Міністерство освіти і науки України
Національний університет “Києво-Могилянська академія”
Факультет інформатики
Кафедра прикладної математики

Магістерська робота

освітній ступінь — магістр

на тему: **«Передбачувальна аналітика в медицині»**

Виконала: студентка 2-го року
навчання,
освітньо-наукової програми
«Прикладна математика», 113.
Фісун Єлизавета Борисівна.
Керівник Крюкова Г.В.
кандидат фіз-мат. наук, доцент.

Рецензент_____
(прізвище та ініціали)
Магістерська робота захищена
з оцінкою_____
Секретар ЕК_____
«____» _____ 2024р.

Київ — 2024

Зміст

ВСТУП **3**
РОЗДІЛ 1: Аналіз виживання **4**
1.1 Censored data 4
1.2 Задача регресії, Крива Каплана-Майєра 6
1.3 Довірчі інтервали, log-rank test, Cox regression 8
РОЗДІЛ 2: Задача класифікації **10**
2.1 Дані, експресія генів 10
2.2 Тестування гіпотез 12
2.3 Візуалізація даних 13
2.4 Виявлення викидів 14
2.5 “Прокляття розмірності” 17
2.6 Mutual information 19
2.7 Two sample Kolmogorov-Smirnov test 21
2.8 Алгоритми зменшення розмірності 22
2.9 Валідація моделі 25
РОЗДІЛ 3: Експерименти **29**
3.1 Підготовка даних 29
3.2 Навчання базових моделей та валідація 30
3.3 Застосування feature selection методів та валідація 31
ВИСНОВКИ **32**
Література **33**

ВСТУП

Передбачувальна аналітика має величезний потенціал для розвитку в галузі медицини. Аналізуючи складні закономірності в медичних даних (генетичних профілях), прогностичні моделі можуть ідентифікувати важливі взаємозв'язки між генами які більш за все провокують хворобу. Окрім цього передбачувальна аналітика має широке застосування для розробки персоналізованого плану лікування на основі генетичного профілю пацієнта та історії хвороби. Незважаючи на те що передбачувальна аналітика дає багато можливостей для розвитку індустрії охорони здоров'я, вона стикається з низкою проблем пов'язаних з даними, а саме складність та варіативність, медичні дані часто отримують з різних джерел. Медичні дані дуже часто можуть містити помилки експериментів, бути неповними, наявність censored data, окрім цього відсутні дані для конкретних хвороб, (stratified sampling problem), що у свою чергу впливає на точність передбачувальної моделі. Висока розмірність даних та досить обмежена кількість зразків призводить до такої проблеми як перенавчання моделі.

За мету даної роботи було поставлено показати особливості моделювання в медицині та відмінності від іншої області застосування, наприклад веб-аналітики та розробити класифікатор для даних експресії генів по Ангельман синдрому.

РОЗДІЛ 1: Аналіз виживання

Передбачувальна аналітика в медицині застосовується для вирішення багатьох задач, а саме виявлення осіб з високим ризиком розвитку певних захворювань, що дозволяє попередити хворобу, прогнозування часу виживання після діагностування, аналіз ефективності різних підходів лікування для специфічних груп пацієнтів, що дозволяє розробляти персоналізований план лікування. Розглянемо дві основні задачі з точки зору машинного навчання це задача регресії та задача класифікації та які виникають проблеми при аналізі медичних даних. Задача регресії полягає в тому щоб визначити зв'язки між залежною змінною y та предикторами x1,x2,...,xn та передбачити неперервну цільову змінну. Нехай ми маємо медичне дослідження, яке тривало 5 років, в якому індивідів лікували від раку, та ми хочемо дати відповідь на наступні питання: скільки часу людина проживе від моменту діагностування до настання події (смерть) враховуючи різні показники здоров'я, чи є різниця рівня виживання між різними групами дослідження (стать, вік інші). На перший погляд здається, що можна застосувати лінійну регресію до даної задачі, але тут виникає певна складність пов'язана з особливістю даних. На щастя деякі пацієнти дожили до кінця дослідження, час виживання для цих пацієнтів є цензурованим (censored data), хоча ми точно знаємо, що вони прожили як мінімум 5 років, але не знаємо справжнього значення. Природа цензурованих даних може бути різною, розглянемо три типи.

1.1 Censored data

Для кожного індивідуму ми розглядаємо час виживання T - survival time (час настання події, наприклад час коли пацієнт помер) та цензорний час C - censoring time (час коли пацієнт вийшов з дослідження або дослідження закінчилось). [1]

Нехай Y - час спостереження, випадкова величина, яка визначається наступним чином

$$Y = min(T, C)$$

Іншими словами якщо подія відбулась перед цензорним часом ($T < C$), тоді ми спостерігаємо час виживання T. Однак якщо ($T > C$), то ми спостерігаємо цензорний час. Також необхідно визначити індикаторну функцію

$$\delta = \begin{cases} 1 & \text{if } T \leq C \\ 0 & \text{if } T > C \end{cases}$$

Right censoring якщо $T \geq Y$ (реальний час виживання як мінімум час дослідження, найбільш частий). [2]

Left censoring якщо $T \leq Y$ (подія вже відбулась до початку дослідження, достатньо рідкісний тип) [3]

Interval censoring якщо ми знаємо, що подія відбулась в певний проміжок часу, але ми не знаємо точно коли.

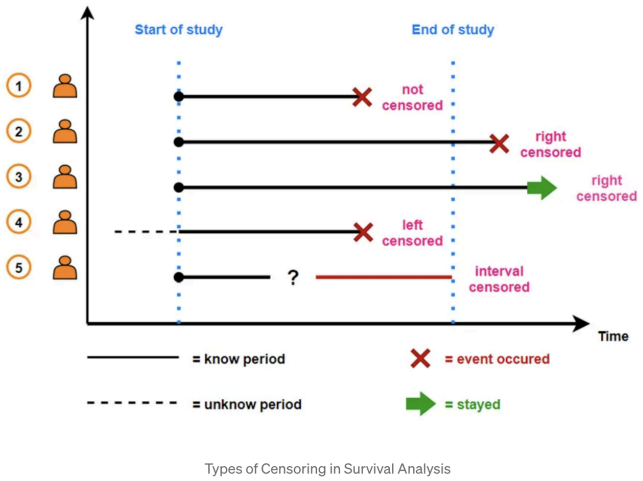

Рис. 1 Типи censored data [4]

Нехтувати такими кейсами є неправильним, адже це є цінна інформація для прогностичного моделювання, але виникає питання як лінійна регресія сприйматиме такі значення. Лінійна регресія ігноруватиме індикаторну

функцію, точніше сприйматиме що всі семпли мають статус 1. Для даної задачі варто застосовувати Kaplan-Meier estimator.

1.2 Задача регресії, Крива Каплана-Майєра

Функція виживання (survival function)

$$S(t) = Pr(T > t)$$

ймовірність, що індивідум проживе більше якогось визначеного часу t, T - випадкова величина, що характеризує час до настання події. Також функцію виживання можна визначити по іншому. [1]

$$S(t) = 1 - F(t)$$

F(t) - кумулятивна функція розподілу, яка виражає ймовірність, що подія відбулась раніше часу t.

$$F(t) = P(T \leq t)$$

S(t) не зростаюча функція відносно часу, приймає значення від 0 до 1. Наприклад ми досліджуємо час виживання пацієнтів після лікування, функція виживання S(6)=0.8, означатиме що з ймовірністю 80% пацієнт проживе як мінімум 6 місяців після лікування.

