УДК 004.056.5 **doi: 10.26906/SUNZ.2022.1.052**

С. Ю. Гавриленко, В. Д. Зозуля

Національний технічний університет “Харківський політехнічний інститут”, Харків, Україна

ДОСЛІДЖЕННЯ МЕТОДІВ ВИЯВЛЕННЯ АНОМАЛІЙ НА ЕТАПІ ПОПЕРЕДНЬОЇ ОБРОБКИ ДАНИХ

Анотація. Предметом дослідження є методи та засоби виявлення аномалій в даних**. Метою статті** є підвищення якості класифікації даних за рахунок виявлення аномалій на етапі їх попередньої обробки. **Завдання**: дослідити методи виявлення аномалій на етапі попередньої обробки даних, визначити поріг прийняття рішень *anomaly_score* для кожного із методів та оцінити якість класифікації до та після *preprocessing*. Використовуваними **методами** є: методи штучного інтелекту, машинного навчання, ансамблеві методи. Отримано такі **результати**: досліджено методи виявлення аномалій: метод стандартного відхилення (*Standard Deviation Method*), метод локального рівня викидів (*Local Outlier Factor*), метод Ізолюючого лісу (*Isolation Forest*). Отримано залежність кількості аномалій від порогу прийняття рішень для кожного із методів. Оцінку якості попередньої обробки даних виконано з використанням класифікаторів на основі методів *KNN* та беггінгу (*Bagging*). Досліджені методи реалізовані програмно з використанням хмарного сервісу *GOOGLE COLAB* на основі *Jupyter Notebook.* **Висновки.** Наукова новизна отриманих результатів полягає у дослідженні методів виявлення аномалій на етапі попередньої обробки даних, вибору мета-алгоритму preprocessing та визначення оптимальних параметрів його налаштування.

Ключові слова: попередня обробка даних, машинне навчання, preprocessing, *Standard Deviation Method, Local Outlier Factor , Random Forest. KNN, Bagging.*

Вступ

Попередня обробка даних (*preprocessing*) – це набір процедур які направлені на підвищення якості самих даних та якості інтелектуального аналізу даних. Вона є основою достовірного аналізу даних, які, зазвичай, мають низьку якість, визначається як нетривіальне завдання в аналізі даних і може становити до 80% загального обсягу зусиль з їх інтелектуального аналізу [1]. Якщо обробка даних не буде виконана, то подальший аналіз в більшості випадків неможливий з-за того, що аналітичні алгоритми просто не можуть працювати і результати їх роботи будуть некоректними. Іншими словами, реалізується принцип *GIGO - garbage in, garbage out* (сміття на вході, сміття на виході).

В цілому, попередня обробка даних складається з п'яти основних завдань: очищення, скорочення, масштабування, перетворення та секціонування даних. Одним із важливих завдань очищення даних є виявлення аномалій.

Об’єктом дослідження є процес виявлення аномалій в даних.

Предметом дослідження є методи виявлення аномалій в вихідних даних .

Постановка проблеми та огляд наукових публікацій. Виявлення аномалій (викидів) – це процес розпізнання під час інтелектуального аналізу даних рідкісних даних, подій або спостережень, які викликають підозри, зважаючи на істотні відмінності від більшої частини даних [1]. Виявлення аномалій містить два напрямки: детектування викидів (*Outlier Detection*) і детектування «новизни» (*Novelty Detection*). Новизна, як правило, з'являється в результаті принципово нової поведінки об'єкта (*sample*) або екземпляру (*example*), наприклад, після проникнення вірусів в комп’ютерну систему. Пошук аномалій може бути як кінцевою метою аналізу та побудови моделей, так і проміжним етапом підготовки. На етапі попередньої обробки даних вирішується завдання детектування викидів та шумів. Шум (*noise*) – це викид «в слабкому сенсі», який розмиває кордони класу (кластера) та заважає чітко розгледіти картину.

У класичній постановці задача детектування аномалій сформулюється так. Нехай дані мають ознакове подання, тобто кожен об'єкт x заданий у вигляді деякого вектору з $\boldsymbol{R}^{\boldsymbol{d}}$. Необхідно у заданій множині X для кожного елемента видати 0, якщо цей об'єкт відноситься до класу нормальних даних, і 1 якщо цей об'єкт є аномальним. Таке завдання відноситься до класу завдань навчання без вчителя, оскільки правильних відповідей на частини вхідних даних не надається. Методи навчання без вчителя [2, 3] широко застосовуються для виявлення аномалій у випадках, коли аномалії рідкісні або аномальна вибірка нерепрезентативна, тобто не покриває всіх можливих випадків аномальної поведінки. Вона показують гарну якість у випадках, коли класи не перетинаються або перетин незначний.

