Лабораторна робота №3

Інтелектуальний аналіз даних за допомогою програмного пакета WEKA. Метод найближчих сусідів.

Вступ

У двох попередніх лабораторних роботах ми розглянули основні концепції інтелектуального аналізу даних.

Перший вивчений нами метод - метод регресійного аналізу використовується для прогнозування чисельного результату (наприклад, вартості будинку), ґрунтуючись на аналогічних відомих прикладах.

Другий метод, метод класифікації, так само відомий як метод класифікаційних дерев або дерев рішень, дозволяє створити дерево для прогнозування можливого значення невідомої величини (у розглянутому прикладі ми спробували передбачити ефективність рекламної компанії BMW).

Крім того, ми познайомилися з методом кластеризації, який дозволяє розбити наявні дані на кластери (групи) і для кожної з груп виявити свої тенденції і правила (в якості конкретного прикладу використовувалися дані про продажі BMW). Всі ці методи дозволяють перетворити наші дані в корисну інформацію, проте кожен з них використовує свій власний аналітичний підхід, свої типи та набори даних, що, власне, і визначає один з найважливіших аспектів інтелектуального аналізу даних: для того, щоб розробити ефективну модель аналізу, необхідно визначити правильний набір вхідних даних.

В цій лабораторній роботі ми познайомимося з ще одним часто вживаним методом інтелектуального аналізу даних - **методом найближчих сусідів**. Цей метод може розглядатися як свого роду комбінація методів класифікації та кластеризації і являє собою потужне знаряддя в боротьбі з неінформативністю даних.

Метод найближчих сусідів

Метод найближчих сусідів, так само відомий як метод колаборативної фільтрації або метод навчання на прикладах, є досить корисним способом інтелектуального аналізу даних, що дозволяє використовувати накопичені приклади з відомим результатом для прогнозування очікуваного результату для ***нових зразків даних.*** Подібне визначення схоже на визначення методів регресійного аналізу та класифікації. Чим же новий метод відрізняється від них? По-перше, як ви пам'ятаєте, метод регресійного аналізу використовується для прогнозування чисельного результату. Це його основна відмінність від методу найближчих сусідів. Метод класифікації, як ми переконалися на розглянутих раніше завданнях, використовує кожне значення для створення дерева рішень, по якому потрібно пройти для визначення кінцевого результату. Подібний підхід може викликати певні складнощі. Задумайтеся, наприклад, про діяльність такої компанії, як *Amazon*, і наданої їй функціональності «*Покупці, які купили **X**, також купили і **Y***». Уявляєте, скільки гілок і вузлів включало б у себе класифікаційне дерево для подібного аналізу, якби компанія *Amazon* використовувала метод класифікації? Список продуктів містить кілька сотень тисяч різних найменувань, так що ви можете уявити собі обсяг результуючого дерева і його точність. Навіть якщо ви дійшли до останньої гілки, то ви з подивом виявите, що вона містить всього 3 продукти, в той час як *Amazon* зазвичай пропонує своїм відвідувачам 12 продуктів. Так що класифікаційне дерево є не надто придатною моделлю для такого аналізу.

Ви побачите, що метод найближчих сусідів в змозі успішно вирішити ці проблеми, особливо проблему аналізу великих масивів даних, таких, якими оперує, наприклад, *Amazon*. Метод не має обмежень по кількості порівнянь і однаково добре працює і з базою даних, яка містить інформацію про 20 користувачів, і з базою даних, що

нараховує 20 мільйонів користувацьких записів. Крім того, ви самі можете визначити, скільки варіантів рішень ви хочете отримати в результаті роботи методу.

Давайте коротко розглянемо математичний алгоритм, використовуваний в методі найближчих сусідів, щоб вам було простіше розібратися, як працює цей метод, і краще зрозуміти його обмеження.

