Передмова

Курс лекцій з розпізнавання образів для студентів спеціальності "Прикладна математика"факультету кібернетики Київського національного університету імені Тараса Шевченка, що читався в 2012-2014 рр. протягом двох семестрів, складається з двох частин, які умовно можна назвати "Розпізнавання як оптимізація" і "Розпізнавання як перевірка статистичних гіпотез". В цей посібник включені лекції з першої частини курсу.

Я глибоко вдячний Сергію Івановичу Ляшку, Володимиру Вікторовичу Семенову та Дмитру Анатолійовичу Номіровському за плідні наукові і методичні дискусії, а також щиро дякую своїм аспірантам В'ячеславу Алєксєєнку та Марині Присяжній за допомогу в роботі.

Хочу віддати особливу шану одному із своїх вчителів, видатному вченому і унікальній людині, покійному професору Юрію Івановичу Петуніну, з яким я мав щастя працювати понад 20 років.

Зауваження і поради щодо лекцій читачі можуть надсилати на адресу dokmed5@gmail.com. Буду вдячний за доброзичливу критику і конструктивні пропозиції.

Київ, 6 жовтня 2014 року
Д.А. Клюшин

Глава 1

Основні поняття розпізнавання образів

1.1. Основні концепції

Розпізнавання образів — це розділ теорії штучного інтелекту, що вивчає методи комп'ютерної класифікації об'єктів, тобто методи автоматичної ідентифікації одного із наперед заданих класів (двох або більше), якому належить об'єкт. Метою перших досліджень у цьому напрямку була реалізація органів зору у роботів, тому, за традицією, об'єкт, що піддається класифікації, називається ***образом***. Образом може бути цифрова фотографія (розпізнавання зображень), буква або цифра (розпізнавання символів), аудіозапис (розпізнавання мови) тощо.

Важливою складовою класифікації є процедура машинного навчання, метою якої є побудова ***вирішального правила або функції***, що класифікує об'єкти за ознаками. Відповідно до того, який аспект розпізнавання образів вибирається за основу, машинне навчання часто розглядають або як геометричну теорію, метою якої є пошук лінійної або нелінійної поверхні, що розділяє задані множини прецедентів у просторі ознак, або як статистичну теорію, метою якої є пошук оптимальної апроксимації вирішальної функції за допомогою функцій із наперед заданої множини і навчальних вибірок.

Другою частиною процесу класифікації є ***процедура узагальнення***, яка полягає у класифікації нової вибірки за допомогою вирішального правила, сформульованого у процесі навчання. Отже, класифікація складається з двох етапів: 1) індуктивного, що полягає у навчанні, тобто отриманні загальної інформації (вирішального правила) з часткових спостережень (навчальних вибірок); і 2) дедуктивного, що полягає у застосування знайденого вирішального правила для класифікації нових об'єктів.

$$\boxed{\textbf{Розпізнавання} = \textbf{навчання} + \textbf{узагальнення}}$$

Перейдемо до опису математичного формалізму, що лежить в основі розпізнавання образів.

Означення 1. Нехай X — множина об'єктів, $\Omega = \{\omega_2, \omega_2, ..., \omega_N\}$ — множина міток класів $C_1, C_2, ..., C_N$, $f : X \to \Omega$ — цільова функція, значення якої відомі лише на скінченній підмножині об'єктів $\{x_1, x_2, ..., x_m\} \subset X$. Пари (x_i, ω_i) називаються ***прецедентами***, а сукупність пар $T = \{(x_i, \omega_i)\}_{i=1}^{m}$ — ***навчальною вибіркою***.

Задача навчання за прецедентами полягає в тому, щоб за вибіркою T відновити функцію f, тобто побудувати вирішальну (індикаторну) функцію $g : X \to \Omega$, що у певному розумінні найкращим чином наближає цільову функцію $f : X \to \Omega$ не лише на об'єктах $\{x_1, x_2, ..., x_m\}$, а й на всій множині X. В задачі класифікації на N диз'юнктних класів множина міток визначається однозначно. У випадку $N = 2$ класифікація називається ***бінарною***.

Означення 2. ***Ознака об'єкта*** — це результат вимірювання числової або категорійної характеристики. З формальної точки зору ознака є відображенням $x : X \to D$, де D — множина допустимих значень ознаки.

Означення 3. Вектор ознак $(x_1, x_2, ..., x_n)$ називається ***ознаковим описом об'єкта***, де $x_i \in D_i$.

В основу теорії розпізнавання образів покладено два постулати.

1. **Постулат про векторну модель:** об'єкт можна подати як елемент векторного простору ознак.

2. **Постулат про компактність:** переважна більшість об'єктів, що належать до одного класу, є більш близькими один до одного, ніж до об'єктів іншого класу, і лежать в області і з відносно простою межею.

Виходячи з цих постулатів, навчання можна описати як задачу пошуку поверхні, що розділяє множини розмічених точок у векторному просторі. Оскільки таких поверхонь може бути безліч, виникає задача про пошук оптимальної за певним критерієм роздільної поверхні. Позначимо множину допустимих роздільних поверхонь як S (наприклад, допустимою множиною поверхонь може бути множина ліній на площині або гіперплощин у багатовимірному євклідовому просторі), а критерій якості розпізнавання, або функцію втрат, як J.

Означення 4. ***Критерій якості розпізнавання*** — це невід'ємний функціонал $\mathcal{J}(s, x)$, який характеризує величину помилки при класифікації об'єкта x за допомогою роздільної поверхні s. Якщо $\mathcal{J}(s, x) = 0$, класифікація називається правильною.

Тоді задачу навчання можна сформулювати так: знайти

$$\arg \min_{s \in S, x \in T} J(s, x)$$

Як правило, критерій якості формулюють за допомогою функції середніх втрат, або емпіричного ризику, яка характеризує кількість помилок.

Означення 5. ***Функція втрат, або емпіричний ризик*** на вибірці T вирішального правила, основаного на роздільній поверхні s, має вигляд

$$J(s,T) = \frac{1}{m}\sum_{i=1}^{m} \mathcal{L}(s, x_i),$$

де функція $\mathcal{L}$ набуває значення 0, якщо об'єкт x класифікується правильно, і 1, якщо об'єкт x класифікується неправильно. У цьому випадку емпіричний ризик $J(s,T)$ дорівнює частоті помилок вирішального правила, основаного на роздільній поверхні s, на об'єктах навчальної вибірки.

Розпізнавання = оптимізація

На практиці якість класифікації часто характеризують такими показниками як точність, чутливість, специфічність тощо. Припустимо, що тестова вибірка складається з P об'єктів класу A (позначення P означає positive — позитивні об'єкти) і N об'єктів класу B (позначення N означає negative — негативні об'єкти). Якщо об'єкт, про який заздалегідь відомо, що він належить до класу A, класифікується як позитивний, то результат називається ***істинно позитивним***. Якщо об'єкт, про який заздалегідь відомо, що він належить до класу B, класифікується як негативний, результат називається ***істинно негативним***. Відповідно, помилкові результати класифікації називаються ***хибно позитивними*** і ***хибно негативними*** (FN — false negative). Позначимо кількість істинно позитивних результатів як TP (true positive) кількість істинно негативних результатів як TN (true negative), кількість хибно позитивних результатів як FP (false positive) і кількість хибно негативних результатів як FN (false negative).

Означення 6. ***Чутливість, або TPR (true positive rate)***, це доля істинно-позитивних результатів серед усіх позитивних результатів, тобто $TPR = \frac{TP}{TP+FN} = \frac{TP}{P}$.

Означення 7. ***Специфічність, або TNR (true negative rate)***, це доля істинно негативних результатів серед усіх негативних результатів, тобто $TNR = \frac{TN}{TN+FP} = \frac{TN}{N}$.

Означення 8. ***Точність*** — це доля правильних результатів серед усіх результатів класифікації, тобто $PR = \frac{TP+TN}{P+N}$.

Якщо долю хибно позитивно результатів позначити як $FPR = \frac{FP}{N}$, то легко бачити, що $FPR + TNR = 1$.

Терміни чутливість, специфічність і точність характерні для класифікації, яка проводиться у медичних дослідженнях. В галузі інформаційного пошуку чутливість називають повнотою (recall), специфічність — релевантністю (relevance)

Означення 9. ***Крива залежності специфічності*** TPR ***від величини*** $1 - TNR$ ***при варіюванні параметрів вирішальної функції, називається ROC-кривою.*** Ця крива характеризує якість бінарної класифікації і називається також кривою помилок, а її аналіз називається ROC-аналізом.

Якісна інтерпретація ROC-кривої виражається площею, обмеженою ROC-кривою і віссю, на якій відкладаються долі хибних позитивних результатів, тобто величина 1-специфічність. Цей показник називається AUC (area under curve — площа під кривою). Якісні класифікатори мають більш високий показник AUC. Якість класифікатора в залежності від значення AUC визначається так: від 0,9 до 1.0 — відміна, від 0,8 до 0.9 — дуже добра, 0,7-0,8 — добра, 0,6-0,7 — задовільна, 0,5-0.6 — незадовільна (де факто, випадковий результат)

1.2. Ймовірнісні концепції розпізнавання образів

Дані про об'єкти з множини X можуть бути неточними або неповними. У цьому випадку одному опису x можуть відповідати разні відповіді. Ймовірнісна постановка полягає у такому: замість невідомої цільової залежності $f(x)$ припускається існування невідомого ймовірнісного розподілу на множині $X \times \Omega$ із щільністю $p(x, \omega)$, з якого випадково і незалежно вибираються спостереження $T = \{(x_i, \omega_i)\}_{i=1}^{m}$. Такі вибірки називаються **простими**.

1.2.1. Принцип максимальної правдоподібності

При ймовірнісній постановці задачі замість моделі алгоритмів $g(x, \theta)$, яка апроксимує невідому залежність $f(x)$, задається модель сумісної щільності розподілу об'єктів і відповідей $\varphi(x, \omega, \theta)$, що апроксимує невідому ймовірність $p(x, \omega)$. Після цього визначається значення параметру θ, при якому вибірка даних T є найбільш правдоподібною, тобто найкраще узгоджується із моделлю щільності. Якщо спостереження у вибірці T є незалежними, то сумісна щільність всіх спостережень дорівнює добутку значень щільності $p(x, \omega)$ для кожного спостереження:

$$p(T) = \prod_{i=1}^{m} p(x_i, \omega_i).$$

Якщо апроксимувати $p(x_i, \omega_i)$ моделлю щільності $\varphi(x_i, \omega_i, \theta)$, отримуємо функцію правдоподібності

$$L(\theta, T) = \prod_{i=1}^{m} \varphi(x_i, \omega_i, \theta).$$

Що більше значення функції правдоподібности, то краще вибірка узгоджується з моделлю. Отже, треба шукати

$$\arg\max_{\theta} L(\theta, T).$$

Описаний вище метод називається принципом правдоподібності.

1.2.2. Мінімізація емпіричного ризику

Замість максимізації функції правдоподібності $L(\theta, T)$ зручніше мінімізувати функціонал $-\ln L(\theta, T)$, оскільки він є адитивним за об'єктами вибірки.

$$-\ln L(\theta, T) = -\sum_{i=1}^{m} \ln \varphi(x_i, \omega_i, \theta) \to \min_{\theta}.$$

Означення 10. Ймовірнісна функція втрат дорівнює

$$\mathcal{L}(a_\theta, x) = -m \ln \varphi(x_i, \omega_i, \theta).$$

Що гірше пара (x_i, ω_i) узгоджується з моделлю φ, то менше значення щільності $\varphi(x_i, y_i, \theta)$ і більше величина втрати $\mathcal{L}(a_\theta, x)$, і навпаки, для багатьох функцій втрат можна підібрати таку модель щільності $\varphi(x, \omega, \theta)$, щоб мінімізація емпіричного ризику була еквівалентною максимізації правдоподібності.

1.3. Перенавчання і здатність до узагальнення

Мінімізація емпіричного ризику має певні особливості, а саме: якщо мінімум функціонала якості $J(s, T)$ досягається на алгоритмі g, це не гарантує, що алгоритм g буде добре наближати цільову залежність на довільній контрольній виборці

$$K = \{(x_i', \omega_i')\}_{i=1}^{n}.$$

Погіршення якості роботи алгоритму на об'єктах,які не входили до навчальної вибірки, може бути наслідком наднавчання.

Приклад 1. Уявімо собі метод, яки просто запам'ятовує об'єкти з навчальних вибірок і розпізнає нові об'єкти лише тоді, коли вони точно збігаються із об'єктами з навчальної вибірки. У цьому випадку емпіричний ризик дорівнює нулю, але точність розпізнавання інших вибірок теж дорівнює нулю. Навчання — це не лише запам'ятовування, але й узагальнення.

Означення 11. ***Узагальнена здатність методу*** μ характеризується мінімальною кількістю помилок, що робить метод на простих навчальних і контрольних вибірках, отриманих з однієї генеральної сукупності X.

Означення 12. Метод навчання μ називається ***слушним***, якшо при заданих достатньо малих числах ε і η ймовірність того, що узагальнена здатність методу більше ε, менше η.

Означення 13. Параметр ε називається ***точністю методу*** μ, а параметр $1–\eta$ — його надійністю. Отримання оцінок типу (1) є основною задачею статистичної теорії навчання.

Коли неможливо отримати теоретичні, застосовуються емпіричні оцінки. Нехай дано вибірку $S = \{(x_i, \omega_i)\}_{i=1}^{M}$. Розіб'ємо її N способами на диз'юнктні підвибірки: навчальну $T_j = \{(x_i, \omega_i)\}_{i=1}^{m}$ і контрольну $K_j = \{(x_i, \omega_i)\}_{i=1}^{n}$, де $n + m = M$.

Означення 14. Для кожного розбиття $j = 1, 2, ..., N$ побудуємо алгоритм $a_j = \mu(T_j)$ і обчислимо кількість помилок. Середня арифметична кількість помилок по всіх розбиттях називається ***оцінкою кросс-валідації***.

Глава 2

Байєсівський метод класифікації

2.1. Байєсівська класифікація з мінімальною ймовірністю помилок

Одним з найбільш популярних підходів до класифікації є байєсівський метод, описаний в численних монографіях, наприклад [5, 14, 3, 11]. Нехай $\omega_1, \omega_2, ..., \omega_N$ — мітки класів $C_1, C_2, ..., C_N$ з відомими *апріорними* ймовірностями $p(\omega_1), p(\omega_2), ..., p(\omega_N)$. Метою бійєсівської класифікації є мінімізація помилок на підставі інформації про щільності розподілів кожного класу. Для цього необхідно знайти для заданого об'єкта x найбільш правдоподібний клас, тобто знайти мітку ω_i, таку що виконується умова

$$p(\omega_i|\, x) > p(\omega_j|\, x) \;\; \forall i \neq j, j = 1, ..., N. \tag{2.1}$$

Інакше кажучи, вирішальне правило відносить об'єкт x до класу C_i, якщо *апостеріорна* ймовірність класу C_i для цього об'єкта є максимальною. Відповідно до цього правила простір ознак Ω розділяється на N областей $\Omega_1, \Omega_2, ..., \Omega_N$, якщо вектор ознак x належить області Ω_i, то він належить класу C_i.

Отже, задача полягає в тому, щоб обчислити апостеріорні ймовірності класів, знаючи їх апріорні ймовірності. Для цього використовується формула Байєса.

Означення 15. *Формула Байєса:*

$$p(\omega_i|\, x) = \frac{p(x|\omega_j)\, p(\omega_i)}{p(x)} \;\; \forall i \neq j, j = 1, ..., N. \tag{2.2}$$

Рівняння (2.2) дозволяє переписати вирішувальне правило (2.1) як правило Байєса з мінімальною ймовірністю помилки.

Означення 16. *Правило Байєса з мінімальною ймовірністю помилки*

$$p(x|\, \omega_i) p(\omega_i) > p(x|\, \omega_j) p(\omega_j) \;\; \forall i \neq j, j = 1, ..., N. \tag{2.3}$$

У випадку бінарної класифікації можна обчислити *відношення правдоподібності*:

$$l(x) = \frac{p(x|\omega_1)}{p(x|\omega_2)}. \tag{2.4}$$

У такому випадку вирішувальне правило набуває такого вигляду.

$$l(x) = \frac{p(x|\omega_1)}{p(x|\omega_2)} > \frac{p(\omega_1)}{p(\omega_2)}. \tag{2.5}$$

Покажемо, що байєсівська класифікація дійсно мінімізує ймовірність помилки [11].

Теорема 1. *Правило Байєса є оптимальним, тобто воно мінімізує ймовірність помилки.*

Доведення. Повернемось до загального випадку з класами $C_1, C_2, ..., C_N$ і запишемо ймовірність помилки. Позначимо подію, що означає помилкову класифікацію як e, а ймовірність помилкової класифікації об'єкта x з класу ω_i як $p(e|\omega_i)$.

$$p(e) = \sum_{i=1}^{N} p(e|\,\omega_i) p(\omega_i)\,. \tag{2.6}$$

Позначимо через $\Omega \backslash \Omega_i$ доповнення області Ω_i. Тоді ймовірність помилкової класифікації виражається формулою

$$p(e|\,\omega_i) = \int\limits_{\Omega \backslash \Omega_i} p(x|\,\omega_i) dx. \tag{2.7}$$

Формула (2.7) дозволяє записати ймовірність помилок як

$$\begin{aligned} p(e) &= \sum_{i=1}^{N} \int\limits_{\Omega \backslash \Omega_i} p(x|\omega_i)\, p(\omega_i)\, dx = \\ &= \sum_{i=1}^{N} p(\omega_i) \left(1 - \int\limits_{\Omega_i} p(x|\omega_i)\, dx \right) = \\ &= 1 - \sum_{i=1}^{N} p(\omega_i) \int\limits_{\Omega_i} p(x|\omega_i)\, dx. \end{aligned}$$

Звідси випливає, що розбиття області Ω на області Ω_i, $1 = 1, ..., N$, що мінімізує ймовірність помилки, еквівалентне максимізації величини $\sum\limits_{i=1}^{N} p(\omega_i) \int\limits_{\Omega_i} p(x|\omega_i)\, dx$.

Це означає, що мінімізація ймовірності помилки еквівалентна максимізації ймовірності правильної класифікації. Позначимо ймовірність правильної класифікації об'єкта x з класу C_j як q. Для того щоб вона була максимальною, слід знайти таку область Ω_i, де величина $p(\omega_i)\, p(x|\omega_i)$ є максимальною.

$$q = \int_{\Omega} \max_i p(\omega_i)\, p(x|\omega_i)\, dx.$$

Отже, ймовірність помилки байєсівської класифікації дорівнює

$$p(e) = 1 - q = 1 - \int_{\Omega} \max_i p(\omega_i)\, p(x|\omega_i)\, dx.$$

Що і потрібно було довести. □

2.2. Байєсівська класифікація з мінімальним середнім ризиком

В попередньому підрозділі ми розглянули задачу мінімізації помилкової класифікації, нехтуючи тим фактом, що помилкова класифікація може бути пов'язана із певними витратами, що залежать від класу. Наприклад, з одного боку, якщо здорова людина буде класифікована як хвора, вона може понести витрати на непотрібні ліки або навіть на лікування від їх побочних наслідків, а з іншого боку, класифікація хворої людини як здорової може призвести до дуже серйозних ускладнень. Отже, необхідно розв'язати задачу мінімізації середнього ризику за допомогою байєсівського підходу.

Уведемо в розгляд величину λij — вартість наслідків неправильної класифікації об'єкта, що належить класу C_i, коли його відносять до класу C_j.

