МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет "Львівська політехніка"

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

Класифікація методом опорних векторів.

МЕТОДИЧНІ ВКАЗІВКИ
до лабораторної роботи № 11

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

Ознайомитися та отримати навички побудови моделей класифікації за допомогою Data Mining GUI бібліотеки Weka. На практиці вивчити роботу методу опорних векторів, навчитися інтерпретувати результати роботу класифікатора.

2. КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ

2.1. Метод опорних векторів

У лабораторній роботі розглядається метод опорних векторів (у WEKA - functions.SMO);

Метод Опорних Векторів (Support Vector Machines, SVM) є популярним методом класичної класифікації та регресії. Даний алгоритм має широке застосування на практиці і може вирішувати як лінійні так і нелінійні задачі. Суть методу Опорних Векторів проста: алгоритм створює лінію або гіперплощину, яка розділяє дані на класи. Основним завданням алгоритму є знайти найбільш правильну лінію, або гіперплощину, що розділяє дані на два класи. На зображенні це показано наочніше:

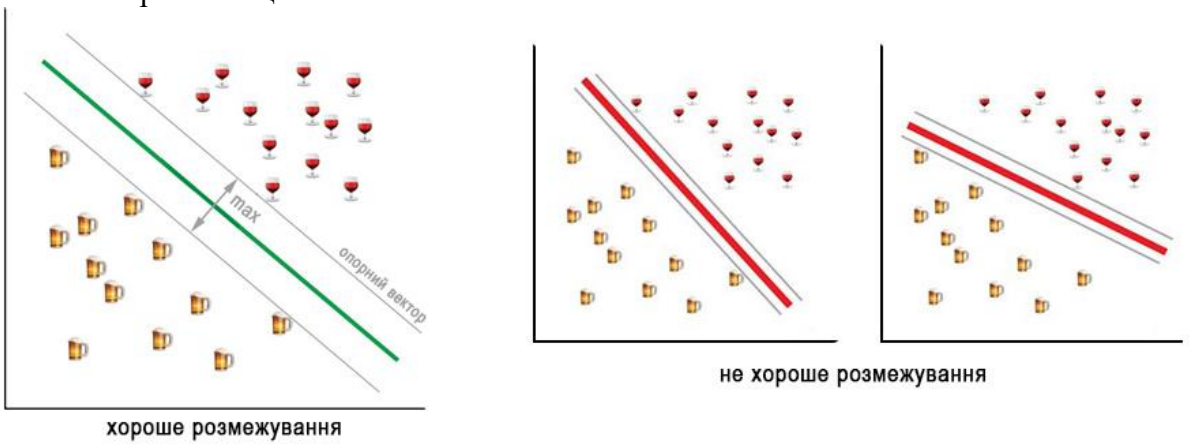

Рис.1 Гіперплощина з хорошим і поганим розмежуванням

Припустимо є набір даних, і потрібно класифікувати і розділити червоні квадрати від синіх кіл. Основною метою в даній задачі буде знайти "ідеальну" лінію, яка розділить набір даних на синій і червоний класи. Втім, немає однієї, унікальної лінії, яка б вирішувала таку задачу. Можна підібрати багато таких ліній, які можуть розділити ці два класи, але яка з двох ліній (суцільна або пунктирна) найкраще розділяє два класи, і підходить під опис "ідеальної" (рис.2.).

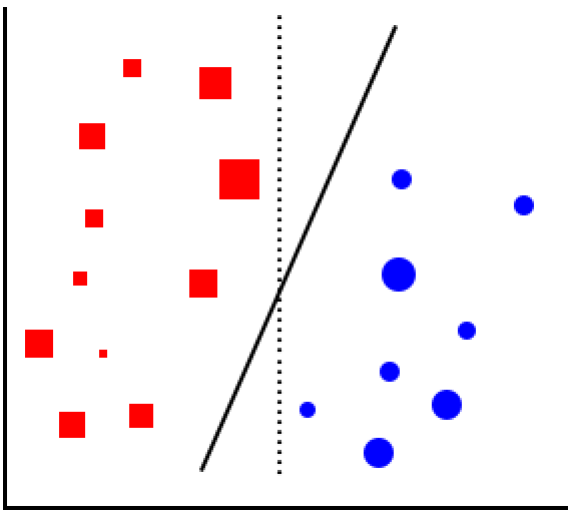

Рис.2. Підбір лінії, що найкраще розділяє класи об’єктів

Алгоритм вибере суцільну пряму, оскільки вона розділяє і відповідно класифікує два класи краще за зелену. У випадку з пунктирною лінією - вона розташована занадто близько до червоного класу.