Як оцінити S(t) враховуючи той факт, що у датасеті присутні цензуровані дані?

Нехай $d_1 < d_2 < \ldots < d_k$, де К - час смерті серед не цензурованих даних та q_k- кількість пацієнтів, які померли в момент часу d_k. Для $k = 1, \ldots, K$ нехай r_k - це кількість живих та присутніх у дослідженні пацієнтів до моменту d_k; це пацієнти під ризиком. За формулою повної ймовірності

$$P(A) = P(A|B)P(B) + P(A|B^c)P(B^c)$$

$$Pr(T > d_k) = Pr(T > d_k | T > d_{k-1}) Pr(T > d_{k-1}) +$$

$$+ Pr(T > d_k | T \leq d_{k-1}) Pr(T \leq d_{k-1})$$

Враховуючи, що $d_{k-1} < d_k$ випливає $Pr(T > d_k | T \leq d_{k-1}) = 0$, оскільки це неможливо щоб пацієнт прожив більше ніж d_k, якщо він не вижив раніше ніж d_{k-1}. Тому маємо

$$S(d_k) = Pr(T > d_k) = Pr(T > d_k | T > d_{k-1}) Pr(T > d_{k-1})$$

застосуємо функцію виживання знову

$$S(d_k) = Pr(T > d_k | T > d_{k-1}) S(d_{k-1})$$

З цього випливає наступне

$$S(d_k) = Pr(T > d_k | T > d_{k-1}) \times \ldots \times Pr(T > d_2 | T > d_1) Pr(T > d_1)$$

Для оцінки кожного множника введемо наступну функцію, яка показує частину пацієнтів під ризиком які вижили після d_j

$$\widehat{Pr}(T > d_j | T > d_{j-1}) = (r_j - q_j)/r_j$$

Отже, Kaplan-Meier для оцінки функції виживання виглядає наступним чином

$$\widehat{S}(d_k) = \prod_{j=1}^{k} \left(\frac{r_j - q_j}{r_j}\right)$$

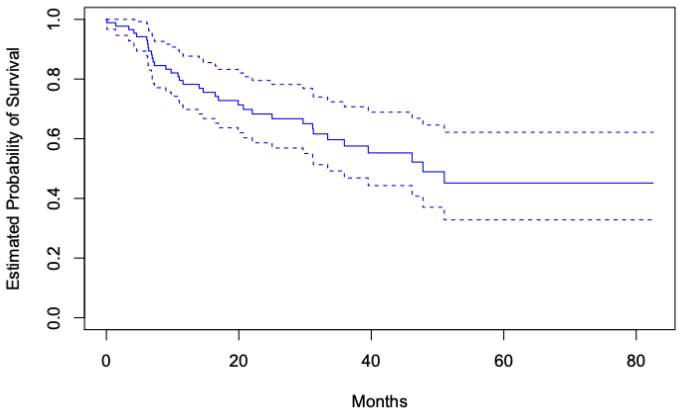

Рис.2 Крива виживання Каплана–Майєра [1]

1.3 Довірчі інтервали, log-rank test, Cox regression

Графік Каплана-Майера відображає функцію виживання, насправді форма цієї кривої показує тенденцію, чим більше “сходинок”, тим вища частота подій, а значить гірший рівень виживання. І навпаки більш плоский схил показує нижчу частоту подій, отже високий рівень виживання. На даному графіку можемо побачити що ймовірність прожити більше 20 місяців складає 71%. Цензуровані дані на графіку відмічаються вертикальною лінією на проміжку виживання. Також можна оцінити довірчі інтервали, які вказують на діапазон, у якому можна очікувати, що знаходиться істинна функція виживання, з певною ймовірністю (зазвичай 95%), це штрихові лінії на рис.2. Розрахувати такі інтервали можна за допомогою формули Greenwood. [5]

$$\widehat{S_{KM}}(t) \pm 1.96\sqrt{\widehat{Var}[\widehat{S_{KM}}(t)]}$$

$$\widehat{Var}[\widehat{S_{KM}}(t)] = (\widehat{S_{KM}}(t))^2 \sum_{f:t_{(f)}\le t} [\frac{m_f}{n_f(n_f - m_f)}]$$

$t_{(f)}$ – впорядкований час настання події

m_f – кількість подій на момент часу $t_{(f)}$

n_f – кількість під ризиком на момент часу $t_{(f)}$

1.96 це критичне значення для точності 95% для нормального розподілу. Окрім цього досить часто визначають таку метрику як медіана виживання це час коли очікується що половина пацієнтів виживуть, тобто ймовірність вижити більше якогось часу t це 50%. Також можна представляти декілька кривих Каплана-Майера, що представляють різні групи (чоловіки та жінки, пацієнти, що приймали таблетку А та пацієнти що приймали таблетку Б та ін.) для порівняння патернів. Якщо криві паралельні, це свідчить про те, що групи мають однакові показники виживання. Якщо криві розходяться

або перетинаються, це вказує на відмінності у виживанні між групами.

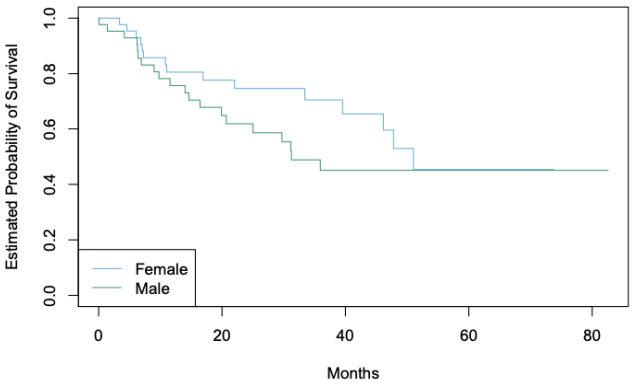

Рис.3 Криві Каплана-Майєра для чоловіків та жінок [1]

Графік Каплана-Майєра дає чудову візуалізацію різниці між групами, але якщо ми хочемо кількісно оцінити цю різницю, потрібно застосувати статистичні тести, а саме log rank test. Він тестує нульову гіпотезу про те що немає різниці між кривими виживання популяції (тобто ймовірність події, що відбудеться в будь-який момент часу, однакова для кожної популяції). Відповідно альтернативна гіпотеза полягає у тому що обидві групи мають різні криві розподілу. По суті, log rank test перевіряє, чи є криві виживання ідентичними (перекриваються) чи ні.

РОЗДІЛ 2: Задача класифікації

У цьому розділі розглянемо задачу класифікації та які виникають особливості її вирішення у медицині.

2.1 Дані, експресія генів

Датасет складається з 9 різних експериментів це 60 семплів (42 - миші, 18 - люди), source - IRB Barcelona, bioteque. Датасет це матриця експресії генів. Експресія генів — це процес, за допомогою якого інформація, закодована в гені, використовується для синтезу функціонального продукту, такого як білок або РНК. Експресія генів була отримана методом секвенування РНК. Матриця показує зміни в експресії генів між двома групами (контрольна група та Ангельман синдром) та наскільки ці відмінності значущі.

Кожен рядок матриці відповідає окремому гену. Це може бути ідентифікатор гена або його назва. Стовпці з префіксом Control та AS відображають контрольну групу та Ангельман синдром групу.

