ЛАБОРАТОРНА РОБОТА № 5
ЗАДАЧА КЛАСИФІКАЦІЇ

Мета роботи

На практиці вивчити роботу алгоритмів класифікації, навчитися інтерпретувати результати роботи класифікаторів і вибирати найкращий метод для вирішення поставленого завдання.

Основні теоретичні відомості

У лабораторній роботі розглядаються наступні методи класифікації (у дужках наведено назву в програмі WEKA, при цьому перше слово перед крапкою означає тип алгоритму класифікації і також вказує назву папки, в якій знаходиться метод):

– 0R чи Zero Rule класифікація, тобто прогнозування середнього значення для числового класу та моди для категоріального (rules.ZeroR);

– класифікація за одним правилом 1R чи One Rule (rules.OneR);

– покриваючий метод PRISM (rules.Prism);

– наївна Байєсова класифікація (bayes.NaiveBayes);

– методи побудови дерев рішень CART (trees.simpleCart) та C4.5 (trees.J48);

– метод опорних векторів (functions.SMO);

– метод k найближчих сусідів kNN (lazy.IBk).

Для додавання найпростіших методів в WEKA необхідно встановити пакети simpleEducationalLearningSchemes і simpleCART.

Параметри налаштування алгоритмів класифікації

Розглянемо параметри налаштування алгоритмів, що використовуються в роботі (табл. 5.1).

Додаткову інформацію про алгоритми, їх параметри і вимоги до оброблюваних даних можна отримати у вікні налаштувань алгоритмів на панелі «About» в програмі WEKA і в джерелах [2-4].

При виконанні завдань до кожної з лабораторних робіт необхідно дослідити вплив параметрів налаштування на результати роботи алгоритмів.3

Для алгоритмів Prism, Id3, ZeroR параметрів, що настроюються, немає.

Таблица 5.1 – Параметри налаштування класифікаторів

Метод	**Параметр**
OneR	*MinBucketSize* – використовується для дискретизації числових атрибутів
NaiveBayes	*displayModelInOldFormat* – відображення побудованої моделі у старому форматі, що підходить, коли атрибут класу приймає багато значень. Новий формат краще у випадку, коли менше класів і багато атрибутів. *useKernelEstimator* – для оцінки числових атрибутів використовувати оціночну функцію відмінну від нормального розподілу. *useSupervisedDiscretization* – використовувати дискретизацію з учителем для перетворення числових атрибутів у номінальні
J48	*binarySplits* – використовувати бінарний поділ на категоріальних атрибутах для побудови дерев. *confidenceFactor* – довірчий рівень, використовується для відсікання гілок (малі значення - сильніше відсікання). *minNumObj* – мінімальна кількість примірників у листі. *reducedErrorPruning* – який алгоритм відсікання гілок використовувати *saveInstanceData* – чи зберігати навчальну інформацію для візуалізації. *subtreeRaising* – чи використовувати операцію підняття піддерев при відсіканні гілок. *unpruned* – чи залишити дерево повним. *useLaplace* – використовувати оціночну функцію Лапласа для підрахунку ймовірностей в листках
SMO	*buildLogisticModels* – чи застосовувати логістичні моделі до виходів (для належної оцінки ймовірностей). *c* – параметр складності С. *kernel* – функція ядра. *epsilon* – параметр точності (не змінювати). *filterType* – чи буде змінена початкова інформація і яким чином (нормалізація або стандартизація). *toleranceParameter* – допустиме відхилення (не змінювати).

Продовження табл.5.1.

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

Методи оцінки помилок класифікації

Розглянемо параметри оцінки побудованої моделі класифікації.

Матриця помилок – це матриця розміру L × L, де L - число класів, ij-й елемент матриці (i - рядок, j - стовпець) дорівнює числу об'єктів з i-го класу, які були віднесені до j-го. Число вірно класифікованих об'єктів дорівнює сумі елементів, що стоять на головній діагоналі.

Результати добре навченого класифікатора покажуть матрицю помилок, в якій найбільші значення стоять на діагоналі матриці, а невеликі значення (в ідеалі нулі) - на інших позиціях.

Розглянемо матрицю помилок для двох класів «так» та «ні» (табл. 5.2).

На діагоналі матриці знаходяться істинно позитивні (true positive, TP) і істинно негативні (true negative, TN) екземпляри. Примірники, які відносяться до класу «так», але були віднесені класифікатором до класу «ні» називаються хибно негативними (false negative, FN). Примірники, які відносяться до класу «ні», але були віднесені класифікатором до класу «так» називаються хибно позитивними (false positive, FP).

