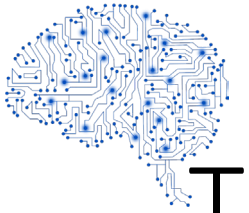
Класифікаційні метрики

Професор, д.е.н. Ставицький А.В.

Задача класифікації

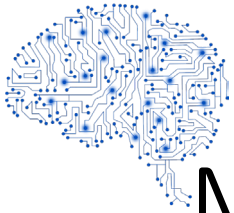
- $X = \{x_1,...,x_l\}$ — вибірка
- $y_i = y(x_i) \in \{0,1\}$, $i = 1,...,l$ — відомі бінарні відповіді
- $a: X \to Y$ — алгоритм, (розв'язувальна функція, стратегія) що наближує y на всій множині об'єктів X
- Питання: як виміряти якість a(x) на вибірці X?
- Інтуїтивна відповідь:

$$\text{Accuracy} = \frac{\#\text{correctly classified items}}{\#\text{all classified items}}$$



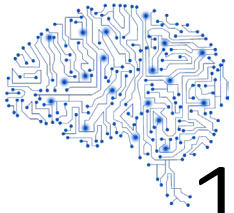
Точність

- Точність, як правило, є кінцевим результатом, яким піклується користувач
- Однак це не дуже корисно для діагностики, оскільки немає інформації, де модель робить помилки
- Відповідь на це запитання "де" є важливою частиною моделювання. Отже, точність - це початок, але не кінець дослідження



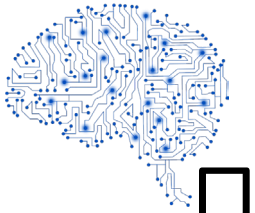
Метрики класифікації

1. Confusion matrix
2. Accuracy
3. Precision and Recall
4. F-Scores
5. Two kinds: binary & multi-class classification
6. Matthews Correlation Coefficient (MCC)
7. Cohen’s Kappa
8. Balanced Accuracy
9. Precision-Recall curve
10. AUC: Receiver operating characteristic (ROC) curve
11. Gini coefficient (Gini index)
12. Log-loss



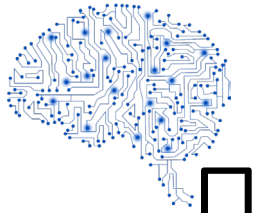
1. Матриця помилок (confusion matrix)

- Матриця помилок надає нам інформацію про те, де ми допускаємо помилки, у вигляді умовної таблиці розподілу
- ij-та позиція матриці дорівнює кількості об'єктів j-го класу, яким алгоритм присвоїв мітку i-го класу



Позначення

- TP = TruePositive, тобто коли фактичним значенням було «так», модель передбачала «так» (тобто правильне передбачення = правильно спрацював)
- FP = FalsePositive, тобто коли фактичним значенням було «ні», модель передбачала «так» (тобто неправильне передбачення = хибно спрацював)
- TN = TrueNegative, тобто коли фактичним значенням було «ні», модель передбачала «ні» (тобто правильне передбачення)
- FN = FalseNegative, тобто коли фактичне значення було "так", модель передбачала "ні" (тобто неправильне передбачення)



Приклад

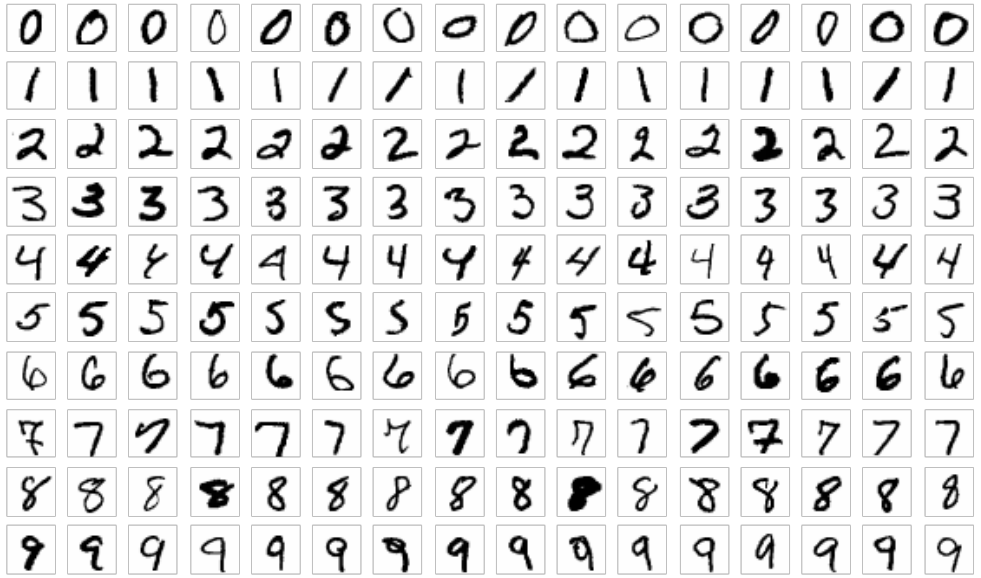
		True label	
		Хвора людина	*Здорова людина*
Pre-dicted label	*Хвора*	True Positive (TP)	False Positive (FP)
	Здорова	False Negative (FN)	True Negative (TN)

Нульова гіпотеза: всі здорові

		Реальна ситуація	
		y=1	y=0
Відпо-відь алго-ритму	y=1	True Positive (TP)	False Positive (FP) = помилка I роду = «хибна тривога»
	y=0	False Negative (FN) = помилка II роду = «пропуск цілі»	True Negative (TN)

Помилки I та II типу

Багатокласова матриця помилок



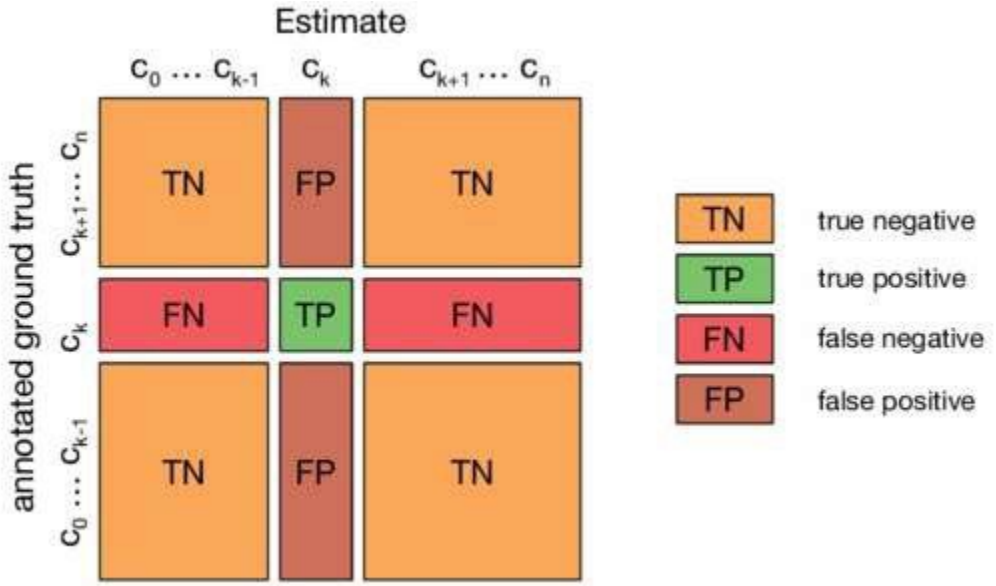
Predictions

Reals

	0	1	2	3	4	5	6	7	8	9
0	902	0	10	5	1	12	3	3	7	3
1	0	1057	8	4	2	6	4	6	14	7
2	14	11	826	25	23	5	17	17	25	4
3	7	6	15	900	2	39	2	16	33	11
4	1	3	13	1	893	3	5	7	3	52
5	14	7	7	25	13	814	29	3	26	15
6	8	5	25	1	21	18	875	3	7	4
7	6	10	10	9	9	1	0	893	0	44
8	9	21	8	24	13	42	10	5	822	31
9	7	6	4	6	40	5	0	39	11	885