В аналогічному завданні навчання з учителем на деякій частині *Xtrain* дані є розміченими, тобто для кожного об'єкта $x \in Xtrain$ відомі мітки $y(x) \in \{0, 1\}$ – чи є об'єкт аномалією. Завдання визначення міток для нових даних *Xtest,* формально, є завданням бінарної класифікації, отже, може вирішуватися за допомогою будь-яких алгоритмів машинного навчання з учителем. Такий підхід є якісним за умови відносно частих аномалій [4-6], бінарна класифікація нестійка до малих чи нерепрезентативним вибірках аномалій.

Однак, можливий і «проміжний варіант», а саме розпізнавання частково з учителем коли всі мітки $y(x)$, $x \in Xtrain$ дорівнюють 0, тобто задані приклади лише нормальних («перевірених», «хороших») даних. Навчившись на одному класі, система може визначати приналежність нових даних до нього, таким чином, визначаючи протилежний клас [7]. Практично всі алгоритми детектування аномалій зводяться до побудови деякої функції *anomaly_score* (x), яка є відхилення від норми, оцінкою ступеня ймовірності

того, що екземпляр є аномальним. Після цього поділ на клас аномалій і клас нормальних даних проводиться бінаризацією за деяким порогом, вибір якого є особливим етапом вирішення задачі. У відсутності міток або апріорної інформації, єдиною наявною інформацією є одновимірний розподіл значень *anomaly_score* на наявних даних, чого для обґрунтованого вибору недостатньо. Найчастіше, відома приблизна частка аномалій в даних; в таких випадках поріг вибирається відповідно квантилю.

Іншим важливим завданням контролю якості даних є аналіз аномалій, тобто визначення джерел аномалій [8]. За способом реалізації виділяють наступні технології виявлення аномалій: класифікація, кластеризація, статистичний аналіз, спектральні методи, гібридні методи. Вищенаведені технології використовують алгоритми на онові ймовірнісного, лінійного та метричного підходів.

До поширених методів розпізнавання можна віднести наступні: метод опорних векторів (*Support Vector Machine, SVM*) *[9],* ізоляційний ліс (*Isolation Forest, IF)[8]*, метод *k* найближчих сусідів (*k-Nearest Neighbors Detector, KNN*) *[10];* метод *k*-середніх (*Average KNN, AKNN*), кластерний метод (*Cluster-based Local Outlier Factor, CLOF*) [11, 12], *DBSCAN* [13, 14] та ін. Ефективність різних методів залежить від даних та параметрів і має слабкі систематичні переваги один перед іншими, якщо порівнювати за багатьма наборами даних та параметрів [15,16].Окрім цього методи виявлення аномалій чутливі до параметрів налаштування моделей.

Дослідження методів виявлення аномалій

Дослідження було виконано з використанням хмарного сервісу *GOOGLE COLAB* на основі *Jupyter Notebook*.

У якості вихідних даних використано програмно згенеровані файли формату **.csv*. В рамках дослідження використано розмічені дані, які визначають приналежність до одного із двох класів.

Розмітка даних використовується лише для оцінки якості класифікації. Подальші дослідження базуються на використанні методів ідентифікації аномалій без вчителя.

Розглянуто такі методи: метод стандартного відхилення (*Standard Deviation Method*), метод локального рівня викидів (*Local Outlier Factor*), метод Ізолюючого лісу (*Isolation Forest*).

Досліджено залежність абсолютної кількості аномалій від порогу прийняття рішень для кожного із методів. Прийняття рішення щодо віднесення об'єкту до аномалій відбувалося за двома алгоритмами: *Entire* та E*ach*. Відповідно до алгоритму *Entire*, аномальність об'єкту оцінювалась для обох класів одночасно. Відповідно до алгоритму E*ach*, аномальність об'єкту оцінювалась окремо по кожному із класів.

Оцінку якості попередньої обробки даних виконано з використанням класифікаторів на основі методу *KNN* та беггінгу (*Bagging*) зі стандартним налаштуванням на розмічених даних. При цьому точність класифікації до попередньої обробки є наступною: для методу *KNN* – **82,3%,** для беггінгу (*Bagging*) – **83,3.**

Основою методу стандартного відхилення є розрахунок середніх значень ряду

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i,$$

та середньоквадратичного відхилення

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2},$$

де N– кількість елементів, x_i – i-й елемент екземпляру.