Означення 17. ***Умовним ризиком*** класифікації об'єкта x до класу C_j називається функціонал

$$\mathcal{L}_j(x) = \sum_{i=1}^{N} \lambda_{ij} p(\omega_i|x)\, dx.$$

Означення 18. ***Середнім ризиком*** класифікації об'єкта x до класу C_j називається функціонал

$$\mathcal{R}_j(x) = \int_{\Omega_j} \mathcal{L}_j(x)\, p(x)\, dx = \int_{\Omega_j} \sum_{i=1}^{N} \lambda_{ij} p(\omega_i|x)\, p(x)\, dx.$$

Означення 19. ***Ризиком*** класифікації об'єкта x називається функціонал

$$\mathcal{R}(x) = \sum_{j=1}^{N} \mathcal{R}_i(x) = \sum_{j=1}^{N} \int_{\Omega_j} \sum_{i=1}^{N} \lambda_{ij} p(\omega_i|x)\, p(x)\, dx.$$

Вирішальне правило формулюється так, щоб мінімізувати ризик на вибраних областях Ω_i. Якщо

$$\int_{\Omega_i} \sum_{j=1}^{N} \lambda_{ij} p(\omega_j|x)\, p(x)\, dx < \int_{\Omega_i} \sum_{j=1}^{N} \lambda_{kj} p(\omega_j|x)\, p(x)\, dx, \quad k = 1, 2, ..., N,$$

то $x \in \Omega_i$.

Таким чином, задача зводиться до пошуку областей Ω_i, на яких досягається мінімум ризику:

$$\int_{\Omega} \min_i \sum_{j=1}^{N} \lambda_{ij} p(\omega_j|x)\, p(x)\, dx.$$

2.3. Оцінка щільності розподілу

В обох варіантах байєсівської класифікації — з мінімізацією ймовірності помилок і ризику відповідно — ми вважали відомими апріорні ймовірності класів $p(\omega_i)$ і умовні ймовірності $p(x|\omega_i)$. На практиці апріорні ймовірності класів $p(\omega_i)$, вважаються відомими, а от умовні ймовірності оцінюються на підставі спостережень — багатовимірних вибірок $X_i = (x_{ij})$, $i = 1, ..., N; j = 1, ..., n_i$.

Таким чином, постає проблема оцінювання невідомої щільності умовної ймовірності. Для цього існують два підходи: параметричний і непараметричний.

2.3.1. Параметрична оцінка щільності розподілу

Параметричний підхід полягає у припущенні, що умовна щільність має певний відомий вигляд $p(x, \theta)$ (наприклад, є рівномірною або нормальною), але з невідомим параметром θ. Цей параметр обчислюється за спостереженнями і наближається числом $\hat{\theta}$.

Приклад 2. Припустимо, що

$$p(x|\omega_i) = \frac{1}{(2\pi)^{\frac{p}{2}}} \exp\left\{ -\frac{1}{2} (x - \mu_i)^T det\left(\Sigma_i^{-1}\right) (x - \mu_i) \right\},$$

де μ_i — математичне сподівання класу C_i, Σ_i — коваріаційна матриця класу C_i.

Об'єкт x належить до класу C_i, якщо на цьому класі мінімізується функція

$$\log\left(p\left(\omega_i|x\right)\right) = \log\left(p\left(x|\omega_i\right)\right) + \log\left(p\left(\omega_i\right)\right) - \log\left(p\left(x\right)\right) = \tag{2.8}$$

$$= -\frac{1}{2}\left(x-\mu_i\right)^T det\left(\Sigma_i^{-1}\right)\left(x-\mu_i\right) - \tag{2.9}$$

$$-\frac{1}{2}\log\left(det\left(\Sigma_i^{-1}\right)\right) - \frac{p}{2}\log\left(2\pi\right) + \log\left(p\left(\omega_i\right)\right) - \log\left(p\left(x\right)\right). \tag{2.10}$$

Зважаючи на те, що рішення повинне залежати від класу, ми можемо знехтувати доданками, що не залежать від класу. В результаті отримуємо функцію

$$g_i\left(x\right) = \log\left(p\left(\omega_i\right)\right) - \frac{1}{2}\log\left(det\left(\Sigma_i^{-1}\right)\right) - \frac{1}{2}\left(x-\mu_i\right)^T det\left(\Sigma_i^{-1}\right)\left(x-\mu_i\right).$$

Вирішальне правило у такому *нормальному дискримінантному аналізі* формулюється так: якщо $g_i\left(x\right) > g_j\left(x\right)\ \forall i \neq j$, то $x \in C_i$.

Маючи навчальну вибірку об'єктів з кожного класу, ми можемо оцінити математичне сподівання і коваріаційну матрицю класу.

$$m = \frac{1}{n_i}\sum_{k=1}^{n_i} x_{ik},$$

$$\Sigma_i = \frac{1}{n_i}\sum_{k=1}^{n_i}\left(x_{ik}-m\right)\left(x_{ik}-m\right)^T.$$

Зауважимо, що в загальному випадку задача пошуку параметрів багатовимірного розподілу може стати дуже складною, тому на практиці часто використовують так званий "наївний"баєсівській підхід, який полягає у припущенні, що всі ознаки об'єктів незалежними і однаково розподіленими випадковими величинами. Звісно, таке припущення є занадто сильним (тому підхід і називають наївним), але воно спрощує задачу відновлення щільності розподілу, зводячи її до одновимірного випадку. В наступному падрозділі ми розглянемо найбільш загальний спосіб розв'язання цієї задачі, в якому немає припущення щодо виду розподілу.

2.3.2. Непараметрична оцінка щільності розподілу

Розглянемо класичну оцінку Парзена [14] для одновимірної щільності ймовірності. Нехай $x_1, x_2, ..., x_N$ — незалежні і однаково розподілені випадкові величини. Оцінимо функцію їх розподілу.

$$\hat{P}_N\left(x\right) = \frac{\sharp\left\{x_k \leq x\right\}}{N}. \tag{2.11}$$

Оцінку щільності ймовірності запишемо відповідно до її означення.

$$\hat{p}_N(x) = \frac{\hat{P}_N(x+h) - \hat{P}_N(x-h)}{2h}. \tag{2.12}$$

Виходячи із властивостей щільності розподілу, необхідно, щоб величина h прямувала до 0, але оптимальний вибір швидкості її збіжності повинен бути узгодженим із статистичними властивостями оцінки.

Перепишемо оцінку (2.12) в такий спосіб.

$$\begin{aligned}\hat{p}_N(x) &= \frac{1}{2h}\int\limits_{x-h}^{x+h} d\hat{P}_N(x) = \\ &= \int\limits_{-\infty}^{\infty} \frac{1}{h} K\left(\frac{x-\xi}{h}\right) d\hat{P}_N(\xi) = \\ &= \frac{1}{hN}\sum_{i=1}^{N} K\left(\frac{x-x_i}{h}\right),\end{aligned} \tag{2.13}$$

де

$$K(y) = \begin{cases} \frac{1}{2}, & \text{якщо } y \le 1, \\ 0, & \text{якщо } y > 1. \end{cases} \tag{2.14}$$

Означення 20. Функція $K(y)$ називається *ядром* оцінки щільності розподілу.

Залишається з'ясувати, як вибрати величину h (ширину вікна) і які властивості повинне мати ядро. Будемо вимагати, щоб оцінка щільності розподілу була асимптотично незсунутою і слушною, тобто математичне сподівання оцінки (2.11) при $N \to \infty$ прямувало до щільності розподілу (асимптотична незсунутість), а сама оцінка (2.11) прямувала до щільності розподілу при $N \to \infty$ за ймовірністю. Зформулюємо умови, що забезпечують ці властивості [14].

1. $\int\limits_{-\infty}^{\infty} K(z)\, dz = 1,$
2. $\int\limits_{-\infty}^{\infty} |K(z)|dz < \infty,$
3. $\sup\limits_{-\infty<z<\infty} |K(z)| < \infty,$
4. $\lim\limits_{h\to 0} |zK(z)| = 0.$

Існує багато ядер, що задовольняють цю властивість. Деякі з них наведені в [14], [11], [2]. Всі вони мають графік, що має максимум в точці x, який спадає в її околі до нуля. Форма ядра та швидкість спадання, власне, обираються з огляду на бажані властивості їхньої гладкості. Площа фігури, обмеженою цим графіком та віссю Ox, дорівнює одиниці. Наприклад, ядро Парзена (2.14) має прямокутний графік.

Прямокутне ядро.

$$K(y) = \begin{cases} \frac{1}{2}, & \text{якщо } y \leq 1, \\ 0, & \text{якщо } y > 1. \end{cases} \tag{2.15}$$

Трикутне ядро.

$$K(y) = \begin{cases} 1 - |x|, & \text{якщо } y \leq 1, \\ 0, & \text{якщо } y > 1. \end{cases} \tag{2.16}$$

Квартичне ядро.

$$K(y) = \begin{cases} \frac{15}{16}\left(1 - x^2\right)x^2, & \text{якщо } y \leq 1, \\ 0, & \text{якщо } y > 1. \end{cases} \tag{2.17}$$

Гауссово ядро.

$$K(y) = \begin{cases} \frac{1}{\sqrt{((\pi))}} \exp(\left(-\frac{x^2}{2}\right)), & \text{якщо } y \leq 1, \\ 0, & \text{якщо } y > 1. \end{cases} \tag{2.18}$$

Ядро Єпанечникова.

$$K(y) = \begin{cases} \frac{3}{4}\frac{\left(1-\frac{x^5}{5}\right)}{\sqrt{((5))}}, & \text{якщо } y \leq \sqrt{5}, \\ 0, & \text{якщо } y > 1. \end{cases} \tag{2.19}$$

Універсального рецепту вибору ширини вікна Парзена h не існує, оскільки вона залежить від вибірки, зокрема, від щільності розташування точок цієї вибірки на числовій вісі. Отже, для оптимального вибору вікна Парзена проводяться додаткові дослідження, як правило, методом ковзного контролю [2].

2.4. Застосування наївного байєсівського підходу

Розглянемо задачу класифікації текстів, які виникають в теорії інформаційного пошуку, наприклад, при фільтрації спама. Розглянемо два варіанти наївного методу Байєса: мультиноміальну модель і модель Бернуллі. Припустимо, що навчальна вибірка складається з текстів у вигляді неупорядкованих векторів термінів із заданого лексікона, які можуть повторюватися. Оскільки

розподіл апостеріорної і умовної ймовірності у цій задачі є дискретним, усі обчислення стають елементарними.

1. transfer, greetings, transfer — спам
2. transfer, transfer, diplomat — спам
3. transfer, winner, telephone — спам
4. transfer, name, information — спам
5. credit, card, transfer — не спам

Припустимо,що ми отримали лист s, що містить терміни transfer, transfer, transfer, credit, card. Застосуємо наївний байєсівський підхід.

Мультиноміальна байєсівська модель

У цій моделі ми будемо шукати найбільш ймовірний клас $\omega_i, i = 1,2$ (ω_1 — спам, ω_2 — не спам) для об'єкту x, який подається як вектор термінів. Для цього необхідно знайти клас, на якому апостеріорна ймовірність $\hat{P}(\omega_i|x)$ досягає максимуму:

$$\arg\max_{i=1,2} P(\omega_i|\, x) = \arg\max_{i=1,2} P(\omega_i) \prod_{k=1}^{n} P(t_k|\omega_i)\,.$$

Тут $P(t_k|\omega_1)$ — умовна імовірність появи терміну t_k у листі-спамі, $P(t_k|\omega_2)$ — умовна імовірність появи терміну t_k у листі-спамі, $P(\omega_1)$ — апріорна імовірність спаму, $P(\omega_2)$ — апріорна імовірність звичайного листа. Ймовірність спаму можна або обчислити за навчальною вибіркою, або використати загальні оцінки, що були отримані фахівцями (наприклад, 0,5 або навіть 0,8). Ми будемо використовувати оцінку за навчальними вибірками.

Оскільки в нашому прикладі з 5 навчальних вибірок 4 належать до спаму, маємо:

$$\hat{P}(\omega_1) = \frac{4}{5}, \hat{P}(\omega_2) = \frac{1}{5}.$$

Якщо в тестовій вибірці зустрічається термін, якого немає в навчальних вибірках, його умовна ймовірність дорівнює нулю. Для того щоб не відкидати листи із рідко вживаними словами, можно використати *згладжування Лапласа*, додавши одиницю до частоти кожного терміну. В такому випадку умовна ймовірність терміну t в класі ω обчислюється так:

$$\hat{P}(t|\omega) = \frac{T_{\omega t} + 1}{\sum\limits_{t' \in L} T_{\omega t'} + M},$$

де M — кількість різних термінів в лексиконі.

Оцінимо умовну ймовірність $\hat{P}(t|\omega)$ для кожного класу. В навчальних вибірках, що класифіковані як спам, міститься 12 слів (нагадуємо, що вони можуть повторюватися), з них слово *transfer* зустрічається 6 разів, в слова *credit* і *card* зовсім не зустрічаються. Лексікон містить 9 різних термінів. Отже, з урахуванням згладжування Лапласа, маємо:

$$\hat{P}(transfer|\omega_1) = \frac{6+1}{12+9} = \frac{7}{21} = \frac{1}{3},$$

$$\hat{P}(credit|\omega_1) = \frac{0+1}{12+9} = \frac{1}{21},$$

$$\hat{P}(card|\omega_1) = \frac{0+1}{12+9} = \frac{1}{21}.$$

В навчальній вибірці, що класифікована як звичайний текст, міститься 3 слова, з них слова *transfer*, *credit* і *card* зустрічаються по одному разу. Отже:

$$\hat{P}(transfer|\omega_2) = \frac{0+1}{3+9} = \frac{1}{12},$$

$$\hat{P}(credit|\omega_2) = \frac{0+1}{3+9} = \frac{1}{12},$$

$$\hat{P}(card|\omega_2) = \frac{0+1}{3+9} = \frac{1}{12}.$$

Тепер ми можемо обчислити апостеріорні ймовірності кожного класу.

$$\hat{P}(\omega_1|s) = \frac{4}{5}\frac{1}{3}\frac{1}{3}\frac{1}{3}\frac{1}{21}\frac{1}{21} = 6,72 * 10^{-5}.$$

$$\hat{P}(\omega_2|s) = \frac{1}{5}\left(\frac{1}{12}\right)^5 = 8,04 * 10^{-7}.$$

Як бачимо, ймовірність того, що тестовий лист належить до спаму, значно вища, за ймовірність протилежної події. Відповідно до вирішального правила наївного байєсівського підходу, доходимо висновку, що ми отримали спам.

2.4.1. Модель Бернуллі

Друга модель наївного байєсівського класифікатора — модель Бернуллі — еквівалентна бінарній моделі, що генерує індикатор для кожного терміну словника: 1, якщо термін є присутнім у документі, і 0, якщо відсутнім. У моделі Бернуллі імовірність $\hat{P}(t|\omega)$ оцінюється як частка об'єктів із класу ω, що мають ознаку t. На противагу йому в мультиноміальній моделі імовірність $\hat{P}(t|\omega)$ оцінюється як частка ознаки t в об'єктах з класу ω. На відміну від мультиноміальної моделі при класифікації тестового документа на основі

моделі Бернуллі головним є факт наявності ознаки, а кількість її входжень ігнорується. Це є слабкістю моделі Бернуллі, оскільки вона може ураховувати рідкі терміни як значущі. З іншого, боку, в мультиноміальній моделі ніяк не враховується інформація про терміни, відсутні в тестовому документі, але в моделі Бернуллі, яка для кожного терміну обчислює індикатор 0 або 1, ця інформація має вагу.

Застосуємо модель Бернуллі для класифікації попереднього листа на підставі навчальних вибірок, наведених вище.

Апріорні ймовірності класів не змінюються.

$$\hat{P}(\omega_1) = \frac{4}{5}, \hat{P}(\omega_2) = \frac{1}{5}.$$

Обчислимо умовні імовірності в моделі Бернуллі (не забуваймо про згладжування Лапласа). Слово *transfer* зустрічається в усіх чотирьох навчальних вибірках, а слова *greetings, diplomat, winner, telephone, name, information* — по одному разу. Оскільки клас ω_1 містить 4 вибірки, а величина M дорівнює двом (термін або присутній, або відсутній), маємо:

$$\hat{P}(transfer|\omega_1) = \frac{4+1}{4+2} = \frac{5}{6},$$

$$\hat{P}(credit|\omega_1) = \frac{0+1}{4+2} = \frac{1}{6},$$

$$\hat{P}(card|\omega_1) = \frac{0+1}{4+2} = \frac{1}{6}.$$

$$\hat{P}(greetings|\omega_1) = \frac{1+1}{4+2} = \frac{2}{3},$$

$$\hat{P}(diplomat|\omega_1) = \frac{1+1}{4+2} = \frac{2}{3},$$

$$\hat{P}(winner|\omega_1) = \frac{1+1}{4+2} = \frac{2}{3},$$

$$\hat{P}(telephone|\omega_1) = \frac{1+1}{4+2} = \frac{2}{3},$$

$$\hat{P}(name|\omega_1) = \frac{1+1}{4+2} = \frac{2}{3},$$

$$\hat{P}(information|\omega_1) = \frac{1+1}{4+2} = \frac{2}{3},$$

Клас ω_2 містить одну вибірку :

$$\hat{P}\left(transfer|\omega_2\right) = \frac{1+1}{1+2} = \frac{2}{3},$$

$$\hat{P}\left(credit|\omega_2\right) = \frac{1+1}{1+2} = \frac{2}{3},$$

$$\hat{P}\left(card|\omega_2\right) = \frac{1+1}{1+2} = \frac{2}{3}.$$

$$\hat{P}\left(greetings|\omega_1\right) = \frac{0+1}{1+2} = \frac{1}{3},$$

$$\hat{P}\left(diplomat|\omega_1\right) = \frac{0+1}{1+2} = \frac{1}{3},$$

$$\hat{P}\left(winner|\omega_1\right) = \frac{0+1}{1+2} = \frac{1}{3},$$

$$\hat{P}\left(telephone|\omega_1\right) = \frac{0+1}{1+2} = \frac{1}{3},$$

$$\hat{P}\left(name|\omega_1\right) = \frac{0+1}{1+2} = \frac{1}{3},$$

$$\hat{P}\left(information|\omega_1\right) = \frac{0+1}{1+2} = \frac{1}{3},$$

Обчислимо апостеріорні ймовірності кожного класу.

$$\begin{aligned}\hat{P}\left(\omega_1|s\right) = \hat{P}\left(\omega_1\right) * \hat{P}\left(transfer|\omega_1\right) * \\ * \hat{P}\left(credit|\omega_1\right) * \\ * \hat{P}\left(card|\omega_1\right) * \\ * \left(1 - \hat{P}\left(greetings|\omega_1\right)\right) * \\ * \left(1 - \hat{P}\left(diplomat|\omega_1\right)\right) * \\ * \left(1 - \hat{P}\left(winner|\omega_1\right)\right) * \\ * \left(1 - \hat{P}\left(telephone|\omega_1\right)\right) * \\ * \left(1 - \hat{P}\left(name|\omega_1\right)\right) * \\ * \left(1 - \hat{P}\left(information|\omega_1\right)\right).\end{aligned} \tag{2.20}$$

Отже,

$$\hat{P}\left(\omega_1|s\right) = \frac{4}{5}\frac{5}{6}\frac{1}{6}\frac{1}{6}\left(\frac{2}{3}\right)^6 = 0,0016.$$

$$\begin{aligned}\hat{P}(\omega_2|s) = \hat{P}(\omega_2) * \hat{P}(transfer|\omega_2) * \\ * \hat{P}(credit|\omega_2) * \\ * \hat{P}(card|\omega_2) * \\ * \left(1 - \hat{P}(greetings|\omega_2)\right) * \\ * \left(1 - \hat{P}(diplomat|\omega_2)\right) * \\ * \left(1 - \hat{P}(winner|\omega_2)\right) * \\ * \left(1 - \hat{P}(telephone|\omega_2)\right) * \\ * \left(1 - \hat{P}(name|\omega_2)\right) * \\ * \left(1 - \hat{P}(information|\omega_2)\right).\end{aligned} \tag{2.21}$$

Отже,

$$\hat{P}(\omega_2|s) = \frac{1}{5}\left(\frac{2}{3}\right)^5\left(\frac{1}{3}\right)^6 = 3,61 * 10^{-5}.$$

Як і при використанні мультиноміальної моделі, ймовірність того, що тестовий лист належить до спаму, значно перевищує ймовірність того, що ми отримали звичайний лист. Отже, це спам.