Алгоритм SVM влаштований таким чином, що він шукає точки на графіку, які розташовані безпосередньо до лінії розділення найближче. Ці точки називаються опорними векторами. Алгоритм обчислює відстань між опорними векторами і роздільною лінією. Ця відстань називається зазором. Основна мета алгоритму - максимізувати відстань зазору. Кращою вважається така лінія, для якої цей зазор є максимально великим (рис.3).

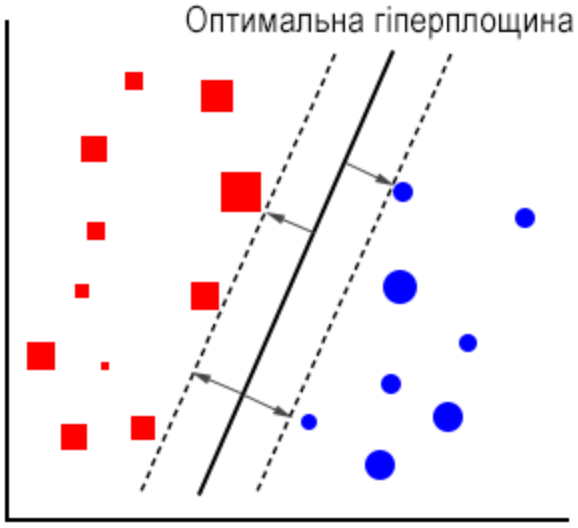

Рис.3. Визначення оптимальної лінії, що розділяє класи

Якщо є набір складніший набір даних, наприклад про пацієнта клініки. Кожен пацієнт може бути описаний різними параметрами, такими як пульс, рівень холестерину, тиск тощо. Кожен з цих параметрів є виміром. SVM відображає ці параметри у багатовимірному просторі вищого виміру, а потім знаходить гіперплощину, щоб розділити класи.

Гіперплощина - це n-1 мірна площина в n-вимірному Евклідовому просторі, яка поділяє простір на дві окремі частини.

Алгоритм SVM є популярним для спам-фільтрів, раніше часто використовували для класифікатора осіб.

Історія та використання SVM

• Вапник та колеги (1992) — основа статистичної теорії навчання Вапника та Червоненкіса 1960-х років.

• Особливості: навчання може бути повільним, але точність висока завдяки їх здатності моделювати складні нелінійні межі рішень (максимізація маржі).

• Використовується як для класифікації, так і для прогнозування.

• Застосування: розпізнавання рукописних цифр, розпізнавання об’єктів, ідентифікація мовця, порівняльний аналіз тестів прогнозування часових рядів.

Алгоритм

• Визначте оптимальну гіперплощину: максимізуйте запас

• Розширте наведене вище визначення для нелінійно роздільних задач: встановіть термін штрафу за неправильну класифікацію.

• Відображення даних у багатовимірному просторі, де їх легше класифікувати за допомогою лінійних поверхонь прийняття рішень: переформулюйте проблему так, щоб дані неявно відображалися в цьому просторі.

2.2. Приклад виконання класифікації у WEKA

Наведений нижче приклад є перекладом та інтерпретацією англомовних вказівок до виконання класифікація на прикладі даних, отриманих вд дилерськиих центрів BMW.

https://www.programmersought.com/article/9237166886/

У нашому випадку було взято датасет bmw-training.arff з 3000 записів та проведено його рощеплення на власне bmw-training.arff (1999 записів) та bmw-test.arff з рештою записів.