	ALIAS	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj	Control_Neu_076_merged	Control_Neu_037A	Control_Neu_023
0	A1BG	39.526470	0.041215	0.493403	0.083532	0.933429	0.999849	46.865182	40.529608	28.710821
1	A1BG-AS1	2.464391	0.467102	2.008447	0.232569	0.816096	0.999849	3.712094	2.251645	0.000000
2	A1CF	2.573820	-0.557334	1.865214	-0.298804	0.765089	0.999849	0.464012	9.006580	0.000000
3	A2M	15.630055	1.451297	0.938394	1.546575	0.121966	0.999849	3.712094	13.509869	8.613246
4	A2M-AS1	8.877927	-1.312203	1.219467	-1.076046	0.281907	0.999849	20.880527	0.000000	17.226492

Рис. 4 Приклад даних з експерименту

По суті, диференціальний аналіз експресії генів спрямований на порівняння середньої експресії гена в групі А з середньою експресією цього гена в групі В.

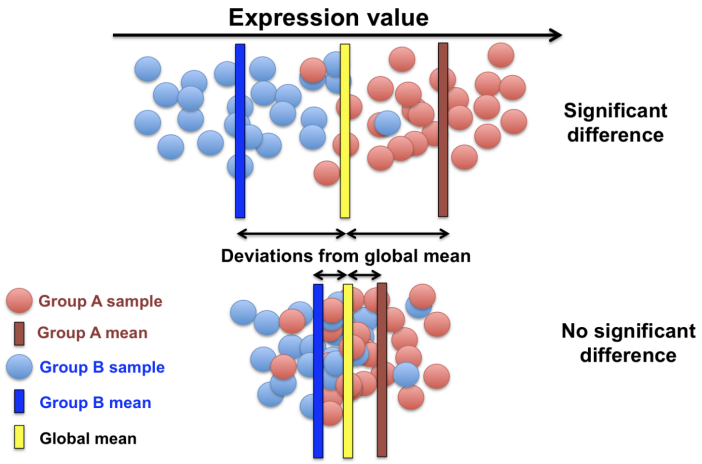

Рис.5 Порівняння середньої експресії гена між різними групами [6]

Log2FoldChange вказує на зміну експресії гена між двома умовами, додатній Log2FoldChange відображає збільшення експресії гена, від'ємний відповідно зниження експресії. Наприклад, logFC = 2 означає, що експресія гена збільшилася в 4 рази, а logFC = -1 означає, що експресія гена зменшилася вдвічі.

P-value використовується для статистичної перевірки гіпотез. Наприклад нульова гіпотеза говорить про те, що різниці в експресії гену немає між двома групами, а альтернативна відповідно, що різниця є. Результат тестування гіпотези потрібно оцінити, тому визначаємо статистичну значущість результатів, для цього визначається рівень значущості альфа. Рівень значущості — це ймовірність відхилити нульову гіпотезу, якщо вона вважається істинною, тобто зробити висновок, що спостережувана різниця середніх є результатом реального ефекту, тоді як насправді це не так. Рівень значущості 0,05 вказує на 5% ризику зробити висновок про існування різниці, коли фактичної різниці немає. Якщо p-значення нижче рівня значущості (наприклад, 0,05), ми можемо відхилити нульову гіпотезу на користь альтернативної гіпотези, тобто ми робимо висновок, що дійсно існує різниця між групами. При тестуванні гіпотез трапляються помилки, розглянемо якими вони бувають та в чому різниця.

2.2 Тестування гіпотез

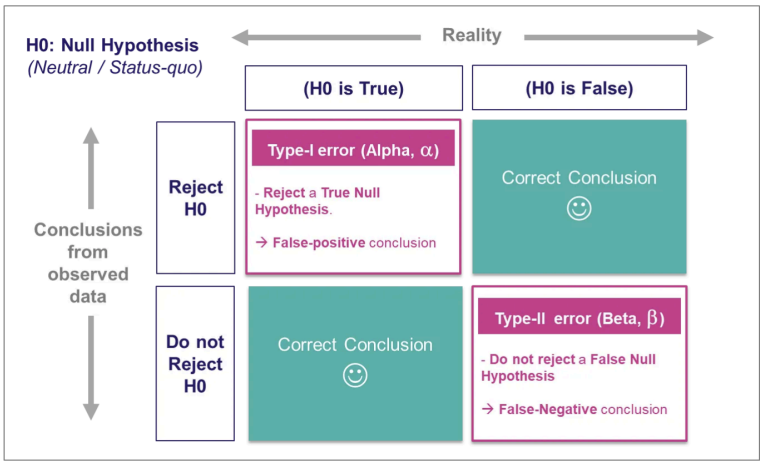

Рис.6 Матриця помилок при тестуванні гіпотез [7]

Зазвичай нульова гіпотеза припускає, що взаємозв'язку чи різниці між семплами немає. Помилка першого роду, також відома як хибнопозитивна, виникає, коли ви відхиляєте нульову гіпотезу, яка насправді є істинною. Іншими словами, ви робите висновок, що ваш результат є статистично значущим, хоча насправді він стався випадково. Рівень значущості альфа визначає ймовірність помилки першого роду. Помилка другого роду означає неможливість відхилення нульової гіпотези, коли вона насправді хибна. Тобто ми робимо висновок що наші результати не є статистично значущими, хоча насправді вони такими є. Відхиляти чи не відхиляти нульову гіпотезу ми визначаємо за допомогою порівняння p-value з α. Для визначення диференціальної експресії генів одночасно тестуються тисячі генів, що в свою чергу спричиняє проблему множинного тестування. Чим більше тестів, тим більша ймовірність отримати значення яке є статистично значущим, хоча насправді таким не є, у висновку ми ризикуємо отримати велику кількість хибнопозитивних результатів. Для того щоб скорегувати проблему множинного тестування застосовується метод Бенджаміні-Хохберга (FDR - false discovery rate) для корегування

рівня значущості альфа. [8]

Основні кроки методу Бенджаміні-Хохберга:

1) Розрахувати p-value для кожного гена.

2) Впорядкувати p-value у зростаючому порядку.

3) Для кожного впорядкованого p-value обчислюється відповідне критичне значення за формулою $(i/m) \cdot Q$; i- порядковий номер p-value; m-загальна кількість генів; Q- рівень FDR (це частка хибнопозитивних результатів серед усіх значущих результатів, значення Q залежить від допустимого рівня помилок першого роду у вашому дослідженні)

4) Порівнюємо p-value з скоригованим p-adjusted, якщо p-value < p-adjusted, то ми відхиляємо нульову гіпотезу. Це означає, що зміна експресії гена є статистично значущою з урахуванням корекції на множинне тестування.