Якщо використання наївного байєсівського класифікатора передбачає дві можливі моделі — мультиноміальну і Бернуллі — виникає питання, як правильно вибрати потрібну модель. З точки зору сили припущень, які лежать в основі моделі, вони еквівалентні — припускається, що всі ознаки є незалежними і ймовірність їх появи на будь-якій позиції однакова. Обидві моделі дають дуже неточну оцінку справжньої умовної ймовірності термінів. Втім, незважаючи на це, вони є достатньо точними щодо рішення про класифікацію [12]. Отже, на передній план виходить їх ефективність при роботі з різними обсягами вибірок і ознак. З цього боку вони розрізняються: мультиноміальна модель краще працює з довгими вибірками і великою кількістю ознак, а модель Бернуллі — з короткими вибірками і невеликою кількістю ознак.

Глава 3

Лінійний дискримінант Фішера

Однією з основних проблем є залежність складності алгоритму розпізнавання від розмірності простору ознак. Що більше розмірність простору ознак, то більше складність алгоритму. З цієї причини бажано зменшити простір ознак, в ідеалі, до числової прямої. Саме ця ідея покладена в основу методу, основаного на використанні лінійної дискримінантної функції Фішера.

Розглянемо дві множини навчальних вибірок з p-вимірного простору: $X_1, X_2, \ldots, X_{n_1}$ і $X_{n_1+1}, X_{n_1+2}, \ldots, X_{n_1+n_2}$. Вважатимемо, що $X_1, X_2, \ldots, X_{n_1}$, $X_{n_1+1}, X_{n_1+2}, \ldots, X_{n_1+n_2}$ — вектори чисел в просторі розмірності p. Дискримінантний метод Фішера полягає у проектуванні цих векторів із простору розмірності p на числову пряму за допомогою лінійної функції $l(X) = w^T X$ й відокремленні двох генеральних сукупностей якомога далі одна від одної за допомогою вектора w з простору розмірності p.

Необхідно знайти вектор $\hat{w}$, що максимізує функціонал $J(w)$, де

$$J(w) = \frac{\left(\bar{Y}_1 - \bar{Y}_2\right)^2}{S_Y^2},\ \bar{Y}_1 = \frac{\sum\limits_{i=1}^{n_1} Y_i}{n_1},\ \bar{Y}_2 = \frac{\sum\limits_{i=n_1+1}^{n_1+n_2} Y_i}{n_2},$$

$$S_Y^2 = \frac{\sum\limits_{i=1}^{n_1} (Y_i - \bar{Y}_1)^2 + \sum\limits_{i=n_1+1}^{n_1+n_2} (Y_i - \bar{Y}_2)^2}{n_1 + n_2 - 2},$$

$$Y_i = w^T \vec{X}_i = \left(w, \vec{X}_i\right),\ \ i = 1, 2, \ldots, n_1 + n_2$$

З інтуїтивної точки зору функція критерію $J(w) = \frac{\left(\bar{Y}_1 - \bar{Y}_2\right)^2}{S_Y^2}$ оцінює різницю між середніми проекцій $\bar{Y}_1 - \bar{Y}_2$ відносно до стандартного відхилення S_Y. Якщо проекції $Y_1, Y_2, \ldots, Y_{n_1}$ і $Y_{n_1+1}, Y_{n_1+2}, \ldots, Y_{n_1+n_2}$ можна відокремити повністю, то величина $\left(\bar{Y}_1 - \bar{Y}_2\right)^2$ повинна бути великою відносно до середнього відхилення S_Y.

Теорема 2. *Вектор $\hat{w}$, що максимізує величину $J(w) = \frac{\left(\bar{Y}_1 - \bar{Y}_2\right)^2}{S_Y^2}$, має вигляд $S_W^{-1}\left(\bar{X}_1 - \bar{X}_2\right)$, де*

$$S_W = \frac{(n_1 - 1) S_1 + (n_2 - 1) S_2}{n_1 + n_2 - 2},$$
$$S_1 = \frac{\sum\limits_{i=1}^{n_1} \left(X_i - \bar{X}_1\right)\left(X_i - \bar{X}_1\right)^T}{n_1 - 1},$$
$$S_2 = \frac{\sum\limits_{i=n_1+1}^{n_1+n_2} \left(X_i - \bar{X}_2\right)\left(X_i - \bar{X}_2\right)^T}{n_2 - 1},$$
$$\bar{X}_1 = \frac{\sum\limits_{i=1}^{n_1} X_i}{n_1},\ \bar{X}_2 = \frac{\sum\limits_{i=n_1+1}^{n_1+n_2} X_i}{n_2}.$$

Доведення. Запишемо середнє вибіркове значення проекції першої вибірки як

$$\bar{Y}_1 = \frac{1}{n_1}\sum_{i=1}^{n_1} Y_i = \frac{1}{n_1}\sum_{i=1}^{n_1}\left(w, \vec{X}_i\right) = \left(w, \frac{1}{n_1}\sum_{i=1}^{n_1} X_i\right) = \left(w, \bar{X}_1\right).$$

Аналогічно,

$$\bar{Y}_2 = \left(w, \bar{X}_2\right).$$

Отже,

$$\begin{aligned}\sum_{i=1}^{n_1}\left(Y_i - \bar{Y}_1\right)^2 &= \sum_{i=1}^{n_1}\left(w^T\vec{X}_i - w^T\bar{X}_1\right)^2 = \\ &= \sum_{i=1}^{n_1}\left(w^T X_i - w^T\bar{X}_1\right)\left(w^T X_i - w^T\bar{X}_1\right)^T = \\ &= \sum_{i=1}^{n_1} w^T\left(X_i - \bar{X}_1\right)\left(X_i - \bar{X}_1\right)^T w = \\ &= w^T\left[\sum_{i=1}^{n_1}\left(X_i - \bar{X}_1\right)\left(X_i - \bar{X}_1\right)^T\right] w.\end{aligned}$$

Крім того,

$$\sum_{i=n_1+1}^{n_1+n_2}\left(Y_i - \bar{Y}_2\right)^2 = w^T\left[\sum_{i=n_1+1}^{n_1+n_2}\left(X_i - \bar{X}_2\right)\left(X_i - \bar{X}_2\right)^T\right] w$$

Таким чином,

$$S_Y^2 = \frac{\sum\limits_{i=1}^{n_1}\left(Y_i - \bar{Y}_1\right)^2 + \sum\limits_{i=n_1+1}^{n_1+n_2}\left(Y_i - \bar{Y}_2\right)^2}{n_1+n_2-2} =$$

$$= \frac{w^T\left[\sum\limits_{i=1}^{n_1}\left(X_i - \bar{X}_1\right)\left(X_i - \bar{X}_1\right)^T\right]w + w^T\left[\sum\limits_{i=n_1+1}^{n_1+n_2}\left(X_i - \bar{X}_2\right)\left(X_i - \bar{X}_2\right)^T\right]w}{n_1+n_2-2} =$$

$$= w^T\left[\frac{\sum\limits_{i=1}^{n_1}\left(X_i - \bar{X}_1\right)\left(X_i - \bar{X}_1\right)^T + \sum\limits_{i=n_1+1}^{n_1+n_2}\left(X_i - \bar{X}_2\right)\left(X_i - \bar{X}_2\right)^T}{n_1+n_2-2}\right]w =$$

$$= w^T\left[\frac{(n_1-1)S_1 + (n_2-1)S_2}{n_1+n_2-2}\right]w = w^T S_W w.$$

Отже,

$$J(w) = \frac{\left(\bar{Y}_1 - \bar{Y}_2\right)^2}{S_Y^2} = \frac{\left(w^T\left(\bar{X}_1 - \bar{X}_2\right)\right)^2}{w^T S_W w}$$

Вектор $\hat{w}$ можна знайти, розв'язавши рівняння

$$\frac{\partial J(w)}{\partial w} = 2\frac{\left(w^T\left(\bar{X}_1 - \bar{X}_2\right)\right)\left(\bar{X}_1 - \bar{X}_2\right)\left(w^T S_W w\right)}{\left(w^T S_W w\right)^2} -$$
$$- \left(w^T\left(\bar{X}_1 - \bar{X}_2\right)\right)^2 \frac{2S_W w}{\left(w^T S_W w\right)^2} = 0.$$

Подальші перетворення приводять до рівняння

$$\left(\bar{X}_1 - \bar{X}_2\right) = \left[\frac{w^T\left(\bar{X}_1 - \bar{X}_2\right)}{w^T S_W w}\right]S_W w.$$

Множачи обидві частині цього рівняння на матрицю S_W^{-1}

$$S_W^{-1}\left(\bar{X}_1 - \bar{X}_2\right) = \left[\frac{w^T\left(\bar{X}_1 - \bar{X}_2\right)}{w^T S_W w}\right]w$$

Оскільки $\frac{w^T\left(\bar{X}_1 - \bar{X}_2\right)}{w^T S_W w}$ — дійсне число, маємо $\hat{w} = cS_W^{-1}\left(\bar{X}_1 - \bar{X}_2\right)$, де c — деяка константа.

Матриця S_W називається *матрицею розкиду всередині класу*. Матриця $S_B = \left(\bar{X}_1 - \bar{X}_2\right)\left(\bar{X}_1 - \bar{X}_2\right)^T$ називається *матрицею розкиду між класами*.

За допомогою матриць S_W і S_B функцію критерію в методі Фішера можна сформулювати за допомогою узагальненого відношення Релея:

$$J(w) = \frac{w^T S_B w}{w^T S_W w}.$$

□

3.0.2. Приклад класифікації

Припустимо, що ми маємо спостереження X_0. Спираючись на дискримінантну функцію $l(X) = \hat{w}^T X$, яку ми знайшли вище, ми можемо віднести це спостереження до певного класу. Вирішальне правило формулюється так: спостереження X_0 належить до генеральної сукупності 1, якщо

$$\hat{Y}_0 = \hat{w}^T X_0 = \left(\bar{X}_1 - \bar{X}_2\right)^T S_W^{-1} X_0 \geq \frac{1}{2}\hat{w}^T(\bar{X}_1 + \bar{X}_2) =$$
$$= \frac{1}{2}\left(\bar{Y}_1 + \bar{Y}_2\right) = \frac{1}{2}\left(\bar{X}_1 - \bar{X}_2\right)^T S_W^{-1}\left(\bar{X}_1 + \bar{X}_2\right).$$

У супротивному випадку

$$\hat{Y}_0 = (\bar{X}_1 - \bar{X}_2)^T S_W^{-1} X_0 < \frac{1}{2}\left(\bar{X}_1 - \bar{X}_2\right)^T S_W^{-1}\left(\bar{X}_1 + \bar{X}_2\right)$$

і спостереження X_0 належить до генеральної сукупності 2.

Якщо $\hat{Y}_0$ менше $\frac{\bar{Y}_1+\bar{Y}_2}{2}$ (ближче до $\bar{Y}_1$), то відносимо X_0 до генеральної сукупності 1 і навпаки.

Зауваження 1. Значне відділення не означає гарної класифікації. З іншого боку, якщо відділення не є значним, то здійснювати класифікацію немає сенсу.

Приклад 3. Розглянемо штучний приклад, в якому дикримінант Фішера легко визначити заздалегідь. Перший клас складається з точок

$$x_1 = \left[\ 1.00,\ \ 1.00\ \right],\ \ x_2 = \left[\ 2.00,\ \ 2.00\ \right],\ \ x_3 = \left[\ 3.00,\ \ 3.00\ \right].$$

Другий клас складається з точок

$$x_4 = \left[\ 1.00,\ \ 2.00\ \right],\ \ x_5 = \left[\ 2.00,\ \ 3.00\ \right],\ \ x_6 = \left[\ 3.00,\ \ 4.00\ \right].$$

Обчислимо коваріаційні матриці першого і другого класів, а також матрицю розкиду всередині класу.

$$S_1 = \begin{bmatrix} 0.667 & 0.667 \\ 0.667 & 0.917 \end{bmatrix},$$

$$S_2 = \begin{bmatrix} 0.667 & 0.667 \\ 0.667 & 0.917 \end{bmatrix},$$

$$S_W = \begin{bmatrix} 0.667 & 0.667 \\ 0.667 & 0.917 \end{bmatrix}.$$

Обернена матриця розкиду всередині класу має вигляд.

$$S_W^{-1} = \begin{bmatrix} 5.50 & -4.00 \\ -4.00 & 4.00 \end{bmatrix}.$$

Отже,

$$\hat{w} = S_W^{-1}(\bar{x}_1 - \bar{x}_2) = \begin{bmatrix} 5.50 & -4.00 \\ -4.00 & 4.00 \end{bmatrix} \left(\begin{bmatrix} 2.00 \\ 2.00 \end{bmatrix} - \begin{bmatrix} 2.00 \\ 3.00 \end{bmatrix} \right) = \begin{bmatrix} 4.00 \\ -4.00 \end{bmatrix}.$$

Звідси випливає, що

$$\hat{y} = \hat{w}^T x = 4x_1 - 4x_2.$$

Отже,

$$y_1 = \hat{w}^T x_1 = \begin{bmatrix} 4.00 & -4.00 \end{bmatrix} \begin{bmatrix} 1.00 \\ 1.00 \end{bmatrix} = 0.00,$$
$$y_2 = \hat{w}^T x_2 = \begin{bmatrix} 4.00 & -4.00 \end{bmatrix} \begin{bmatrix} 2.00 \\ 2.00 \end{bmatrix} = 0.00,$$
$$y_3 = \hat{w}^T x_3 = \begin{bmatrix} 4.00 & -4.00 \end{bmatrix} \begin{bmatrix} 3.00 \\ 3.00 \end{bmatrix} = 0.00.$$

$$\bar{Y}_1 = \frac{y_1 + y_2 + y_3}{3} = 0.00,$$

$$y_4 = \hat{w}^T x_4 = \begin{bmatrix} 4.00 & -4.00 \end{bmatrix} \begin{bmatrix} 1.00 \\ 2.00 \end{bmatrix} = -4.00,$$
$$y_5 = \hat{w}^T x_5 = \begin{bmatrix} 4.00 & -4.00 \end{bmatrix} \begin{bmatrix} 2.00 \\ 3.00 \end{bmatrix} = -4.00$$
$$y_6 = \hat{w}^T x_6 = \begin{bmatrix} 4.00 & -4.00 \end{bmatrix} \begin{bmatrix} 3.00 \\ 4.00 \end{bmatrix} = -4.00,$$

$$\bar{Y}_2 = \frac{\bar{y}_4 + \bar{y}_5 + \bar{y}_6}{3} = -4.00.$$

Таким чином,

$$\frac{\bar{Y}_1 + \bar{Y}_2}{2} = \frac{0.00 - 4.00}{2} = -2.00.$$

Припустимо, що маємо нове спостереження

$$x_0 = \begin{bmatrix} 2.00 \\ 2.70 \end{bmatrix}.$$

Тоді

$$\hat{y}_0 = \hat{w}^T x_0 = \begin{bmatrix} 4.00 & -4.00 \end{bmatrix} \begin{bmatrix} 2.00 \\ 2.70 \end{bmatrix} = 8.00 - 10.80 = -2.80 < -2.00.$$

Отже, спостереження належить до першої генеральної сукупності.

3.1. Нелінійний дискримінант Фішера

Цікаво, що лінійний дискримімант Фішера був винайдений у 1936 р. [17], а його узагальнення на нелінійні випадки — лише у 1999 році [18]. Якщо множини точок з навчальної вибірки не допускають лінійне розділення, можна скористатися відображення вихідного простору ознак X у новий простір ознак F за допомогою деякої функції φ, так що $w = \varphi(x)$. У такому випадку задача класифікації зводиться до максимізації функціонала

$$J(\vec{w}) = \frac{\vec{w}^{\mathrm{T}} \vec{S}_B^{\varphi} \vec{w}}{\vec{w}^{\mathrm{T}} \vec{S}_W^{\varphi} \vec{w}}, \tag{3.1}$$

де

$$\vec{S}_B^{\varphi} = (\vec{m}_2^{\varphi} - \vec{m}_1^{\varphi})(\vec{m}_2^{\varphi} - \vec{m}_1^{\varphi})^{\mathrm{T}} \tag{3.2}$$

$$\vec{S}_W^{\varphi} = \sum_{i=1}^{2} \sum_{n=1}^{l_i} (\varphi(\vec{x}_n^i) - \vec{m}_i^{\varphi})(\varphi(\vec{x}_n^i) - \vec{m}_i^{\varphi})^{\mathrm{T}}, \vec{m}_i^{\varphi} = \frac{1}{l_i} \sum_{j=1}^{l_i} \varphi(\vec{x}_j^i). \tag{3.3}$$

Безпосереднє застосування лінійного дискримінанта Фишера у просторі F може виявитися недоречним з огляду на складність обчислень або велику вимірність простору F. Тому, введемо у розгляд ядро $K(\vec{x}, \vec{y}) = \varphi(\vec{x}) \cdot \varphi(\vec{y})$ і скористаємось розвиненням

$$\vec{w} = \sum_{i=1}^{l} \alpha_i \varphi(\vec{x}_i).$$

Зважаючи на те, що

$$\vec{w}^{\mathrm{T}} \vec{m}_i^{\varphi} = \frac{1}{l_i} \sum_{j=1}^{l} \sum_{k=1}^{l_i} \alpha_j K(\vec{x}_j, \vec{x}_k^i) = \vec{\alpha}^{\mathrm{T}} \vec{M}_i,$$

де

$$(\vec{M}_i)_j = \frac{1}{l_i} \sum_{k=1}^{l_i} k(\vec{x}_j, \vec{x}_k^i).$$

Перепишемо чисельник фукнкціонала $J(\vec{w})$ у такий спосіб:

$$\vec{w}^{\mathrm{T}} \vec{S}_B^{\varphi} \vec{w} = \vec{w}^{\mathrm{T}} (\vec{m}_2^{\varphi} - \vec{m}_1^{\varphi})(\vec{m}_2^{\varphi} - \vec{m}_1^{\varphi})^{\mathrm{T}} \vec{w} \tag{3.4}$$

$$= \vec{\alpha}^{\mathrm{T}} \vec{M} \vec{\alpha}, \tag{3.5}$$

де $\vec{M} = (\vec{M}_2 - \vec{M}_1)(\vec{M}_2 - \vec{M}_1)^{\mathrm{T}}$.

Знаменник можна переписати аналогічним чином:

$$\vec{w}^{\mathrm{T}} \vec{S}_W^{\varphi} \vec{w} = \vec{\alpha}^{\mathrm{T}} \vec{N} \vec{\alpha},$$

де

$$\vec{N} = \sum_{j=1}^{2} \vec{K}_j (I - L) \vec{K}_j^{\mathrm{T}},$$

а n-тий і m-тий компоненти $\vec{K}_j$ визначаються як $K(\vec{x}_n, \vec{x}_m^j)$, I — одинична матрицы, і L — матриця, заповнена числами $1/l_j$. Цю тотожність можна вивести, застосувавши розвинення $\vec{w}$ і означення S_W^{φ} та $\vec{m}_i^{\varphi}$ у виразі $\vec{w}^{\mathrm{T}} \vec{S}_W^{\varphi} \vec{w}$.

Зауваження 2. Виконайте цю вправу самостійно!

Перепишемо рівняння для функціонала J як

$$J(\vec{\alpha}) = \frac{\vec{\alpha}^{\mathrm{T}} \vec{M} \vec{\alpha}}{\vec{\alpha}^{\mathrm{T}} \vec{N} \vec{\alpha}}.$$

Застосовуючи умови Ейлера, отримуємо таке рівняння.

$$(\vec{\alpha}^{\mathrm{T}} \vec{M} \vec{\alpha}) \vec{N} \vec{\alpha} = (\vec{\alpha}^{\mathrm{T}} \vec{N} \vec{\alpha}) \vec{M} \vec{\alpha}.$$

Розв'язуючи це рівняння, отримуємо:

$$\vec{\alpha} = \vec{N}^{-1} (\vec{M}_2 - \vec{M}_1).$$

Для того щоб уникнути сингулярності, додамо незначне збурення [18]:

$$\vec{N}_{\epsilon} = \vec{N} + \epsilon \vec{I}.$$

Отже, образом точки $\vec{x}$ є точка

$$\vec{y} = (\vec{w}, \varphi(\vec{x})) = \sum_{i=1}^{l} \alpha_i k(\vec{x}_i, \vec{x}).$$

Глава 4

Метод опорних векторів з жорстким зазором

В двокласовому дискримінантному методі Фішера на площині ми шукали таку пряму, проекція на яку забезпечувала б максимальне розділення точок. Природно поставити нову задачу: знайти таку гіперплощину, яка б розділяла два класи так, що відстань від неї до найближчої точки з кожного класу була максимальною. Ця ідея інтуїтивно зрозуміла — якщо у нас є вибір ліній, які розділяють дві множини точок, то найкращим класифікатором була б лінія, максимально віддалена від цих множин. В такому випадку емпірична помилка класифікації стає мінімальною, а здатність алгоритму до узагальнення стає максимальною.