Матриця помилок класифікації з n класами

- При розгляді класу k (0 ≤ k ≤ n) можна отримати чотири різні результати класифікації:
- справжньо позитивний (зелений)
- справжньо негативний (помаранчевий)
- хибно позитивний (коричневий)
- помилково негативний (червоний)

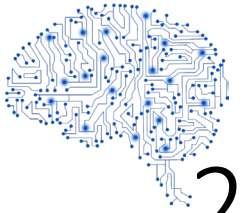


$$tp_i = c_{ii}$$

$$fp_i = \sum_{l=1}^{n} c_{li} - tp_i$$

$$fn_i = \sum_{l=1}^{n} c_{il} - tp_i$$

$$tn_i = \sum_{l=1}^{n} \sum_{k=1}^{n} c_{lk} - tp_i - fp_i - fn_i$$



2. Парадокс точності - 1

- Парадокс точності - це парадоксальний висновок про те, що точність не є хорошою метрикою для прогнозних моделей при класифікації в прогностичній аналітиці

- Це тому, що проста модель може мати високий рівень точності, але бути занадто грубою, щоб бути корисною

Парадокс точності - 2

- Звичайна метрика: $$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- Проблеми на незбалансованих вибірках: є 1075 пацієнтів, 1020 із яких наш класифікатор визначив правильно (True Positive = 20, True Negative = 1000), і 55 неправильно (False Negative = 5, Positive = 50)

  $accuracy$ = 94,88%

	$y = 1$	$y = 0$
$a(x) = 1$	20	50
$a(x) = 0$	5	1000

- Але примітивний класифікатор, який вважатиме всіх користувачів здоровими, дасть краще значення цієї метрики!

$$accuracy = \frac{0 + 1050}{0 + 1050 + 0 + 25} = 97{,}67\%$$

- Причому він нічого не може передбачити!

Точність з кількома оцінювачами

$$r^{\star}_{ik} = \sum_{l=1}^{q} w_{kl} r_{il}$$

$$\text{accuracy} = = \frac{1}{n'} \sum_{i=1}^{n'} \sum_{k=1}^{q} \frac{r_{ik}(r^{\star}_{ik} - 1)}{r_i(r_i - 1)}$$

q is the total number of categories

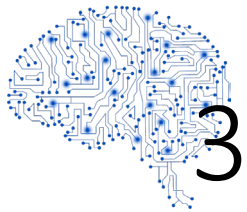
w_{kl} is the weight associated with two raters assigning an item to categories k and l

r_{il} is the number of raters that assigned item i to category l

n' is the number of items that were coded by two or more raters

r_{ik} is the number of raters that assigned item i to category k

r_i is the number of raters that assigned item i to any category

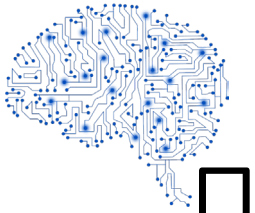


3. Метрики precision (точність) і recall (повнота)

- Точність і повнота походять із сфери пошуку інформації. Під час пошуку інформації потрібно надати користувачам записи, які відповідають їх пошуковому запиту, а не записи, які не мають значення

- Припустимо, наприклад, що ми запускаємо пошукову систему.
- Наша пошукова система повертає 30 сторінок, з них 20 релевантних, а 10 не мають значення
- Наша пошукова система також не повертає 40 інших потрібних сторінок

- У цьому прикладі точність - це відсоток важливих результатів, які ми повернули: 20/30 або 2/3
- Тим часом „повнота” інформації, яку ми повернули, щодо усіх можливих відповідних результатів становить лише: 20/60 або 1/3



Правильний баланс

- Потрібний пошук правильного балансу між точністю та повнотою
- Початкові користувачі менше дбають про повноту, а більше про точність, оскільки їх в основному цікавить пошук кількох звернень, які задовольняють їх потреби
- Досвідчені користувачі тим часом надають пріоритет повноті, вони можуть терпіти лише кілька помилкових спрацьовувань

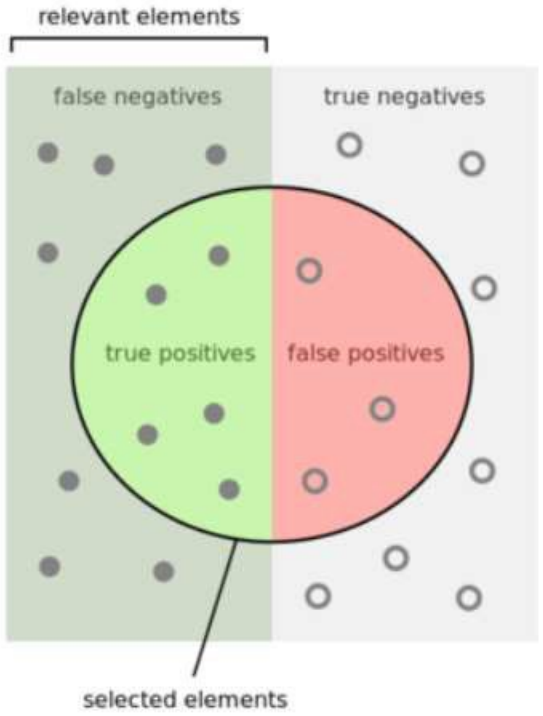
- У той же час між точністю і повнотою є принципово обернений зв’язок!

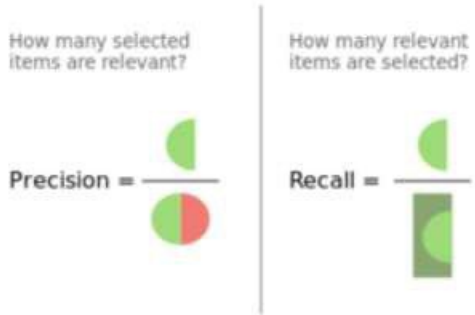
Приклад - 1

	$y = 1$	$y = 0$
$a(x) = 1$	20	50
$a(x) = 0$	5	1000

- precision (точність) характеризує здатність відрізняти потрібний клас від інших; наскільки можна довіряти класифікатору
- precision ∈ [0; 1] (=«доля дійсно хворих серед усіх тих, кого порахували хворими»)

- Точність класифікатора: 28,57 %
- Точність примітивного класифікатора («усі здорові»): 0 %
- **Precision = Confidence = Positive predictive value**



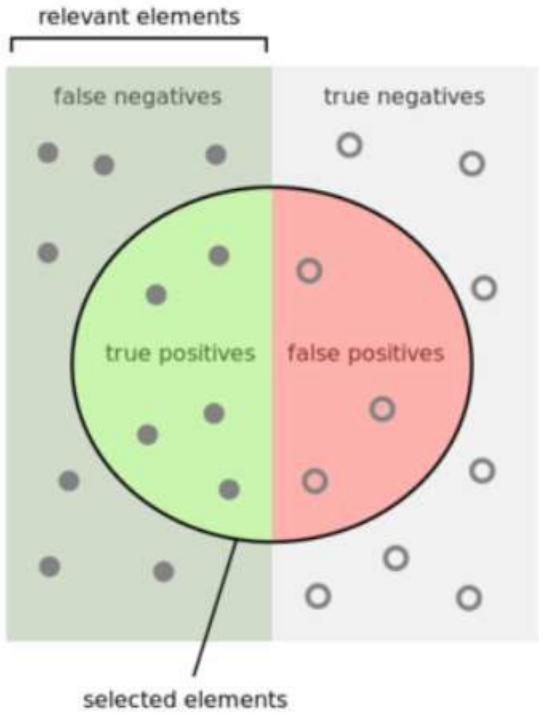


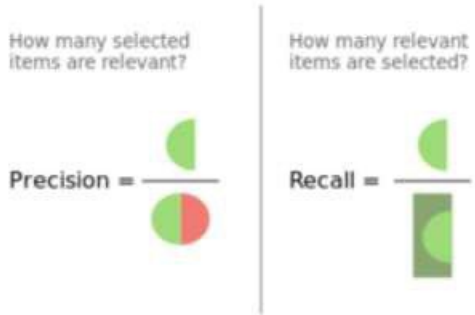
Приклад - 2

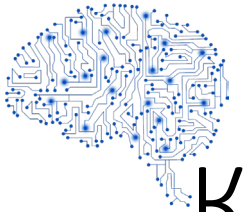
	$y = 1$	$y = 0$
$a(x) = 1$	20	50
$a(x) = 0$	5	1000

