0;

dataGridView2.RowCount = matrs.GetLength(1);

dataGridView2.ColumnCount = matrs.GetLength(1);

for (l = 0; l < matrs.GetLength(1); l++)

{

for (k = 0; k < matrs.GetLength(1); k++)

{

sumxy = 0;

for (i = 0; i < sumRow; i++)

{

sumxy = sumxy + matrs[i, l] \* matrs[i, k];

}

avrxy = sumxy / sumRow;

koefkorel = (avrxy - avr[k] \* avr[l]) / (Math.Sqrt(disp[l]) \* Math.Sqrt(disp[k]));

dataGridView2.Rows[l].Cells[k].Value = koefkorel;

}

dataGridView2.Columns[l].HeaderText = dataGridView1.Columns[l+1].HeaderText;

dataGridView2.Rows[l].HeaderCell.Value = dataGridView1.Columns[l+1].HeaderText;

}

rez = new double[dataGridView2.RowCount, dataGridView2.ColumnCount];

for (i = 0; i < dataGridView2.RowCount; i++)

{

for (j = 0; j < dataGridView2.ColumnCount; j++)

{

rez[i, j] = Convert.ToDouble(dataGridView2.Rows[i].Cells[j].Value);

}

}

// Визначник матриці, ХІ - квадрат

const double xitable = 24.996;

double det = Determ(rez);

Console.WriteLine(det);

textBox3.Text = Convert.ToString(det);

double xi = -(sumRow - 1 - 1 / 6 \* (2 \* matrs.GetLength(1)+ 5)) \* Math.Log(det, Math.E);

textBox4.Text = Convert.ToString(xi);

if (xi > xitable)

{

textBox1.Text = "В масиві незалежних змінних існує мультиколінеарність.";

button5.Enabled = true;

}

else

{

MessageBox.Show("Аналіз завершено. Мультиколінеарності неіснує!");

}

button3.Enabled = false;

}

public static double[,] GetMinor(double[,] matrix, int row, int column)

{

double[,] buf = new double[matrix.GetLength(0) - 1, matrix.GetLength(0) - 1];

for (int i = 0; i < matrix.GetLength(0); i++)

for (int j = 0; j < matrix.GetLength(1); j++)

{

if ((i != row) || (j != column))

{

if (i > row && j < column) buf[i - 1, j] = matrix[i, j];

if (i < row && j > column) buf[i, j - 1] = matrix[i, j];

if (i > row && j > column) buf[i - 1, j - 1] = matrix[i, j];

if (i < row && j < column) buf[i, j] = matrix[i, j];

}

}

return buf;

}

public static double Determ(double[,] matrix)

{

if (matrix.GetLength(0) != matrix.GetLength(1)) throw new Exception(" Число строк в матрице не совпадает с числом столбцов");

double det = 0;

int Rank = matrix.GetLength(0);

if (Rank == 1) det = matrix[0, 0];

if (Rank == 2) det = matrix[0, 0] \* matrix[1, 1] - matrix[0, 1] \* matrix[1, 0];

if (Rank > 2)

{

for (int j = 0; j < matrix.GetLength(1); j++)

{

det += Math.Pow(-1, 0 + j) \* matrix[0, j] \* Determ(GetMinor(matrix, 0, j));

}

}

return det;

}

public static double[,] One1(double[,] matrix, int len)

{

double[,] ob = new double[len, len];

for (int i = 0; i < len; i++)

{

for (int j = 0; j < len;j++ )

{

if (i == j)

{ ob[i, j] = 1; }

else

{ ob[i, j] = 0; }

}

}

double arg;

int i1;

for (int j = 0; j < len; )

{

for (int i = 0; i < len; )

{

if (i == j)

{ goto k; }

arg = matrix[i, j] / matrix[j, j];

for (i1 = 0; i1 < len; )

{

matrix[i, i1] = matrix[i, i1] - matrix[j, i1] \* arg;

ob[i, i1] = ob[i, i1] - ob[j, i1] \* arg;

i1++;

}

k:

i++;

}

j++;

