МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет "Львівська політехніка"

Інтелектуальний аналіз даних за допомогою програмного пакета WEKA та MS Excel.

Класифікація методом k-найближчих сусідів.

МЕТОДИЧНІ ВКАЗІВКИ
до лабораторної роботи № 10

з курсу **"Системи інтелектуального аналізу та візуалізації даних"**

для студентів за освітньою програмою Комп'ютерні науки (Проектування і програмування інтелектуальних систем та пристроїв)

Затверджено на засіданні кафедри
"Системи автоматизованого проектування"

Протокол N 1 від 28.08.2023р.

ЛЬВІВ 2023

1. МЕТА РОБОТИ

Мета роботи – навчитися класифікувати дані за допомогою методу k-найближчих сусідів. Вивчити теоретичні основи методу та для реалізації аналізу даних навчитися використовувати програму WEKA та Excel.

2. КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ

2.1. Метод k-найближчих сусідів.

Цей метод дозволяє відповісти на таке запитання : «Якщо людина купує автомобіль BMW M5, які ще товари вона може придбати?» Як свідчать статистичні дані, якщо людина купує BMW M5, то досить часто вона бере відповідну за кольором барсетку (подібні дослідження називаються «аналіз ринкового кошика»). Використовуючи подібні дані, дилерський центр може розмістити рекламу відповідних товарів або подати на друк оголошення про знижки на барсетки і сумки відповідного кольору або їх безкоштовну пропозицію всім покупцям M5 в якості одного із способів підвищення кількості продажів.

Алгоритм k-найближчих сусідів (K-Nearest Neighbours (KNN)) — популярний метод машинного навчання, який використовується для завдань класифікації та регресії. Він спирається на ідею, що подібні точки даних, як правило, мають подібні атрибути або значення.

Під час фази навчання алгоритм k-найближчих сусідів зберігає весь набір навчальних даних як еталон. Роблячи прогнози, він обчислює відстань між точкою вхідних даних і всіма навчальними прикладами, використовуючи вибрану метрику відстані, наприклад евклідову відстань.

Далі алгоритм визначає K найближчих сусідів до точки вхідних даних на основі їх відстані. У разі класифікації алгоритм призначає найпоширенішу мітку класу серед K сусідів як прогнозовану мітку для точки вхідних даних.

Алгоритм k-найближчих сусідів є простим і легким для розуміння, що робить його популярним вибором у різних областях. Однак на його продуктивність може вплинути вибір K і метрики відстані, тому для досягнення оптимальних результатів необхідне ретельне налаштування параметрів.

Алгоритм k-найближчих сусідів можна використовувати як для задач класифікації, так і для прогнозування регресії. Однак він більш широко використовується в проблемах класифікації в промисловості. Щоб оцінити будь-яку техніку, ми зазвичай розглядаємо 3 важливі аспекти:

1. Легкість інтерпретації результату
2. Час розрахунку
3. Передбачувана сила

Розглянемо кілька прикладів, щоб розмістити k-найближчих сусідів у шкалі:

	Logistic Regression	CART	Random Forest	KNN
1. Ease to interpret output	2	3	1	3
2. Calculation time	3	2	1	3
3. Predictive Power	2	2	3	2

Класифікатор k-найближчих сусідів відповідає всім параметрам. Він звичайно використовується через легкість інтерпретації та короткий час обчислення.

Розглянемо простий випадок, щоб зрозуміти цей алгоритм. Нижче показано розташування червоних кіл (RC) і зелених квадратів (GS):

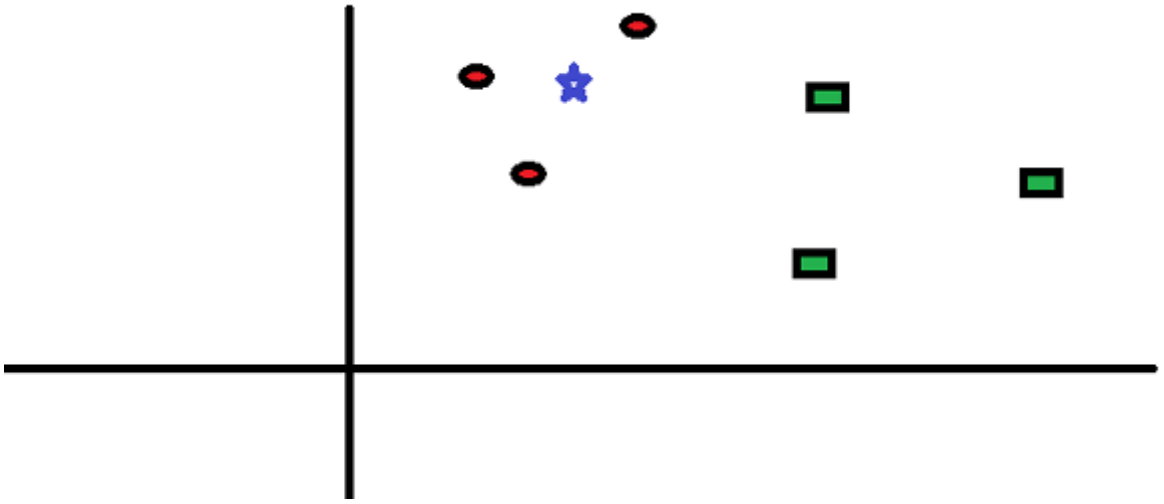

Рис.1. Розташування червоних кіл (RC) і зелених квадратів (GS)