2.3 Візуалізація даних

Аналізуючи дані виникає природне бажання їх візуалізувати, але експресія генів характерна тим, що це дуже високорозмірні дані. Теплова карта (heatmap) допомагає відобразити рівні експресії генів та визначити закономірності та відмінності між різними групами. В тепловій карті за рядок відповідає ген, а за колонку семпл. Колір клітинки відповідає значенню експресії — зазвичай, червоний колір означає високу експресію, а зелений або синій — низьку експресію, білий колір означає що немає різниці між експресією здорового та хворого зразка. [9]

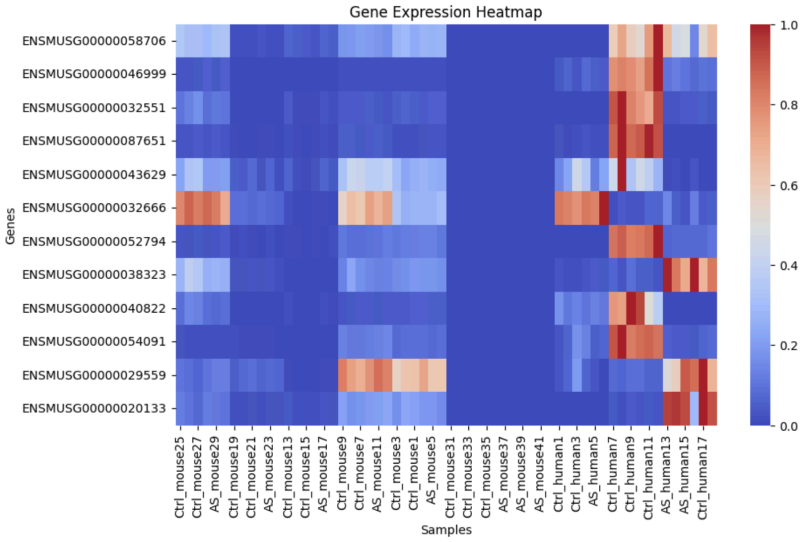

Рис.7 Теплова карта експресії генів

Для того щоб краще побачити патерни теплову карту розбивають на кластери, таким чином по боках утворюються дендограми, які вказують на результати кластеризації генів та семплів.

2.4 Виявлення викидів

Однією з основних проблем при аналізі даних є наявність викидів, тобто спостережень, які значно відрізняються від більшості даних, вони можуть бути або значно вищими, або значно нижчим за інші точки даних. Такі екстремальні значення впливають на результати алгоритмів машинного навчання, а саме вони спотворюють точність прогностичних моделей. Викиди можуть виникати з різних причин, від помилок у введенні даних до справжніх аномалій. Їх наявність можна пояснити такими факторами, як: похибки вимірювання, помилки обробки даних та справжні екстремальні спостереження. Існують різноманітні статистичні методи (z-score, IQR) та ML (Isolation Forest) підходи для детекції аутлаєрів. У своїй роботі я використовувала IQR та Isolation Forest, розглянемо їх детальніше. IQR означає інтерквартильний діапазон, який є різницею між

q_3 (75-й процентиль) і q_1 (25-й процентиль). IQR використовується для опису середніх 50% значень у порядку від найнижчого до найвищого. Метод IQR обчислює нижню та верхню межі для виявлення викидів. Будь-яке значення нижче нижньої межі та вище верхньої межі вважається викидом.

$$Lower\ Bound\ =\ q_1 - 1.5 \cdot IQR$$

$$Upper\ Bound\ =\ q_3 + 1.5 \cdot IQR$$

$$IQR = q_3 - q_1$$

IQR зазвичай застосовується для кожної змінної (гена) окремо, не враховує взаємозв’язки між генами. Важливо розрізняти одновимірні та багатовимірні аутлаєри. Одновимірні аутлаєри — це точки даних, які значно відрізняються від інших значень у межах одного гена. Багатовимірні аутлаєри — це точки даних, які відрізняються від інших точок у багатовимірному просторі, враховуючи взаємозв’язки між кількома генами. Для виявлення багатовимірних викидів просто IQR не допоможе, для цього потрібно застосувати ідею функції статистичної глибини. Функції глибини були розроблені, щоб розширити однофакторні поняття медіани, квантилів, рангів до набору багатовимірних даних. Statistical function depth (статистична глибина функцій) оцінює, наскільки "глибоко" (тобто "всередині") знаходиться точка відносно розподілу даних. Вона надає ранг кожній точці даних, базуючись на її положенні відносно інших точок. Чим глибше точка знаходиться в розподілі, тим більш "центральною" вона є і, відповідно, менш ймовірно бути аутлаєром. Однією з відомих функцій глибини є відстань Махаланобіса, яка вимірює відстань між точкою і центром багатовимірного розподілу, враховуючи коваріацію між змінними, можна сказати що це багатовимірне узагальнення z-score. [12]

$$d_n = \sqrt{(x_n - \widehat{\mu_X})C^{-1}(x_n - \widehat{\mu_X})^T}$$

$x -$ спостережуваний вектор, $\mu -$ арифметичне середнє вектора

$C^{-1} -$ обернена коваріаційна матриця

Isolation Forest — це алгоритм машинного навчання без вчителя для виявлення аномалій, особливо ефективний для даних великої розмірності (Fei Tony Liu and Zhi-Hua Zhou, 2008). Підхід використовує двійкові дерева для виявлення аномалій, що призводить до лінійної складності часу та низького використання пам'яті, що добре підходить для обробки великих наборів даних. Він працює за принципом, що аномалії є рідкісними та виразними, тому їх легше виділити з решти даних. Розглянемо основні кроки isolation forest алгоритму. [11]

1) Алгоритм починається зі створення набору ізоляційних дерев, як правило, сотні або навіть тисячі, кожне з яких побудовано на випадково відібраній підмножині початкового набору даних.

2) Ізольовані дерева вводять випадковість у кожному вузлі дерева, вибирається випадкова функція з набору даних. Потім вибирається випадкове розділене значення в діапазоні значень цієї конкретної функції. Ця випадковість допомагає гарантувати, що аномалії, які, як правило, відрізняються від більшості точок даних, не приховані в окремих гілках дерева.

3) Точки даних потім спрямовуються вниз по гілках дерева ізоляції на основі їхніх значень ознак. Якщо значення точки даних для вибраної функції падає нижче значення розділення, воно переходить до лівої гілки. В іншому випадку він переходить до правої гілки. Цей процес триває рекурсивно, доки точка даних не досягне кінцевого вузла, який просто представляє ізольовану точку даних.

4) Ключова концепція ізоляційних лісів полягає в довжині шляху точки даних через ізоляційне дерево. Аномалії, оскільки вони відрізняються від

більшості, як правило, легше ізолювати. Їм потрібно менше випадкових поділів, щоб досягти кінцевого вузла, оскільки вони, ймовірно, вийдуть за межі типового діапазону значень для вибраних функцій.

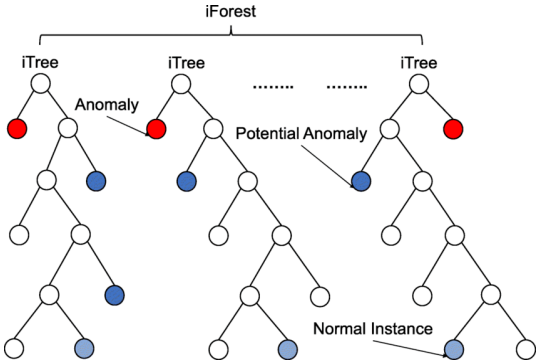

Рис.8 Isolation Forest [10]

Оцінка аномалії

$$s(x,m) = 2^{\frac{-E(h(x))}{c(m)}}$$

$$c(m) = \begin{cases} 2H(m-1) - \frac{2(m-1)}{n} & \text{if } m > 2 \\ 1 & \text{if } m - 2 \\ 0 & \end{cases}$$

2.5 “Прокляття розмірності”