}

for (int j = 0; j < len; j++)

{

for (int i = 0; i < len; i++)

{

double arg\_2;

if (i == j)

{

arg\_2 = matrix[i, j];

for (i1 = 0; i1 < len; )

{

matrix[i, i1] = matrix[i, i1] / arg\_2;

ob[i, i1] = ob[i, i1] / arg\_2;

i1++;

}

}

}

}

return ob;

}

private void button5\_Click(object sender, EventArgs e)

{

button5.Enabled = false;

const double fish = 2.740;

label8.Text = "Обернена матриця";

textBox1.Text = " ";

//Обернена матриця

double[,] ober = One1(rez, rez.GetLength(0));

for (int i = 0; i < dataGridView2.RowCount; i++)

{

for (int j = 0; j < dataGridView2.ColumnCount; j++)

{

dataGridView2.Rows[i].Cells[j].Value = ober[i,j];

rez[i, j] = ober[i, j];

}

}

//Коефіцієнти фішера

double[] fisher = new double[rez.GetLength(0)];

for (int i = 0; i < dataGridView2.RowCount; i++)

{

fisher[i] = (ober[i, i] - 1) \* (sumRow - rez.GetLength(0)) / (rez.GetLength(0)-1);

}

for (int i = 0; i < dataGridView2.RowCount; i++)

{

textBox5.Text += Convert.ToString(Math.Round(fisher[i], 2)) + "\r\n";

label7.Text += dataGridView2.Columns[i].HeaderText + "\r\n";

}

bool check = false;

for (int i = 0; i < dataGridView2.RowCount; i++)

if (fisher[i] > fish)

{

textBox1.Text += dataGridView2.Columns[i].HeaderText + " - незалежна змінна, що мультиколінеарна з іншими. \r\n";

button4.Enabled = true;

check = true;

}

if (check == false)

{

MessageBox.Show("Аналіз завершено. Мультиколінеарності неіснує!");

}

}

private void button4\_Click(object sender, EventArgs e)

{

textBox1.Text = " ";

double[,] matrparkoef = new double[rez.GetLength(0), rez.GetLength(1)];

label8.Text = "Матриця частинних коефіцієнтів кореляції";

for (int i = 0; i < dataGridView2.RowCount; i++)

{

for (int j = 0; j < dataGridView2.ColumnCount; j++)

{

matrparkoef[i, j] = -rez[i, j] / (Math.Sqrt(rez[i, i] \* rez[j, j]));

dataGridView2.Rows[i].Cells[j].Value = matrparkoef[i, j];

}

}

//System.Threading.Thread.Sleep(3000);

double[,] krst = new double[rez.GetLength(0), rez.GetLength(1)];

label8.Text = "Матриця критеріїв Стьюдента";

const double st = 2.093;

for (int i = 0; i < dataGridView2.RowCount; i++)

{

for (int j = 0; j < dataGridView2.ColumnCount; j++)

{

dataGridView2.Rows[i].Cells[j].Value = " ";

}

}

for (int i = 0; i < dataGridView2.RowCount; i++)

{

for (int j = 0; j < dataGridView2.ColumnCount; j++)

{

if (j > i)

{

krst[i, j] = matrparkoef[i, j] \* Math.Sqrt(sumRow - rez.GetLength(0) - 1) / Math.Sqrt(1 - matrparkoef[i, j] \* matrparkoef[i, j]);

dataGridView2.Rows[i].Cells[j].Value = krst[i, j];

if (krst[i, j] > st)

{

dataGridView2.Rows[i].Cells[j].Style.BackColor = Color.DarkSeaGreen ;

textBox1.Text += "Між незалежна змінними " + dataGridView2.Columns[i].HeaderText + " і " + dataGridView2.Columns[j].HeaderText + " існує мультиколінеарність. \r\n";

}

}

}

}

button4.Enabled = false;

}

}

}

Список використаних джерел

1 <http://www.ukrstat.gov.ua/>

2 http://www.grandars.ru/student/statistika/korrelyacionno-regressionnyy-analiz.html