Ви маєте намір дізнатися клас блакитної зірки (BS). BS може бути або RC, або GS і нічим іншим. «K» в алгоритмі k-найближчих сусідів — це найближчий сусід, у якого ми хочемо взяти голос. Скажімо, K = 3. Отже, тепер ми створимо коло з центром BS настільки великим, щоб охоплювати лише три точки даних на площині:

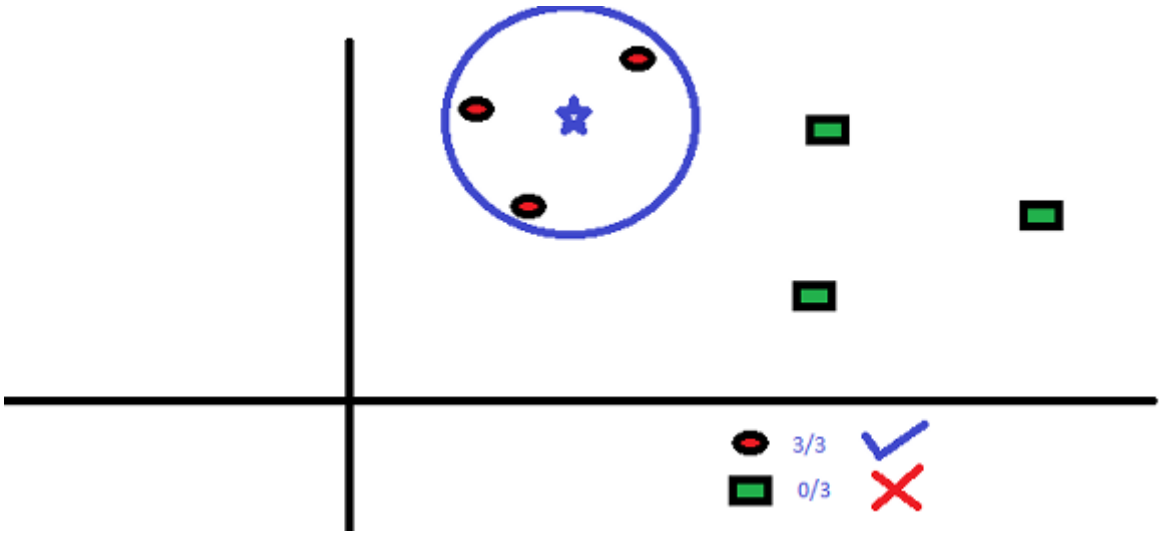

Рис.2. Класифікація нового об’єкту при k=3

Три найближчі точки до BS – це RC. Отже, з достатньою впевненістю можна сказати, що БС має належати до класу RC. Тут вибір став очевидним, оскільки всі три голоси від найближчого сусіда дісталися РК. Вибір параметра K є дуже важливим у цьому алгоритмі. Далі ми розберемо фактори, які слід враховувати, щоб зробити висновок про найкращий K.

Як вибрати фактор K? По-перше, давайте спробуємо зрозуміти вплив k-найближчих сусідів на алгоритм. Якщо ми розглянемо останній приклад, зберігаючи постійними всі 6 тренувальних спостережень, задане значення K дозволяє нам встановити межі для кожного класу. Ці межі рішень фактично відокремлюють, наприклад, RC від GS. Подібним чином розглянемо вплив значення «K» на межі цих класів. Далі показано чіткі межі, які розділяють два класи, кожен з яких відповідає різним значенням K.