Припустимо, що N навчальних вибірок з класів C_1 і C_2, які мають мітку ω, що дорівнює 1, якщо $x_i \in C_1$ і $\omega = -1$, якщо $x_i \in C_1$ відповідно, містять n вибіркових значень, тобто $\vec{x}_i = \{x_1, x_2, ..., x_n\}$, $i = 1, ..., N$. Для простоти припустимо, що множини точок є лінійно роздільними і позначимо мітку точки x_i як ω_i.

Розглянемо роздільну лінію

$$g(\vec{x}) = (\vec{w}, \vec{x}) + b \tag{4.1}$$

де $\vec{w}$ — n-вимірний вектор, ортогональний роздільній гіперплощині, b — параметр зсуву цієї гіперплощини і

$$g(\vec{x}_i) = \omega_i, i = 1, ..., N. \tag{4.2}$$

Таким чином, функція $g(x)$ набуває значення 1 на навчальних вибірках з класу C_1 і значення -1 на навчальних вибірках з класу C_2. До того ж, оскільки ми припустили, що множини навчальних вибірок є лінійно роздільними, жодна з точок не лежить на прямій, тобто

$$g(\vec{x}_i) \neq 0, i = 1, ..., N.$$

Уведемо в розгляд важливі поняття, які стануть у нагоді в майбутньому.

Означення 21. Точки, що є найближчими до роздільної гіперплощини, називаються *опорними.*

Означення 22. Величина $\omega\left((\vec{w},\vec{x}_i)+b\right)$ називається *функціональним зазором.*

Означення 23. Відстань від роздільної гіперплощини до найближчої навчальної точки називається *зазором.*

Означення 24. Максимальна ширина смуги, що можна провести паралельно роздільній гіперплощини через опорні точки, називається *геометричним зазором.*

Позначимо євклідову відстань між деякою точкою $\vec{x}_i$ і роздільною гіперплощиною символом r_i. Оскільки найменша відстань між точкою і гіперплощиною визначається перпендикуляром до площини, паралельним вектору $\vec{w}$, то одиничний вектор у цьому напрямку має вигляд $\frac{\vec{w}}{|\vec{w}|}$. Проекція точки $\vec{x}_i$ на роздільну гіперплощину обчислюється так.

$$\vec{x}_i' = \vec{x}_i - \omega_i r\frac{\vec{w}}{|\vec{w}|}.$$

Проекція $\vec{x}_i'$ лежить на гіперплощині і задовольняє рівнянню $(\vec{w},\vec{x}_i')+b=0$. Отже,

$$\left(\vec{w},\vec{x}_i - \omega_i r\frac{\vec{w}}{|\vec{w}|}\right)+b=0 \tag{4.3}$$

Звідси випливає, що

$$r_i = \omega_i\frac{(\vec{w},\vec{x}_i)+b}{|\vec{w}|}. \tag{4.4}$$

Множник ω_i визначає положення точки x_i відносно роздільної гіперплощини: якщо $\omega_i=1$, то $x_i\in C_1$, а якщо $\omega_i=-1$, то $x_i\in C_2$. Множення правої частини рівняння (4.4) на будь-яке додатне число не впливає на точність класифікації, отже, можна для зручності оптимізації провести нормалізацію геометричного зазору так, щоб він не був меншим одиниці на жодній навчальній точці і досягав одиниці на опорних точках. Цього можна досягти, поклавши $|\vec{w}|=1$. Тоді геометричний і функціональний зазори збігаються, і можна сформулювати таке обмеження:

$$\omega_i\left((\vec{w},\vec{x}_i)+b\right)\geq 1. \tag{4.5}$$

Виходячи із рівності (4.5), доходимо висновку, що геометричний зазор дорівнює $\rho=\frac{2}{|\vec{w}|}$

Таким чином, отримуємо наступну задачу оптимізації

$$\arg \max_{\omega_i((\vec{w},\vec{x}_i)+b)\geq 1} \frac{2}{|\vec{w}|}. \tag{4.6}$$

Це еквівалентно такій задач мінімізації:

$$\arg \min_{\omega_i((\vec{w},\vec{x}_i)+b)\geq 1} \frac{(\vec{w},\vec{w})}{2}. \tag{4.7}$$

Задача мінімізації квадратичної функції при лінійних обмеженнях називається *задачею квадратичної оптимізації.*

Стандартним способом розв'язання задачі квадратичної мінімізації є її зведення до двоїстої задачі, у якій кожному обмеженню $\omega_i\left((\vec{w},\vec{x}_i)+b\right)\geq 1$ прямої задачі ставиться у відповідність шуканий множник Лагранжа λ_i. За теоремою Куна-Таккера [6], задача (4.7) еквівалентна опуклій задачі пошуку сідлової точки функції Лагранжа без обмежень.

Двоїста задача без обмежень. Знайти множники Лагранжа λ_i, $i = 1, ..., N$, що задовольняють умову

$$J\left(\vec{w}, b, \vec{\lambda}\right) = \frac{(\vec{w},\vec{w})}{2} - \sum_{i=1}^{N} \lambda_i \left(\omega_i\left((\vec{w},\vec{x}_i)+b\right)-1\right) \to \min_{\vec{w},b} \max_{\lambda_i}. \tag{4.8}$$

Тут $\vec{x}^T\vec{y}$ — це скалярний добуток векторів $\vec{x}$ і $\vec{y}$. Відповідно до теореми Куна-Таккера, розв'язком цієї задачі є сідлова точка. Ця точка задовольняє такі умови.

$$\frac{\partial J\left(\vec{w}, b, \vec{\lambda}\right)}{\partial \omega} = 0, \tag{4.9}$$

$$\frac{\partial J\left(\vec{w}, b, \vec{\lambda}\right)}{\partial b} = 0, \tag{4.10}$$

$$\lambda_i\left(\omega_i\left((\vec{w},\vec{x}_i)+b\right)-1\right) = 0, \quad i = 1, ..., N, \tag{4.11}$$

$$\lambda_i \geq 0 \quad i = 1, ..., N. \tag{4.12}$$

Обчислюючи частинні похідні (4.9, 4.10), маємо

$$\vec{w} = \sum_{i=1}^{N} \lambda_i \omega_i x_i, \tag{4.13}$$

$$\sum_{i=1}^{N} \lambda_i \omega_i = 0. \tag{4.14}$$

Підставляючи ці вирази в (4.8), отримуємо еквівалентну двоїсту задачу відносно множників Лагранжа λ_i.

Двоїста задача відносно множників Лагранжа. Знайти множники Лагранжа λ_i, $i = 1, ..., N$ за яких величина

$$J(\lambda) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_i \lambda_j \omega_i \omega_j (\vec{x}_i, \vec{x}_j) \tag{4.15}$$

досягає максимуму за умов

1. $\sum\limits_{i=1}^{N} \lambda_i \omega_i = 0$;
2. $\lambda_i \geq 0 \;\; i = 1, ..., N$.

Знайшовши множники Лагранжа, ми можемо визначити розв'язок задачі (4.7):

$$\begin{gathered} \vec{w} = \sum_{i=1}^{N} \lambda_i \omega_i \vec{x}_i, \\ b = \omega_k - (\vec{w}, \vec{x}_k) \;\; \text{для таких } x_k, \text{ що } \lambda_k \neq 0. \end{gathered} \tag{4.16}$$

Зважаючи на умову (4.11), легко бачити, що всі множники Лагранжа, які не відповідають опорним точкам, тобто точкам, що задовольняють умову $\omega_i ((\vec{w}, \vec{x}_i) + b) - 1 = 0$, дорівнюють нулю. Тому розмірність задачі оптимізації визначається лише кількістю опорних точок, яка, зазвичай, є невеликою. Врешті, вирішальна функція набуває такого вигляду.

$$g(\vec{x}) = sign \left(\sum_{k=1}^{M} \lambda_k \omega_k (\vec{x}_k, \vec{x}) + b \right), \tag{4.17}$$

де M — кількість опорних точок.

Приклад 4. Для ілюстрації обчислень розглянемо тривіальний приклад з навчальними вибірками, що є числами: точка $x_1 = 0$ з міткою 1 і $x_2 = 1$, $x_3 = 2$ з міткою -1.

Побудуємо двоїсту задачу. Функціонал якості має такий вигляд:

$$J(\lambda) = \lambda_1 + \lambda_2 + \lambda_3 - \frac{1}{2} \left(\lambda_1^2 - 4\lambda_2 \lambda_3 + 4\lambda_3 \right)$$

З огляду на те, що

$$\lambda_1 = \lambda_2 + \lambda_3$$

отримуємо

$$\frac{\partial J(\lambda)}{\partial \lambda_2} = 2 - 2\lambda_2 + 2\lambda_3 = 0, \tag{4.18}$$

$$\frac{\partial J(\lambda)}{\partial \lambda_3} = 2 + 2\lambda_2 - 4\lambda_3 = 0. \tag{4.19}$$

Таким чином, розв'язком двоїстої задачі є трійка

$$\lambda_1 = 2, \lambda_2 = 2, \lambda_3 = 0.$$

Звідси випливає, что опорними векторами є точки $x_1 = 0$ і $x_1 = 1$, а параметри роздільної лінії дорівнюють

$$w = 2, b = 1.$$

Вирішальна функція набуває вигляд:

$$g(x) = -2x + 1.$$

Межа між класами задається виразом

$$g(x) = sign(-2x + 1),$$

тобто роздільною точкою є

$$x = \frac{1}{2}.$$

Досі ми припускали, що точки можна розділити гіперплощиною. Втім, це ідеальний випадок, який рідко зустрічається на практиці. Для того щоб класифікувати множини, що не можна розділити гіперплощиною, можна застосувати два підходи: 1) штрафувати помилки (метод опорних векторів з м'яким зазором) і 2) використати ядро, щоб відобразити точки у спрямляючий простір (нелінійний метод опорних векторів).

Глава 5

Метод опорних векторів з м'яким зазором

Якщо застосувати до лінійно нероздільних множин точок метод опорних векторів з жорстким зазором, описаний в попередньому розділі, задача класифікації не матиме розв'язку. Для того щоб урахувати лінійну нероздільність навчальних вибірок, уведемо в обмеження (4.5) невід'ємні фіктивні змінні, які відіграють роль штрафу.

$$\omega_i \left((\vec{w}, \vec{x}_i) + b \right) \geq 1 - \xi_i, \quad i = 1, ..., N. \tag{5.1}$$

Завдяки фіктивним змінним розв'язок оптимізаційної задачі завжди існує. В цьому випадку смуга, що розділяє навчальні вибірки не буде порожньою, як у методі опорних векторів з жорстким зазором. Вона буде містити навчальні вибірки, які можуть бути класифіковані неправильно. Якщо $0 < \xi_i < 1$, то навчальні вибірки класифікується правильно, хоча і максимум функціонального зазору не досягається. Якщо $\xi \geq 1$, то навчальна вибірка класифікується неправильно. Для того щоб мінімізувати кількість помилок, треба мінімізувати функціонал

$$J\left(\vec{w}, b, \vec{\xi}, \vec{\lambda}, \vec{\mu}\right) = \frac{1}{2}||\vec{w}||^2 + C\sum_{i=1}^{N} \xi_i^p -$$

$$- \sum_{k=1}^{N} \lambda_i \left(\omega_i \left((\vec{w}, \vec{x}_i) + b \right) - 1 + \xi_i \right) - \sum_{k=1}^{N} \mu_i \xi_i \to \min_{\vec{w}, b} \max_{\lambda_i \mu_i}, \tag{5.2}$$

де $\vec{\lambda}$ і $\vec{\mu}$ — вектори невід'ємних множників Лагранжа.

Показник p може набувати два значення. Якщо $p = 1$, то описуваний метод називається L_1-методом опорних векторів, а якщо $p = 2$, то L_2-методом [16]. Вибір цього параметру залежить від характеру навчальних вибірок, тому поки що ми приділимо основну увагу загальній схемі методів, не віддаючи переваги жодному з них.

5.1. L_1-метод опорних векторів з м'яким зазором

Відповідно до теореми Куна-Таккера, розв'язком задачі (5.2) є сідлова точка, яка задовольняє такі умови.

$$\frac{\partial J\left(\vec{w}, b, \vec{\xi}, \vec{\lambda}, \vec{\mu}\right)}{\partial \omega} = 0, \tag{5.3}$$

$$\frac{\partial J\left(\vec{w}, b, \vec{\xi}, \vec{\lambda}, \vec{\mu}\right)}{\partial b} = 0, \tag{5.4}$$

$$\frac{\partial J\left(\vec{w}, b, \vec{\xi}, \vec{\lambda}, \vec{\mu}\right)}{\partial \xi} = 0, \tag{5.5}$$

$$\lambda_i \left(\omega_i \left((\vec{w}, \vec{x}_i) + b\right) - 1 + \xi_i\right) = 0, \quad i = 1, ..., N, \tag{5.6}$$

$$\mu_i \xi_i = 0, \quad i = 1, ..., N, \tag{5.7}$$

$$\lambda_i \geq 0, \mu_i \geq 0, \xi_i \geq 0, \quad i = 1, ..., N. \tag{5.8}$$

Обчислюючи частинні похідні (5.3 – 5.5), маємо

$$\vec{w} = \sum_{i=1}^{N} \lambda_i \omega_i x_i, \tag{5.9}$$

$$\sum_{i=1}^{N} \lambda_i \omega_i = 0, \tag{5.10}$$

$$\lambda_i + \mu_i = C, \quad i = 1, ..., N. \tag{5.11}$$

Підставляючи ці вирази в (5.2), отримуємо еквівалентну двоїсту задачу відносно множників Лагранжа λ_i.

Двоїста задача відносно множників Лагранжа. Знайти множники Лагранжа λ_i, $i = 1, ..., N$, за яких величина

$$J(\lambda) = \sum_{1=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_i \lambda_j \omega_i \omega_j \left(\vec{x}_i, \vec{x}_j\right) \tag{5.12}$$

досягає максимуму за умов

1. $\sum\limits_{i=1}^{N} \lambda_i \omega_i = 0;$

2. $C \geq \lambda_i \geq 0 \;\; i = 1, ..., N.$

Як бачимо, L_1-метод опорних векторів з м'яким зазором відрізняється від методу з жорстким зазором лише обмеженням $C \geq \lambda_i \geq 0 \;\; i = 1, ..., N.$

Розглянемо можливі варіанти значень множників Лагранжа λ_i.

1. $\lambda_i = 0$. У цьому випадку точка x_i класифікується правильно.

2. $0 < \lambda_i < C$. У цьому випадку $\omega_i \left((\vec{w}, \vec{x}_i) + b \right) - 1 + \xi_i = 0$ і $\xi_i = 0$. Отже, $\omega_i \left((\vec{w}, \vec{x}_i) + b \right) = 1$, тобто точка $\vec{x}_i$ є опорним вектором. Назвемо його *вільним опорним вектором.*

3. $\lambda_i = C$. У цьому випадку $\omega_i \left((\vec{w}, \vec{x}_i) + b \right) - 1 + \xi_i = 0$ і $\xi_i \geq 0$. Отже, знову маємо $\omega_i \left((\vec{w}, \vec{x}_i) + b \right) = 1$, тобто точка $\vec{x}_i$ є опорним вектором. На відміну від попереднього випадку назвемо його *зв'язаним опорним вектором.*

Вирішальна функція в методі опорних векторів із м'яким зазором має такий самий вигляд, як і в методі з жорстким зазором.

$$\begin{gathered} g\left(\vec{x}\right) = \sum_{k=1}^{M} \lambda_k \omega_k \left(\vec{x}_k, \vec{x}\right) + b, \\ b = \omega_k - \left(\vec{w}, \vec{x}_k\right), \;\; \text{де } \vec{x}_k \text{ — вільний опорний вектор } \lambda_k \neq 0. \end{gathered} \tag{5.13}$$

Оскільки $\lambda_i \neq 0$ лише для опорних векторів, сумування у вирішальній функції відбувається лише по опорних векторах. Вирішальне правило формулюється так:

$$\vec{x}_i \in C_1, \text{ якщо } g\left(\vec{x}_i\right) > 0, \tag{5.14}$$

$$\vec{x}_i \in C_2, \text{ якщо } g\left(\vec{x}_i\right) < 0. \tag{5.15}$$

Якщо $g\left(\vec{x}_i\right) = 0$, то вектор $\vec{x}_i$ не класифікується.

5.2. L_2-метод опорних векторів з м'яким зазором

Як ми зауважили вище, вибираючи в (5.2) параметр $p = 2$, можна побудувати метод L_2-метод опорних векторів з м'яким зазором. У цьому методі задача набуває такий вигляд.

$$J\left(\vec{w}, b, \vec{\lambda}\right) = \frac{1}{2}||\vec{w}||^2 + C\sum_{i=1}^{N} \xi_i^2 - \frac{C}{2}\sum_{k=1}^{N} \lambda_i^2 \to \min_{\vec{w}, b} \max_{\lambda_i}, \tag{5.16}$$

за умови

$$\omega_i \left((\vec{w}, h\left(\vec{x}_i\right)) + b \right) \geq 1 - \xi_i, i =, ..., N. \tag{5.17}$$

де $\vec{w} = (w_1, ..., w_l)$ — l-вимірний вектор, а $h(\vec{x})$ — функція, що відображає вектор $\vec{x} = (x_1, ..., x_m)$ в l-вимірний простір, ξ_i — фіктивні змінні, C — параметр зазору. Отже, ідея L_2-метод опорних векторів з м'яким зазором полягає у зменшенні вимірності задачі.

Задача мінімізації функціонала у L_2-методі опорних векторів з м'яким зазором формулюється так.

$$J\left(\vec{w}, b, \vec{\xi}, \vec{\lambda}\right) = \frac{1}{2}||\vec{w}||^2 + C\sum_{i=1}^{N}\xi_i^2 -$$
$$-\sum_{i=1}^{N}\lambda_i\left(\omega_i\left((\vec{w}, h\left(\vec{x}_i\right)\right) + b\right) - 1 + \xi_i\right). \tag{5.18}$$

На відміну від L_1-метод опорних векторів з м'яким зазором в L_2-методі множники Лагранжа для фіктивних змінних не потрібні, оскільки для оптимального розв'язку виконується умова $C\xi_i = \lambda_i \geq 0$.

Відповідно до теореми Куна-Таккера, розв'язком задачі (5.19) є сідлова точка, яка задовольняє такі умови.

$$\frac{\partial J\left(\vec{w}, b, \vec{\xi}, \vec{\lambda}\right)}{\partial \omega} = 0, \tag{5.19}$$

$$\frac{\partial J\left(\vec{w}, b, \vec{\xi}, \vec{\lambda}\right)}{\partial b} = 0, \tag{5.20}$$

$$\frac{\partial J\left(\vec{w}, b, \vec{\xi}, \vec{\lambda}\right)}{\partial \xi} = 0, \tag{5.21}$$

$$\lambda_i\left(\omega_i\left((\vec{w}, h\left(\vec{x}_i\right)\right) + b\right) - 1 + \xi_i\right) = 0, \quad i = 1, ..., N. \tag{5.22}$$

Обчислюючи частинні похідні (5.19– 5.20), маємо

$$\vec{w} = \sum_{i=1}^{N}\lambda_i\omega_i h\left(\vec{x}_i\right), \tag{5.23}$$

$$C\xi_i - \lambda_i = 0, \tag{5.24}$$

$$\sum_{i=1}^{N}\omega_i\lambda_i = 0. \tag{5.25}$$

Підставляючи ці вирази в (5.18), отримуємо еквівалентну двоїсту задачу відносно множників Лагранжа λ_i.