Математичний алгоритм методу найближчих сусідів

Алгоритм методу найближчих сусідів в чомусь схожий з алгоритмом, використовуваним в методі кластеризації. Метод визначає відстань між невідомою точкою і всіма відомими точками даних. Визначення відстані - цілком тривіальна процедура, що легко виконується в рамках електронних таблиць, так що досить потужний комп'ютер впорається з цим завданням практично миттєво. Найпростіший і найбільш поширений спосіб визначення відстані - це нормалізована евклідова відстань.

Евклідова відстань *(Евклідова метрика) — формула традиційної відстані між двома точками*

$\mathbf{p} = (p_1, p_2, \ldots, p_n)$ *та* $\mathbf{q} = (q_1, q_2, \ldots, q_n)$

для Евклідового простору:

$$\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}.$$

Позначається $\|\mathbf{p} - \mathbf{q}\|$.

Опис звучить складніше, ніж власне обчислення. Звернемося до конкретного прикладу і спробуємо визначити, який товар схильний придбати покупець № 5.

Математична модель методу найближчих сусідів

Покупець	Вік	Дохід	Куплений продукт
1	45	46k	Книга
2	39	100k	TV
3	35	38k	DVD
4	69	150k	Акустична система для автомобіля
5	58	51k	???

Крок 1: Формула для визначення відстані

Відстань = SQRT(((58 - Вік)/(69-35))^2) + ((51000 - Дохід)/(150000-38000))^2)

Крок 2: Розрахунок балів

Покуцпець	Бали	Куплений продукт
1	.385	Книга
2	.710	TV
3	.686	DVD
4	.941	Акустична система для автомобіля
5	0.0	???

Щоб відповісти на питання, який товар з найбільшою ймовірністю придбає покупець № 5, ми скористалися алгоритмом методу найближчих сусідів, наведеним вище, і отримали в результаті в якості найбільш вірогідної покупки книгу. Справа в тому, що відстань між покупцем № 5 і покупцем № 1 менше (значно менше), ніж відстань між покупцем № 5 і будь-яким іншим покупцем. Ґрунтуючись на цій моделі, можна

стверджувати, що поведінка покупця № 5 з великою часткою ймовірності співпаде з поведінкою найближчого до нього покупця.

Однак корисність методу найближчих сусідів цим не вичерпується. Цей алгоритм може бути розширений таким чином, що замість одного найближчого сусіда можна було б визначати будь-яку кількість досить близьких відповідників. Таке розширення алгоритму називається ***N*** найближчих сусідів (наприклад, три найближчих сусіда). Наприклад, якщо в розглянутому вище прикладі потрібно визначити два найбільш вірогідних придбання покупця № 5, то відповідь буде книга і DVD. Якщо потрібно визначити 12 найбільш вірогідних покупок, то слід скористатися алгоритмом 12 найближчих сусідів.

Крім того, алгоритм не обмежується визначенням найбільш вірогідної покупки, він може бути використаний для отримання бінарної відповіді **так/ні**. Якщо в розглянутому вище прикладі ми замінимо значення в останньому стовпці на «Так, Ні, Так, Ні» (для покупців з першого по четвертий), то метод одного найближчого сусіда визначить «Так» у якості найбільш прогнозованої відповіді покупця № 5, метод двох найближчих сусідів теж видасть «Так» (покупці № 1 і № 3 відповіли «Так»), і метод трьох найближчих сусідів теж спрогнозує позитивну відповідь (покупці № 1 і № 3 відповіли «Так», покупець № 2 відповів «Ні», так що середнє значення дорівнює «Так»).

Останнє питання, на яке потрібно відповісти, перш ніж приступити до використання методу найближчих сусідів у практичних завданнях, це вирішити, скільки сусідів потрібно для нашої моделі. Ну що ж, не на всі питання можна легко знайти відповідь. 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.

Набір даних для WEKA

Набір даних, який ми будемо аналізувати методом найближчих сусідів, вам вже знайомий - це той же самий набір даних, який ми використовували для вивчення методу класифікації в попередній роботі, а саме, дані рекламної компанії вигаданого дилера BMW з продажу розширеної дворічної гарантії своїм постійним покупцям.