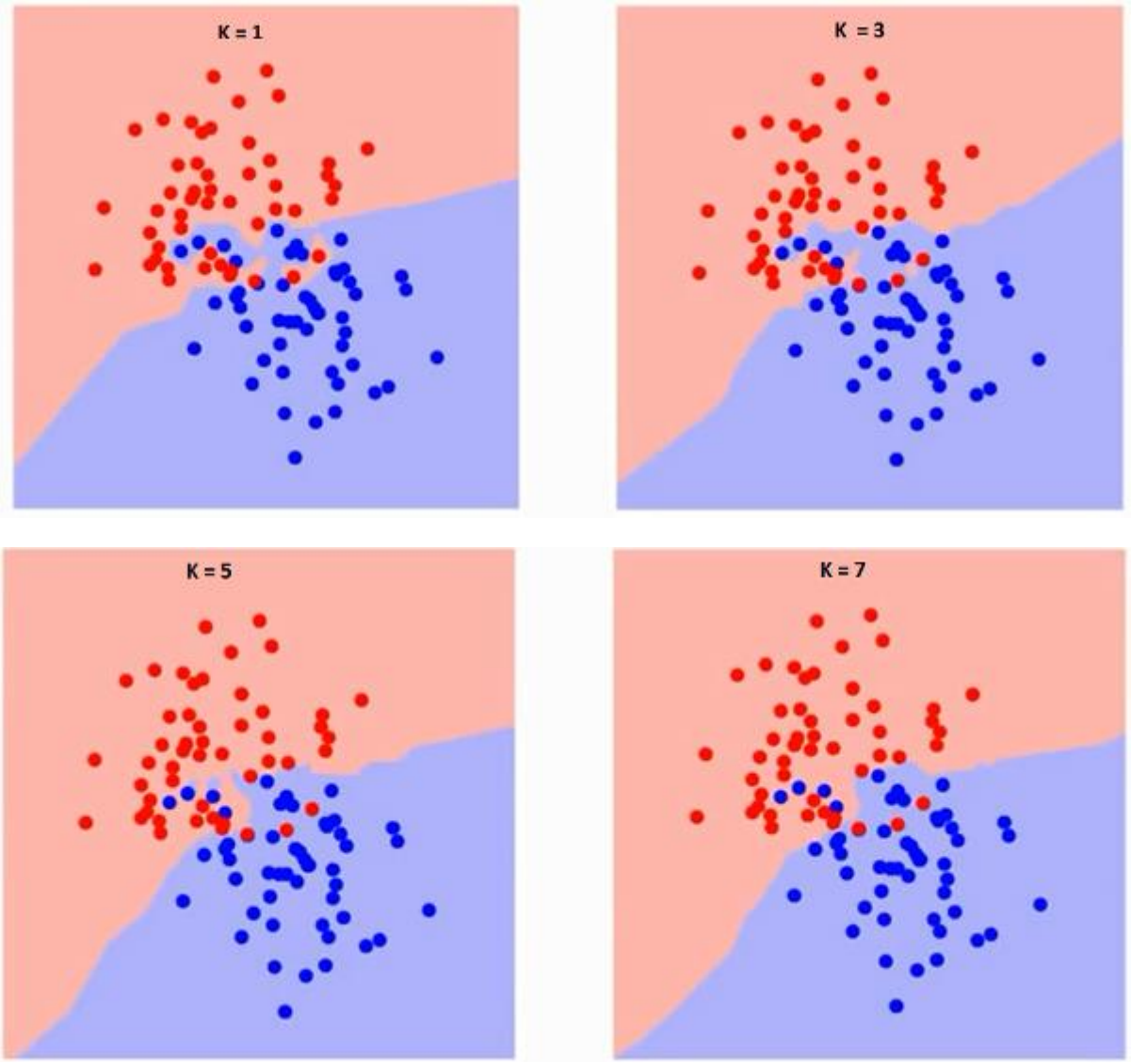

Рис.3. Результати класифікація при різних k

Якщо ви уважно спостерігаєте, ви можете побачити, що межа стає більш гладкою зі збільшенням значення K. Зі збільшенням K до нескінченності вона нарешті стає повністю синьою або повністю червоною залежно від загальної більшості. Частота помилок навчання та частота помилок перевірки є двома параметрами, які нам потрібні для доступу до різних значень K. Нижче наведена крива частоти помилок навчання зі змінним значенням K:

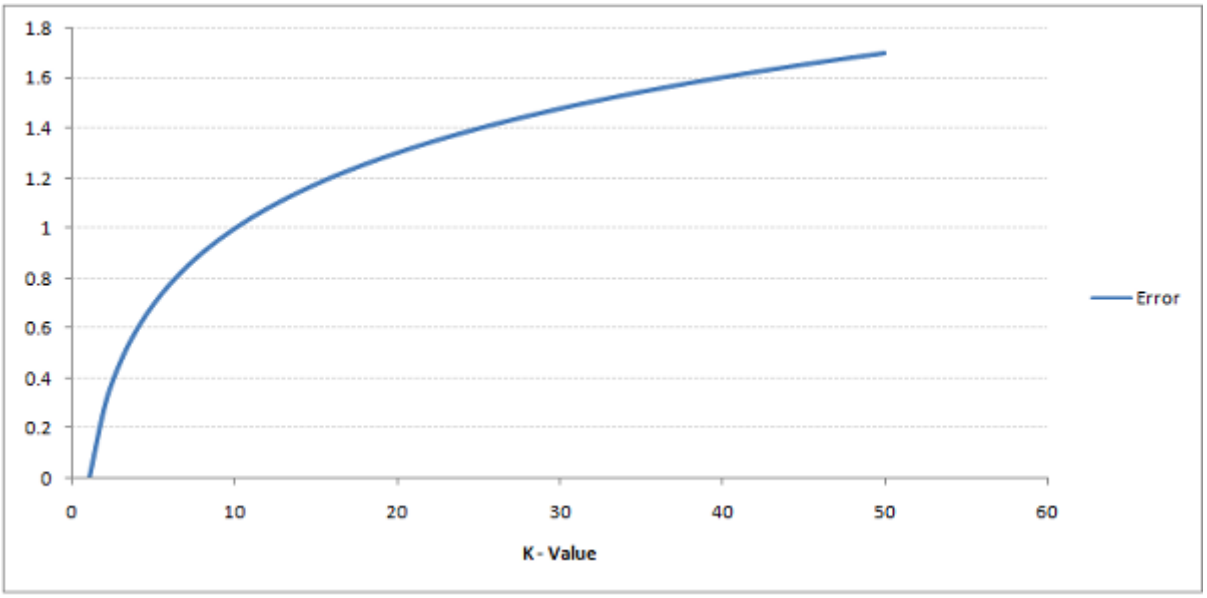

Рис.4. Крива частоти помилок навчання зі змінним значенням K

Як бачите, частота помилок при K=1 завжди дорівнює нулю для навчальної вибірки. Це пояснюється тим, що найближчою точкою до будь-якої точки навчальних даних є вона сама. Тому прогноз завжди точний з K=1. Якби крива помилок перевірки була подібною, наш вибір K був би 1. Нижче наведена крива помилок перевірки зі змінним значенням K:

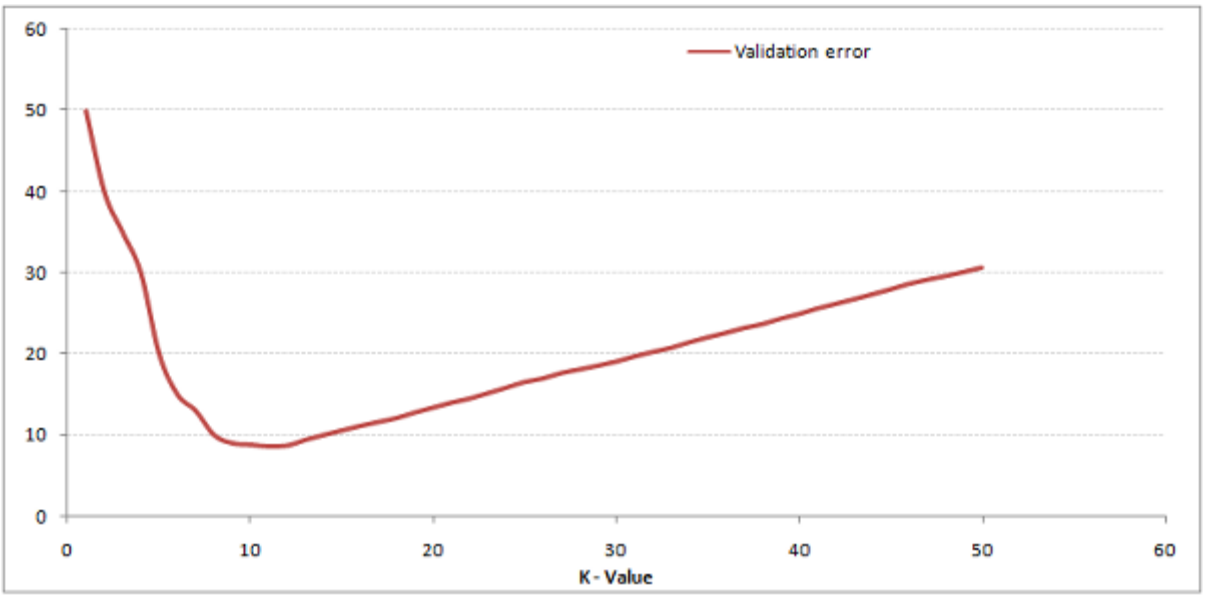

Рис.5. Крива помилок перевірки зі змінним значенням K

Це робить історію більш зрозумілою. При K=1 ми переобладнали межі. Таким чином, частота помилок спочатку зменшується і досягає мінімуму. Після точки мінімуму він збільшується зі збільшенням K. Щоб отримати оптимальне значення K, ви можете відокремити навчання та перевірку від початкового набору даних. Тепер побудуйте криву помилок перевірки, щоб отримати оптимальне значення K. Це значення K слід використовувати для всіх прогнозів.

Алгоритму k-найближчих сусідів:

Ми можемо реалізувати модель k-найближчих сусідів, виконавши наведені нижче дії.

1. Завантажте дані
2. Ініціалізуйте значення k

3. Щоб отримати прогнозований клас, повторіть від 1 до загальної кількості точок навчальних даних

- Обчислити відстань між тестовими даними та кожним рядком навчального набору даних. Тут ми будемо використовувати евклідову відстань як нашу метрику відстані, оскільки це найпопулярніший метод. Інші функції або метрики відстані, які можна використовувати, це Манхеттенська відстань, відстань Мінковського, Чебишева, косинус тощо. Якщо є категоричні змінні, можна використовувати відстань Хеммінга.
- Відсортуйте обчислені відстані в порядку зростання на основі значень відстані
- Отримати k перших рядків із відсортованого масиву
- Отримайте найчастіший клас із цих рядків
- Повернути передбачений клас

2.2. Реалізація методу найближчих сусідів у WEKA

Набір даних, який ми будемо аналізувати методом найближчих сусідів містить дані рекламної компанії вигаданого дилера BMW з продажу розширеної дворічної гарантії своїм постійним покупцям.

Наведемо тут короткий опис цього набору даних. Дилерський центр має дані про 3000 продажів розширеної гарантії. Цей набір має **такі атрибути:**

- **розподіл за доходами**
  0 = $ 0 - $ 30k,
  1 = $ 31k-$ 40k,
  2 = $ 41k-$ 60k,
  3 = $ 61k-$ 75k,
  4 = $ 76k-$ 100k,
  5 = $ 101k-$ 150k,
  6 = $ 151k-$ 500k,
  7 = $ 501k+
- **рік/місяць покупки першого автомобіля BMW,**
- **рік/місяць покупки останнього автомобіля BMW,**
- **чи скористався клієнт розширеною гарантією.**

Файл даних для аналізу методом найближчих сусідів за допомогою пакету WEKA

```
@attribute IncomeBracket {0,1,2,3,4,5,6,7}
@attribute FirstPurchase numeric
@attribute LastPurchase numeric
@attribute responded {1,0}

@data

4,200210,200601,0
5,200301,200601,1
...
```

Завантажимо файл **bmw-training.arff** в WEKA, виконавши в закладці **Preprocess** вже знайомі нам кроки. Вікно WEKA має виглядати так, як показано на рис. 6.