Поріг визначення аномалій, зазвичай, визначається як

$$anomaly\_score = x_i \pm 2\sigma.$$

Результати дослідження методу стандартного відхилення (*Standard Deviation Method*) для виявлення аномалій в вихідних даних наведено в табл. 1.

Таблиця 1 – **Результати виявлення аномалій в даних методом стандартного відхилення**

Стандартне відхилення	Кількість аномалій, алгоритм *Entire*	Кількість аномалій, алгоритм E*ach*	Точність класифікації, після preprocessing, метод *KNN*		Точність класифікації, після preprocessing, метод *Bagging*	
			алгоритм *Entire*	алгоритм E*ach*	алгоритм *Entire*	алгоритм E*ach*
3,5	1	1	93,33	93,33	90,3	91,21
3,25	1	4	93,33	93,33	90,3	91,82
3	2	5	93,33	93,33	89,7	90,61
2,75	4	9	93,33	93,33	**92,12**	90
2,5	10	16	93,33	93,33	89,7	90,61
2,25	**18**	**29**	**93,33**	**93,64**	90,61	92,12
2	36	51	93,33	93,64	90	91,82
1,75	72	91	93,64	93,33	91,21	**93,33**
1,5	153	152	93,03	92,42	91,82	91,82
1,25	263	253	92,42	93,33	90,3	92,12
1	390	371	93,64	93,03	82,42	92,12
0,75	515	483	93,03	93,03	53,64	92,12
0,5	601	584	92,12	93,03	80,91	92,12

Відповідно до результатів, поріг аномалій може бути визначено наступним чином:Для методу *KNN* як

$$anomaly\_score = x_i \pm 2,25\sigma .$$

Для методу беггінгу та алгоритму *Entire*, як

$$anomaly\_score = x_i \pm 2,75\sigma .$$

Для методу беггінгу та алгоритму E*ach*, як

$$anomaly\_score = x_i \pm 1,75\sigma .$$

Попередня обробка даних з метою вилучення аномалій в даних методом стандартного відхилення надала можливість підвищити точність класифікації для методу *KNN* – до 11,3%, для беггінгу – до 10%.

При цьому, ідентифікація аномалій обома алгоритмами суттєво не вплинула на якість ідентифікації для методу *KNN*, та надала можливість підвищити точність класифікації до 1,2% для методу беггінгу та алгоритму E*ach*.

Алгоритм *Local Outlier Factor* (*LOF*) – це метод неконтрольованого виявлення аномалій. *LOF* використовує ідею порівняння локальної щільності точки із середньою локальною щільністю її *k*-найближчих сусідів. Він вважає, що аномалії – це об'єкти, які знаходяться в областях з низькою щільністю або низькою локальною щільністю, чим їх сусіди [11]. Алгоритм є якісним у ситуаціях, коли щільність даних не однакова у всьому наборі даних і показує хороші результати при виявленні локальних викидів, однак має високу обчислювальну складність.

Відповідно до результатів, поріг аномалій може бути визначено наступним чином.:

Для методу *KNN* як

$$anomaly\_score = x_i \pm 2,25\sigma .$$

Для методу беггінгу та алгоритму *Entire*, як

$$anomaly\_score = x_i \pm 2,75\sigma .$$

Для методу беггінгу та алгоритму E*ach*, як

$$anomaly\_score = x_i \pm 1,75\sigma .$$

Попередня обробка даних з метою вилучення аномалій в даних методом стандартного відхилення надала можливість підвищити точність класифікації для методу *KNN* – до 11,3%, для беггінгу – до 10%.

При цьому, ідентифікація аномалій обома алгоритмами суттєво не вплинула на якість ідентифікації для методу *KNN*, та надала можливість підвищити точність класифікації до 1,2% для методу беггінгу та алгоритму E*ach*.

Алгоритм *Local Outlier Factor* (*LOF*) – це метод неконтрольованого виявлення аномалій. *LOF* використовує ідею порівняння локальної щільності точки із середньою локальною щільністю її *k*-найближчих сусідів. Він вважає, що аномалії – це об'єкти, які знаходяться в областях з низькою щільністю або низькою локальною щільністю, чим їх сусіди [11]. Алгоритм є якісним у ситуаціях, коли щільність даних не однакова у всьому наборі даних і показує хороші результати при виявленні локальних викидів, однак має високу обчислювальну складність.