Наведемо тут ще раз короткий опис цього набору даних.

Дилерський центр має дані про 4500 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
...
```

Реалізація методу найближчих сусідів у WEKA

Чому ми вирішили використовувати той же самий набір даних, яким ми скористалися для вивчення методу класифікації? Тому що, якщо ви пам'ятаєте отриманий нами результат, метод класифікації на цьому наборі даних виявився недостатньо точним (59% точності - з таким же успіхом можна просто намагатися вгадати шукану відповідь). Тепер ми спробуємо побудувати модель з більш високою точністю прогнозування та надати нашому вигаданому дилеру інформацію, придатну для практичного використання.

Завантажимо файл **bmw-training.arff** в WEKA, виконавши в закладці **Preprocess** вже знайомі нам кроки. Вікно WEKA має виглядати так, як показано на рис. 1.

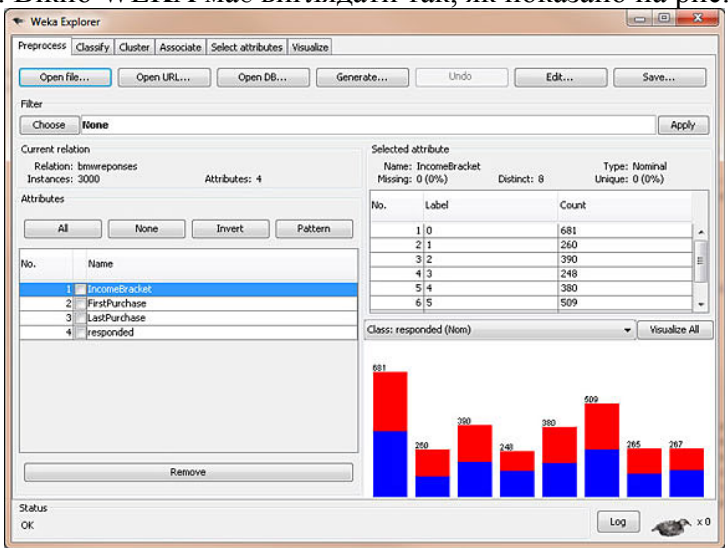

Рис. 1. Дані дилерського центру BMW для аналізу методом найближчих сусідів за допомогою WEKA

Точно так само, як ми виконали це для методів регресійного аналізу та класифікації в попередніх роботах, ми повинні відкрити закладку **Classify**. В 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).

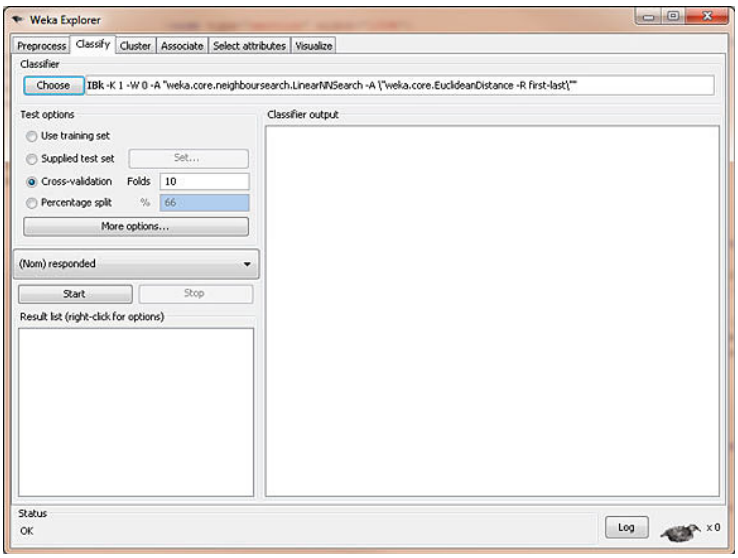

Рис. 2. Алгоритм методу найближчих сусідів для набору даних BMW

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 в WEKA. Натисніть кнопку **Start** і дозвольте WEKA виконати всі необхідні обчислення. На рис. 3 показано, як має виглядати вікно WEKA по закінченні обчислень, далі приведена результуюча модель.

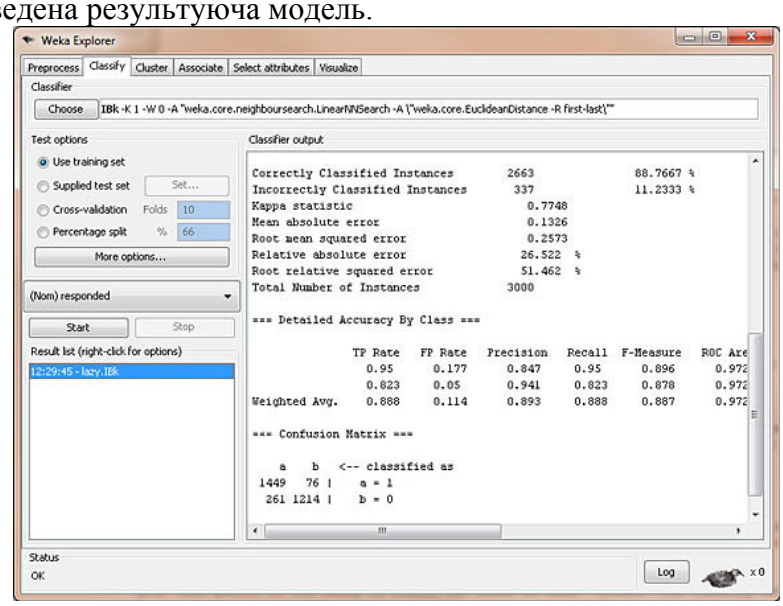

Рис 3. Модель методу найближчих сусідів для набору даних BMW

Результат обчислень IBk