Головним завданням аналізу даних про експресію генів із великою кількістю генів і малою вибіркою є вилучення інформації, пов’язаної з захворюваннями, із величезної кількості надлишкових даних і шуму. Розглянемо таку проблему в аналізі даних як “прокляття розмірності” (curse of dimensionality).[13] Прокляття розмірності описує вибухову природу збільшення розмірності даних і, як результат, експоненціальне збільшення обчислювальних зусиль, необхідних для їх обробки та/або аналізу. Цей термін вперше ввів Річард Е. Беллман, щоб пояснити збільшення об’єму евклідового простору, пов’язане з додаванням додаткових вимірів у сфері динамічного програмування. Коли ми говоримо про збільшення розмірності, мається на увазі збільшення кількості ознак, які використовуються для опису даних, у контексті моєї задачі це різні

гени. Зі збільшенням розмірності кількість точок даних, необхідних для хорошої роботи будь-якого алгоритму машинного навчання, зростає експоненціально. Причина полягає в тому, що нам знадобиться більше точок даних для будь-якої комбінації атрибутів, щоб будь-яка модель машинного навчання була дійсною. Наприклад, скажімо, щоб модель працювала добре, нам потрібно принаймні 10 точок даних для кожної комбінації значень ознак. Якщо припустити, що у нас є один бінарний атрибут, то нам знадобиться 2^1x 10 = 20 точок даних. Для 2 двійкових атрибутів ми матимемо 2^2 унікальних значень і потребуватимемо 2^2 x 10 = 40 точок даних. Таким чином, для k-кількості двійкових ознак нам знадобиться 2^k x 10 точок даних. Hughes (1968) у своєму дослідженні дійшов висновку, що при фіксованій кількості навчальних вибірок прогностична здатність будь-якого класифікатора спочатку зростає зі збільшенням кількості вимірів, але після певного значення кількості вимірів продуктивність погіршується. Таким чином, феномен прокляття розмірності також відомий як феномен Hughes.

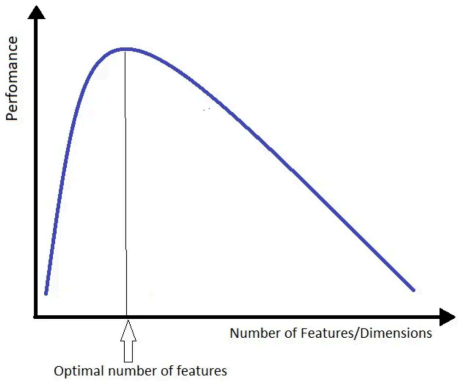

Рис. 9 феномен Hughes [14]

Висока розмірність у даних призводить до такої проблеми як перенавчання. Розглянемо фільтраційні методи виявлення найважливіших ознак (Mutual information, Kolmogorov-Smirnov test) з метою покращення якості моделі.

2.6 Mutual information

У теорії ймовірностей та теорії інформації взаємна інформація (MI) двох випадкових величин є мірою взаємної залежності між двома змінними, у моїй задачі це ген та цільова змінна (статус хвороби).[15] По суті, MI це міра того, скільки «інформації» ви можете отримати про одну змінну, спостерігаючи за іншою змінною. MI між двома випадковими величинами X та Y може бути визначена наступним чином [16]

$I(X;Y) = H(X) - H(X|Y)$

$H(X) := -\sum_{x \in X} p(x) \cdot log\, p(x)$

$H(x)$ - ентропія, визначає міру невизначеності випадкової величини. Логарифм з основою 2 та ентропія вимірюється в бітах. Графік ентропії має увігнуту форму та має наступні властивості: $H(x) = 1$, коли $p = \frac{1}{2}$, $H(x) = 0$, коли $p = 1$ або 0.

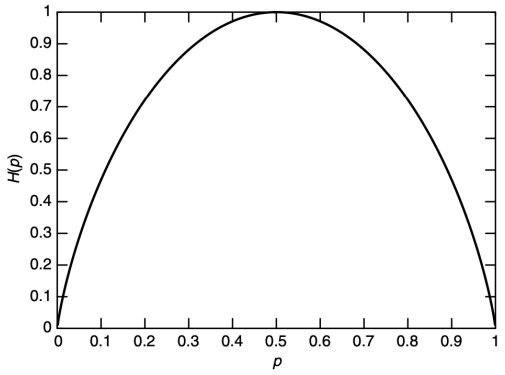

Рис. 10 Ентропія [16]

Ентропію можна також визначити іншим чином, а саме як очікуване значення випадкової величини $log\frac{1}{p(x)}$,

$H(X) = E_p log\frac{1}{p(x)}$

Ми визначили ентропію однієї випадкової величини, зараз ми розширимо це поняття до пари випадкових величин. Спільна ентропія $H(X, Y)$ пари

дискретних випадкових величин (X, Y) з спільним розподілом $p(x, y)$ визначається як

$$H(X, Y) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \cdot log\, p(x, y)$$

$$p(x, y) = P(X = x, Y = y) = P((X = x)\, and\, (Y = y))$$

Якщо події x та y незалежні, тобто настання однієї події не впливає на ймовірність настання іншої події, то $p(x, y) = p(x) \cdot p(y)$. Якщо події залежні, тоді $p(x, y) = p(x) \cdot p(y|x)$, де $p(y|x)$ – це умовна ймовірність настання події y, за умови що подія x вже відбулась.

Також введемо означення умовної ентропії $H(Y|X)$

$$H(Y|X) = \sum_{x \in X} p(x) \cdot H(Y|X = x) = -\sum_{x \in X} p(x) \sum_{y \in Y} p(y|x) \cdot log\, p(y|x) =$$

$$= -\sum_{x \in X} \sum_{y \in Y} p(x, y) \cdot log\, p(y|x) = -\, E\, log\, p(Y|X)$$

Отже, MI можна тепер довести чому формула представлена вище є правдивою

$$I(X; Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y)\, log\, \frac{p(x,y)}{p(x) \cdot p(y)} = \sum_{x,y} p(x, y) \cdot log\, \frac{p(x|y)}{p(x)} =$$

$$= -\sum_{x,y} p(x, y) \cdot log\, p(x) + \sum_{x,y} p(x, y) \cdot log\, p(x|y) =$$

$$= -\sum_{x} p(x) \cdot log\, p(x) - (-\sum_{x,y} p(x, y) \cdot log\, p(x|y)) = H(X) - H(X|Y).$$

Тому mutual information $I(X; Y)$ це зменшення невизначеності X завдяки інформації Y. На діаграмі показаний зв'язок між ентропією та MI.

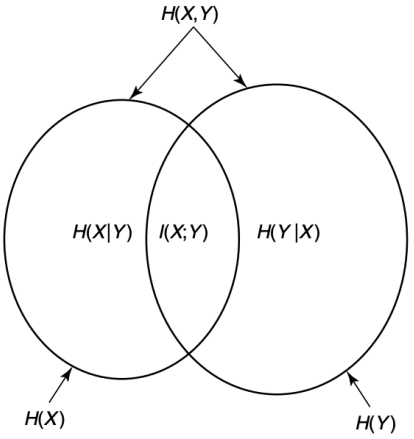

Рис. 11 Зв’язок між ентропією та MI [16]

2.7 Two sample Kolmogorov-Smirnov test

Two sample Kolmogorov-Smirnov test - це непараметричний статистичний тест, який визначає чи дві незалежні групи мають однаковий розподіл, у нашому випадку це клас здорових X та хворих Y. Непараметричний означає, що він не очікує певного розподілу даних. Тест K-S кількісно визначає відстань між функціями емпіричного розподілу двох класів. Для двох вибірок X та Y розмірами m та n відповідно їх емпіричні функції розподілу визначаються наступним чином [17]

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n} I(x_i \leq x)$$

$$G_m(y) = \frac{1}{m}\sum_{i=1}^{m} I(y_i \leq y)$$

I це індикаторна функція, яка повертає 1 якщо умова правдива та 0 в іншому випадку. емпірична функція розподілу має ступінчастий вид та приймає значення [0,1].