LOF заснований на методі найближчих сусідів (*kNN*) і описаний у роботі [12]. Досяжність точки *x* щодо точки *y* визначається так

$$R_k(x, y) = \max(p(x.y), D_k(y)), \qquad (1)$$

де *Dk* (*y*) – відстань від точки *y* до *k* найближчого сусіда.

Тоді оцінку *LOF* визначається як

$$LOF_k(x, y) = \underset{y \in N_k(x)}{mean} \frac{AR_k(x)}{AR_k(y)},$$

де $ARk(x)$ – середня досяжність точки даних *x* щодо *k* своїх найближчих сусідів, $ARk(y)$ –множина *k* найближчих сусідів точки даних *x*. Якщо *LOF* приймає значення, близькі до 1, тоді об'єкт x буде вважатися нормальним, однак якщо *LOF* >> 1, у такому разі досліджуваний об'єкт визнається аномалією. *LOF* визначить аномалію з урахуванням щільності даної області.

Результати дослідження виявлення аномалій в вихідних даних методом *LOF* наведено в табл. 2.

Таблиця 2 – **Результати виявлення аномалій в даних методом *Local Outlier Factor***

Кількість класифікаторів (кластерів)	Кількість аномалій, алгоритм *Entire*	Кількість аномалій, алгоритм E*ach*	Точність класифікації, після препоцессінгу, метод *KNN*		Точність класифікації, після препоцессінгу,, метод *Bagging*	
			алгоритм *Entire*	алгоритм E*ach*	алгоритм *Entire*	алгоритм E*ach*
1	148	153	93,03	92,73	89,7	91,21
5	17	31	**93,64**	**93,64**	89,7	91,52
10	11	27	93,33	93,64	89,39	**90,91**
15	12	29	93,33	93,64	**90,91**	90,3
20	15	29	93,33	93,33	91,21	90,61
25	18	29	93,33	93,33	91,21	91,82
30	18	29	93,33	93,33	91,21	91,82
35	20	30	93,33	93,33	90,61	91,21
40	20	29	93,33	93,33	90,61	90
45	20	30	93,33	93,64	90,61	91,82
50	19	30	93,33	93,64	91,52	89,39
100	17	24	93,33	93,64	90,3	92,42

Отримані результати дозволили підібрати оптимальну кількість сусідів на етапі попередньої обробки даних

Так для методу *KNN* оптимальним є п'ять сусідів для обох алгоритмів. Для методу беггінгу та алгоритму *Entire* оптимальним є 35 сусідів, алгоритму E*ach* – 10 сусідів.

Вилучення аномалій на етапі попередньої обробки даних методом *LOF* надало можливість підвищити точність класифікації для методу *KNN* – до 11,3%, для беггінгу – до 7,6%. При цьому, ідентифікація аномалій обома алгоритмами суттєво не вплинула на якість ідентифікації для методу *KNN*, та беггінгу.

Метод ізоляційного лісу (*Isolation Forest*) [9, 17] є технікою машинного навчання без вчителя, заснованої на принципі Монте-Карло.

Метод поєднує дерева прийняття рішень [14], використовуючи евристики, пов'язані з навчанням дерев прийняття рішень.

Робота алгоритму полягає у побудові випадкового бінарного дерева, в якому корінь – весь простір ознак, а вузол представлений випадковою ознакою та порогом розбиття.

Поріг розбиття вибирається з рівномірного розподілу на відрізку від мінімального до максимального значення обраної ознаки. При тотожному збігу об'єктів у вузлі робота алгоритму завершується.

Глибина листя в дереві буде відповідати значенню алгоритму *anomaly_score*.

Оцінка аномалії об'єкту x визначається як [9]:

$$anomaly\_score(x,n) = 2^{-\frac{E(h(x))}{c(n)}},$$

де $c(n)$ визначається таким чином:

$$c(n) = 2H(n-1) - (2(n-1)/n),$$

$H(i)$ визначається як:

$$H(i) = \ln(i) + \gamma .$$

$h(x)$ – довжина шляху до спостереження x, $c(n)$ – середня довжина шляху безуспішного пошуку у двійковому дереві пошуку, а n – кількість точок даних.

Якщо s набуває значення, близькі до 1, тоді об'єкт x з великою ймовірністю буде аномалією, однак якщо s приймає значення менше 0.5, то досліджуваний об'єкт визнається нормальним. *Isolation Forest* чутливий лише до глобальних викидів та слабко справляється з локальними викидами.