З обмежень (5.23–5.25) доходимо висновку, що сідлова точка повинна задовольняти умову $\lambda_j = 0$ або

$$\omega_j \left(\sum_{i=1}^{N} \omega_i \lambda_i \left(K\left(\vec{x}_i, \vec{x}_j\right) + \frac{\delta_{ij}}{C} \right) + b \right) - 1 = 0, \tag{5.26}$$

де $K\left(\vec{x}, \vec{y}\right) = h^T\left(\vec{x}\right) h\left(\vec{y}\right)$ — ядро, δ_{ij} — символ Кронекера. Як бачимо, ядро являє собой матрицю, розмір якої визначається розміром вектора $g\left(\vec{x}\right)$ в *просторі ознак*. Приклади різних ядер, в тому числі таких, що не є матрицею, будуть наведені в наступному розділі.

Відповідно, вирішальна функція має вигляд:

$$\begin{aligned} g\left(\vec{x}\right) &= \sum_{i=1}^{N} \lambda_i \omega_i K\left(\vec{x}_i, \vec{x}\right) + b, \\ b &= \omega_k - \sum_{i=1}^{N} \lambda_i \omega_i \left(K\left(\vec{x}_k, \vec{x}_i\right) + \frac{\delta_{ij}}{C} \right), \end{aligned} \tag{5.27}$$

Підставляючи вирази (5.23–5.25) в (5.27), отримуємо двоїсту задачу відносно множників Лагранжа.

$$J\left(\vec{\lambda}\right) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \omega_i \omega_j \lambda_i \lambda_j \left(K\left(\vec{x}_i, \vec{x}_j\right) + \frac{\delta_{ij}}{C} \right) \to \max_{\lambda_i}, \tag{5.28}$$

за умови, що

$$\sum_{i=1}^{N} \lambda_i \omega_i = 0, ; \quad \lambda_i \geq 0, \quad i = 1, ..., N. \tag{5.29}$$

Як бачимо, відмінність L_2-методу опорних векторів із м'яким зазором від методу із жорстким зазором полягає у доданку $\frac{\delta_{ij}}{C}$ і зменшенні виміру вихідної задачі. Завдяки тому, що доданок $\frac{\delta_{ij}}{C}$ додається до кожного діагонального елемента матриці $K\left(\vec{x}_i, \vec{x}_j\right)$, вона є додатно визначеною. Це значно підвищує стійкість обчислень порівняно із L_1-методом опорних векторів із м'яким зазором.

5.3. Нелінійний метод опорних векторів

В попередньому розділі ми зіткнулися з поняттям ядра $K\left(\vec{x}, \vec{y}\right)$, що породжувалося скалярним добутком $h^T\left(\vec{x}\right) h\left(\vec{y}\right)$, за допомогою якого можна було перейти з вихідного n-вимірного простору у l-вимірний *випрямляючий простір ознак*. Якщо ядро задовольняє умови Мерсера

1. $K\left(\vec{x}, \vec{y}\right) = K\left(\vec{y}, \vec{x}\right)$(симетричність);

2. $\int\limits_X K(\vec{x},\vec{y})\,h(\vec{x})\,h(\vec{y})\,d\vec{x}d\vec{y} \geq 0$ для будь-якої функції $h(\vec{x})$, що діє із простору ознак в простір $\mathbb{R}$ (додатна визначеність),

то навчаючі вибірки без дублікатів допускають розділення поліномами у випрямляючому просторі [2, 8].

За допомогою ядра двоїсту задачу в випрямляючому просторі можна сформулювати так:

$$J\left(\vec{\lambda}\right) = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\omega_i\omega_j\lambda_i\lambda_j K(\vec{x}_i,\vec{x}_j), \tag{5.30}$$

за умови, що

$$\sum_{i=1}^{N}\lambda_i\omega_i = 0,; \quad 0 \leq \lambda_i \leq C, \quad i = 1,...,N. \tag{5.31}$$

Завдяки додатній визначеності ядра задача квадратичної оптимізації (5.30, 5.31) є опуклою, а отже, має єдиний розв'язок.

Умови Куна-Таккера у випрямляючому просторі набувають такий вигляд.

$$\lambda_i\left(\omega_i\left(\sum_{j=1}^{N}\omega_j\lambda_j K(\vec{x}_i,\vec{x}_j) + b - 1\right) + \xi_i\right) = 0, \quad i = 1,...,N, \tag{5.32}$$

$$(C - \lambda_i)\,\xi_i = 0 \;\; i = 1,...,N, \tag{5.33}$$

$$\lambda_i \geq 0, \xi_i \geq 0, \quad i = 1,...,N. \tag{5.34}$$

Вирішальна функція задається формулою

$$\begin{gathered} g(\vec{x}) = \sum_{k=1}^{M}\lambda_k\omega_k K(\vec{x}_i,\vec{x}) + b, \\ b = \omega_k - \sum_{i=1}^{M}\lambda_k\omega_k K(\vec{x}_i,\vec{x}_k), \end{gathered} \tag{5.35}$$

де x_k — деякий вільний опорний вектор, M — кількість опорних векторів.

Вирішальне правило формулюється так:

$$\vec{x}_i \in C_1, \text{ якщо } g(\vec{x}_i) > 0, \tag{5.36}$$

$$\vec{x}_i \in C_2, \text{ якщо } g(\vec{x}_i) < 0. \tag{5.37}$$

Якщо $g(\vec{x}_i) = 0$, то вектор $\vec{x}_i$ не класифікується.

Наведемо список найбільш широко вживаних ядер.

1. Лінійне ядро: $K(\vec{x},\vec{y}) = (\vec{x},\vec{x})$.

2. Поліноміальне ядро степеня d: $K(\vec{x},\vec{y}) = ((\vec{x},\vec{x})+1)^d$, $d \in \mathbb{N}$.

3. Радіальне ядро: $\exp\left(-\gamma||\vec{x}-\vec{y}||^2\right)$, $\gamma > 0$.

Приклад 5. Нехай $x = (x_1, x_2)$ і d=2. В такому випадку поліноміальне ядро записується як

$$K(x,y) = 1 + 2x_1y_1 + 2x_2y_2 + 2x_1y_1x_2y_2 + x_1^2y_1^2 + x_2^2y_2^2 = (h,(x),h(y)),$$

де

$$h(x) = \left(1, \sqrt{2}x_1, \sqrt{2}x_2, \sqrt{2}x_1x_2, x_1^2, x_2^2\right).$$

Отже, воно задовольняє умови Мерсера.

Приклад 6. Розглянемо одновимірний приклад бінарної класифікації з поліноміальним ядром при d=2 і трьома навчальними точками: $x_1 = 0$ з міткою $\omega_1 = 1$, $x_2 = -1$ з міткою $\omega_2 = -1$ і $x_3 = 1$ з міткою $\omega_3 = -1$.

$$\frac{\partial J(\lambda)}{\partial \lambda_2} = 2 - 2\lambda_2 + 2\lambda_3 = 0, \tag{5.38}$$

$$\frac{\partial J(\lambda)}{\partial \lambda_3} = 2 + 2\lambda_2 - 4\lambda_3 = 0. \tag{5.39}$$

,

Двоїста задача має вигляд:

$$\begin{aligned} J(\lambda) = \lambda_1 + \lambda_2 + \lambda_3 - \frac{1}{2}(&\omega_2\omega_2\lambda_2\lambda_2(x_2x_2+1)^2 + \\ &+ \omega_2\omega_3\lambda_2\lambda_3(x_2x_3+1)^2 + \omega_3\omega_2\lambda_3\lambda_2(x_3x_2+1)^2 + \\ &+ \omega_3\omega_3\lambda_3\lambda_3(x_3x_3+1)^2) = \\ &= \lambda_1 + \lambda_2 + \lambda_3 - 2\lambda_2^2 - 2\lambda_3^2, \end{aligned} \tag{5.40}$$

$$\lambda_1 - \lambda_2 - \lambda_3 = 0, \tag{5.41}$$

Таким чином,

$$\lambda_1 = \lambda_2 + \lambda_3.$$

Знайдемо розв'язок системи:

$$\frac{\partial J(\lambda)}{\partial \lambda_2} = 2 - 4\lambda_2 = 0,$$

$$\frac{\partial J(\lambda)}{\partial \lambda_3} = 2 - 4\lambda_3 = 0.$$

Отже,

$$\lambda_1 = 4, \lambda_2 = 2, \lambda_3 = 2.$$

Звідси випливає, що всі точки є опорними векторами, а параметр зсуву b дорівнює 1.

Вирішальна функція має вигляд:

$$g(x) = 2x^2 - 1.$$

Звідси випливає, що межею між класами у випрямляючому просторі є парабола, що проходить через точки $-\frac{\sqrt{2}}{2}$ і $\frac{\sqrt{2}}{2}$, так що точка класу C_1 лежать вище цієї параболи, а точки класу C_2 — нижче.

Глава 6

Нейронні мережі

Ідея застосувати для класифікації штучні нейронні мережі, що імітують природні процеси, які відбуваються у мозку людини, досить природна. Зважаючи на складність мозку, це можливо здійснити, лише зробивши серйозні спрощення. Нейронним мережам присвячено дуже багато робіт (див. [13], [15] та інші монографії). В нашому курсі ми зосередимося на оптимізаційній складовій цієї теорії.

6.1. Персептрон Розенблатта

Модель нейронної мережі схематично описує роботу мозку як сукупності нейронів, які можуть перебувати у двох станах: збудженому і незбудженому. Кожний нейрон отримує сигнали від інших нейронів, обчислює їх лінійну комбінацію і відповідно змінює свій потенціал. Якщо обчислений потенціал перевищує порогове значення, нейрон збуджується. У такій моделі навчання можна звести до обчислення коефіцієнтів лінійних комбінацій потенціалів, що змінюються у часі.

На вхід нейрона надходить вектор ознак $\vec{x} = (x_1, x_2, ..., x_n)$. На підставі цього вектору і вектору ваг $\vec{w} = (w_1, w_2, ..., w_n)$, обчислюється лінійна комбінація

$$g(\vec{x}) = \sum_{i=1}^{n} w_i x_i + w_0.$$

Нейрон переходить у збуджений стан, якщо вихідний сигнал відмінний від нуля. Ступінь збудження монотонно залежить від стану, обмежена знизу і зверху, і стрімко змінюється в інтервалі значень від $\min \sum_{i=1}^{N} w_i x_i$ до $\max \sum_{i=1}^{N} w_i x_i$. Прикладами активаційних функцій є сходинка, сходинка з лінійним порогом, гіперболічний тангенс та сігмоїдна функція.

Результат класифікації обчислюється за таким вирішальним правилом.

$$\omega = \begin{cases} 1, \text{ якщо } g(\vec{x}) > 0, \\ 0, \text{ якщо } g(\vec{x}) < 0. \end{cases} \tag{6.1}$$

Як бачимо, нейрон працює як лінійна дискримінантна функція.

6.1.1. Алгоритм Розенблатта

Одна з перших нейронних мереж була запропонована Ф.Розенблаттом у 1957 р. і отримала назву *персептрон*. Алгоритм Розенблатта призначений для поступового навчання персептрона безпомилковій бінарній класифікації об'єктів шляхом ітераційного уточнення ваг лінійної комбінації. Особливістю алгоритму Розенблатта є його циклічність — об'єкти для класифікації подаються від першого до останнього, потім знову з першого до останнього і так далі, поки кількість помилок не буде дорівнювати нулю.

Алгоритм Розенблатта

1. $\vec{w}^{(0)} := 0$ (вектор ваг).
2. $m := 0$ (лічильник помилок).
3. $k := 0$ (лічильник ітерацій).
4. $l := 0$ (лічильник корекцій).
5. Якщо $m > 0, k := k \ mod \ N + 1$, подати на вхід вектор ознак k-го об'єкта $\vec{x}^{(k)}$; інакше перейти на крок 11.
6. Обчислити лінійну комбінацію $g\left(\vec{x}^{(k)}\right) = \sum_{i=1}^{n} w_i^{(k)} x_i^{(k)} + w_0^{(k)}$.
7. Якщо $\vec{x}^{(k)} \in C_1$ і $g\left(\vec{x}^{(k)}\right) > 0$, то перейти на крок 5.
8. Якщо $\vec{x}^{(k)} \in C_2$ і $g\left(\vec{x}^{(k)}\right) < 0$, то перейти на крок 5.
9. Якщо $\vec{x}^{(k)} \in C_1$ і $g\left(\vec{x}^{(k)}\right) < 0$ (помилкова класифікація), то $m := m + 1; l := l + 1; w^{(k)} := w^{(k-1)} + x^{(k)}$ і перейти на крок 4.
10. Якщо $\vec{x}^{(k)} \in C_2$ і $g\left(\vec{x}^{(k)}\right) > 0$ (помилкова класифікація), то $m := m + 1; l := l + 1; w^{(k)} := w^{(k-1)} - x^{(k)}$ і перейти на крок 4.
11. Вихід: вектор w.

Відповідь на питання, чи збігається процес навчання за алгоритмом Розенблатта, тобто чи дорівнює кількість помилок нулю після навчання, дає теорема Новікова [8].

Теорема 3. *Якщо навчальні вибірки можна розділити смугою шириною* 2δ *і всі вони лежать в кулі радіусу* $R = \max\limits_{1\le i\le N} ||\vec{x}_i||$*, то навчання за алгоритмом Розенблатта збігається і кількість корекцій не перевищує* $\left(\frac{R}{\delta}\right)^2$.

Доведення. Позначимо напрямний вектор роздільної смуги як v, а мітки класів C_1 і C_2 як $\omega_1 = -1$ і $\omega_2 = 1$. Це означає, що корекція вектору ваг $w^{(k-1)}$ здійснюється шляхом додавання до нього вектору $\omega_i\vec{x}_i$, такого що $\left(\vec{w}^{(k-1)}, \vec{v}\right) < 0$ і $||\omega_i\vec{x}_i|| \le R$.

Отже,

$$\left(\vec{w}^{(k)}, \vec{v}\right) = \left(\vec{w}^{(k-1)}, \vec{v}\right) + (\omega_i\vec{x}_i, \vec{v}) \ge \left(\vec{w}^{(k-1)}, \vec{v}\right) + \delta||\vec{v}||.$$

Звідси випливає, що

$$\left(\vec{w}^{(k)}, \vec{v}\right) \ge k\delta||\vec{v}||.$$

З нерівності Коші-Буняковського маємо:

$$||\vec{w}^{(k)}|| \ge k\delta. \tag{6.2}$$

З іншого боку,

$$||\vec{w}^{(k)}||^2 = ||\vec{w}^{(k)} + \omega_i\vec{x}_i||^2 \le ||\vec{w}^{(k-1)}||^2 + ||\omega_i\vec{x}_i||^2 \le ||\vec{w}^{(k-1)}||^2 + R^2. \tag{6.3}$$

Нерівності (6.2) і (6.3) є сумісними лише за умови, що

$$k \le \left(\frac{R}{\delta}\right)^2.$$

Таким чином, кількість кроків алгоритму є скінченною. □

Приклад 7.

6.2. Оптимізаційна трактовка персептрону Розенблатта

Алгоритм Розенблатта допускає оптимізаційну інтерпретацію [9]. Уведемо в розгляд кусково-лінійну функцію

$$J(w) = \sum_{x\in Y} \delta_x(w, x),$$

де Y — множина неправильно класифікованих об'єктів,

$$\delta_x = \begin{cases} -1, \text{ якщо } x \in C_1, \\ 1, \text{ якщо } x \in C_2. \end{cases}$$

У такому випадку класифікація об'єктів за допомогою роздільної гіперплощини, яка визначається коефіцієнтами $w = (w_0, w_1, ..., w_n)$, еквівалентна задачі

$$J(w) = \sum_{x \in Y} \delta_x(w, x) \to \min_w.$$

Мінімізуємо цей функціонал за методом градієнтного спуску:

$$w^{(k)} = w^{(k-1)} - \rho^{(k-1)} \frac{dJ(w)}{dw} = w^{(k-1)} - \rho^{(k-1)} \sum_{x \in Y} x \delta_x.$$

Таким чином, алгоритм Розенблатта є різновидом алгоритму градієнтного спуску. Для його збіжності ряд $\sum\limits_{k=0}^{\infty} |\rho_k|$ повинен розбігатися, а ряд $\sum\limits_{k=0}^{\infty} |\rho_k^2|$ — збігатися.

6.3. Багатошаровий персептрон

Шаром персептрону є суматор, який імітує роботу нейронів. Узагальнюючи цей факт, можна дати наступне означення.

Означення 25. *Шаром* штучної нейронної мережі є сукупність її елементів, що імітують накопичення потенціалу нейрону.

Зауваження 3. Елементи штучної нейронної мережі, на які надходять ознаки об'єкта, а також елементи, що зберігають результати розпізнавання, також називають вхідним і віхідним шарами, але вони не використовуються для класифікації мереж. Тому, незважаючи на наявність трьох фізичних шарів, персептрон Розенблатта називають **одношаровим**.

Теорема Новікова дає підстави сподіватися, що нейронні мережі можна навчити розв'язувати складні задачі розпізнавання образів. Втім, нагадаємо, що основним припущенням цієї теореми є лінійна роздільність навчальних множин. У випадку множин, що не можна розділити гіперплощиною, виникають серйозні проблеми. Розглянемо, наприклад, чотири точки: $A_1 = (0, 0)$, $A_2 = (1, 1)$, $B_1 = (0, 1)$ і $B_2 = (1, 0)$. Будемо вважати, що точки A_1 і A_2 належать класу A, а точки B_1 і B_2 — класу B. В ціх точках легко впізнати значення логічної функції XOR, а в класах A і B — класи одиниць і нулів. Ці класи неможливо розділити лінією на площині. Намагаючись побудувати лінійну роздільну функцію для цієї задачі за допомогою одношарового персептрону Розенблатта, ми будемо перебирати усі лінійні комбінацію точок $w_1 x_1 + w_2 x_2$ і жодна з них не зможе описати таку лінію, щоб точки A_1 і A_2 лежали з одного боку, а точки B_1 і B_2 — з іншого. Для виходу з цієї ситуації було запропоновано побудувати багатошарові штучні нейронні мережі.

Згадаємо, що операцію XOR можна виразити за допомогою операцій OR і AND.

$$A\ XOR\ B = A\ OR\ B\ AND\ (NOT\ A\ AND\ NOT\ B)$$

Логічна функція OR набуває значення, 0, якому відповідає точка $A_1 = (0,0)$, що утворює клас A, і 1, якому відповідають точки $B_1 = (0,1)$, $B_2 = (1,0)$ і $B_3 = (1,1)$. Неважно зауважити, що ці точки розділяються прямою $x_1 + x + 2 = \frac{1}{2}$.

Логічна функція AND набуває значення, 0, якому відповідає точка $A_1 = (0,0), A_2 = (1,0), A_3 = (0,1)$, що утворюють клас A, і 1, якому відповідає точка $B_1 = (1,1)$. Неважно зауважити, що ці точки розділяються прямою $x_1 + x + 2 = \frac{3}{2}$.

Таким чином, застосувавши суперпозицію двох одношарових персептронів Розенблатта, ми розділимо точки із задачі про функцію XOR двома лініями. Спочатку побудуємо лінійну роздільну функцію для операції OR, на виході якої отримуємо значення y_1 та y_2, які надходять на вхід наступного персептрона (додаткового шару нейронів). В термінах значень y_1 і y_2 розв'язок задачі про функцію XOR записується як $y_1 - y_2 = \frac{1}{2}$.

6.4. Оптимізаційна трактовка багатошарової нейронної мережі

Побудову багатошарової штучної нейронної мережі для багатокласової класифікації можна описати як оптимізаційну задачу [10]. Отримана таким чином штучна нейронна мережа називається персептроном Румпельхарта. Позначимо кількість класів як m, кількість навчальних вибірок як N, кількість прихованих шарів в нейронній мережі як L, кількість нейронів на l-му шарі як K_l. На вхід нейронної мережі подається вектор ознак $\vec{x}_i = \left(x_1^{(i)}, ..., x_n^{(i)}\right)$, $i = 1, ..., N$, на виході очікується вектор $\vec{y}_i = \left(y_1^{(i)}, ..., y_m^{(i)}\right)$, $i = 1, ..., N$.