Kolmogorov-Smirnov statistic визначається наступним чином

$$D = max\,|F_n(x) - F_m(y)|$$

Kolmogorov-Smirnov test використовує статистику D для перевірки нульової гіпотези H_0: Розподіли X та Y однакові, а альтернативна H_1: розподіли X та Y не однакові. Якщо $D < D_{crit}$, то немає суттєвої різниці між позитивним та негативним класом при рівні значущості α. Якщо $D \geq D_{crit}$, то присутня суттєва різниця між позитивним та негативним класом при рівні значущості $1 - \alpha$. $D_{crit} -$ табличне значення при відповідному рівні значущості α.

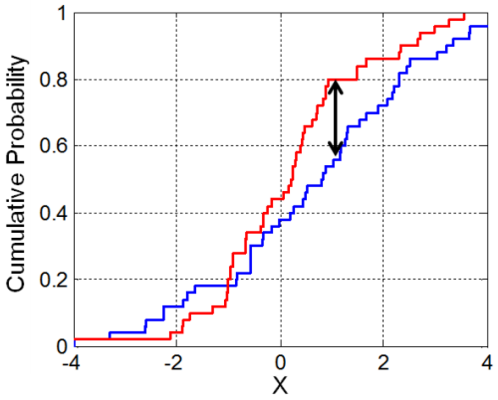

Рис 12. Kolmogorov-Smirnov statistic (Wikipedia)

2.8 Алгоритми зменшення розмірності

Окрім фільтраційних методів для визначення важливих предикторів також застосовуються алгоритми по зменшенню розмірності: PCA та T-SNE. PCA (Principal Component Analysis) - алгоритм машинного навчання (навчання без вчителя) метою якого є зменшення розмірності даних, зберігаючи при цьому якомога більше варіативності (інформації) запропонований Karl Pearson 1901. PCA ідентифікує набір ортогональних осей (головних компонент), які фіксують максимальну дисперсію в даних. Головні компоненти є лінійними комбінаціями вихідних змінних у наборі даних і впорядковані в порядку зменшення важливості. Перший головний компонент фіксує найбільшу варіацію даних, але другий головний

компонент фіксує максимальну дисперсію, ортогональну до першого головного компонента, і так далі.

Основні кроки PCA:

1) Стандартизація даних (шляхом віднімання середнього значення та ділення на стандартне відхилення для кожної ознаки. Це гарантує, що всі фічі будуть порівнювані) та виявлення викидів, так як PCA чутливий до них та може давати хибні результати.
2) Обчислення коваріаційної матриці, коваріаційна матриця для набору даних із n ознаками — це матриця n x n, яка підсумовує зв'язки між усіма парами ознак.
3) Знаходження власних значень і власних векторів: Власні вектори коваріаційної матриці вказують напрямки найбільшої варіативності даних, а власні значення кількісно визначають важливість кожного головного компонента.
4) Сортування власних векторів за зменшенням власних значень: Власні вектори, що відповідають найбільшим власним значенням, зберігають найбільше інформації про датасет.
5) Проектування даних на новий простір: Дані трансформуються в новий простір, сформований першими кількома власними векторами.

Окрім зменшення розмірності, що ускладнює модель, PCA використовується для візуалізації даних, зменшивши кількість вимірів до 2 або 3, які легше представити графічно.

Може виникнути логічне запитання навіщо нам ще інший алгоритм для зменшення розмірності, якщо PCA добре справляється зі своєю задачею. А відповідь полягає в тому, що PCA не вміє працювати з не лінійно сепарабельними даними, і в цьому нам допоможе T-SNE. T-distributed Stochastic Neighbor Embedding це метод машинного навчання візуалізації даних, розроблений Лоренсом ван дер Маатеном і Джефрі Гінтоном. Його

основна ідея полягає в тому щоб відобразити кожну точку багатовимірного простору в дво або тривимірну точку евклідового простору так, що подібні об'єкти розташовуються поруч, а не схожі об'єкти відповідають віддаленим точкам з високою ймовірністю. Опишемо основні кроки T-SNE [18]

1) Створення розподілу ймовірностей, який представляє схожість між сусідами. Схожість точки даних x_j до точки даних x_i є умовною ймовірністю $p(j|i)$ того, що x_i вибере x_j своїм сусідом. Обираємо дві точки з датасету та обчислюємо Евклідову відстань між ними $|x_i - x_j|$ а потім перетворюємо її у ймовірність подібності точок користуючись наступною формулою:

$$p(j|i) = \frac{exp(-||x_i - x_j||^2 / 2\sigma_i^2)}{\sum_{k \neq i} exp(-||x_i - x_k||^2 / 2\sigma_i^2)}$$

$\sigma_i -$ дисперсія Гаусівського розподілу з центром у точці x_i

2) Обчислення відповідних подібностей точок у 2D просторі, по суті це проекція.

$$q(j|i) = \frac{exp(-||y_i - y_j||^2)}{\sum_{k \neq i} exp(-||y_i - y_k||^2}$$

Умовні ймовірності $p(j|i)$ та $q(j|i)$ мають бути однаковими, тобто різниця між нима має бути нуль для ідеального представлення подібності точок даних у просторах різної розмірності. Враховуючи цей факт алгоритм намагається мінімізувати відстань між умовними ймовірностями.

3) Застосування дивергенції Кульбака-Лейблера як функцію втрат для мінімізації розбіжності між розподілами ймовірностей у високовимірному просторі (справжніми даними) і відповідним розподілом у низьковимірному просторі (змодельованими даними).

Цільова функція KL дивергенції мінімізується за допомогою градієнтного спуску.

$$D_{KL}(P||Q) = \sum_{x \in X} P(x)\, log(\frac{P(x)}{Q(x)})$$

P - це матриця ймовірностей схожостей у високо вимірному просторі, Q - це матриця ймовірностей схожостей у низько вимірному просторі.

4) Fine-tuning: розрахунок перплексії - це параметр, який який контролює кількість найближчих сусідів (локальність), яка впливає на формування ймовірностей схожості між точками у високовимірному просторі.

$$Perp(P_i) = 2^{H(P_i)}$$

$H(P_i)$ - ентропія розподілу ймовірностей схожості для даної точки i.

2.9 Валідація моделі

Фінальним етапом є валідація моделі, розглянемо різні методи, а саме cross-validation, bootstrapping, jackknife. Cross-validation — це метод оцінки ефективності моделі машинного навчання, який полягає в багаторазовому розбитті вихідних даних на навчальні та тестові підмножини. Головною метою крос-валідації є забезпечення об'єктивної оцінки точності моделі на нових, невідомих даних, зменшуючи ймовірність перенавчання. Існують різні види cross validation, я використовувала k-fold. Вихідні дані розбиваються на k рівних підмножин, які називаються фолдами (folds). Процес навчання і тестування виконується k разів. На кожному кроці один з фолдів використовується. Якщо використовується (5-fold cross-validation),

то дані розбиваються на 5 частин. Модель тренується 5 разів, кожного разу використовуючи інший фолд як тестовий і решту 4 фолди як навчальні. тестова вибірка, а решта k−1 фолдів використовуються як навчальна вибірка. В кінці оцінюється ефективність моделі як середнє значення метрик (таких як точність, F-score), отриманих на кожному фолді.