Одним із ваших завдань є порівняти отримані результати у першоджерелі із 4500 записами та результатами, отриманими на 3000 записів.

Набір даних, який буде застосований для прикладу класифікаційного аналізу, містить інформацію (bmw-training.arff, bmw-test.arff), зібрану центром продажу компанії BMW. Центр починає рекламну компанію, пропонуючи розширену дворічну гарантію своїм постійним клієнтам. Подібні компанії вже проводилися, так що центр продажу має у розпорядженні 3000 екземплярів даних щодо попередніх продажів з розширеною гарантією. Цей набір даних охоплює наступні 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 у форматі Attribute-Relation File Format (ARFF) буде виглядати наступним чином, див. Лістинг 1.

Лістинг 1. Файл даних для класифікаційного аналізу у Weka

```
@attribute IncomeBracket {0,1,2,3,4,5,6,7} % Групи за доходом
@attribute FirstPurchase numeric % перша покупка
@attribute LastPurchase numeric % остання покупка
@attribute responded {1,0} % відгук
@data
4,200210,200601,0
5,200301,200601,1
...
```

При побудові моделі класифікації набір даних зазвичай ділять так, щоб частина даних використовувалася для побудови моделі (навчання), частина – для перевірки її коректності (тестування), щоб переконатися, що модель не є навченою тільки під конкретний набір даних.

Розділіть вибраний набір даних на два файли *.arff в співвідношенні «2/3» (навчальні дані bmw-training.arff) та «1/3» (тестові дані bmw-test.arff) від загальної кількості даних. Завантажте файл «2/3» в програмний пакет Weka.

Коли файл з навчальними даними готовий, його потрібно завантажити у Weka. Запустіть Weka і виберіть опцію Explorer. У результаті відкриється закладка Preprocess у вікні Explorer. Натисніть кнопку Open File і виберіть створений вами ARFF-файл. Вікно Weka Explorer з завантаженими даними показано на Рисунку 1. Зауваження: в пропонованому навчальному файлі містяться 1999 записів.