Результати дослідження наведено в табл. 3.

Таблиця 3 – **Результати виявлення аномалій в даних методом *Isolation Forest***

Кількість класифікаторів (дерев рішень) ансамблю	Кількість аномалій, алгоритм *Entire*	Кількість аномалій, алгоритм E*ach*	Точність класифікації, після препроцессінгу, метод *KNN*		Точність класифікації, після препроцессінгу,, метод *Bagging*	
			Алгоритм *Entire*	алгоритм E*ach*	Алгоритм *Entire*	алгоритм E*ach*
1	313	173	93,33	92,42	88,79	90,61
5	256	139	**93,64**	92,12	88,79	93,03
10	228	135	93,33	92,12	89,7	92,12
20	177	139	93,33	92,12	90	93,33
50	152	123	93,64	92,73	90	93,03
100	130	115	93,64	92,42	92,12	**93,64**
150	135	112	93,64	**92,73**	91,52	92,42
200	135	109	93,64	92,73	91,52	92,73
300	135	109	93,33	92,73	**92,12**	92,12
500	142	115	92,42	93,03	90,61	92,73

Отримані результати дозволили підібрати оптимальну кількість класифікаторів ансамблю на основі алгоритму *Isolation Forest*, які забезпечують найбільшу точність класифікаторів. Отримано, що на етапі попередньої обробки даних на основі алгоритму *Isolation Forest* для методу *KNN* оптимальним є використання 50 класифікаторів для обох алгоритмів, а для беггінгу оптимальним є використання 100 дерев рішень для обох алгоритмів. Попередня обробка даних методом *Isolation Forest* надала можливість підвищити точність класифікації для методу *KNN* – до 11,4%, для беггінгу – до 11,3%.

Висновки

У роботі розглянуто методи попередньої обробки даних.

Досліджено методи:

стандартного відхилення (*Standard Deviation Method*),

локального рівня викидів (*Local Outlier Factor*),

ізолюючого лісу (*Isolation Forest*) для виявлення аномалій в даних.

Для кожного із методів отримано залежність абсолютної кількості аномалій від порогу прийняття рішень. Крім того, оцінка прийняття рішення щодо віднесення об'єкту до аномалій відбувалося за двома алгоритмами.

Відповідно до першого алгоритму, за наявності нерозмічених даних, аномальність об'єкту оцінювалась для обох класів одночасно. Відповідно до другого алгоритму, за наявності розмічених даних аномальність об'єкту оцінювалась окремо за кожним із

класів. Отримано, що попередня обробка даних з метою вилучення аномалій в даних методом стандартного відхилення надала можливість підвищити точність класифікації для методу *KNN* – до 11,3%, для беггінгу – до 10%. Вилучення аномалій на етапі попередньої обробки даних методом *LOF* дозволило підвищити точність класифікації для методу *KNN* – до 11,3%, для беггінгу – до 7,6%.

Більш якісним виявився алгоритм попередньої обробки даних *Isolation Forest*. Тестування показало збільшення точності класифікації для методу *KNN* – до 11,4%, для беггінгу – до 11,3%. При цьому, для методу *KNN* оптимальним є використання 50 дерев рішень для обох алгоритмів, а для беггінгу оптимальним є використання 100 дерев рішень для обох алгоритмів.

Досліджені методи реалізовані програмно з використанням хмарного сервісу *GOOGLE COLAB* на основі *Jupyter Notebook*

Проведені експерименти підтвердили працездатність методу *Isolation Forest*., що надає можливість рекомендувати його для практичного використання на етапі попередньої обробки даних з метою підвищення їх точності.