Нехай на i-й навчальній вибірці мережа видає результат $\hat{\vec{y}}_i$. Він може збігатися або не збігатися із очікуваним результатом $\vec{y}_i$. Позначимо помилку на i-й навчальній вибірці як

$$\epsilon_i = \frac{1}{2}\sum_{j=1}^{m}\left(\hat{\vec{y}}_j^{(i)} - \vec{y}_j^{(i)}\right)^2,$$

а функціонал помилок як

$$J(w) = \sum_{i=1}^{N}\epsilon_i = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{m}\left(\hat{\vec{y}}_j^{(i)} - \vec{y}_j^{(i)}\right)^2, \tag{6.4}$$

Навчання мережі зводиться до розв'язання оптимізаційної задачі: знайти

$$\arg\min_w J(w).$$

6.4.1. Алгоритм зворотнього розповсюдження помилки

Позначимо функцію активації нейрона як $f(s)$. Перетворення вхідного сигналу на j-му нейроні l-го шару обчислимо як $x_j^{(l)} = f\left(s_j^{(l)}\right)$, де $s_j^{(l)} = \left(w_j^{(l)}, x\right)$. Корекцію ваги w_{ij} на l-му шарі обчислимо за методом градієнтного спуску:

$$\Delta w_{ij}^{(l)} = -\rho\frac{\partial J}{\partial w_{ij}^{(l)}} = -\rho\frac{\partial J}{\partial x_j^{(l)}}\frac{\partial x_j^{(l)}}{\partial s_j^{(l)}}\frac{\partial s_j^{(l)}}{\partial w_{ij}^{(l)}} = -\rho\delta_j^{(l)}x_i^{(l)},$$

де $\delta_j^{(l)} = \frac{\partial J}{\partial x_j^{(l)}}\frac{\partial x_j^{(l)}}{\partial s_j^{(l)}}$ — помилка на j-му нейроні l-го шару.

Виразимо помилку на j-му нейроні l-го шару через помилку на $(l+1)$-му шарі.

$$\delta_j^{(l)} = \frac{\partial J}{\partial x_j^{(l)}}\frac{\partial x_j^{(l)}}{\partial s_j^{(l)}} = \left(\sum_{k=1}^{K_{l+1}}\frac{\partial J}{\partial x_k^{(l+1)}}\frac{\partial x_k^{(l+1)}}{\partial s_k^{(l+1)}}\frac{\partial s_k^{(l+1)}}{\partial x_j^{(l+1)}}\right)\frac{\partial x_j^{(l)}}{\partial s_j^{(l)}}.$$

Враховуючи, що $\frac{\partial s_k^{(l+1)}}{\partial x_j^{(l+1)}} = w_{jk}^{(l+1)}$ і $\frac{\partial x_j^{(l)}}{\partial s_j^{(l)}} = f'\left(s_j^{(l)}\right)$, маємо:

$$\delta_j^l = \left(\sum_{k=1}^{K_{l+1}}\delta_k^{(l+1)}w_{jk}^{(l+1)}\right)f'\left(s_j^{(l)}\right).$$

Отже,

$$\delta_j^l = \left(\sum_{k=1}^{K_{l+1}}\delta_k^{(l+1)}w_{jk}^{(l+1)}\right)f'\left(s_j^{(l)}\right), \quad l = L-1, L-2, ..., 1.$$

На останньому шарі

$$\delta_j^L = \left(x_j^{(L)} - y_j\right)f'\left(s_j^{(l)}\right).$$

Назва алгоритму зворотнього розповсюдження помилки пояснюється тим, что корекція ваги на кожному шарі обчислюється через корекції на наступних шарах, що можна трактувати як зворотнє розповсюдження помилки від останнього шару до першого.

6.4.2. Приклади активаційних функцій

Для того щоб обчислити помилку для кожного нейрона, треба продиференціювати його активаційну функцію. Знаючи цю функцію заздалегідь, можна спростити обчислення в алгоритмі зворотнього розповсюдження помилок [15].

Логістична функція

Логістична функція j-го нейрона на l-му шарі має вигляд

$$f\left(s_j^{(l)}\right) = \frac{1}{1+\exp\left(-as_j^{(l)}\right)},\ a>0. \tag{6.5}$$

Похідна цієї функції дорівнює

$$f'\left(s_j^{(l)}\right) = \frac{a\exp\left(-as_j^{(l)}\right)}{\left(1+\exp\left(-as_j^{(l)}\right)\right)^2},\ a>0. \tag{6.6}$$

Враховуючи, що за означенням $f\left(s_j^{(l)}\right) = x_j^{(l)}$, можна переписати похідну як

$$f'\left(s_j^{(l)}\right) = ax_j^{(l)}\left(1-x_j^{(l)}\right). \tag{6.7}$$

Отже, для довільного нейрону на прихованому шарі в алгоритмі зворотнього розповсюдження помилки з логістичною функцією активації маємо:

$$\delta_j^l = \left(\sum_{k=1}^{K_{l+1}}\delta_k^{(l+1)}w_{jk}^{l+1}\right)f'\left(s_j^{(l)}\right) = ax_j^{(l)}\left(1-x_j^{(l)}\right)\sum_{k=1}^{K_{l+1}}\delta_k^{(l+1)}w_{jk}^{(l+1)} \tag{6.8}$$

Гіперболічний тангенс

Активаційна функція j-го нейрона на l-му шарі, що задається гіперболічним тангенсом, має вигляд

$$f\left(s_j^{(l)}\right) = a\tanh\left(bs_j^{(l)}\right),\ a>0,\ b>0. \tag{6.9}$$

Похідна цієї функції дорівнює

$$f'\left(s_j^{(l)}\right) = ab\left(1-\tanh^2\left(bs_j^{(l)}\right)\right) = \frac{b}{a}\left(a-s_j^{(l)}\right)\left(a+s_j^{(l)}\right). \tag{6.10}$$

Отже, для довільного нейрону на прихованому шарі в алгоритмі зворотнього розповсюдження помилки з логістичною функцією активації маємо:

$$\delta_j^l = \left(\sum_{k=1}^{K_{l+1}}\delta_k^{(l+1)}w_{jk}^{l+1}\right)f'\left(s_j^{(l)}\right) = \frac{b}{a}\left(a-s_j^{(l)}\right)\left(a+s_j^{(l)}\right)\sum_{k=1}^{K_{l+1}}\delta_k^{(l+1)}w_{jk}^{(l+1)} \tag{6.11}$$

6.4.3. Метод стохастичного градієнта

Мінімізація функціонала помилок (6.4) — не єдиний спосіб навчання штучних нейронних мереж. Замість мінімізації сумарної помилки можна мінімізувати емпіричний ризик, що задається диференційовною функцією [2]:

$$\arg\min_w J(w) = \sum_{i=1}^{N}\mathcal{L}\left(\omega_i\left(w,\vec{x}_i\right)\right), \tag{6.12}$$

де N — кількість навчальних вибірок, ω_i — мітка i-го класу.

Застосуємо для розв'язання оптимізаційної задачі (6.12) метод градієнтного спуску:

$$w^{(k+1)} = w^k - \rho_k \frac{\partial J(\vec{w})}{\partial \vec{w}} = w^k - \rho_k \mathcal{L}'(\omega_i(w, \vec{x}_i)) x_i \omega_i, \quad k = 1, 2, ... \tag{6.13}$$

де $\frac{\partial J(\vec{w})}{\partial \vec{w}} = \left(\frac{\partial J(\vec{w})}{\partial w_j}, ..., \frac{\partial J(\vec{w})}{\partial w_n} \right)^T$.

Для цього методу є слушною теорема [8].

Теорема 4. *Якщо виконуються умови:*

1) *навчальні вибірки $(\vec{x}_i; \omega_i)$ є незалежними і мають однаковий розподіл F,*

2) *для будь-якого значення w випадкова величина $\frac{\partial J(\vec{w})}{\partial \vec{w}}$, що залежить від випадкового вектора $(\vec{x}; \omega)$ з розподілом F, має скінчене математичне сподівання і дисперсію,*

3) *$\rho_k > 0$ при всіх k і $\rho_k \to 0$ при $k \to \infty$,*

4) *ряд $\sum\limits_{k=1}^{\infty} \rho_i$ розбігається,*

5) *ряд $\sum\limits_{k=1}^{\infty} \rho_i^2$ збігається.*

то із ймовірністю 1 ітераційна процедура стохастичного градієнтного спуску (6.13) збігається до локального мінімуму математичного сподівання помилки.

Глава 7

Метод потенціальних функцій

Ідея методу потенціальних функцій має фізичну природу. Як відомо, згідно закону Кулона вплив заряду на точку убуває пропорційно квадрату відстані до неї. Отже, потенціал може використовуватися як оцінка віддаленості точки від заряду. У цьому розумінні метод потенціальних функцій є різновидом методу найближчого сусіди. Якщо поле утворене декількома зарядами, потенціал у кожній точці поля дорівнює сумі потенціалів, створюваних у цій точці кожним із зарядів. Якщо заряди, що утворюють поле, розташовані компактно, потенціал поля досягає максимального значення в центрі тяжіння множини точок і убуває в міру видалення від нього. Таким чином, якщо точку в просторі ознак інтерпретувати як заряд, то ці міркування можна формалізувати і звести до обчислення певної суми функцій, що описують потенціали зарядів — точок навчальної множини.

Розглянемо задачу бінарної класифікації. Кожному образу поставимо у однозначну відповідність точку в просторі ознак X. В подальшому будемо припускати, що класи C_1 і C_2 не перетинаються. Як правило, топологічні властивості простору ознак допускають застосування малої леми Урисона, з якої випливає, що в просторі ознак X існує неперервна функція Φ, яка набуває значення більше нуля в точках класу C_1 і менше нуля в точках класу C_2. Зауважимо, що таких функцій може бути багато (і навіть нескінченно багато). В ході навчання для кожного образу R_k, якому відповідає точка x_k, обчислюється функція $K(x, x_k)$, яка називається потенціальною, а побудова роздільної функції Φ за навчальною послідовністю образів $A_1, A_2, ..., A_k, ...$ зводиться до побудови послідовності потенціальних функцій $K(x, x_1), K(x, x_2), ..., K(x, x_k), ...$, що збігаються до функції Φ.

Інтуїтивна ідея методу полягає у тому, щоб обчислити загальні потенціали обох класів:

$$K_1(x) = \sum_{x_k \in C_1} K(x, x_k),$$

$$K_2(x) = \sum_{x_k \in C_2} K(x, x_k),$$

а потім знайти роздільну функцію у вигляді їх різниці

$$\Phi(x) = K_1(x) - K_2(x).$$

Якщо при класифікації нової точки x^* виконується умова

$$K_1(x) > K_2(x),$$

то $x^* \in C_1$, інакше $x^* \in C_2$.

Звідси випливає, що роздільна функція є додатною на точках з класу C_1 і від'ємною на точках з класу C_2.

Математична формалізація цієї ідеї зводиться до відновлення певної функції за допомогою рекурентних процедур і обґрунтування їх збіжності.

7.1. Загальна схема

Метод потенціальних функцій грунтується на припущенні, що існує система функцій $\varphi_1, \varphi_2, ..., \varphi_k, ...$, яка дозволяє для будь-яких двох диз'юнктних класів знайти таке число N, що функцію Φ можна подати у вигляді

$$\Phi(x) = \sum_{k=1}^{N} c_k \varphi_k(x). \tag{7.1}$$

Як правило простір X є гільбертовим, тому в новому просторі існує повна система функцій $\varphi_1, \varphi_2, ..., \varphi_k, ...$, за допомогою яких функцію Φ можна подати у вигляді ряду Фур'є

$$\Phi(x) = \sum_{k=1}^{\infty} c_k \varphi_k(x). \tag{7.2}$$

Прагнучи подати роздільну функцію Φ у вигляді суми (а не ряду), уведемо в розгляд N-вимірний простір Y, на який простір ознак X відображається за правилом $y_k = \varphi_k(x)$, $k = 1, 2, ..., N$. Таким чином, роздільна функція Φ відображається в лінійну функцію $\sum_{k=1}^{N} a_k y_k(x)$ і набуває значення більше нуля в точках класу C_1 і менше нуля в точках класу C_2. Легко бачити, що в просторі Y функція Φ є лінійною (відносно точок y).

Потенціальна функція розглядається як функція двох векторів

$$K(x, x^*) = \sum_{k=1}^{\infty} \alpha_k^2 \varphi_k(x) \varphi_k(x^*), \tag{7.3}$$

де $\varphi_1, \varphi_2, ..., \varphi_k, ...$ — лінійно-незалежна система функцій; α_k^2 — дійсні числа, що не обертаються в нуль одночасно; x^* — точка, яка обчислюється в ході навчання.

Вважатимемо, що функції φ_k і $K(x, x^*)$ обмежені на просторі ознак. Першому образу P_1 ставимо у відповідність точку ознак x_1, за якою будується потенційна функція

$$\Phi_1(x) = \begin{cases} K(x, x_1), & \text{якщо } x \in C_1, \\ -K(x, x_1), & \text{якщо } x \in C_2. \end{cases} \tag{7.4}$$

На k-му кроці отримуємо потенціальну функцію $\Phi_k(x)$. На (k+1)-му кроці, обчислюючи потенціал в точці x_{k+1} можемо отримати наступні значення потенціалу:

$$x_{k+1} \in K_1 \Rightarrow \Phi_k(x_{k+1}) > 0 \Rightarrow \Phi_{k+1}(x) = \Phi_k(x),$$

$$x_{k+1} \in K_2 \Rightarrow \Phi_k(x_{k+1}) < 0 \Rightarrow \Phi_{k+1}(x) = \Phi_k(x),$$

$$x_{k+1} \in K_1 \Rightarrow \Phi_k(x_{k+1}) < 0 \Rightarrow \Phi_{k+1}(x) = \Phi_k(x) + \Phi(x, x_{k+1}),$$

$$x_{k+1} \in K_2 \Rightarrow \Phi_k(x_{k+1}) > 0 \Rightarrow \Phi_{k+1}(x) = \Phi_k(x) - \Phi(x, x_{k+1}).$$

Таким чином,

$$\Phi_k(x) = \sum_{x_i^- \in V_1} K(x, x_i) + \sum_{x_j^+ \in V_2} K(x, x_j), \tag{7.5}$$

де x_i — точки з класу C_1, на яких класифікатор робить помилку.

Корекцію роздільної функції можна здійснювати по-різному, тому існують декілька алгоритмів, заснованих на методі потенціальних функцій.

Вважатимемо, що початкове наближення функції $\Phi_0(x)$ в точці $x_{(0)}$ дорівнює нулю. Тоді корекцію роздільної функції можна здійснити так:

$$\Phi_{k+1}(x) = \Phi_k(x) + \alpha_{k+1}\mathrm{sgn}\left[\Phi(x_{k+1}) - \Phi_k(x_{k+1})\right] K(x, x_{k+1}), \tag{7.6}$$

де $\Phi(x_{k+1})$ — істинне значення роздільної функції в точці x_{k+1}; α_k — будь-яка послідовність чисел, що задовольняє умовам:

$$\alpha_k \to 0, \ \sum_{k=1}^{\infty} \alpha_k = \infty, \ \sum_{k=1}^{\infty} \alpha_k^2 < \infty.$$

Інакше кажучи,

$$\{\alpha_k\} \in (c_0 \cap l_2) \setminus l_1.$$

Це еквівалентно обчисленню коефіцієнтів розвинення функції $\Phi(x)$ за таким правилом:

$$c_k^{(n+1)} = c_{k+1}^{(n)} + \alpha_{k+1}\mathrm{sgn}\left[\Phi(x_{k+1}) - \sum_{k=1}^{N} c_k \varphi_k(x_{k+1})\right] \varphi_k(x_{k+1}), \tag{7.7}$$

Збіжність цього ітераційного процесу обґрунтовується теоремою 5.

Теорема 5. *Нехай x_k — послідовність незалежних випадкових точок з простору ознак X, $P(x)$ — щільність ймовірності появи цих точок, а $\Phi(x)$ — функція, що має вигляд*

$$\Phi(x) = \sum_{k=1}^{N} c_k \varphi_k(x).$$

Тоді послідовність функцій $\Phi_k(x)$, $k = 1, 2, ...$, що задаються рекурентними співвідношеннями (7.6) задовольняють умову

$$P\left\{\lim_{k\to\infty} \int_X |\Phi(x) - \Phi_i(x)| P(x)\, dx = 0\right\} = 1 \tag{7.8}$$

Другий варіант виглядає так:

$$\Phi_0(x) = 0,$$

$$\Phi_{k+1}(x) = \Phi_k(x) + \frac{1}{\lambda}\left[\Phi(x_{k+1}) - \Phi_k(x_{k+1})\right] K(x, x_{k+1}), \tag{7.9}$$

де $\lambda > \frac{1}{2}\max K(x, x^*)$.

Це еквівалентно обчисленню коефіцієнтів розвинення функції $\Phi(x)$ за таким правилом:

$$c_{k+1}^{(n+1)} = c_{k+1}^{(n)} + \frac{1}{\lambda}\left[\Phi(x_{k+1}) - \sum_{k=1}^{N} c_k \varphi_k(x_{k+1})\right] \varphi_k(x_{k+1}). \tag{7.10}$$

Збіжність цієї процедури випливає з теореми 6.

Теорема 6. *За умов теореми 5 послідовність функций $\Phi_k(x)$, $k = 1, 2, ...$, що визначаються співвідношенням (7.8) задовольняє умову*

$$P\left\{\lim_{k\to\infty} \int_X (\Phi(x) - \Phi_i(x))^2 P(x)\, dx = 0\right\} = 1. \tag{7.11}$$

Зауваження 4. Варіанти методу потенціальних функцій, які ґрунтуються на формулах (7.6) і (7.9) називаються *машинною* реалізацією, а варіанти, які ґрунтуються на формулах (7.7) і (7.10) називаються *персептронною* реалізацією. Вибір назви для персептронною реалізації пояснюється тим, що цю схему можно описати за допомогою персептрона Розенблатта.

7.2. Геометрична інтерпретація

Нехай в просторі X існує функція $\Phi(x)$, що розділяє множини $A \subset C_1$ і $B \subset C_2$ і задовольняє умови (7.1) і (7.2). Тоді в просторі Y існує роздільна гіперплощина Γ, що проходить через початок координат із направляючим

вектором $\vec{c}$. Відобразимо множину B симетрично відносно початку координат у множину B' (тобто замінимо кожний вектор x вектором $-x$) і отримаємо множину $S = A \cup B'$. За припущенням множини A і B' розділяються гіперплощиною Γ, тобто множина S лежить по один бік від площини Γ. Розглянемо послідовності точок $M = \{x_1, x_2, ..., x_n, ...\}$ з простору X і їх образи $M^* = \{y_1, y_2, ..., y_n, ...\}$ в просторі Y.

Потенціальна функція у просторі Y може бути подана як

$$U(Z, Z^*) = ZZ^* \tag{7.12}$$

де

$$Z = \{z_1, z_2, ...\} = \{\alpha_1\varphi_1(x), \alpha_2\varphi_2(x), ...\}$$

і

$$Z^* = \{z_1^*, z_2^*, ...\} = \{\alpha_1\varphi_1(x^*), \alpha_2\varphi_2(x^*), ...\}.$$

Отже, співвідношення (7.5) можна переписати як

$$\Phi_k(Z) = \sum_{Z^{p(-)} \in M^*} (Z, Z^p) \tag{7.13}$$

Виправлення помилки відбувається тоді, коли $\Phi_k(Z) < 0$. Отже, корекція помилки на (k+1)-му кроці відбувається, коли

$$z_{k+1} \sum_{m=1}^{k} z_m < 0 \tag{7.14}$$

Таким чином, перша точка $Z^{(1)}$ з множини M^* призводить до побудови площини $U_1(Z) = \left(Z, Z^1\right) = 0$ із напрямляючим вектором $Z^{(1)}$. Якщо наступна точка з множини M^* лежить у тому ж підпросторі, що й напрямляючий вектор Z^1, то помилка відсутня і роздільна площина не уточнюється. Якщо ж наступна точка потрапляє у протилежний підпростір, відбувається корекція. При цьому попередній напрямляючий вектор складається із вектором точки, що вимагала корекції, і їхня сума береться як новий напрямляючий вектор.