```
=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances        2663               88.7667 %
Incorrectly Classified Instances       337               11.2333 %
Kappa statistic                          0.7748
Mean absolute error                      0.1326
Root mean squared error                  0.2573
Relative absolute error                 26.522  %
Root relative squared error             51.462  %
```

```
Total Number of Instances            3000
=== Detailed Accuracy By Class ===
               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.95      0.177      0.847     0.95      0.896      0.972    1
                 0.823     0.05       0.941     0.823     0.878      0.972    0
Weighted Avg.    0.888     0.114      0.893     0.888     0.887      0.972
=== Confusion Matrix ===
   a    b   <-- classified as
 1449   76 |    a = 1
 261 1214 |    b = 0
```

Як це співвідноситься з моделлю, яку ми отримали за допомогою методу класифікації? У моделі, що використовує метод найближчих сусідів, показник точності дорівнює 89% - 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 76 хибно-позитивних розпізнавань (2.5%) і 261 хибно-негативних розпізнавань (8.7%). 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$ - (2.5%*$3) - (8.7%*400) = $365.

Отже, помилкове розпізнавання помиляється на користь дилера.

Порівняємо цей показник з даними моделі класифікації:

400$ - (17.2%*$ 3) - (23.7%*$400) = $304.

Як ви бачите, використання більш точної 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, таких, наприклад, якими володіє *Amazon*. Маючи дані про 20 мільйонів користувачів нескладно отримати достатньо точний результат - у базі потенційних покупців *Amazon* 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 *Amazon*, 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, так як ви, цілком очевидно, володієте великими грошима. По-друге, подібні обчислення - ідеальне завдання для хмарних систем, так як в 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 *Amazon*, 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.

Контрольні запитання

1. Які інші назви використовуються для методу найближчих сусідів?
2. Основна відмінність регресійного аналізу від методу найближчих сусідів.
3. Математичний алгоритм методу найближчих сусідів.
4. Яким чином можна розширити алгоритм методу найближчих сусідів? (N найближчих сусідів).