- recall (повнота) демонструє здатність алгоритму виявляти потрібний клас
- recall ∈ [0; 1] (=«доля правильно виявлених хворих серед усіх хворих»)

- Повнота класифікатора: 80 %
- Повнота примітивного класифікатора («усі здорові»): 0 %
- **Recall = Sensitivity = True Positive Rate = Hit Rate**

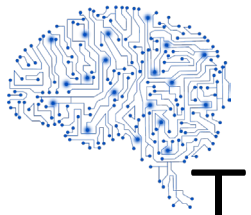






Кого шукати: хворих чи здорових?

- З точки зору алгоритмів машинного навчання - немає ніякої різниці, але з точки зору більшості метрик (точність, повнота тощо) різниця є
- Тому основна рекомендація - щоб була вірна інтерпретація. Зазвичай, це збігається з тим, що «найменший клас» позначають за 1
- Наприклад, рідкісну хворобу. Тоді повнота- який відсоток хворих ми знайшли, точність - який відсоток із знайдених є хворими



Точність-повнота з багатьма класами

- Показники можна "усереднити" по всіх класах багатьма можливими способами
- Якщо класи відрізняються по потужності, то при зваженому усередненні малі за кількістю класи практично ніяк не впливатимуть на результат, оскільки їх внесок в середні TP, FP, FN і TN буде незначний
- У випадку зі звичайним усередненням кожен клас має однакову вагу у підсумковій метриці

Приклад

- Зважене середнє recall = 100R(A)/126 + 9R(B)/126 + 8R(C)/126 + 9R(D)/126 = 0.576
- Звичайне середнє = 126 / (126 + 104) = 0.548

predictions →

	A	B	C	D	
A	100	80	10	10	TP: 100, FN: 100 R(A) = 100 / 200
B	0	9	0	1	TP: 9, FN: 1 R(B) = 9/10
C	0	1	8	1	TP: 8, FN: 2 R(C) = 8/10
D	0	1	0	9	TP: 9, FN: 1 R(D) = 9/10

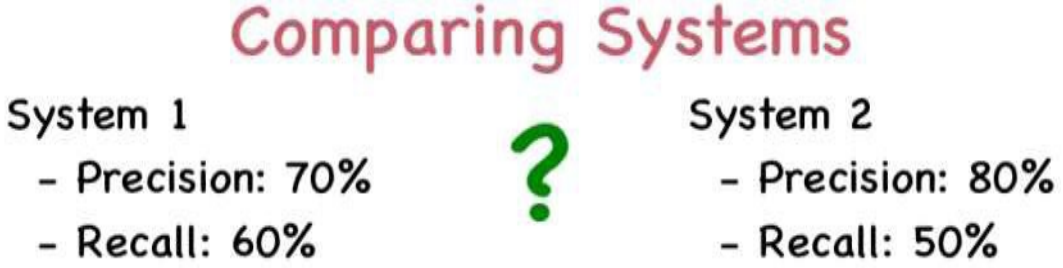
Recall = TP / (TP + FN)

average recall = R(A) + R(B) + R(C) + R(D) / 4 = 0.775

the number of classes

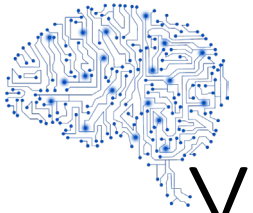
Усереднення точності та повноти – 1

- Арифметичне середнє:
  A =(precision + recall)/2
- Модифікація:
  A=precision + recall – 1



Наприклад:

- примітивний класифікатор: якщо precision=0,05, recall=1, то A=0,525
- непоганий класифікатор: якщо precision=0,525, recall=0,525, то A=0,525



Усереднення точності та повноти – 2

- мінімум:
  M=min (precision, recall)

Наприклад:

- примітивний класифікатор:
  якщо precision=0,05, recall=1, то M=0,05
  якщо precision=0,2, recall=1, то M=0,2
- непоганий класифікатор:
  якщо precision=0,525, recall=0,525, то M=0,525
  якщо precision=0,2, recall=0,3, то M=0,2

Усереднення точності та повноти – 3

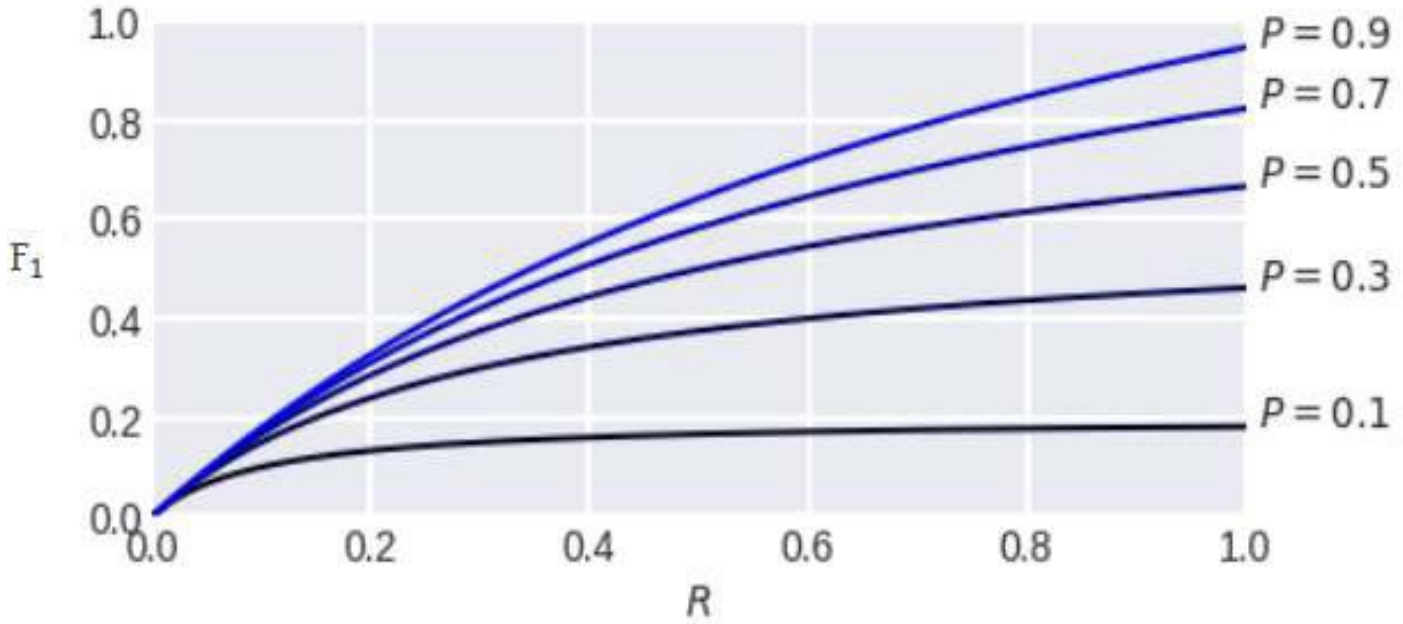
- Гармонічне середнє: $F_1 = \frac{2}{\frac{1}{precision}+\frac{1}{recall}} = \frac{2*precision*recall}{precision+recall}$

Наприклад:

- примітивний класифікатор:
  якщо precision=0,05, recall=1, то $F_1 \approx 0{,}095$
  якщо precision=0,2, recall=1, то $F_1 = 0{,}33$

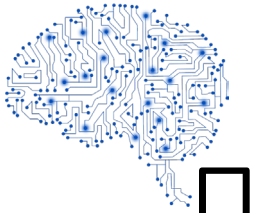
- непоганий класифікатор:
  якщо precision=0,525, recall=0,525, то $F_1 = 0{,}525$
  якщо precision=0,2, recall=0,3, то $F_1 = 0{,}24$

Залежність F_1-міри від повноти при фіксованій точності



4. F-scores

- F-міра: $$F_\beta = (1 + \beta^2) \frac{precision * recall}{\beta^2 * precision + recall}$$
- β визначає вагу точності в усередненій (агрегованій) метриці
- На практиці, зазвичай, беруть β =0,5 або β =2, коли хочуть надати перевагу одній із складових
- Оскільки F-міра, очевидно, приймає тільки невід'ємні значення, то точка (precision=0, recall=0) є точкою мінімуму
- Коли змінні дорівнюватимуть 1, тоді максимум F_β дорівнює 1

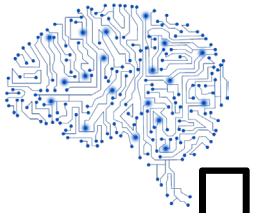


Питання

- Нехай хочемо зменшити складські витрати, прогнозуючи, коли товар, що швидко псується, закінчиться у магазині.
- Які помилки потрібно зменшити більше: хибні спрацьовування чи хибні пропуски? Яке потрібно взяти β?

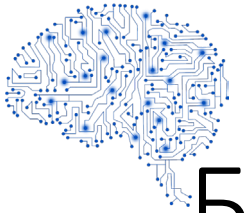
$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN}$$

- Якщо $\beta^2 < 1$, то вплив precision буде більшим, інакше – більше впливатиме r$ecall$



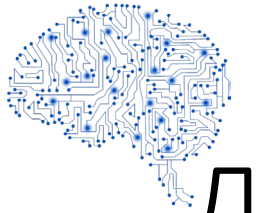
Питання

- Завдання: служба безпеки аеропорту хоче мінімізувати пропуск терористів на борт літака
- Які помилки потрібно зменшити більше: хибні спрацьовування чи хибні пропуски?
- Має збільшитися повнота
- Як це вплине на точність? Чому будемо брати $\beta=2$?



Багато класів

- F-scores можна визначити при багатьох класах
- Показники можуть бути "усереднені" для всіх класів у безліч можливих способів

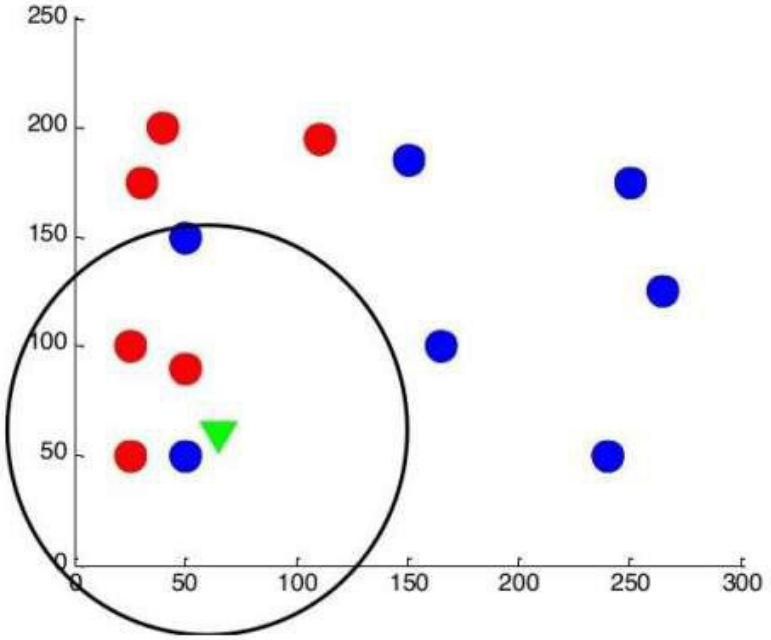


Два типи класифікаційних алгоритмів

- Належність класу: Алгоритми, такі як SVM та KNN, створюють результат належності класу. Наприклад, у двійковій задачі класифікації результати будуть або 0, або 1
- Імовірнісний результат: Алгоритми, такі як логістична регресія, випадковий ліс, посилення градієнта, Adaboost тощо, дають імовірнісні результати. Перетворення вихідних даних імовірності у дані належності до класу - це лише питання створення порогової ймовірності

Оцінка належності та класифікатор у методі k найближчих сусідів

- Звідки може братися на виході ймовірність приналежності об’єкту до певного класу?
- Якщо серед сусідів об’єкту всі відносяться до одного класу, то він упевнений у класифікації та дає високу оцінку, інакше – оцінка знижується
- А якщо ці класи нерівноцінні? Наприклад, щось із цих кружечків – золоті піщинки, а щось – звичайний пісок? Тоді поріг спрацьовування потрібно зменшувати



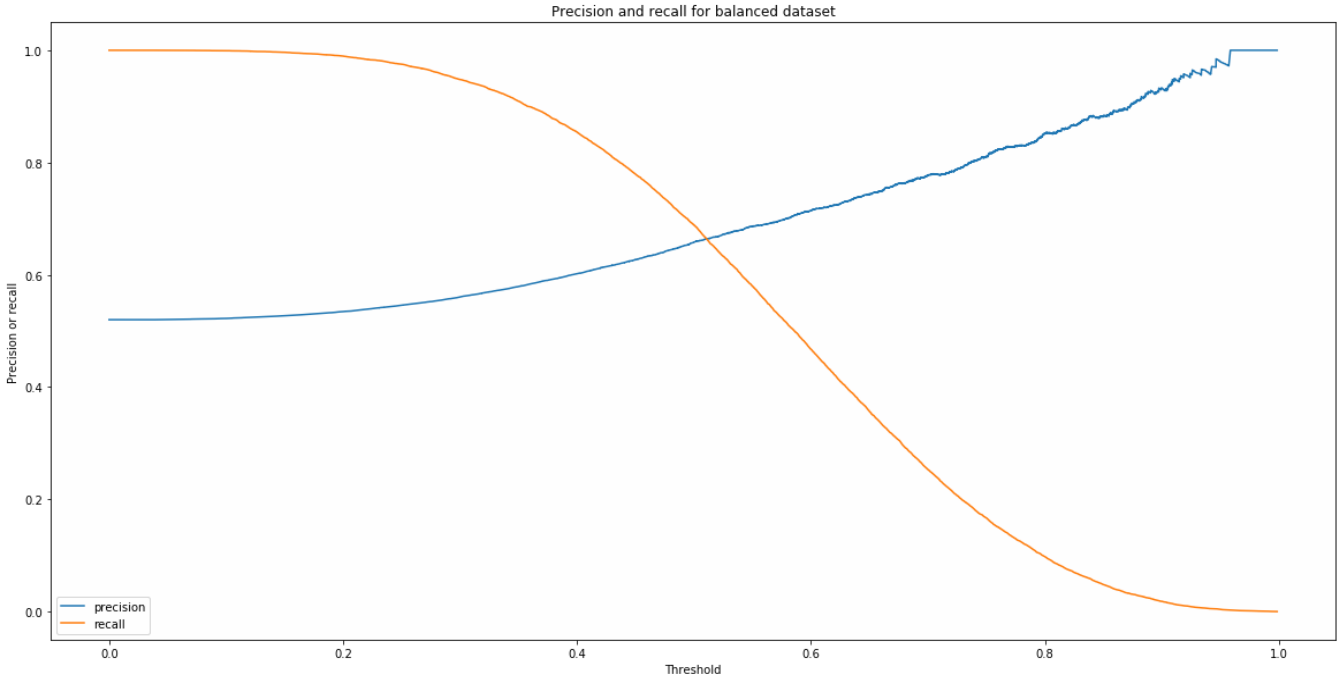
https://habrahabr.ru/company/yandex/blog/206058/

$$a(x) = \left[\sum_{i=1}^{k}[y^{(i)} = 1] > k/2\right]$$

Порогова ймовірність

- $b(x)$ – оцінка належності до класу 1
- t - порогове значення

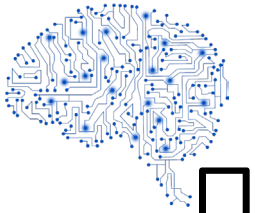
$$a(x) = [b(x) > t]$$



5. Коефіцієнт кореляції Метьюса (MCC)