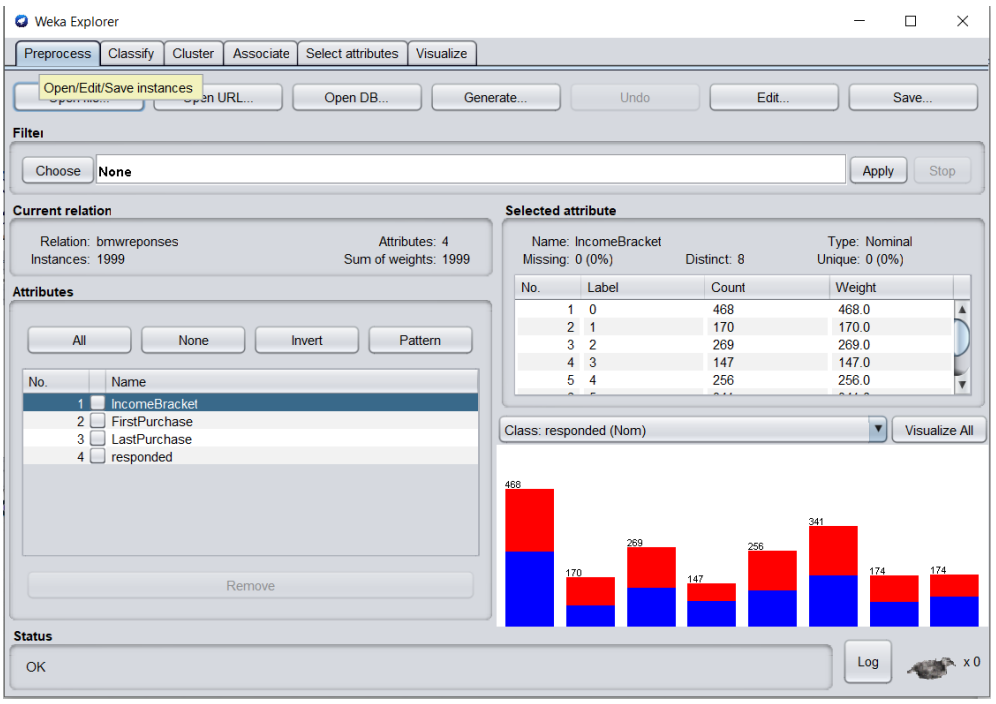

Рис. 6. Дані дилерського центру BMW для аналізу методом найближчих сусідів за допомогою WEKA

Точно так само, як ми виконали це для методів регресійного аналізу та класифікації в попередніх роботах, ми повинні відкрити закладку **Classify**. У панелі **Classify** потрібно вибрати опцію **lazy**, а потім **Ibk** (тут ***IB*** означає **Instance-Based** - навчання на прикладах, а ***k*** вказує на кількість сусідів, поведінку яких ми хочемо дослідити).

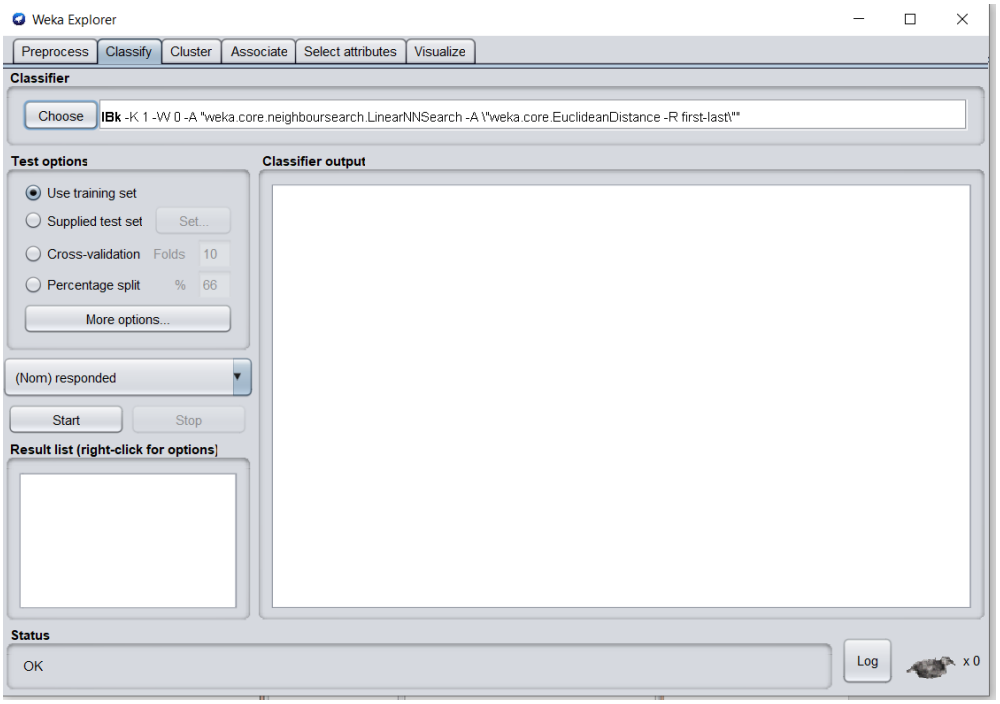

Рис. 7. Алгоритм методу найближчих сусідів для набору даних BMW

Тепер ми готові приступити до створення нашої моделі в WEKA. Переконайтеся, що ви вибрали опцію **Use training set**, щоб використовувати набір даних, який ми тільки що завантажили у WEKA. Натисніть кнопку **Start** і дозвольте WEKA виконати всі необхідні обчислення. На рис. 8 показано, як має виглядати вікно WEKA по заверешенні обчислень, далі приведена результуюча модель.