Таблиця 5.2 – Матриця помилок для двох класів

		Передбачений клас		
		Так	**Ні**	
Реальний клас	**Так**	істинно позитивні (TP)	хибно негативні (FN)	P=TP+FN
	Ні	хибно позитивні (FP)	істинно негативні (TN)	N=FP+TN
		P'=TP+FP	N'=FN+TN	

Розглянемо вирази для розрахунку параметрів точності класифікації для кожного з класів. Для випадків класифікації, в яких кількість класів більше двох, для розрахунків приймається, що клас, що розглядається, є класом «так», а всі інші класи об'єднуються та утворюють клас «ні».

Параметр «*TP rate*» (чутливість, *sensitivity*) або «recall» (ефективність) розраховується наступним чином:

$$TP\ rate = recall = \frac{TP}{TP + FN} = \frac{TP}{P}.$$

Для класу, що розглядається, значення цього параметру дорівнює відсотку вірно класифікованих об'єктів класу (отримується діленням діагонального елементу матриці помилок на суму елементів в його рядку). Іншими словами, параметр чутливості показує долю позитивних екземплярів, які були вірно розпізнані.

Параметр «FP rate» розраховується за формулою:

$$FP\ rate = \frac{FP}{FP + TN} = \frac{FP}{N}.$$

Його значення дорівнює відсотку об'єктів інших класів, які помилково були занесені в клас, що розглядається (якщо з матриці викреслити рядок класу, що розглядається, то значення дорівнюватиме сумі елементів стовпця цього класу, поділеній на суму всіх елементів).

Параметр «TN rate» (специфічність, specificity) дорівнює:

$$TN\ rate = \frac{TN}{TN + FP} = \frac{TN}{N}.$$

та показує частину негативних екземплярів, які були вірно розпізнані.

Параметр «precision» (точність) розраховується як:

$$precision = \frac{TP}{TP + FP} = \frac{TP}{P'}.$$

Його значення дорівнює відсотку вірно класифікованих об'єктів із всіх об'єктів, віднесених алгоритмом до класу, що розглядається (відношення діагонального елемента до суми елементів стовпця).

Параметр «F-measure» - це середнє гармонійне Precision і Recall:

$$F - measure = 2 * \frac{precision * recall}{precision + recall}.$$

Параметри success rate (accuracy, recognition rate – частка успішних спроб, точність, коефіцієнт розпізнавання) та error rate (misclassification – частка помилок, помилкова класифікація):

$$success\ rate = \frac{TP + TN}{TP + TN + FP + FN};$$

$$error\ rate = 1 - success\ rate = \frac{FP + FN}{TP + TN + FP + FN}.$$

Також буває важливо оцінити втрати і виграші (costs and benefits), пов'язані з класифікаційної моделлю. Втрати, пов'язані з хибно позитивними передбаченнями (як наприклад, невірне передбачення, що хворий пацієнт є здоровим), набагато більші, ніж хибно негативні передбачення (здоровий пацієнт віднесений до хворих). В таких випадках ми можемо віддати перевагу одному типу помилки над іншим шляхом призначення їм різних значень втрат, пов'язаних з перейменуванням класів.

Наприклад, розглянемо задачі видачі кредиту банком. Втрати при видачі кредиту неплатнику набагато вище, ніж втрати від невидачі кредиту особі, яка зуміла би виплатити свій борг.

Розглянемо тепер проблему нерівномірного розподілу класів, коли важливий для задачі клас є рідкісним в вибірці. Це означає, що розподіл набору даних відображає значну більшість негативних примірників і меншість позитивних примірників.

Наприклад, у задачі розпізнавання шахрайства цікавим для нас класом (позитивним) є клас «шахрайство», який з'являється набагато рідше, ніж негативний клас «не шахрайство». У медичній задачі до такого рідкісного класу може бути віднесений клас «злоякісна

пухлина».

Припустимо, що ми навчили класифікатор класифікувати медичний набір даних, в якому цільовим атрибутом є атрибут «злоякісна пухлина», який може приймати значення «так» і «ні».

Параметр точності success rate рівний 97% може показати, що класифікатор досить точний. Однак якщо у всій вибірці було тільки 3% примірників, що відносяться до злоякісних пухлин? Ясно, що в такому випадку точність розпізнавання в 97% не може бути прийнятною. У цьому випадку класифікатор може правильно розпізнавати більшість негативних екземплярів (не злоякісна) і помилково класифікувати всі позитивні примірники (злоякісна).