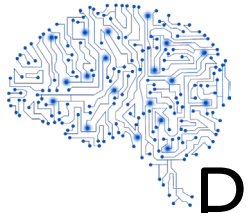
$$\mathrm{MCC} = \frac{\mathrm{TP} \cdot \mathrm{TN} - \mathrm{FP} \cdot \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP}) \cdot (\mathrm{TP} + \mathrm{FN}) \cdot (\mathrm{TN} + \mathrm{FP}) \cdot (\mathrm{TN} + \mathrm{FN})}}$$

- Подібно до коефіцієнта кореляції, діапазон значень MCC лежить
- від -1 до +1
- Модель з оцінкою +1 - ідеальна модель, а -1 - погана модель



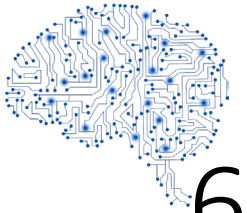
Приклад оцінки належності до спаму

- Потрібно передбачити, чи є даний електронний лист спамом
- Сортуємо листи по оцінці b(x) ймовірності того, що лист є спамом:
- Отримуємо ранжований список листів
- Поріг вибирається залежно від нашої стратегії
- Поріг може багаторазово переглядатися



Розрахунок

- **Нехай лише 100 листів із 1 млн є спамом**
- **Приклад (поріг**=0,8): TP=90, FP=10, FN=10, TN=999890
- Recall (=Sensitivity) = $\frac{TP}{TP + FN} = \frac{90}{90 + 10} = 0,9$
- Specificity = $\frac{TN}{TN + FP} = \frac{999890}{999890 + 10} = 1$
- Precision = $\frac{TP}{TP + FP} = \frac{90}{90 + 10} = 0,9$
- F1 score = $2^{*}\frac{precison * recall}{precison + recall} = 2^{*}\frac{0,9 * 0,9}{0,9 + 0,9} = 0,9$
- MCC=$\frac{TP * TN - FP * FN}{\sqrt{(TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)}} =$
- $= \frac{90 * 999890 - 10 * 10}{\sqrt{(90 + 10) * (90 + 10) * (999890 + 10) * (999890 + 10)}} = 0,9$



6. Каппа Коена (Cohen’s Kappa)

- Коефіцієнт каппа Коена — це статистика, яка вимірює узгодження рішень двох експертів про якісні (категоріальні) об’єкти
- Скажімо, два експерти класифікують електронні листи — це спам чи не спам
- Зазвичай вважається, що це більш надійна міра, ніж простий підрахунок відсотка співпадань думок (=рішень) експертів, оскільки каппа Коена враховує можливість випадкового співпадання їх рішень
- У класичному варіанті каппа Коена підраховує узгодженість рішень двох експертів, кожен з яких класифікує N предметів на M взаємовиключних категорій

Каппа Коена

$$\kappa = \frac{p_0 - p_e}{1 - p_e}$$

- де p_0 - емпірична ймовірність згоди на присвоєння класу будь-якій вибірці (спостерігається співвідношення збігів),
- p_e - це очікувана угода, коли обидва експерти призначають мітки випадковим чином (коефіцієнт домовленості).

- Каппа Коена - це число від -1 до 1. Оцінки вище 0,8, як правило, вважаються хорошою згодою; нуль або нижче означає відсутність згоди (практично випадкові мітки)

Приклад – 1

- Два члени кредитного комітету оцінювали 50 заявок на отримання кредиту. Результати їх рішень зведемо до таблиці

		Експерт 2	
		дати кредит	не давати кредит
Експерт 1	дати кредит	20	5
	не давати кредит	10	15

- Експерт 1 сказав «Так» 25 заявникам і «Ні» також 25 заявникам, тобто рекомендував дати кредит в 50% випадків
- Експерт 2 сказав «Так» 30 заявникам і «Ні» 20 заявникам, тобто рекомендував дати кредит в 60% випадках

Приклад – 2

		Експерт 2	
		дати кредит	не давати кредит
Експерт 1	дати кредит	20	5
	не давати кредит	10	15

- observed agreement ratio p_0 =(20+15)/50=0,7
- Рахуючи ці події незалежними, ймовірність того, що експерти одночасно скажуть «Так» дорівнює 0,5*0,6=0,3
- Аналогічно ймовірність того, що вони одночасно скажуть «Ні» дорівнює 0,5*0,4=0,2
- Тоді спільна ймовірність випадкової згоди рішень (chance agreement ratio): p_e =0,3+0,2=0,5

$$\kappa = \frac{p_0 - p_e}{1 - p_e} = \frac{0,7 - 0,5}{1 - 0,5} = 0,4$$

Каппа Коена як метрика класифікації

- Оскільки використання точності (Accuracy) викликає сумнів у задачах з сильним дисбалансом класів, треба її значення дещо перенормувати:

$$\kappa = \frac{\text{Accuracy} - \text{Accuracy}_{\text{chance}}}{1 - \text{Accuracy}_{\text{chance}}}$$

	$a = 0$	$a = 1$
$y = 0$	m_{00}	m_{01}
$y = 1$	m_{10}	m_{11}

$$\text{Accuracy} = \frac{m_{00} + m_{11}}{m}$$

$$\text{Accuracy}_{\text{chance}} = \frac{m_{00} + m_{01}}{m}\frac{m_{00} + m_{10}}{m} + \frac{m_{10} + m_{11}}{m}\frac{m_{01} + m_{11}}{m}$$

- Червоним виділена ймовірність вгадати клас 0, а синім - клас 1
- Припускаємо, що це незалежні події. Імовірність приналежності до класу k можна оцінити по матриці помилок як частку об'єктів класу k. Аналогічно, ймовірність видати мітку оцінюємо як частку таких міток у відповідях побудованого алгоритму

7. Збалансована точність (Balanced Accuracy)

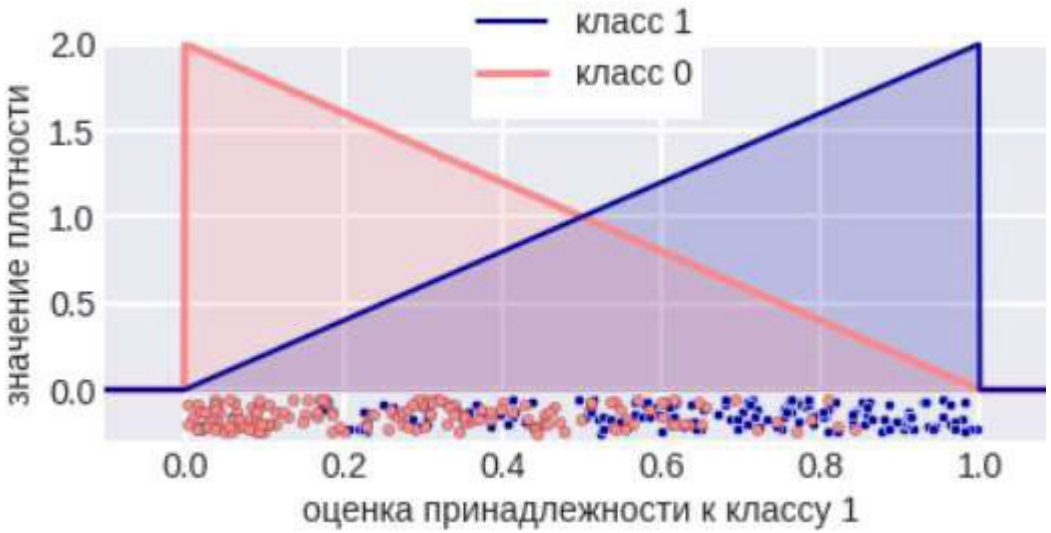
- У разі дисбалансу класів є спеціальний аналог точності — збалансована точність:

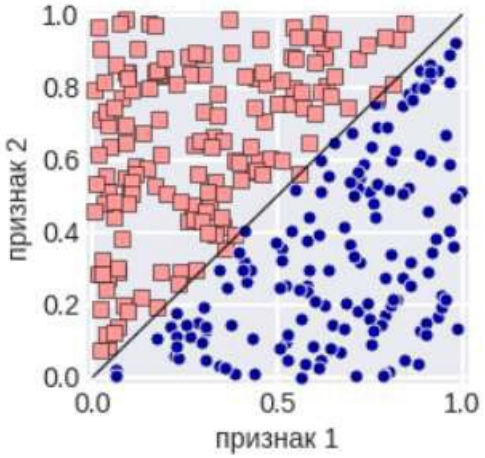
$$\mathrm{BA} = \frac{R_1 + R_0}{2} = \frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} + \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}\right)$$

- Це середнє повноти всіх класів
- Якщо в бінарній задачі класифікації представників двох класів приблизно порівну, то TP + FN ≈ TN + FP ≈ l/2 і збалансована точність приблизно дорівнює точності звичайній (Accuracy)

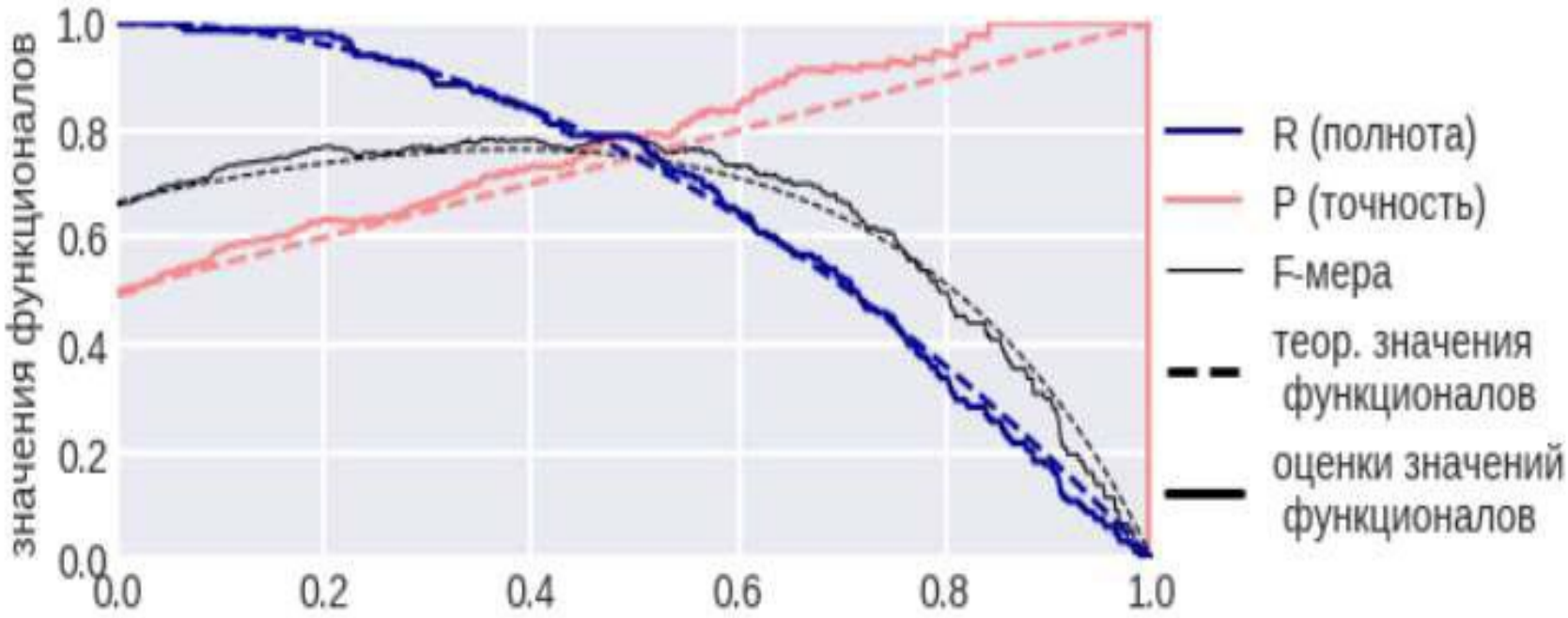
Модельний приклад – 1

- Розв’язуємо задачу бінарної класифікації.
- Нехай наш алгоритм видає оцінки b(x) приналежності об’єкту x до класу 1 на відрізку [0; 1]
- Нехай функції щільності розподілу класів на оцінках, породжених цим алгоритмом, є лінійними: за відповідями алгоритму b(x) об’єкти x класу 0 розподілені зі щільністю f(x)=2-2x, а об’єкти класу 1 – зі щільністю f(x)=2x
- Інтуїтивно зрозуміло, що алгоритм має певну роздільну здатність: більшість об'єктів класу 0 мають оцінку менше 0.5, а більшість об'єктів класу 1 – більше 0.5
- Рішення залежить тільки від першої ознаки (при другій коефіцієнт дорівнює нулю)