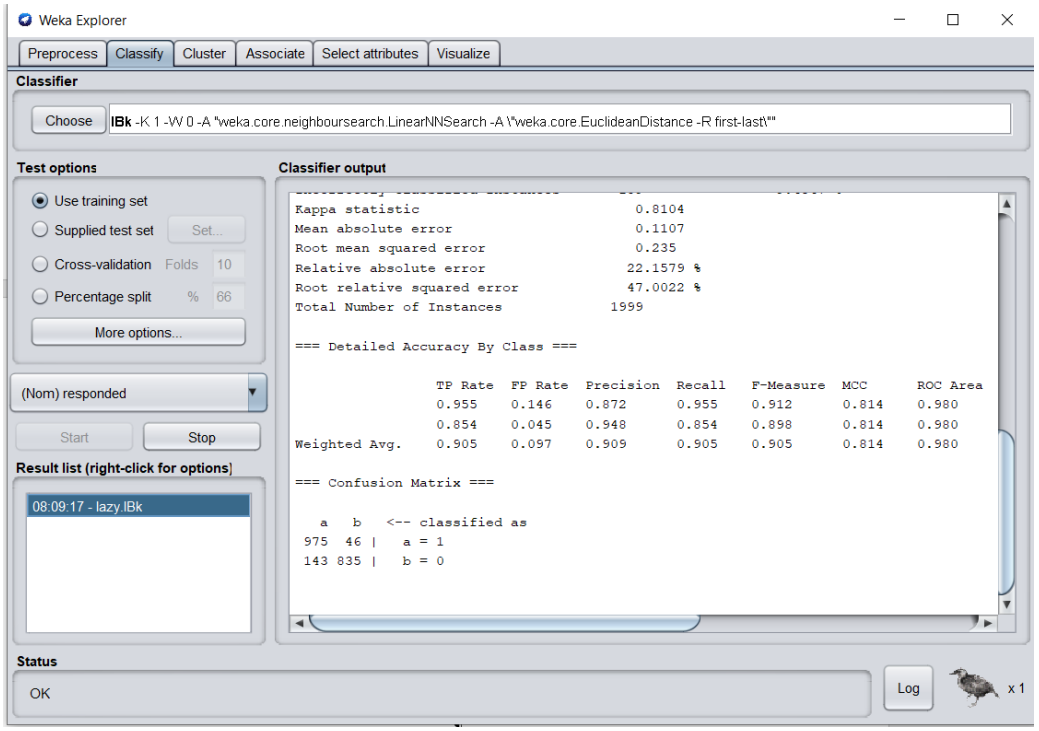

Рис. 8. Модель методу найближчих сусідів для набору даних BMW

Результат обчислень IBk

```
=== Run information ===

Scheme:       weka.classifiers.lazy.IBk -K 1 -W 0 -A
"weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""
Relation:     bmwreponses
Instances:    1999
Attributes:   4
              IncomeBracket
              FirstPurchase
              LastPurchase
              responded
Test mode:    evaluate on training data

=== Classifier model (full training set) ===

IB1 instance-based classifier
using 1 nearest neighbour(s) for classification


Time taken to build model: 0 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.38 seconds

=== Summary ===

Correctly Classified Instances        1810               90.5453 %
Incorrectly Classified Instances       189                9.4547 %
Kappa statistic                          0.8104
Mean absolute error                      0.1107
```

```
Root mean squared error                  0.235
Relative absolute error                 22.1579 %
Root relative squared error             47.0022 %
Total Number of Instances               1999

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0.955    0.146    0.872      0.955    0.912      0.814    0.980     0.977     1
               0.854    0.045    0.948      0.854    0.898      0.814    0.980     0.975     0
Weighted Avg.  0.905    0.097    0.909      0.905    0.905      0.814    0.980     0.976

=== Confusion Matrix ===

   a   b   <-- classified as
 975  46 |   a = 1
 143 835 |   b = 0
```

Як це співвідноситься з моделлю, яку ми отримали за допомогою методу класифікації? У моделі, що використовує метод найближчих сусідів, показник точності дорівнює 90% - зовсім непогано для початку, зважаючи на те, що точність попередньої моделі становила всього 59%. Практично 90% точності - це цілком прийнятний рівень. Давайте розглянемо результати роботи методу в термінах помилкових визначень, щоб на конкретному прикладі побачити, як саме WEKA може використовуватися для вирішення реальних проблем.

Результати використання моделі на нашому наборі даних показують, що у нас є 46 хибно-позитивних розпізнавань (1.53%) і 143 хибно-негативних розпізнавань (4.77%). У нашому випадку хибно-позитивне розпізнавання означає, що модель вважає, що даний покупець придбає розширену гарантію, хоча насправді він відмовився від покупки. Хибно-негативне розпізнавання, у свою чергу, означає, що згідно з результатами аналізу даний покупець відмовиться від розширеної гарантії, а насправді він її купив. Припустимо, що вартість кожної рекламної листівки, що розсилається дилером, становить $3, а купівля однієї розширеної гарантії приносить йому 400$ доходу. Таким чином, помилки помилкового розпізнавання в термінах витрат і доходів нашого дилера будуть виглядати наступним чином:

400$ - (1.53%*$3) - (4.74%*400) = **$381**.

Отже, помилкове розпізнавання помиляється на користь дилера.

Порівняємо цей показник з даними моделі класифікації:

400$ - (18.0%*$ 3) - (22.3%*$400) = **$310**.

Як ви бачите, використання точнішої моделі підвищує потенційний дохід дилера на 20%.

В якості самостійної вправи спробуйте змінити кількість найближчих сусідів в моделі (для цього розкрийте список параметрів, клацнувши правою кнопкою мишки на полі "IBk-K 1 ...."). Ви можете вибрати довільне значення для параметра "KNN" (До найближчих сусідів). Ви побачите, що точність моделі підвищується в міру додавання сусідів.