Таким чином нам необхідні інші параметри, що дозволяють оцінити наскільки добре класифікатор розпізнає позитивні примірники і негативні. Для цієї задачі можуть бути використані параметри sensitivity та specificity.

Розглянемо наступний приклад навчання класифікатора.

Таблиця 5.3 – Приклад навчання класифікатора

Клас	Так	Ні	Всього	Розпізнавання (%)
Так	90	210	300	90/300=30,00 (sensitivity)
Ні	140	9560	9700	9560/9700=98,56 (specificity)
Всього	230	9770	10000	9650/10,000=96,50 (середнє)

Можна помітити, що хоча класифікатор показав загальну високу точність класифікації, його можливості правильно розпізнавати позитивний (рідкісний) клас дуже низькі (що видно з величини параметра чутливості). У той же час параметр специфічності високий, що означає, що класифікатор може точно розпізнавати негативні класи.

Параметри precision і recall також широко використовуються в класифікації. Precision може бути розглянутий як міра точності / влучності (тобто який відсоток примірників віднесених до позитивних такими і є), тоді як recall - це міра повноти (який відсоток позитивних примірників віднесені до позитивних).

що кожен екземпляр, який класифікатор відніс до класу С, насправді належить класу С. Однак, це нічого не говорить нам про кількість примірників клас С, які класифікатор неправильно класифікував. Ідеальне значення параметра recall в 1,0 для класу С означає, що кожен екземпляр класу С був віднесений класифікатором до класу С, але це нічого не говорить нам про те, скільки інших

екземплярів були неправильно класифіковані та віднесені до класу С.

Існує тенденція зворотного взаємозв'язку між параметрами precision і recall, тобто існує можливість збільшити один параметр за рахунок зменшення іншого. Приміром, наш медичний класифікатор може досягти високого значення параметра precision відносячи всі екземпляри класу злоякісних пухлин до класу злоякісних, але при цьому може мати низьке значення параметра recall відносячи до класу злоякісних також негативні екземпляри. Зазвичай ці два параметри розглядаються сумісно.

Альтернативний шлях застосування цих параметрів - це їх об'єднання в одному параметрі F-measure (F1 score або F-score).

Розглянемо ще один параметр оцінки точності класифікації. У табл. 5.4 представлений приклад матриці помилок задачі класифікації з трьома класами.

Таблиця 5.4 – Приклад №1 матриці помилок

		Передбачений клас			
		A	**B**	**C**	**Total**
Реальний клас	**A**	88	10	2	100
	B	14	40	6	60
	C	18	10	12	40
	Total	120	60	20	

У цьому прикладі тестова вибірка містить 200 примірників (сума дев'яти елементів матриці). Класифікатор для тестової вибірки передбачив 120 екземплярів класу А, 60 - класу В, 20 - класу С, і при цьому 88 +40 +12 = 140 з них правильно класифіковані. Відсоток правильно класифікованих об'єктів для цього прикладу дорівнює 70%.

Що якби на цій самій вибірці працював би випадковий класифікатор, який передбачив би таку ж кількість примірників кожного класу. Розглянемо таблицю 5.5.

У першому рядку 100 екземплярів класу А розділені в такій же пропорції як (120:60:20). Точно так само розділені примірники другої і третього рядка. Загальні значення в рядках і стовпцях не змінилися, а змінилися значення матриці. Таким чином, випадковий класифікатор дає 60 +18 +4 = 82 правильно класифікованих примірників.

Таблиця 5.5 – Приклад №2 матриці помилок

		Передбачений клас			
		A	**B**	**C**	**Total**
Реальний клас	**A**	60	30	10	100
	B	36	18	6	60
	C	24	12	4	40
	Total	120	60	20	

Параметр Каппа розраховує це очікуване значення (виводячи його з успішності класифікатора) і виражає результат у вигляді відношення: у чисельнику 140-82 = 58 примірників поліпшення в порівнянні з випадковим прогнозуванням, а в знаменнику – все можливе поліпшення 200-82 = 118. Для наведеного вище прикладу параметр Каппа дорівнює 49,2%. Максимальне значення параметра Каппа 100% для ідеального передбачення, а мінімальне 0 - для випадкового. Загалом, можна сказати, що статистичний параметр Каппа використовується для оцінки згоди між прогнозованою і спостережуваною категоризацією набору даних з поправкою на випадковість.

Критерії порівняння роботи класифікаторів

На додаток до параметрів, заснованих на точності, класифікатори можуть бути порівняні за такими наступними параметрами.