Для розуміння метрики F-score, потрібно ввести поняття матриці невідповідностей (confusion matrix). [19]

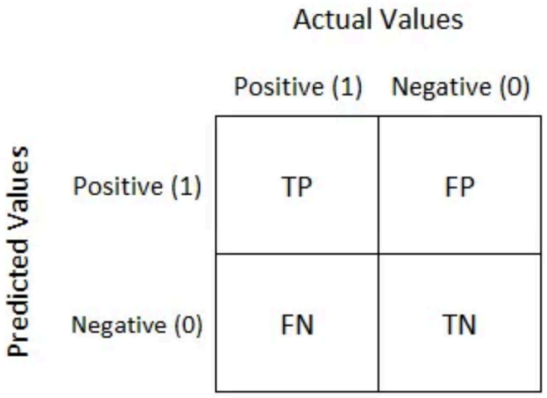

Рис. 13 Confusion matrix

Інтерпретація:

TP - модель передбачила позитивний клас і це є правдивим.

TN - модель передбачила негативний клас і це є правдивим.

FP - модель передбачила позитивний клас, але фактично клас був негативним. (помилка першого роду)

FN - коли модель передбачила негативний клас, але фактично клас був позитивним. (помилка другого роду)

Precision (точність) визначає, яку частку з передбачених позитивних випадків модель передбачила правильно.

$$Precision = \frac{TP}{TP+FP}$$

Recall (повнота) визначає, яку частку фактичних позитивних випадків модель змогла правильно передбачити.

$$Recall = \frac{TP}{TP+FN}$$

F-score є середньозваженим значенням точності та повноти. Ця метрика дозволяє знайти баланс між точністю та повнотою.

$$F1 - score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

Bootstrapping — це процедура повторної вибірки, яка використовує дані з однієї вибірки для створення розподілу вибірки шляхом повторного взяття випадкових вибірок із відомої вибірки з заміною, це дозволяє включати дане спостереження до даної невеликої вибірки більше одного разу. [20]

Нехай X={x1,x2,..., xn} це вихідний набір даних розміру n. Створюємо B підвибірок X* з вихідного набору даних шляхом випадкового вибирання з заміною. Кожна підвибірка X*_b (де b = 1,2,...,B) також має розмір n і може містити потвори елементів. Для кожної підвибірки обчислюється статистичний параметр.

$$\widehat{\theta^*}_b = f(X_b^*)$$

$f -$ функція яка обчислює статистичний параметр з підвибірки $f(X_b^*)$

Після обчислення статистичного параметра для всіх B підвибірок отримуємо набір оцінок. Використовуємо набір оцінок для оцінки розподілу параметра сігма. Основні показники, які можна обчислити з цього розподілу.

Середнє значення:

$$\widehat{\theta_{boot}} = \frac{1}{B}\sum_{b=1}^{B} \widehat{\theta_b^*}$$

Стандартна похибка

$$SE_{boot} = \sqrt{\frac{1}{B-1}\sum_{b=1}^{B}(\widehat{\theta_b^*} - \widehat{\theta^*_{boot}})^2}$$

Bootstrapping не вимагає припущень про розподіл даних та є ефективним методом оцінки статистичних параметрів для малих вибірок.

Метод jackknife це крос-валідаційна техніка, яка базується на ідеї послідовного видалення окремих спостережень з набору даних, вперше була запропонована Морісом Кенуйлем (1949 р.) для оцінки байеса. [20] Байєсом називається різниця між очікуваним значенням цього оцінювача та справжнім значенням параметра, що оцінюється. Джон В. Тьюкі розширив використання jackknife, включивши в нього оцінку дисперсії (1958 р.) Розглянемо основні кроки методу jackknife. Нехай $X = \{x_1, x_2, \ldots, x_n\}$ це вихідний набір даних розміру n. Для кожного і-ого спостереження створюється підвибірка $X_{(i)}$ яка містить усі спостереження окрім і-ого. Кількість таких підвибірок дорівнює n.

$$X_{(i)} = \{x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n\}$$

Для кожної підвибірки $X_{(i)}$ обчислюється статистична оцінка $\widehat{\theta_{(i)}}$.

Jackknife оцінка стандартної похибки, яка визначається як:

$$SE_{jack} = \sqrt{\frac{n-1}{n}\sum_{i=1}^{n}(\widehat{\theta_{(i)}} - \widehat{\theta_{jack}})^2}$$

Jackknife зазвичай менш обчислювально інтенсивний, оскільки потребує обробки n підвибірок. Bootstrap потребує більше обчислювальних ресурсів через велику кількість псевдовибірок.

РОЗДІЛ 3: Експерименти

3.1 Підготовка даних

Оскільки дані різних видів (миші та люди), то я робила експерименти на мишах, людях та на комбінованому датасеті. Також я спочатку спробувала baseline модель, а вже потім застосовувала різні техніки feature selection. Розглянемо основні кроки при підготовці даних. Спочатку я зробила приведення генів до одного формату за допомогою апі веб-ресурсу gProfiler, який використовується для аналізу та обробки списку генів. Оскільки у різних експериментах використовувались різні типи неймспейсів.

```
[ ]  gp = GProfiler(return_dataframe=True)
     converted_GSE16889 = gp.convert(organism='mmusculus',
                 query=GSE16889_list,
                 target_namespace='ENSG')
```

Наступним кроком потрібно знайти спільний список генів між експериментами. На людях вийшло 11747 спільних генів між трьома експериментами, на мишах - 3236 та на комбінованому - 2399. Для комбінованого датасету потрібно було зробити конвертацію генів між різними організмами.

```
[ ]  gp = GProfiler(return_dataframe=True)
     converted = gp.orth(organism='hsapiens',
                 query=GSE67122_list,
                 target='mmusculus')
```

Далі дані було нормалізовано, оскільки у різних експериментах було застосовано різні шкали та також цей етап є важливим та ключовим для алгоритмів, що базуються на відстані (наприклад SVM). Для нормалізації був застосований MinMaxScaler. Він зменшує дані в заданому діапазоні, зазвичай від 0 до 1, мінімум фічі дорівнює нулю, а максимум дорівнює одиниці. Детекція аутлаєрів є не менш важливим кроком при обробці медичних даних. Було застосовано IQR метод, якщо цей семп був визначений як аутлайєр у більше ніж 200 генах, тоді він був видалений з датасету. На датасеті людей було виявлено 4 викиди, на мишах - 6, на комбінованому датасеті було підвищено поріг до 700 і знайдено 12 викидів. Перейдемо до навчання класифікатора за допомогою різних алгоритмів та валідації.