Після k корекцій напрямляючий вектор дорівнює $\sum_{m=1}^{k} Z_m$ і гіперплощина, що проходить через початок координат, приймається як роздільна.

7.3. Оптимізаційна інтерпретація

Розглянемо N функцій $\Phi(\vec{c}, x_i), i = 1, ..., N$, де x — випадкова величина і $\vec{c} = (c_1, c_2, ..., c_N)$. Запишемо систему рівнянь регресії відносно коефіцієнтів c_i:

$$M(\Phi(\vec{c}, x_i)) = 0, \tag{7.15}$$

де $M(\Phi(\vec{c}, x_i))$ — математичне сподівання функції $\Phi(\vec{c}, x_i)$ за величиною x.

7.3.1. Процедура Роббінса-Монро

Припустимо, що розподіл ймовірності випадкової величини x є невідомим, але за послідовними точками $x_1, x_2, ...$ для будь-якого числа c можна обчислити значення функції $\Phi(\vec{c}, x_i)$. За таких умов Г.Роббінс і С.Монро запропонували спеціальну рекурентну процедуру:

$$c_i^{(n+1)} = c_i^{(n)} + \gamma_i \Phi(\vec{c}, x_i), i = 1, 2, ..., N, \tag{7.16}$$

де $\sum_{i=1}^{\infty} \gamma_i = \infty$ і $\sum_{i=1}^{\infty} \gamma_i^2 < \infty$.

В скінченовимірному випадку персептронний варіант методу потенціальних функцій може бути поданий як спеціальний варіант процедури Роббінса-Монро і, відповідно, зведений до мінімізації функціонала

$$J() \to \min_c \tag{7.17}$$

де $J() = M(G(\vec{c}, x))$ і G - невід'ємна функція, що обертається на нуль, коли $\sum_{i=1}^{N} c_i \varphi_i(x) = \Phi(x)$.

Легко бачити, що в такому випадку персептронний варіант методу потенціальних функцій є різновидом методу стохастичного градієнта.

Глава 8

Логістична регресія

Розглянемо новий разновід лінійного класифікатора — метод логістичної регресії, який дозволяє оцінити ймовірність успіху в схемі випадкових випробувань (множині незалежних спостережень) і класифікувати об'єкти за цією ймовірністю. Цей метод має два варіанти: бінарний і мультиноміальний.

8.1. Бінарна логістична регресія

Введемо позначення: $Y_i = 1$ — подія відбулася; $Y_i = 0$ — подія не відбулася; $P(Y_i = 1) = p_i\,, \quad P(Y_i = 0) = 1 - p_i$ — імовірності подій $Y_i = 1$ та $Y_i = 0$; $E(Y_i) = 0 \cdot (1 - p_i) + 1 \cdot (p_i) = p_i$ — математичне сподівання події Y_i.

Задача полягає у прогнозуванні $P(Y_i = 1) = p_i$ на основі попередніх відомостей. Чи можна тут скористатися лінійною регресією $E(Y_i\,|X_i) = \beta_0 + \beta_1 X_i = p_i$? Нажаль, за умови дискретності даних виникають деякі проблеми, а саме:

1. Лінійна регресія не задовольняє умову $0 \leq E\,(Y_i|X_i) = p_i \leq 0$.

2. Необхідною умовою для лінійної регресії є сталість дисперсії відгуків. У випадку ж дискретних даних маємо, що $D(Y_i) = p_i(1 - p_i)$, тобто її значення залежить від X_i.

3. Ще одна необхідна умова лінійної регресії – нормальний розподіл похибки: $\varepsilon_i = Y_i - (\beta_0 + \beta_1 X_i)$. Коли ж $Y_i = 1$, маємо: $\varepsilon_i = 1 - (\beta_0 + \beta_1 X_i)$.

Коли дані є дискретними, графік очікуваних відгуків має вигляд деякої S-кривої, яку називають логістичною кривою. Модель логістичної регресії має вигляд:

$$E(Y_i\,|X_i\,) = p_i = \frac{e^{(\beta_0+\beta_1 X_i)}}{1 + e^{(\beta_0+\beta_1 X_i)}}.$$

Досліджувані параметри знаходяться у степені експоненти, тому спочатку потрібно звести модель до вигляду, коли p_i будуть залежати від X_i лінійно, а

потім повернутися до оригінального вигляду моделі щоб відобразити реальну залежність. Такі перетворення мають вигляд:

$$p = \frac{e^{(\beta_0+\beta_1 X)}}{1+e^{(\beta_0+\beta_1 X)}},$$

$$1-p = \frac{1+e^{(\beta_0+\beta_1 X)}}{1+e^{(\beta_0+\beta_1 X)}} - \frac{e^{(\beta_0+\beta_1 X)}}{1+e^{(\beta_0+\beta_1 X)}} = \frac{1}{1+e^{(\beta_0+\beta_1 X)}},$$

$$\frac{p}{1-p} = e^{(\beta_0+\beta_1 X)},$$

$$\frac{p}{1-p} \neq 1,$$

тому $\ln(\frac{p}{1-p}) = \beta_0 + \beta_1 X$.

Тобто, маємо лінійну залежність від X. Тепер наблизимо імовірність події p_i її частотою h_i. З рівності

$$\ln(\frac{h_i}{1-h_i}) = \beta_0 + \beta_1 X$$

знайдемо регресійні коефіцієнти і повернувшись до вихідної моделі зможемо обчислити передбачувану імовірність

$$\hat{p} = \frac{e^{(\beta_0+\beta_1 X)}}{1+e^{(\beta_0+\beta_1 X)}}.$$

8.2. Оптимізаційна інтерпретація

Найбільш поширеним способом обчислення коефіцієнтів логистичної регресії є метод максимальної правдоподібності. Цей метод полягає в обчисленні максимуму функцію правдоподібності, що виражає імовірність спільної появи результатів вибірки $Y_1, Y_2, ..., Y_k$.

$$\arg\max_{\theta} L\left(Y_1, Y_2, ..., Y_n, \theta\right),$$

де θ — невідмий параметр.

Функція ln є монотонною, тому можна максимізувати не функцію L, а її натуральний логарифм $ln\left(L\right)$, оскільки максимуми обох функцій збігаються. Апроксимуємо ймовірність події логіт-функцією:

$$P\left(Y=1|x\right) = f\left(z\right) = \frac{1}{1+e^{-z}},$$

де $z = \sum_{i=1}^{N} w_i x_i$ і w_i — регресійні коефіцієнти.

У бінарному випадку логарифмічна функція правдоподібності дорівнює:

$$L\left(\vec{w}\right) = \sum_{i=1}^{N} \left(Y_i \ln P_i\left(\vec{w}\right) + \left(1 - Y_i\right) P_i \ln\left(1 - \vec{w}\right)\right)$$

Градієнт функції правдоподібності дорівнює:

$$g = \sum_{i=1}^{N} \left(Y_i - P_i\right) X_i,$$

а гессиан функції правдоподібності дорівнює:

$$H = -\sum_{i=1}^{N} P_i \left(1 - P_i\right) X_i^T X_i,$$

де X_i — рядок матриці пояснювальних змінних.

Завдяки негативній визначеності гессіану логарифмічна функція має єдиний глобальний максимум. Для розв'язання оптимізаційної задачі можна застосовувати метод Ньютона-Рафсона або градієнтні методи, як от: метод градієнта, метод покоординатного спуску або метод спряжених градієнтів.

Зауваження 5. Неважно помітити, що логістичну регресію можна подати у вигляді персептрона Розеблатта із сигмоїдною функцією активацією та вагами, що є регресійними коефіцієнтами.

Приклад 8. Для ілюстрації покажемо, як метод логістичної регресії можна використати для класифікації. Для цього вводиться точка відсічення, відносно якої здійснюється класифікація. Це ймовірність успіху, така що ймовірність успіху для навчальних вибірок з першого класу менша, ниж у вибірок другого класу. За замовченням, точка відсічення дорівнює 0,5.

Розглянемо одну задачу з медичної практики. Проаналізуємо вибірку хворих, які були прооперовані з приводу раку передміхурової залози. Після операції в крові цих хворих був виміряний показник CA-125 — відомий діагностичний маркер раку передміхурової залози. Оцінимо ймовірність успіху, яким в цій схемі є відсутність метастазів протягом 5 років після операції. Метод логістичної регресії дозволяє для кожного пацієнта обчислити ймовірність появи метастазів, а також розділити групи хворих на два класи: групу ризику, в якому ймовірність появи метастазів перевищує 0.5, і групу позитивного прогнозу, у членів якого вона не перевищує 0,5.

В аналізі використовувалися дані про маркер CA-125 и виживаність 60 хворих. Точка відсічення дорівнювала 0.5. Обчислення показали, що цій точці відповідає концентрація CA-125, що дорівнює 12,171.

Таблица 8.1. Специфічність і чутливість диференціальної діагностики за другою групою показників

Подія	N	0	1	Всього	Показник
0	38	0	38	100,00	Специфічність
1	5	17	22	77,27	Чутливість
Всього	43	17	60	91,67	Точність

8.3. Множинна логістична регресія

Множинна логістична регресія використовує лінійну функцію відклику $f(\beta_k, x_i)$, що оцінює ймовірність того, що i-та навчальна вибірка відповідає k-му результату:

$$f(\beta_k, x_i) = \vec{\beta}_k \vec{x}_i,$$

де $\vec{\beta}_k$ — рядок регресійних коефіцієнтів, що відповідає k-му результату, а $\vec{x}_i$ — i-та навчальна вибірка.

8.3.1. Сукупність незалежних бінарних моделей

Множинну іальну логістичну регресію з N результатами можна подати як сукупність $N-1$ незалежних моделей бінарної логістичної регресії, в яких кожному результату відповідає окрема множина пояснювальних змінних.

$$\ln \frac{P(Y_i = 1)}{P(Y_i = K)} = \beta_1 X_i,$$
$$\ln \frac{P(Y_i = 2)}{P(Y_i = K)} = \beta_2 X_i,$$
$$\ln \frac{P(Y_i = K-1)}{P(Y_i = K)} = \beta_{K-1} X_i.$$

Елементарні перетворення призводять до таких результатів:

$$P(Y_i = 1) = P(Y_i = K) e^{\boldsymbol{\beta}_1 \cdot \vec{X}_i},$$
$$P(Y_i = 2) = P(Y_i = K) e^{\boldsymbol{\beta}_2 \cdot \vec{X}_i},$$
$$\cdots\cdots$$
$$P(Y_i = K-1) = P(Y_i = K) e^{\boldsymbol{\beta}_{K-1} \cdot \vec{X}_i}.$$

Оскільки сума усіх ймовірностей повинна дорівнювати одиниці, отримуємо:

$$P(Y_i = K) = \frac{1}{1 + \sum k = 1^{K-1} e^{\beta_k X_i}}.$$

Отже,

$$P(Y_i = 1) = \frac{e^{\beta_1 \cdot \vec{X}_i}}{1 + \sum\limits_{k=1}^{K-1} e^{\beta_k \cdot \vec{X}_i}},$$

$$P(Y_i = 2) = \frac{e^{\beta_2 \cdot \vec{X}_i}}{1 + \sum\limits_{k=1}^{K-1} e^{\beta_k \cdot \vec{X}_i}},$$

$$\cdots\cdots$$

$$P(Y_i = K-1) = \frac{e^{\beta_{K-1} \cdot \vec{X}_i}}{1 + \sum\limits_{k=1}^{K-1} e^{\beta_k \cdot \vec{X}_i}}.$$

Як і в методі бінарної логістичної регресії, оцінка параметрів множинної логістичної регресії здійснюється за методом максимальної правдоподібності, тобто зводиться до розв'язання оптимізаційної задачі відносно регресійних коефіцієнтів.

8.3.2. Узагальнення бінарної моделі

Інший спосіб побудови моделі множинної логістичної регресії полягає у безпосередньому узагальненню бінарної моделі за допомогою множника нормалізації.

$$\ln P(Y_i = 1) = \beta_1 \cdot \vec{X}_i - \ln Z, \tag{8.1}$$

$$\ln P(Y_i = 2) = \beta_2 \cdot \vec{X}_i - \ln Z, \tag{8.2}$$

$$\cdots\cdots \tag{8.3}$$

$$\ln P(Y_i = K) = \beta_K \cdot \vec{X}_i - \ln Z. \tag{8.4}$$

Нормалізуючи множник дозволяє задовольнити умову:

$$\sum_{k=1}^{K} P(Y_i = k) = 1.$$

Шляхом елементарних перетворень легко отримати такі рівняння:

$$P(Y_i = 1) = \frac{1}{Z} e^{\beta_1 \cdot \vec{X}_i},$$

$$P(Y_i = 2) = \frac{1}{Z} e^{\beta_2 \cdot \vec{X}_i},$$

$$\cdots\cdots$$

$$P(Y_i = K) = \frac{1}{Z} e^{\beta_K \cdot \vec{X}_i}.$$

Суммуючи ці рівняння до прирівнюючи їх до одиниці, маємо:

$$1 = \sum_{k=1}^{K} P(Y_i = k) = \sum_{k=1}^{K} \frac{1}{Z} e^{\beta_k \cdot \vec{X}_i} = \frac{1}{Z} \sum_{k=1}^{K} e^{\beta_k \cdot \vec{X}_i}.$$

Таким чином, знаходимо множник нормалізації:

$$Z = \sum_{k=1}^{K} e^{\beta_k \cdot \vec{X}_i}.$$

За рівнянями (8.1)—(8.4), знаходимо ймовірності кожного результату:

$$P(Y_i = 1) = \frac{e^{\beta_1 \cdot \vec{X}_i}}{\sum_{k=1}^{K} e^{\beta_k \cdot \vec{X}_i}},$$

$$P(Y_i = 2) = \frac{e^{\beta_2 \cdot \vec{X}_i}}{\sum_{k=1}^{K} e^{\beta_k \cdot \vec{X}_i}},$$

$$\cdots\cdots$$

$$P(Y_i = K) = \frac{e^{\beta_K \cdot \vec{X}_i}}{\sum_{k=1}^{K} e^{\beta_k \cdot \vec{X}_i}}.$$

Інакше кажучи,

$$P(Y_i = c) = \frac{e^{\beta_c \cdot \vec{X}_i}}{\sum_{k=1}^{K} e^{\beta_k \cdot \vec{X}_i}}$$

Таким чином, задача знаходження регресійних коефіцієнтів зводиться до оптимізаційної задачі:

$$\arg\max_{\beta_k} S(c, \beta_1 \cdot \vec{X}_i, \ldots, \beta_K \cdot \vec{X}_i).$$

Глава 9

Метод найближчого сусіди

Перш, ніж розпочати аналіз методу найближчого сусіди, нагадаємо, що наші міркування завжди ґрунтуються на двох постулатах:

1. **Постулат про векторну модель:** об'єкт можна подати як елемент векторного простору ознак.
2. **Постулат про компактність:** переважна більшість об'єктів, що належать до одного класу, є більш близькими один до одного, ніж до об'єктів іншого класу, і лежать в області і з відносно простою межею.

Метод найближчого сусіди є найбільш інтуїтивно зрозумілим алгоритмом класифікації, адже порівняння за принципом подібності — це дуже розповсюджений і природний спосіб розпізнавання. Як кажуть: "Якщо хтось ходить как качка і крякає як качка, значить, це і є качка". За правилом найближчого сусіди об'єкт x належить тому класу, якому належить найближчий об'єкт із навчальної вибірки (прецедент).

Очевидно, що основними поняттями методу найближчого сусіди є поняття, що описують "близькість"об'єктів у просторі ознак. Отже, це можуть бути або метрика, або міра близькості.

Нагадаємо означення метрики.

Означення 26. Нехай X — довільна множина. Відображення $\rho : X \times X \to R^+$ називається метрикою, якщо $\forall x, y, z \in X$ воно має такі властивості (аксіоми метрики):

1. $\rho(x, y) = 0 \Leftrightarrow x = y$ (аксіома тотожності);
2. $\rho(x, y) = \rho(y, x)$ (аксіома симетрії);
3. $\rho(x, y)\,\rho(x, z) + \rho(z, y)$ (нерівність трикутника).

Означення 27. Значення метрики $\rho(x, y)$ називається відстанню меж векторами x і y.

Відповідно до постулату про векторну модель, кожному об'єкту можна поставити у однозначну відповідність вектор $\vec{x} = (x_1, x_2, ..., x_n)$ у векторному просторі ознак X і ввести у цьому просторі метрику, перетворивши його на метричний простір. Вістань між схожими об'єктами повинна бути малою, а між несхожими — великою. Отже, відносно кожного об'єкта y елементи навчальної вибірки можна упрядкувати за зростанням відстані до нього.

$$\rho(x_{(1)}, y) \leq \rho(x_{(2)}, y) \leq \cdots \leq \rho(x_{(n)}, y),$$

$x_{(i)}$ — i-й елемент варіаційного ряду, або i-та порядкова статистика. Інакше кажучи, $x_{(i)}$ — об'єкт навчальної вибірки, що є i-м сусідом объекта y.

Отже, задачу пошуку найближчого сусіди можна сформулювати так:

$$\arg\max_{x_i \in X} \rho\left(x_i, y\right). \tag{9.1}$$

9.1. Нормалізація даних

Зауважимо, що ознаки, які використовуються при обчисленні відстані (наприклад, за евклідовою метрикою або будь-якою іншою), можуть коливатися в досить широкому діапазоні. Якщо одна з ознак коливається в більш широкому діапазоні, ніж інші, вона буде мати більш значний вплив на відстань, ніж інші. Якщо ми виходимо з припущення, що ознаки мають однакову вагу, така ситуація є небажаною. Для того щоб уникнути її, перед обчисленням відстані виконують нормалізацію даних. Існує кілька способів нормалізації даним. Найбільш поширеними є мінімаксна нормалізація та стандартизація.

Позначимо мінімальне значення k-ї ознаки у навчальній виборці як

$$x_k^{\min} = \min_{i==1,...,n} x_k^i,$$

а максимальне як

$$x_k^{\max} = \max_{i==1,...,n} x_k^i.$$

Мінімаксна нормалізація k-ї ознаки i-го об'єкта здійснюється так:

$$\overline{x}_k^{(i)} = \frac{x_k - x_k^{\min}}{x_k^{\max} - x_k^{\min}}.$$

Позначимо середнє вибіркове k-ї ознаки як m_k:

$$m_k = \frac{1}{n} \sum_{i=1}^{n} x_k^{(i)},$$

а вибіркову оцінку стандартного відхилення k-ї ознаки як s_k

$$s_k = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} \left(x_k^{(i)} - m_k\right)^2}$$

Стандартизація k-ї ознаки здійснюється так:

$$\overline{x}_k = \frac{x_k - m_k}{s_k}.$$

Інакше кажучи, m_k — це середнє значення k-ї ознаки у навчальній виборці, а s_k — його стандартне відхилення.

Після нормалізації всі ознаки об'єктів будуть змінюватися у відрізку $[0, 1]$. Іноді зручно, щоб ознаки коливалися в інтервалі $[-1, 1]$ із центром в нулі. Для цього використовують іншу форму мінімаксної нормалізації:

$$\overline{x}_k^{(i)} = \frac{x_k - \frac{x_k^{\max} + x_k^{\min}}{2}}{x_k^{\max} - x_k^{\min}}.$$

Після стандартизації всі ознаки об'єктів будуть мати наближено стандартизований нормальний розподіл, тобто приблизно нормальний розподіл із нульовим математичним сподіванням і одиничною дисперсією.

Нормалізація і стандартизація даних урівноважують вплив кожної ознаки. Але незважаючи на те, що обидва ці методи є достатньо ефективними. Перевагу слід віддати стандартизації, тому що вона дає можливість застосувати статистичний аналіз і отримати обґрунтовані оцінки щодо відхилення об'єктів один від одного (наприклад, виявити так звані викиди — об'єкти, що статистично значущо відрізняються від об'єктів того ж самого класу).

9.1.1. Метод k найближчих сусідів

Переваги і недоліки методу найближчого сусіди очевидні. Основні переваги — простота і швидкість. Основний недолік — нестійкість. Дійсно, зважаючи на те, що об'єкти навчальної вибірки є випадковими, нема жодних гарантій, що найближчий прецедент – це не випадковий сусід, а неодмінний супутник об'єкта, що класифікується. З цієї причини для підвищення точності класифікації часто визначають не одного, а k його наближчих сусідів і відносять об'єкт до того класу, до якого належить більшість серед цих сусідів. Цю схему можна інтерпретувати як голосування.