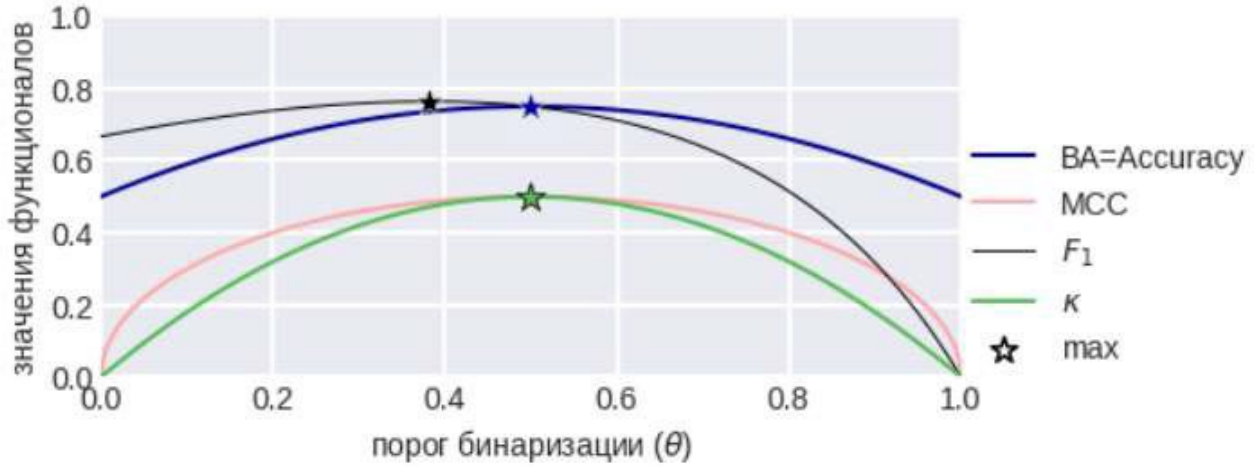


Модельний приклад – 2

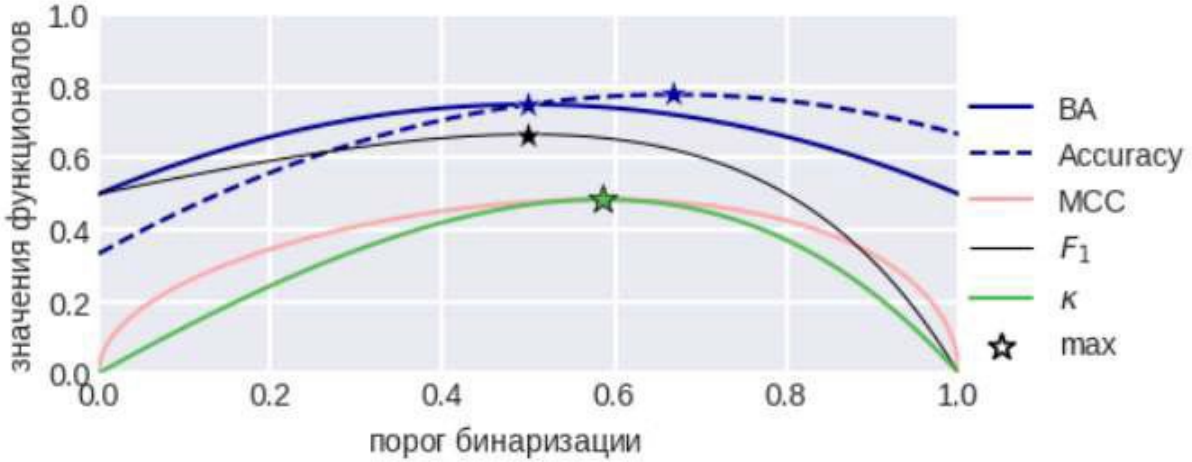


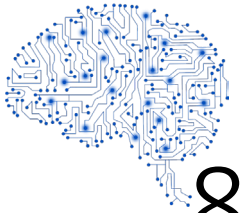
Модельний приклад: вибірка незбалансована

- Збалансована вибірка



- Незбалансована вибірка





8. Precision-Recall-крива

- Відсортуємо об'єкти по зростанню оцінки b(x)
- Переберемо всі пороги класифікації, почавши з максимального: $t_l = b(x_l) > \ldots > t_1 = b(x_1) > t_0 = b(x_1) - e$
- Для кожного порога порахуємо точність і повноту
- Нанесемо відповідну точку в осях «повнота-точність»
- З'єднаємо точки, отримавши Precision-Recall-криву

Приклад

- Відсортували об'єкти по зростанню оцінки $b(x)$:

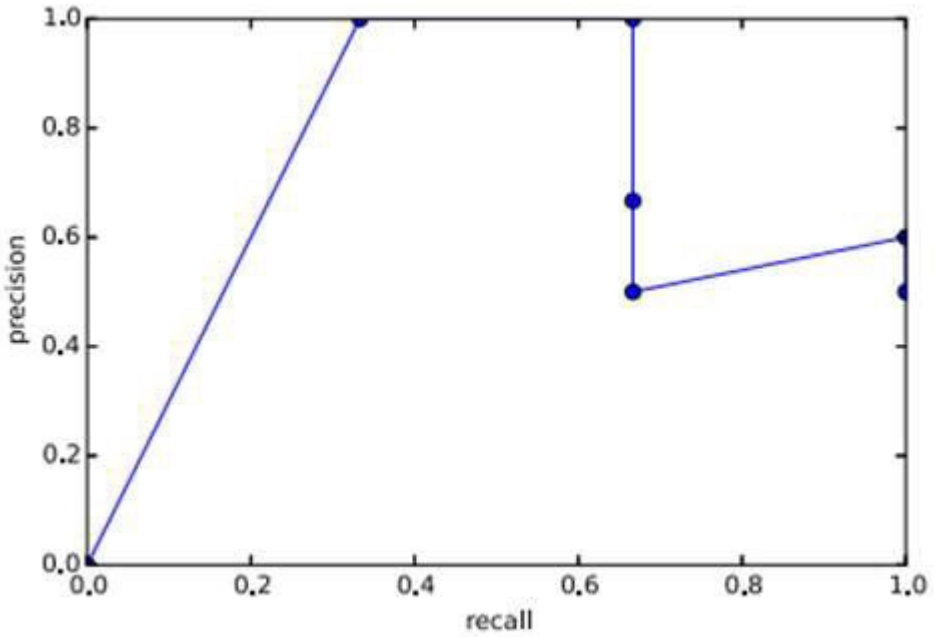
$b(x)$	0.14	0.23	0.39	0.52	0.73	0.90
y	0	1	0	0	1	1

- Переберемо всі пороги класифікації, почавши з максимального
- Нанесемо відповідну точку в осях «повнота-точність»

Отримання Precision-Recall-кривої

- Якщо поріг t=1, то класифікатор всіх вважає здоровими – точка (0, 0)
- Ліва точка кривої: завжди (0, 0) (не вгадуємо жодного хворого)
- Якщо поріг t=0, то класифікатор всіх вважає хворими – точка (1, 0.5)
- Права точка: (1, l+ / l), l+ – число об'єктів класу 1 (хворих) у вибірці
- Якщо вибірка ідеально роздільна, то крива пройде через точку (1, 1)
- Чим більше площа під кривою (AUC), тим краще класифікатор

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

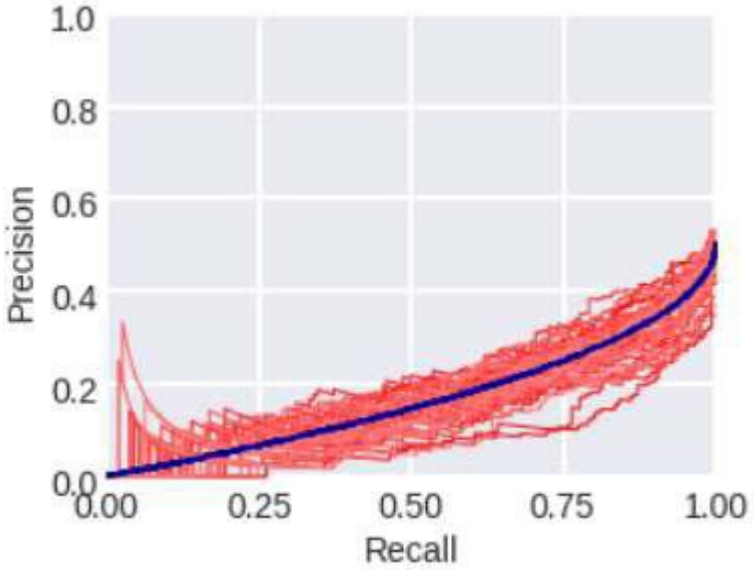


$$\frac{TP}{TP+FN}$$

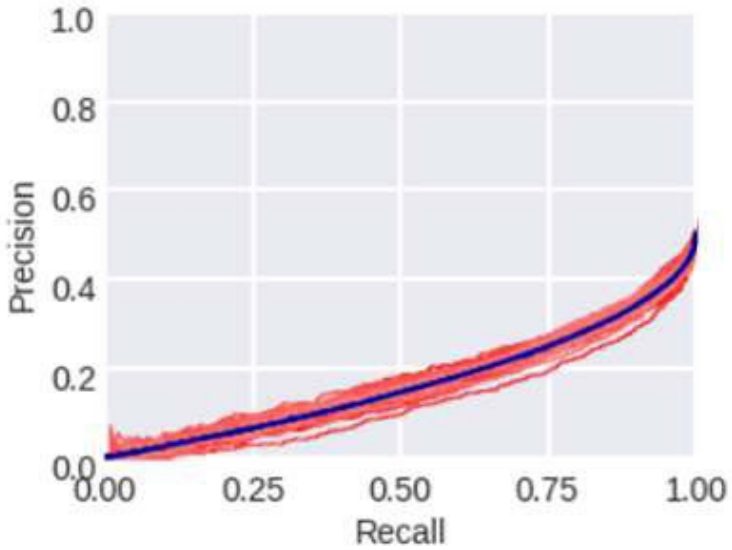
$b(x)$	0.14	0.23	0.39	0.52	0.73	0.90
y	0	1	0	0	1	1