3.2 Навчання базових моделей та валідація

Датасет	Класифікатор	Валідація	Точність
Human	Logistic regression	Cross-validation	0.37
Human	Logistic regression	Jackknife	0.21
Human	SVM (sigmoid)	Cross-validation	0.49
Human	SVM (sigmoid)	Jackknife	0.57
Human	Random forest	Cross-validation	0.41
Human	Random forest	Jackknife	0.36
Mouse	Logistic regression	Cross-validation	0.29
Mouse	Logistic regression	Jackknife	0.14
Mouse	SVM (sigmoid)	Cross-validation	0.34
Mouse	SVM (sigmoid)	Jackknife	0
Mouse	Random forest	Jackknife	0.27
Mouse	Random forest	Cross-validation	0.37
Combined	Logistic regression	Cross-validation	0.24
Combined	Logistic regression	Jackknife	0.15
Combined	SVM (sigmoid)	Cross-validation	0.31
Combined	SVM (sigmoid)	Jackknife	0
Combined	Random forest	Cross-validation	0.24
Combined	Random forest	Jackknife	0.21

3.3 Застосування feature selection методів та валідація

Аналізуючи отримані результати ми приходимо до висновку, що модель перенавчається і погано себе показує на нових даних, це спричинено високою розмірністю та малою кількістю семплів, що було досить очікувано. Тому наступним кроком було застосовано різні feature selection методи та повторено експерименти. Для валідації моделі використовувались cross-validation та jackknife, а для оцінки моделі наступні метрики: accuracy, precision, recall та F-1 score.

Датасет	Feature-selection method	Класифікатор	Валідація	Accuracy	Precision	Recall	F1-score
Mouse	MI	Logistic regression	Cross-validation	0.3	0.27	0.3	0.28
Mouse	MI	Logistic regression	Jackknife	0	0	0	0
Mouse	MI	SVM (sigmoid)	Cross-validation	0.46	0.47	0.47	0.47
Mouse	MI	SVM (sigmoid)	Jackknife	0	0	0	0
Mouse	MI	Random forest	Cross-validation	0.41	0.32	0.33	0.32
Mouse	MI	Random forest	Jackknife	0.33	0.19	0.19	0.19
Mouse	K-S	Logistic regression	Cross-validation	0.42	0.42	0.42	0.42
Mouse	K-S	Logistic regression	Jackknife	0.22	0.11	0.11	0.11
Mouse	K-S	SVM (sigmoid)	Cross-validation	0.33	0.41	0.42	0.41
Mouse	K-S	SVM (sigmoid)	Jackknife	0	0	0	0
Mouse	K-S	Random forest	Cross-validation	0.73	0.72	0.72	0.72
Mouse	K-S	Random forest	Jackknife	0.81	0.38	0.38	0.38

ВИСНОВКИ

У роботі було розв’язано два типи задач передбачувальної аналітики в медицині, а саме задача регресії на прикладі аналізу виживання та задача класифікації. Прогнозуючи час виживання пацієнтів було показано, що не можна застосовувати лінійну регресію, оскільки вона не бере до уваги censored data, незважаючи на те що точність моделі може бути високою, насправді вона не відображає дійсності. У такому випадку варто застосовувати криву Каплана-Майера або Cox regression.

Розробляючи класифікатор для передбачення хвороби Ангельман синдром по експресії генів теж було показано, що звичайні методи машинного навчання відразу не можна застосувати через проблему високої розмірності даних та малої кількості семплів. Для покращення точності моделі було застосовані два фільтраційних методи: Mutual information та Kolmogorov-Smirnov test, які покращили точність моделі. Найкращий результат показав наступний експеримент: на мишах, після застосування Kolmogorov-Smirnov test Random forest дав точність 0.8, хоча якщо проаналізувати precision та recall ми маємо великий шанс помилок першого та другого роду. Для передбачувальної аналітики в медицині валідація та оцінка моделей є дуже важливою складовою, застосування стандартного train-validation-test є неефективним через малу кількість даних, тому варто використовувати такі техніки як jackknife та cross-validation, які оцінюють модель на більш широкому спектрі можливих сценаріїв, що допомагає визначити, наскільки добре модель може працювати з реальними даними.

Література

1) James, Gareth, et al. *An Introduction to Statistical Learning: With Applications in Python*. Springer International Publishing, 2023.
2) Wahutu M, Vesely SK, Campbell J, Pate A, Salvatore AL, Janitz AE. Pancreatic Cancer: A Survival Analysis Study in Oklahoma. J Okla State Med Assoc. 2016 Jul-Aug;109(7-8):391-398. PMID: 27885308; PMCID: PMC5119763.
3) Christine Wohlfahrt-Veje, Annette Mouritsen, Casper P. Hagen, Jeanette Tinggaard, Mikkel Grunnet Mieritz, Malene Boas, Jørgen Holm Petersen, Niels E. Skakkebæk, Katharina M. Main, Pubertal Onset in Boys and Girls Is Influenced by Pubertal Timing of Both Parents, The Journal of Clinical Endocrinology & Metabolism, Volume 101, Issue 7, 1 July 2016.
4) Leboo, Ian. “Introduction to Survival Analysis | by Ian Leboo.” *Medium*, January2023,https://medium.com/@Statistician_Leboo/introduction-to-survival-analysis-992cd4520d4f.
5) Kleinbaum, David G., and Mitchel Klein. Survival Analysis: A Self-Learning Text, Third Edition. Edited by David G. Kleinbaum, Springer, 2012.

6) https://biocorecrg.github.io/CRG_Bioinformatics_for_Biologists/differential_gene_expression.html

7) https://pub.towardsai.net/understanding-type-i-and-type-ii-errors-in-hypothesis-testing-956999e60e17

8) Benjamini Y, Hochberg Y (1995) Controlling the false discovery rate: a practical and powerful approach to multiple hypothesis testing. J R Stat Soc B 57:289–300

9) https://biostatsquid.com/heatmaps-simply-explained/

10) Regaya, Y., Fadli, F. & Amira, A. Point-Denoise: Unsupervised outlier detection for 3D point clouds enhancement. Multimed Tools Appl 80, 28161–28177 (2021). https://doi.org/10.1007/s11042-021-10924-x

11) https://medium.com/@limyenwee_19946/unsupervised-outlier-detection-with-isolation-forest-eab398c593b2

12) Nieto-Reyes, Alicia, and Javier Cabrera. "Statistical Depth based Normalization and Outlier Detection of Gene Expression Data." arXiv preprint arXiv:2206.13928 (2022).

13) https://towardsdatascience.com/curse-of-dimensionality-a-curse-to-machine-learning-c122ee33bfeb

14) G. Hughes, “On the mean accuracy of statistical pattern recognizers,” in IEEE Transactions on Information Theory, vol. 14, no. 1, pp. 55–63, January 1968, doi: 10.1109/TIT.1968.1054102

15) El-Gawady, A.; Makhlouf, M.A.; Tawfik, B.S.; Nassar, H. Machine Learning Framework for the Prediction of Alzheimer's Disease Using Gene Expression Data Based on Efficient Gene Selection. Symmetry 2022, 14, 491. https://doi.org/10.3390/sym14030491

16) Cover, Thomas M., and Joy A. Thomas. Elements of information theory. Wiley, 2006.

17) Qiang Su, Yina Wang, Xiaobing Jiang, Fuxue Chen, Wen-cong Lu, "A Cancer Gene Selection Algorithm Based on the K-S Test and CFS", BioMed Research International, vol. 2017, Article ID 1645619, 6 pages, 2017. https://doi.org/10.1155/2017/1645619

18) https://towardsdatascience.com/t-sne-clearly-explained-d84c537f53a

19) https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

20) https://arxiv.org/pdf/1606.00497