СПИСОК ЛІТЕРАТУРИ

1. Cui Z. G., Cao Y. Wu, Liu H.N, Qiu, Z. F., Chen, C. W. Research on preprocessing technology of building energy consumption monitoring data based on machine learning algorithm. *Build. Sci*. 2018, Vol. 34 (2), C. 94–99.
2. Bernhard Schölkopf, Robert C Williamson, Alex J Smola et al. Support vector method for novelty detection *Advances in Neural Information Processing Systems*, Denver, United States, 2000, P. 582–588.
3. Zhou Chong, Paffenroth Randy C. Anomaly detection with robust deep autoencoders, *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,Halifax*, Canada, 2017, P. 665–674.
4. Adrian Alan Pol, Gianluca Cerminara, Cécile Germain et al. Detector monitoring with artificial neural networks at the CMS experiment at the CERN Large Hadron Collider. *Computing and Software for Big Science.* 2019,*Vol*. 3 (1), *P*. 3-8.
5. Stankevicius Mantas, Marcinkevicius Virginijus, Rapsevicius Valdas. Comparison of Supervised Machine Learning Techniques for *CERN CMS Of*49 *fline Data Certification*. *Doctoral Consortium/Forum@ DB&IS,* 2018, *P*. 170–176.,
6. Adrian Alan Pol, Virginia Azzolini, Gianluca Cerminara et al. Anomaly detection using Deep Autoencoders for the assessment of the quality of the data acquired by the CMS experiment, *EPJ Web of Conferences*, *EDP Sciences,* 2019, *Vol*. 214, *P*. 5.
7. Maxim Borisyak, Artem Ryzhikov, Andrey Ustyuzhanin et al. (1 + epsilon)-class Classification: an Anomaly Detection Method for Highly Imbalanced or Incomplete Data Sets, *Journal of Machine Learning Research*, 2020, *Vol*. 21(72), *P*. 1–22.
8. Гавриленко С.Ю., Шевердін І. В. Розробка методу ідентифікації стану комп'ютерної системи на основі алгоритму «*Isolation Forest*», *Радіоелектроніка, інформатика, управління*, 2021, №.1(56), P. 105-116.
9. Support vector method for novelty detection / Bernhard Schölkopf,Robert C Williamson, Alex J Smola et al., *Advances in Neural InformationProcessing System*, *Denver*, *United States*, 2000, *P*. 582–588.
10. Большаков А.С., Губанкова Е.В. Обнаружение аномалий в компьютерных сетях с использованием методов машинного обучения, *Телекоммуникационные устройства и системы*, 2020. Т. 10. № 1. С. 37-42.
11. *Breunig M. M. et al. LOF*: *identifying density-based local outliers*, *ACM sigmodrecord – ACM*, 2000, *Vol*. 29 (2), *P*. 93-104.
12. Breunig, M. M ., Kriegel, H.P., Ng, R.T ., Сандер, Дж.. *LOF*: определение локальных выбросов на основе плотности,. *Материалы Международной конференции ACM SIGMOD* 2000 г. по управлению данными. *SIGMOD*, 2000, С. 93–104.
13. Zhou, H., Wang, P., & Li, H. Research on adaptive parameter determination in DBSCAN algorithm, *Journal of Xi'an University of Technology*, 2014, 28(3), *P*.289-292.
14. Karami A., Johansson R. Choosing DBSCAN parameters automatically using differential evolution, *International Journal of Computer Applications*, 2014, *Vol*. 91(7), *P*.1-11.
15. Arthur Zimek, Erich Schubert. Outlier Detection. Encyclopedia of Database Systems, *Springe,r* New York, 2017. P.96-106.
16. Dai, Zhifeng and Xiaomin Chang. Predicting Stock Return with Economic Constraint: Can Interquartile Range Truncate the Outliers, *Mathematical Problems in Engineering*, 2021, P. 1-12.
17. Liu, Fei Tony, Ting, Kai Ming and Zhou, Zhi-Hua. Isolation forest, *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, December* 2008, *P*. 413–422.

Надійшла (received) 16.12.2021
Прийнято до друку (accepted for publication) 26.01.2022

Investigation of methods for detecting anomalies at the stage of data pre-processing

S. Gavrylenko, V. Zozulia

Abstract. **The subject** of the research is the methods and means of detecting anomalies in data. **The purpose** of the article is to improve the quality of data classification by detecting anomalies at the pre-processing stage. **Task**: to investigate methods for detecting anomalies at the stage of data preprocessing, to determine the decision threshold for each of the methods and to evaluate the quality of classification before and after preprocessing. **Methods used are**: artificial intelligence methods, machine learning, ensemble methods. The following **results were obtained**: anomaly detection methods were studied: Standard Deviation Method, Local Outlier Factor method, Isolation Forest method. The dependence of the number of anomalies on the decision threshold for each of the methods is obtained. The evaluation of the quality of data preprocessing was performed using classifiers based on the KNN and Bagging methods. The studied methods are implemented programmatically using the GOOGLE COLAB cloud service based on Jupyter Notebook. **Conclusions**. The scientific novelty of the results obtained lies in the study of anomaly detection methods at the stage of data preprocessing, the choice of a preprocessing meta-algorithm and the determination of its optimal settings.

Keywords: data preprocessing, machine learning, preprocessing, Standard Deviation Method, Local Outlier Factor, Random Forest, KNN, Bagging.