Швидкість роботи - іншими словами обчислювальні витрати, пов'язані з навчанням і застосуванням даного класифікатора.

Стійкість до помилок, робастність - можливість класифікатора робити правильні передбачення на зашумлених даних або даних з пропущеними значеннями. Даний параметр зазвичай оцінюється за допомогою синтетичних наборів даних із внесеними шумами і втраченими значеннями атрибутів.

Масштабованість - можливість будувати ефективний класифікатор на великих вибірках. Даний параметр оцінюється за допомогою наборів даних, що збільшуються.

Інтерпретованість - рівень розуміння та можливості проникнути в суть даних, які надає класифікатор. Інтерпретованість є суб'єктивним параметром і тому його важко оцінити. Дерева рішень та класифікаційні правила можуть бути легко інтерпретовані, проте їх інтерпретованість зменшується з їх ускладненням.

Інтерпретація результатів класифікації в WEKA

Розглянемо результати роботи класифікаторів в WEKA (Classifier output).

Секція «*Run information*» містить наступну інформацію:

– метод класифікації (scheme);

– назва набору даних, на якому проводилося навчання (relation);

– кількість примірників у вихідній вибірці (instances);

– атрибути, що характеризують об'єкти вибірки (attributes);

– відомості про тестову вибірку (test mode).

Секція «*Classifier model*» містить параметри налаштованого класифікатора і час, затрачений для побудови моделі. Залежно від типу класифікатора дана область буде містити різну інформацію:

– для алгоритмів, що будують правила, будуть відображені отримані правила;

– для байєсівських класифікаторів будуть перераховані розраховані ймовірності для всіх можливих комбінацій атрибут-значення-клас;

– для класифікаторів, заснованих на побудові дерев, відображається текстове представлення отриманого дерева; в дужках навпроти кожного листа вказана кількість примірників, які до нього віднесені; якщо в лист потрапляють екземпляри декількох класів, через слеш буде вказана кількість примірників, які відносяться до домішок;

– для функціональних методів виводяться значення коефіцієнтів побудованої функціональної моделі;

– для методу k найближчих сусідів відображаються налаштування класифікатора.

Секція «*Predictions*» буде відображена, якщо в налаштуваннях тестування класифікатора обрана опція «Output predictions». У ній для всіх примірників тестової вибірки будуть відображені результати класифікації, отримані за допомогою навченого класифікатора.

Секція оцінки побудованої моделі «*Evaluation*» містить кілька підпунктів.

«*Summary*» містить загальну статистику роботи класифікатора:

– кількість та відсоток правильно і неправильно класифікованих примірників (Correctly and Incorrectly Classified Instances), загальна кількість примірників (Total Number of Instances);

– параметр Каппа (Kappa statistic);

– статистичні параметри помилки класифікації (Mean absolute

error, Root mean squared error, Relative absolute error, Root relative squared error).

«*Detailed Accuracy By Class*» містить наступні параметри точності класифікації по кожному з класів: TP Rate, FP Rate, Precision, Recall, F-Measure, ROC Area.

«*Confusion Matrix*» містить матрицю помилок.

Завдання на лабораторну роботу

1. Для індивідуального завдання (додаток Б) вирішите задачу класифікації за допомогою всіх розглянутих в лабораторній роботі алгоритмів.
2. Змінюючи параметри налаштування алгоритмів, спробуйте досягти найліпшої якості навчання класифікаторів.
3. Порівняйте отримані моделі за допомогою модуля Experimenter.
4. В звіті надайте результати роботи кожного алгоритму, його налаштування, а також результати порівняння

Контрольні питання

1. У чому полягає задача класифікації? Наведіть практичний приклад.
2. Опишіть один із розглянутих методів класифікації.
3. Що таке навчання з учителем і без учителя? До якого типу належить задача класифікації?
4. Задача класифікації є описовою або прогнозуючою і чому?
5. Навіщо потрібні дві вибірки: навчальна і тестова?
6. Які існують підходи для поділу вихідної вибірки на навчальну і тестову?
7. Як оцінити якість побудованої моделі класифікації?
8. Що таке матриця помилок? Як її інтерпретувати?
9. Що означають параметри чутливість, специфічність, точність? Як їх розрахувати?
10. Що таке параметр Каппа? Що він показує?
11. Що таке аналіз втрати-виграші? Навіщо він?
12. Як порівняти роботу двох класифікаторів?
13. Що таке ансамблі класифікаторів і адаптивний бустінг? Для чого вони застосовуються?

Зміст звіту

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