Є два способи голосування: без ваг та з вагами. Якщо усі об'єкти вважаються рівноправними, можно нехтувати вагами і застосувати алгоритм (9.1). Якщо ж треба урахувати вагу, то доцільно зважити на фізичний закон обернених квадратів:

$$\arg\max_{x_i \in X} \sum_{i=1}^{n} \frac{1}{\rho^2(x_i, y)},$$

9.2. Вибір числа сусідів k

Вибір числа сусідів — важлива задача, від якої залежить стійкість алгоритму. Єдиного рецепту вибору параметра k немає, оскільки він сильно залежить від даних. Якщо вибрати граничні значення, то метод найближчих сусідів стає або нестійким ($k = 1$), або виродженим ($k = m$), тобто втрачає здатність до узагальнення. Для пошуку параметра k, близького до оптимального, як правило, використовують метод кросс-валідації по k-блокам. Для цього навчальна вибірка випадковим чином розбивається на k диз'юнктних блоків. Один з блоків стає тестовою вибіркою, а решта — навчальною. Позначимо вихідну навчальну вибірку як X, а вибірку, що грає роль тестової як $\vec{x}_k$. Тоді навчальна вибірка в ході кросс-валідації — це різниця $X \setminus \vec{x}_k$ Шуканий параметр k визначається як розв'язок задачі

$$\arg\min_k \epsilon_k,$$

де $\epsilon_k = \frac{1}{k}\sum_{i=1}^{k} \mathcal{J}\left(X \setminus \vec{x}_k, x_k\right)$ — середня помилка класифікації на тестових вибірках.

9.3. Розпізнавання викидів

На практиці постулат по компактність виконується лише наближено, тобто деякі об'єкти можуть бути оточені сусідами того самого класу, а деякі об'єкти можуть лежати далеко від інших.

Означення 28. Об'єкт, що оточений сусідами того самого класу, називається ***типовим***.

Означення 29. Об'єкт, що віддалений від інших об'єктів того самого класу, називається ***викидами***, або ***шумом***.

Розпізнавання типових об'єктів і шуму має велике значення. По-перше, типові об'єкти можна видалили з навчальної вибірки без зменшення точності класифікації. Це дозволяє зменшити розмір задачі. Во-друге, шум сильно впливає на точність класифікації, тому видалення шуму значно підвищує специфичність і чутливість алгоритму.

Часто об'єкти задаються не вектором, а матрицею, де стовпчики матриці є випадковими вібірками (наприклад, набором вимірювань k-ї ознаки). Це типова ситуація в медичних дослідженнях, коли у пацієнта беруть n клітин, у яких вимірюють m ознак. Визначення викидів і упорядкування таких об'єктів стає набагато складнішою задачею. Для розв'язання цієї задачі використовується, зокрема, поняття статистичної глибини.

Глава 10

Методи розпізнавання за статистичною глибиною

Як показано у попередній главі, задачі класифікації об'єктів часто зводяться до ранжування багатовимірних вибірок. У відповідності із загальновизнаною термінологією, методи багатовимірного ранжування розділяються на маргінальні, редуковані, часткові і умовні. Маргінальні методи упорядковують виборки за окремими компонентами. Редуковані методи обчислюють відстань кожної вибірки від центру розподілу. Часткове ранжування має на увазі розділення вибірок на групи однакових вибірок. В умовних методах здійснюється упорядкування вибірок за обраним компонентом, що впливає на інші.

Велику популярність серед методів багатовимірного ранжування отримав підхід, що ґрунтується на концепції статистичної глибини вибірок відносно центру розподілу і відповідних методах пілінгу. Ці методи дозволяють урахувати геометричні властивості багатовимірних розподілів і є відносно простими для обчислень. Розглянемо один з них, що використовує еліпси Петуніна.

10.1. Еліпсоїд Петуніна

Не обмежуючи загальності, опишемо алгоритм побудови еліпса Петуніна на площині, а потім перенесемо його в простір R^m при $m > 2$. Вихідними даними для алгоритму є множина багатомірних векторів $M_n = \{\vec{x}_1, ..., \vec{x}_n\}$, де $\vec{x}_n = (x_n, y_n)$.

Еліпс Петуніна. На першому етапі побудуємо опуклу оболонку точок $M_n = \{(x_1, y_1), ..., (x_n, y_n)\}$. Знайдемо вершини опуклої оболонки (x_k, y_k) и (x_l, y_l), що лежать на діаметрі опуклої оболонки, тобто вершини, найбільш віддалені одна від одної. З'єднаємо точки (x_k, y_k) і (x_l, y_l) відрізком L. Знайдемо вершини опуклої оболонки (x_r, y_r) і (x_q, y_q), найбільш віддалені від L. З'єднаємо точки (x_r, y_r) і (x_q, y_q) відрізками L_1 и L_2, паралельними до відрізку L. Проведемо через точки (x_k, y_k) і (x_l, y_l) відрізки L_3 и L_4, перпендикуляр-

ні до відрізку L. Перетини відрізків L_1, L_2, L_3 і L_4 утворюють прямокутник П, сторони якого мають довжини a і b.

Будемо вважати, що $a \leq b$. Переведемо лівий нижній кут прямокутника в початок нової системи координат з осями Ox' и Oy' за допомогою повороту і паралельного переносу. Точки (x_1, y_1), (x_2, y_2), ..., (x_n, y_n) перейдуть в точки (x'_1, y'_1), (x'_2, y'_2), ..., (x'_n, y'_n). Відобразимо точки (x'_1, y'_1), (x'_2, y'_2), ..., (x'_n, y'_n) в точки $(\alpha x'_1, y'_1)$, $(\alpha x'_2, y'_2)$, ..., $(\alpha x'_n, y'_n)$, де $\alpha = \frac{a}{b}$. В результаті отримуємо сукупність точок, що лежать у квадраті S.

Обчислимо центр (x'_0, y'_0) квадрата S і знайдемо відстані $r_1, r_2, ..., r_n$ від нього до кожної точки $(\alpha x'_1, y'_1)$, $(\alpha x'_2, y'_2)$, ..., $(\alpha x'_n, y'_n)$. Найбільше число $R = \max(r_1, r_2, ..., r_n)$ визначає коло з центром в точці (x'_0, y'_0) і радіусом R. В результаті всі точки $(\alpha x'_1, y'_1)$, $(\alpha x'_2, y'_2)$, ..., $(\alpha x'_n, y'_n)$ опиняються всередині кола з радіусом R. Розтягуючи це коло вздовж осі Ox' з коефіцієнтом $\beta = \frac{1}{\alpha}$ і виконуючи зворотні перетворення повороту і переносу, отримуємо еліпс Петуніна.

Еліпсоїд Петуніна. У m-мірному просторі на першому кроці знайдемо дві вершини опуклої оболонки $\vec{x}_k$ і $\vec{x}_l$, що лежать на її діаметрі. З'єднаємо точки $\vec{x}_k$ і $\vec{x}_l$ відрізком L. Повернемо і перенесемо систему координат, щоб діаметр опуклої оболонки лежав на осі Ox'_1. Побудуємо найменший прямокутний паралелепіпед, що містить точки $\vec{x}'_1, ..., \vec{x}'_n$.

Стискаючи прямокутний паралелепіпед, відобразимо точки в гіперкуб. Знайдемо центр $\vec{x}_0$ гіперкуба й обчислимо відстані $r_1, r_2, ..., r_n$ від нього до кожної точки. Знайдемо найбільше число $R = \max(r_1, r_2, ..., r_n)$ і побудуємо гіперкулю з центром у точці $\vec{x}_0$ і радіусом R. Застосовуючи до цієї гіперкулі зворотні операції розтягування, повороту і переносу, одержимо еліпсоїд Петуніна в m-вимірному просторі.

У результаті на кожнім вкладеному еліпсоїді лежить по одній точці з вибірки, тобто відбувається їхнє ранжування.

Теорема 1. Якщо вектори $\vec{x}_1, \vec{x}_2, ..., \vec{x}_n$ є незалежними й однаково розподіленими випадковими векторами з генеральної сукупності G, E_n — довірчий еліпсоїд, що містить точки $\vec{x}_1, \vec{x}_2, ..., \vec{x}_n$, і $\overrightarrow{x}_{n+1} \in G$, то $P(\vec{x}_{n+1} \in E_n) = \frac{n}{n+1}$.

10.2. Нова міра близькості між багатомірними вибірками

За аналогією з p-статистикою, для одномірного випадку сконструюємо міру близькості, використовуючи як варіаційний ряд багатомірні вибірки, побудовані при ранжуванні за допомогою еліпсоїдів Петуніна. Варіаційному ряду вибірок $\vec{x}_{(1)} \preceq \vec{x}_{(2)} \preceq ... \preceq \vec{x}_{(n)}$ поставимо у відповідність послідовність вкладених еліпсоїдів $E_{(1)} \subset E_{(2)} \subset ... \subset E_{(n)}$. Як випливає з теореми, імовірність того, вибірка $\vec{x}$ з багатомірної генеральної сукупності G задовольняє умові

$\vec{x}_{(i)} \preceq \vec{x} \preceq \vec{x}_{(j)}$, дорівнює імовірності потрапити між еліпсами $E_{(i)}$ і $E_{(j)}$, тобто $\frac{j-i}{n+1}$.

Ця обставина дозволяє побудувати p-статистику для багатомірного випадку.

Нагадаємо основні визначення. Нехай $x = (x_1, ..., x_n) \in G$ — вибірка з генеральної сукупності G і p — деякий відомий чи невідомий показник, значення якого можуть залежати від вибірки x. Розглянемо дві неперервні функції $a(u_1, ..., u_n)$ і $b(u_1, ..., u_n)$ від n змінних $u_1, ..., u_n$, що задовольняють нерівності $a(u_1, ..., u_n) \leq b(u_1, ..., u_n)\ \forall (u_1, ..., u_n) \in R^n$. Випадковий інтервал $(a(u_1, ..., u_n), b(u_1, ..., u_n)) = (a, b)$ називається довірчим інтервалом для p, що відповідає рівню значущості β, якщо $P(p \in (a, b)) = 1 - \beta, (0 \leq \beta \leq 1)$; при цьому числа $a = a(x_1, ..., x_n)$, $b = b(x_1, ..., x_n)$ називаються довірчими межами для p, що відповідають рівню значущості β.

Означення 30. Інтервали $(a_k, b_k) = (a_k(x_1, ..., x_k), b_k(x_1, ..., x_k))$, $k = 1, 2, ...$ називаються ***асимптотичними інтервалами*** для показників $p_i, i = 1, 2, ..., k, ...$, що відповідають рівню значущості β, якщо

$$\lim_{k \to \infty} P(p_k \in (a_k(x_1, ..., x_k), b_k(x_1, ..., x_k))) = 1 - \beta, \tag{10.1}$$

а кінці цих інтервалів $a_k(x_1, ..., x_k)$ и $b_k(x_1, ..., x_k)$ називаються асимптотичними довірчими межами.

Означення 31. Величина β називається ***асимптотичним рівнем значущості*** послідовності (a_k, b_k), $k = 1, 2, ...$

Означення 32. Якщо $p_k = p\ \forall k = 1, 2, ...$, то інтервал (a_k, b_k) називається ***асимптотичним довірчим інтервалом*** показника p, а величина β — ***асимптотичним рівнем значущості інтервала*** (a_k, b_k).

Позначимо як H гіпотезу про рівність непрервних функцій розподілу $F_1(u)$ и $F_2(u)$ генеральних совокупностей багатовимірних випадкових величин G_1 и G_1 відповідно. Нехай $(\vec{x}_1, \vec{x}_2, ..., \vec{x}_n) \in G_1$ и $(\vec{x}'_1, \vec{x}'_2, ..., \vec{x}'_n) \in G_2$, $\vec{x}_{(1)} \preceq \vec{x}_{(2)} \preceq ... \preceq \vec{x}_{(n)}$, $\vec{x}'_{(1)} \preceq \vec{x}'_{(2)} \preceq ... \preceq \vec{x}'_{(n)}$ — відповідні варіаційні ряди. Припустимо, що $F_1(u)$=$F_2(u)$. Позначимо як A_{ij}, $k = 1, 2, ..., m$ випадкову подію, яка полягає у тому, що $\vec{x}'_k$ потрапляє в область $E_{(j)} \backslash E_{(i)}$. Якщо $F_1(u)$=$F_2(u)$, імовірність p_{ij} цієї події обчислюється за формулою:

$$p_{ij}^{(n)} = \frac{j-i}{n+1}. \tag{10.2}$$

Покладемо

$$p_{ij}^{(1)} = \frac{h_{ij}m + g^2/2 - g\sqrt{h_{ij}(1-h_{ij})m + g^2/4}}{m + g^2}, \tag{10.3}$$

$$p_{ij}^{(2)} = \frac{h_{ij}m + g^2/2 + g\sqrt{h_{ij}(1-h_{ij})m + g^2/4}}{m+g^2}, \qquad (10.4)$$

де h_{ij} — частота події A_{ij} у m випробуваннях і g=3. Розглянемо довірчі інтервали $I_{ij}^{(n)} = \left(p_{ij}^{(1)}, p_{ij}^{(2)}\right)$. Загальна кількість інтервалів $I_{ij}^{(n)}$ дорівнює $N = \frac{n(n-1)}{2}$. Позначимо через L — кількість тих інтервалів $I_{ij}^{(n)}$, що містять імовірності $p_{ij}^{(n)}$. Покладемо $h = \rho(\vec{x}, \vec{x}') = \frac{L}{N}$. Оскільки h — частота випадкової події $B = \left\{p_{ij}^{(n)} \in I_{ij}^{(n)}\right\}$, що має імовірність $p(B) = 1 - \beta$, то, поклавши $h_{ij} = h, m = N$ і $g = 3$ у формулах (10.3),(10.4), одержуємо довірчий інтервал $I^{(n)} = \left(p^{(1)}, p^{(2)}\right)$ для імовірності $p(B)$. Статистика h називається p-статистикою. Вона є мірою близькості $\rho(\vec{x}, \vec{x}')$ між вибірками $\vec{x}$ і $\vec{x}'$.

Якщо гіпотеза H є істинною, то схема випробувань, у якій можуть з'являтися події $A_{ij}^{(k)}$, називається узагальненою схемою Бернуллі, а якщо гіпотеза H є хибною, то схема випробувань називається модифікованою схемою Бернуллі. У загальному випадку, коли може бути істинною будь-яка гіпотеза, як $F_1(u) = F_2(u)$, так і $F_1(u), \neq F_2(u)$, ця схема випробувань називається MP-схемою.

Теорема 7. *Якщо в узагальненій схемі випробувань Бернуллі виконуються умови* $n = m$, $0 < \lim\limits_{n\to\infty} p_{ij}^{(n)} = p_0 < 1$ *і* $0 < \lim\limits_{n\to\infty} \frac{i}{n+1} = p^* < 1$, *то асимптотичний рівень значущості* β *послідовності довірчих інтервалів* $I_{ij}^{(n)}$ *для імовірностей* $p_{ij}^{(n)}$, *побудованих за правилом 3s, не перевищує 0,05.*

Теорема 8. *Якщо вибірки* $\vec{x} = (\vec{x}_1, ..., \vec{x}_n) \in G_1$ *і* $\vec{x}' = (\vec{x}'_1, ..., \vec{x}'_m) \in G_2$ *мають однаковий обсяг, те асимптотичний рівень значущості інтервалу* $I^{(n)} = \left(p^{(1)}, p^{(2)}\right)$, *побудований за правилом 3s при* $g = 3$ *за допомогою формул (10.3), (10.4), не перевищує 0,05.*

10.3. Обчислювальний експеримент

В межах обчислювального експерименту за допомогою статистичного пакета R було проведено попарне порівняння вибірок з генеральних сукупностей, що мають двовимірний нормальний розподіл, вектори математичних сподівань яких дорівнюють (0,0), (1,1), (2,2) і (3,3) відповідно, а коваріаційна матриця є одиничною. Крім того, було проведено попарне порівняння вибірок з генеральних сукупностей, що мають двовимірний нормальний розподіл, вектори математичних сподівань яких $(0, 0)$, а на діагоналі коваріаційної матриці стоять дисперсії $(1, 1)$, $(2, 2)$, $(3, 3)$ і $(4, 4)$ (позадіагональні елементи дорівнюють нулю). Для експериментів генерувалися вибірки обсягом 300 елементів і обчислена середня міра близькості.

Таблиця 1. — Усереднена міра близькості між вибірками

Центри	Міра близькості	Дисперсії	Міра близькості
(0,0)—(0,0)	0,922	(1,1)—(1,1)	0,928
(0,0)—(1,1)	0,395	(1,1)—(2,2)	0,428
(0,0)—(2,2)	0,120	(1,1)—(3,3)	0,289
(0,0)—(3,3)	0,063	(1,1)—(4,4)	0,223

Як бачимо, міра близькості монотонно убуває в міру збільшення відстані між центрами розподілів при фіксованій дисперсії, а також у міру збільшення дисперсії при фіксованому центрі.

Зауваження 6. Запропонована міра близькості дозволяє перевірити гіпотезу зсуву і масштабу при однаковому розподілі кутів векторів, проведених з центра розподілу до точок, але, наприклад, якщо точки двох генеральних сукупностей розподілені однаково, але в протилежних секторах кола, то для їхнього правильного розпізнавання необхідно враховувати розподіл кутів векторів.

Литература

[1] *Айзерман М.А., Браверман Э.М., Розоноэр Л.И.* Метод потенциальных функций в теории обучения машин. — М.: Наука, 1970.

[2] *Воронцов К.В.* Машинное обучение. (Курс лекций). — М.: ВмиК МГУ, 2009.

[3] *Вапник В.Н., Червоненкис А.Я.* Теория распознавания образов. — М.: Наука, 1974.

[4] *Главач В., Шлезингер М.И.* Десять лекций по статистическому и структурному распознаванию образов. — К.: Наукова думка, 2004.

[5] *Дуда Р., Харт П.* Распознавание образов и анализ сцен. — М.: Мир, 1976.

[6] *Ермольев Ю.М., Ляшко И.И., Михалевич В.С., Тюптя В.И.* Математические методы исследования операций. — К: Вища школа, 1979.

[7] *Клюшин Д.А., Петунин Ю.И.* Доказательная медицина: применение статистических методов. — М.: Вильямс, 2008.

[8] *Мерков А.Б.* Распознавание образов. Введение в методы статистического обучения. — М.: Едиториал УРСС, 2011.

[9] *Местецкий Л.М.* Математические методы распознавания образов. (Курс лекций). — М.: ВмиК МГУ 2004.

[10] *Лепский А.Е., Броневич А.Г.* Математические методы распознавания образов. (Курс лекций). — Таганрог: Южный федеральный университет, 2009.

[11] *Webb A.* Statistical Pattern Recignition. — NY: John Wiley and Sons, 2002.

[12] *Маннинг К., Рагхаван П., Шютце Х.* Введение в информационный поиск. — М: Вильямс, 2011.

[13] *Уоссермен Ф.* Нейрокомпьютерная техника: Теория и практика. — М.: Мир, 1992.

[14] *Фукунага К.* Введение в статистическую теорию распознавания образов. — М.: Наука, 1979.

[15] *Хайкин С.* Нейронные сети: Полный курс. — М.: «Вильямс», 2006.

[16] *Abe S.* Support vector machines for pattern classification. — London: Springer, 2009.

[17] *Fisher R.A.* The Use of Multiple Measurements in Taxonomic Problems // Annals of Eugenics. — 1936 Т. 7. — С. 179-188.

[18] *Mika S., Raetsch G., Weston J., Schoelkopf B., Muller K.-R.* Fisher discriminant analysis with kernels. - In: Neural Networks for Signal Processing IX, Y.-H. Hu, J. Larsen, E. Wilson, and S. Douglas, Eds. Piscataway, NJ: IEEE, 1999. — P. 41–48.

[19] *Robbins H., Monro S.* A stochastic approximation method // Ann. Math. Statist. — 1951. — v. 51. — P.400–407.