Зверніть увагу на певні недоліки моделі найближчих сусідів. Корисність цього методу цілком очевидна, коли мова йде про значні наборах даних, такі, наприклад, якими володіє *Rozetka*. Маючи дані про 20 мільйонів користувачів нескладно отримати достатньо точний результат - у базі потенційних покупців *Rozetka* напевно знайдеться людина, чиї уподобання схожі з вашими. Модель, заснована на такому значному обсязі даних, безумовно, буде відрізнятися високою точністю прогнозів. З іншого боку, модель стає практично марною, якщо у вас є лише кілька записів для порівняння. На початкових етапах розвитку он-лайн магазинів електронної комерції, їх власники могли

використовувати дані приблизно про 50 покупців. На такому невеликому наборі даних рекомендації, отримані за допомогою методу найближчих сусідів, не збігалися з дійсними покупками, так як вподобання вашого найближчого сусіда були дуже далекі від ваших уподобань.

Остання проблема, пов'язана з використанням методу найближчих сусідів полягає в тому, що цей метод є високозатратним з точки зору проведення обчислень. У випадку з компанією *Rozetka*, котра володіє даними про 20 мільйонів покупців, щоб визначити найближчих сусідів, конкретного покупця необхідно порівняти з кожним з решти 20 мільйонів. По-перше, якщо ваш бізнес налічує 20 мільйонів клієнтів, то подібні обчислення не викличуть у вас серйозних проблем, оскільки ви, цілком очевидно, володієте великими грошима. По-друге, подібні обчислення - ідеальне завдання для хмарних систем, оскільки у цьому випадку обчислювальні процеси будуть виконуватися паралельно на декількох десятках комп'ютерів, а після обчислення всіх відстаней, результати будуть порівнюватися між собою для визначення найближчих наборів даних (як, наприклад, це робить *Google MapReduce*). По-третє, на практиці таких масштабних обчислень не буде потрібно. Якщо необхідно визначити, чи куплю я одну книгу, то для цього зовсім не обов'язково порівнювати мене з усіма 20 мільйонами користувачів *Rozetka*, достатньо буде знайти найближчого сусіда серед покупців книжок. Подібний підхід дозволяє використовувати лише частину бази даних і скоротити обсяг обчислень.

Запам'ятайте: інтелектуальний аналіз даних не зводиться до простого механізму завантаження вхідних даних та отримання бажаного результату на виході. Необхідно провести досить ретельне дослідження даних для вибору найбільш відповідної моделі для аналізу. Крім того, зменшення обсягу вхідних даних дозволить скоротити час, необхідний для виконання всіх розрахунків. Далі, отриманий результат необхідно проаналізувати з точки зору його точності, тільки після цього ви можете схвалити застосування вашої аналітичної моделі в реальній практиці.

2.3. Параметри налаштування алгоритму найближчих сусідів

Розглянемо параметри налаштування алгоритмів, що використовуються в лабораторній роботі (табл.1).

Таблиця 1. Параметри налаштування класифікаторів

Метод	**Параметр**
IBk	*KNN – кількість сусідів.* *crossValidate* – чи буде використовуватися для вибору оптимальної кількості сусідів крос-перевірка hold-one-out. *distanceWeighting* – метод вибору вагових коефіцієнтів для відстаней. *meanSquared* – чи використовується середньоквадратична помилка, чи середня абсолютна помилка для крос-перевірки під час вирішення завдання регресії. *nearestNeighbourSearchAlgorithm* – алгоритм пошуку найближчих сусідів. *windowSize* – максимальна кількість примірників, дозволених в навчальному пулі. Додавання додаткових примірників понад цього значення призведе до видалення старих екземплярів. Значення 0 означає відсутність межі

2.4. Реалізація алгоритму K-найближчих сусідів в Excel

Змоделюємо наступний бізнес-сценарій. Компанія, яка видає кредитні картки, починає маркетингову кампанію. Компанія хоче передбачити, чи прийме новий клієнт Анастасія конкретну пропозицію кредитної картки, виходячи з того, як відповіли інші схожі клієнти.

Припустимо, що вже є вирішальні фактори, що визначають вірогідність прийняття рішення.

Відкрийте файл lab10.xlsx. Частина аркуша виглядає так, як показано на рис.9. У стовпці E зберігаються відповіді людей, з якими контактував банк. Цифра 1 означає прийняття пропозиції, а 0 означає відмову. Клітинки G1:J9 виділено, де нам потрібно знайти ймовірні відповіді Анастасії на основі різних значень K.

	A	B	C	D	E	F	G	H	I	J
1	Name	Age	Income	Cards	Response			Age	Income	Cards
2	N1	71	42518	6	0		Ananstasiya	22	140000	4
3	N2	37	93544	5	0		K	Anastasiya's likely response:		
4	N3	37	111629	4	0		1			
5	N4	49	92334	8	0		3			
6	N5	57	86135	10	1		5			
7	N6	50	86394	5	1		7			
8	N7	80	10128	9	0		9			
9	N8	43	64159	3	0		11			
10	N9	36	70324	1	1					

Рис. 9. Огляд даних для аналізу даних KNN

Перше завдання — визначити, як обчислити відстань між Анастасією та будь-якою іншою особою на основі трьох атрибутів: віку, доходу та кількості карток. Давайте використаємо евклідову відстань. Дотримуйтеся цих інструкцій, щоб завершити обчислення:

1. Введіть “Distance” в комірку F1. В комірці F2, введіть таку формулу:
=SQRT((B2-$H$2)^2+(C2-$I$2)^2+(D2-$J$2)^2)
2. Автозаповніть з F2 по F201. Частина нашого аркуша виглядає так, як рис.10.