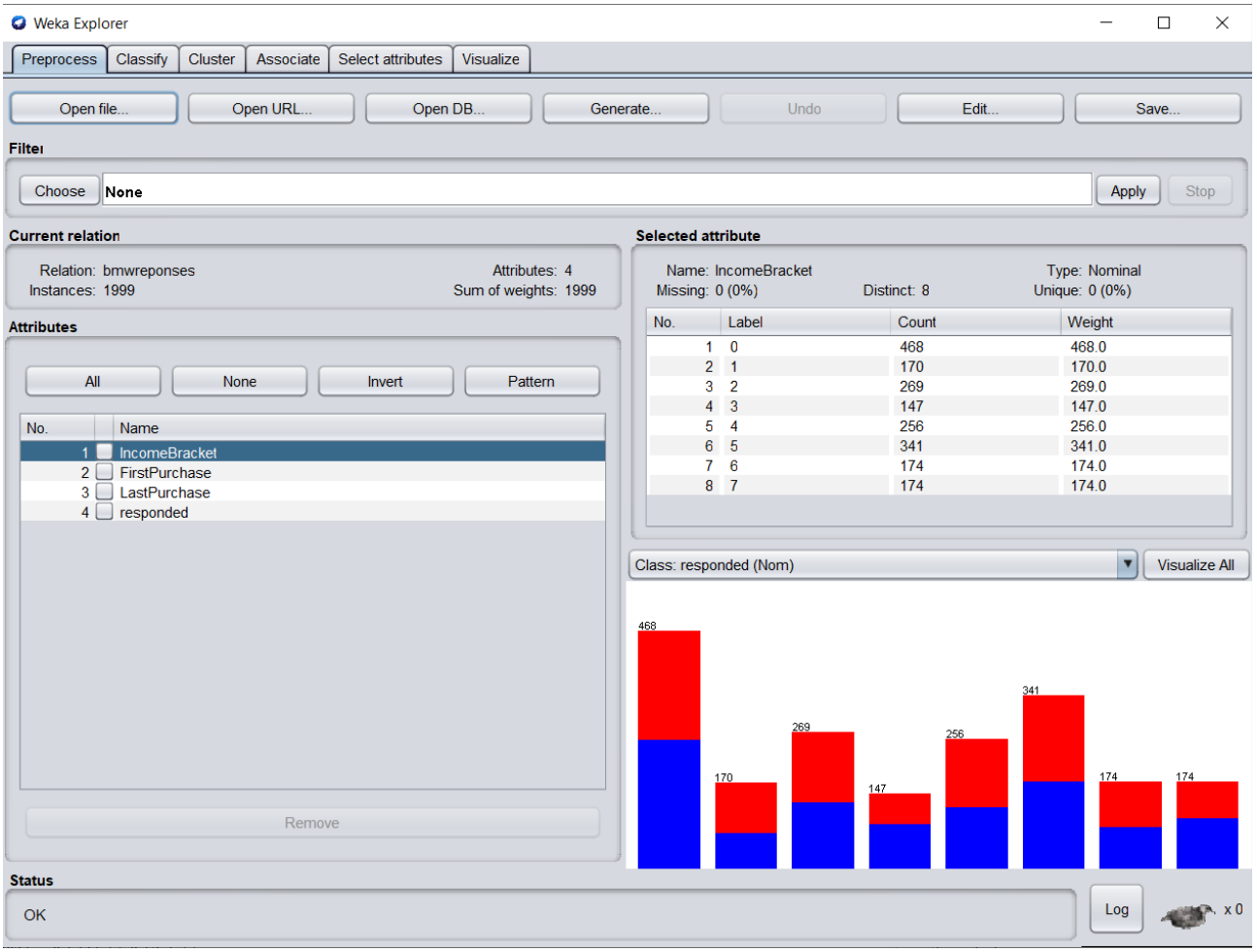

Рисунок 1. Дані дилерського центру BMW

У цьому вікні ви можете перевірити дані, на підставі яких ви збираєтесь будувати модель. У лівій частині вікна Explorer показані атрибути даних (Attributes), які відповідають заголовкам стовпців таблиці, також вказано кількість екземплярів даних (Instances), тобто рядків таблиці. Якщо виділити мишкою однин з заголовків стовпців, тоді в правій частині вікна з'являться значення відповідного атрибуту для різних екземплярів даних. Також існує можливість візуального аналізу даних за допомогою кнопки Visualize All.

2.3. Параметри налаштування алгоритму

Розглянемо параметри налаштування алгоритму, який використовується у лабораторній роботі (табл.1).

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

Таблиця 1. Параметри налаштування класифікаторів

Метод	**Параметр**
SMO	*buildLogisticModels* – чи застосовувати логістичні моделі до виходів (для належної оцінки ймовірностей). *c – параметр складності C.* *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).

3. ЛАБОРАТОРНЕ ЗАВДАННЯ

1. Для індивідуального завдання вирішіть задачу класифікації за допомогою методу опорних векторів (functions.SMO).

2. Змінюючи параметри налаштування алгоритму, спробуйте досягти найвищої якості навчання класифікатора.

3. Здійсніть класифікацію методом опорних векторів із датасетом використаним у WEKA у Excel за допомогою пакету Excel2SVM (https://www.bioinformatics.org/Excel2SVM/)

4. Порівняйте отримані результати від різних систем.

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

4. КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке опорні векторні машини?
2. Що таке опорні вектори в SVM?
3. Як працюють опорні векторні машини?
4. Яка геометрична інтуїція стоїть за SVM?
5. Що вам відомо про Hard Margin SVM і Soft Margin SVM?
6. Що таке трюк з ядром у SVM і чим він корисний?
7. Що впливає на межі прийняття рішень у SVM?
8. Яка різниця в ідеї використання опорної векторної машини для регресії та класифікації?
9. Яка мінімально можлива кількість опорних векторів для N-вимірного набору даних?
10. Як мати справу з кількома класами за допомогою SVM?
11. Порівняйте K-Nearest Neighbors (KNN) і SVM
12. Коли SVM не є хорошим підходом?
13. Чи дає SVM якийсь імовірнісний результат?

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.