	A	B	C	D	E	F	G	H	I	J
1	Name	Age	Income	Cards	Response	Distance		Age	Income	Cards
2	N1	71	42518	6	0	97482.01234	Ananstasiya	22	140000	4
3	N2	37	93544	5	0	46456.00243	K	Anastasiya's likely response:		
4	N3	37	111629	4	0	28371.00397	1			
5	N4	49	92334	8	0	47666.00781	3			
6	N5	57	86135	10	1	53865.01171	5			
7	N6	50	86394	5	1	53606.00732	7			
8	N7	80	10128	9	0	129872.013	9			
9	N8	43	64159	3	0	75841.00291	11			
10	N9	36	70324	1	1	69676.00147				

Рис. 10. *K-NN дані для передбачення відповіді Анастасії*

3. Використаємо функцію SMALL, щоб знайти найближчих сусідів на основі заданих значень K. Введіть «Small» у клітинку K3 та наведену нижче формулу у клітинку K4:

=SMALL(F$2:F$201,G4)

Функція SMALL повертає k-е найменше значення в масиві.

Оскільки G4 = 1, попередня формула знаходить перше найменше значення в F2:F201.

4. Автозаповніть з комірки K4 до комірки K9.

На прикладі k = 3 (що відповідає клітинці G5): оскільки G5 = 3, формула в клітинці K5 знаходить третє найменше значення в F2:F201.

Будь-яка точка даних, чия відстань до Анастасії менша або дорівнює значенню в клітинці K5, є найближчим сусідом Анастасії.

5. Введіть число 1 у клітинку L3 і число 0 у клітинку M3. У стовпці L записується номер відповіді 1, тоді як у стовпці M записується номер відповіді 0.

6. Введіть наступну формулу в клітинку L4:

=COUNTIFS($E$2:$E$201,L$3,$F$2:$F$201,"<="&$K4)

7. Автозаповніть клітинки L4 до M4, а потім разом автозаповніть до L9:M9.

Поточний результат показано на рис.11.

G	H	I	J	K	L	M
	Age	Income	Cards			
Ananstasiya	22	140000	4			
K	Anastasiya's likely response:			Small	1	0
1				6276.024	0	1
3				11815	1	2
5				20409.02	3	2
7				22209.01	4	3
9				22575.02	5	4
11				24622.01	5	6

Рис. 11. Підрахуйте голоси за 1 і 0

8. На підставі чисел 1 і 0, тобто кількості голосів за прийняття та заперечення відповідно, відповідь Анастасії можна визначити за такою формулою, яку слід ввести в клітинку H4 (зверніть увагу, що клітинки H4, I4 і J4 є об'єднані, тому H4 представляє всі три клітинки):

=IF(L4>M4,$L$3,$M$3)

9. Автозаповніть клітинки H4 до клітинки H9.

На рис.12. представлено кінцевий результат. Результат показує, що за іншого значення K прогноз відповіді Анастасії на основі більшості голосів її «сусідів» буде іншим. Майте на увазі, що K-NN — це ймовірнісна модель, тобто вона обчислює ймовірність того, як Анастасія відреагує на основі попередніх відповідей її сусідів. Наприклад, коли k дорівнює 5 (5 є звичайним вибором для K), K-NN передбачає, що вона, ймовірно, прийме пропозицію кредитної картки. Ймовірність її прийняття становить 3/5. Тим не менш, вона має 2/5 ймовірності відхилити пропозицію отримати нову кредитну картку. Зверніть увагу, коли k = 1, ймовірність її відхилення становить 100%. Однак це не гарантує, що вона не прийме пропозицію кредитної картки.

G	H	I	J	K	L	M
	Age	Income	Cards			
Ananstasiya	22	140000	4			
K	Anastasiya's likely response:			Small	1	0
1	0			6276.024	0	1
3	0			11815	1	2
5	1			20409.02	3	2
7	1			22209.01	4	3
9	1			22575.02	5	4
11	0			24622.01	5	6

Рис. 12. K-NN передбачає відповіді Анастасії на основі різних значень K

Неважко помітити, що значення K має значення в моделі K-NN. Про те, яке значення K є найкращим, можна сперечатися, і немає певного правила. Загальне розуміння полягає в тому, що K має бути непарним числом.

3. ЛАБОРАТОРНЕ ЗАВДАННЯ

1. Для індивідуального завдання вирішите задачу класифікації з використанням методу найближчих сусідів двома способами – спершу за допомогою Weka, потім – за допомогою Excel. Вирішіть, скільки сусідів потрібно для вашої моделі. Вам буде потрібно декілька експериментальних спроб для того, щоб визначити, яка кількість сусідів є оптимальною. Крім того, якщо ви використовуєте модель для отримання бінарного результату (0 або 1), то очевидно, що вам буде потрібна парна кількість сусідів.

2. Змінюючи параметри налаштування алгоритму, спробуйте досягти найкращої якості навчання класифікаторів.

3. Порівняйте результати отримані в обидвох системах.

4. У звіті надайте результати роботи кожного алгоритму, його налаштування, а також результати порівняння.

4. КОНТРОЛЬНІ ЗАПИТАННЯ

1. Поясніть роботу методу найближчих сусідів.
2. Чому значення k має значення?
3. Як вибрати оптимальний фактор k?
4. Як оцінити якість побудованої моделі класифікації?
5. Де застосовується метод найближчих сусідів?
6. Плюси та мінуси методу.
7. У яких випадках необхідно здійснювати нормалізацію даних.
8. Наскільки метод чутливий до аномалій?

5. ЗМІСТ ЗВІТУ

1. Тема і мета роботи.
2. Завдання до роботи.
3. Результати виконання завдань.
4. Висновки, що відображують результати виконання роботи та їх критичний аналіз.