Модельна задача

- PR-крива в модельній задачі: теоретична (синя) і емпіричні (червоні) для вибірки із певної кількості об’єктів



Для вибірки із 300 об’єктів



Для вибірки із 3000 об’єктів

9. ROC («reciever operating characteristic»)-крива

- по вісі X: False Positive Rate, доля хибних позитивних спрацьовувань («хибна тривога»):

$$FPR = \frac{FP}{FP + TN}$$

- FPR = 1- TNR, де TNR називається специфічністю (specificity) алгоритму

$$TNR = \frac{TN}{TN + FP}$$

- по вісі Y True Positive Rate, доля правильних позитивних спрацьовувань (класифікацій):

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

- TRP називається чутливістю (sensitivity) (=повнотою) алгоритму

ROC-крива – 2

- ROC-криву будують не по абсолютним значенням (TP і FP), а по відносним — часткам (rates), вираженим у відсотках:

$$TPR = \frac{TP}{TP + FN} \cdot 100\,\%$$

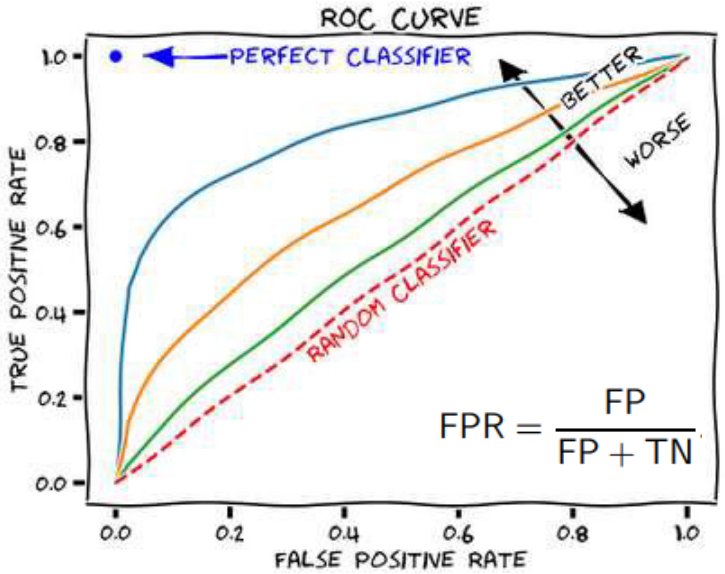
$$FPR = \frac{FP}{TN + FP} \cdot 100\,\%$$

- Для кожного значення порогу класифікації, яке змінюється від 0 до 1 з певним кроком (скажімо, 0.01) розраховуємо TPR і FPR
- Альтернативно поріг можна рахувати для кожного наступного значення прикладу із вибірки

ROC-крива - 3

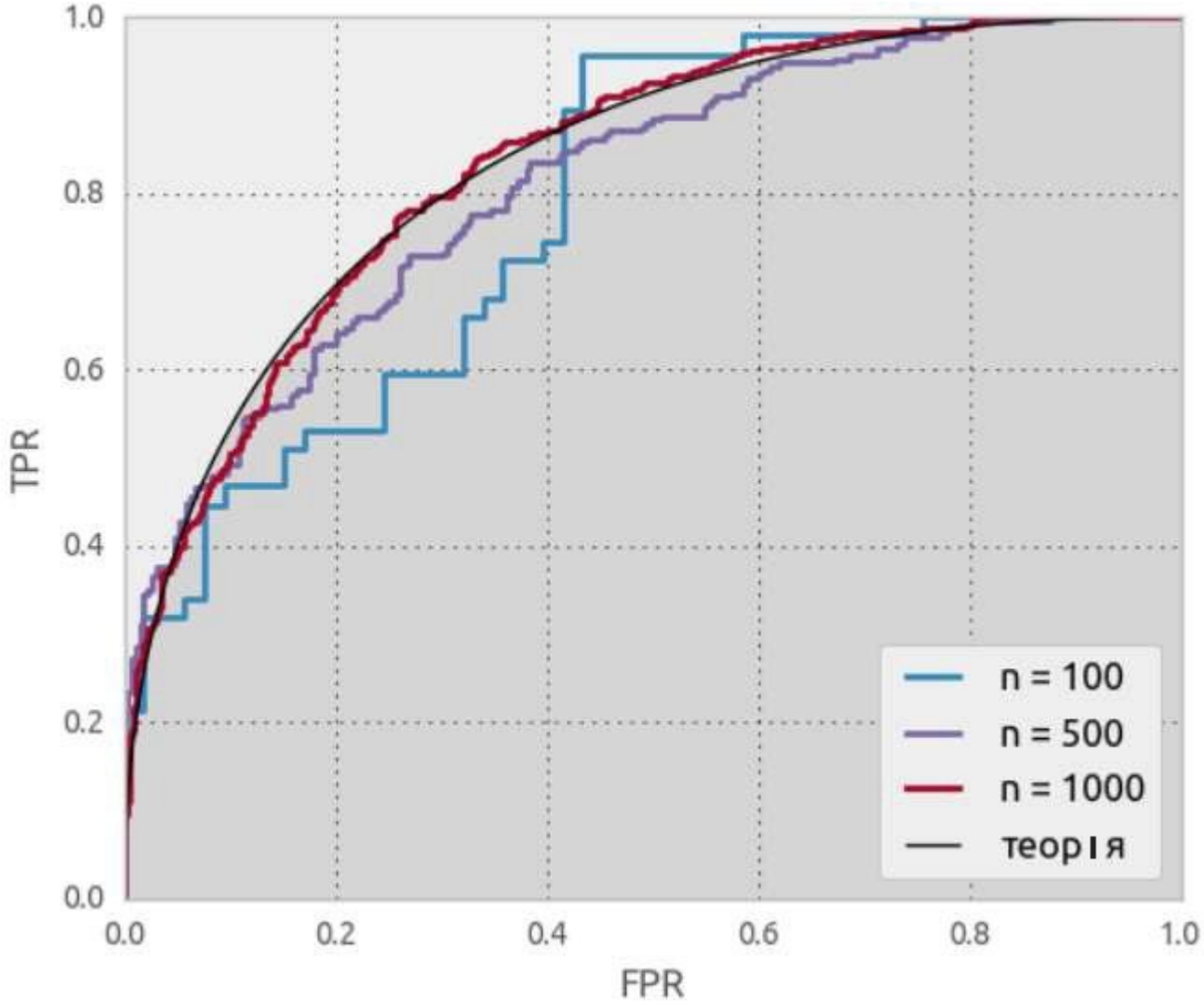
- Найкращий з можливих методів прогнозування (ідеальний класифікатор) дасть точку у верхньому лівому куті або координату (0,1) простору ROC, що представляє 100% чутливість (без помилкових негативів) та 100% конкретність (без помилкових спрацьовувань)
- Завжди: (0,0) ==> (1,1)
- Випадкове відгадування дасть точку вздовж діагональної лінії (так звана недискримінаційна лінія) від лівого нижнього до правого верхнього кута
- Інтуїтивно зрозумілим прикладом випадкового вгадування є рішення перегортанням монет
- У міру збільшення розміру вибірки точка ROC випадкового класифікатора прагне до діагональної лінії
- У разі збалансованої монети вона буде прагнути до точки (0,5, 0,5)

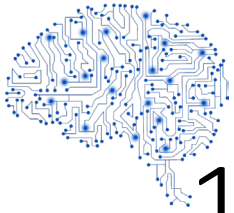
$$TPR = \frac{TP}{TP + FN}$$



ROC-крива для модельної задачі

- При збільшенні обсягу вибірки ROC-криві, побудовані по вибірці, будуть збігатися до теоретичної кривої, побудованої для розподілу



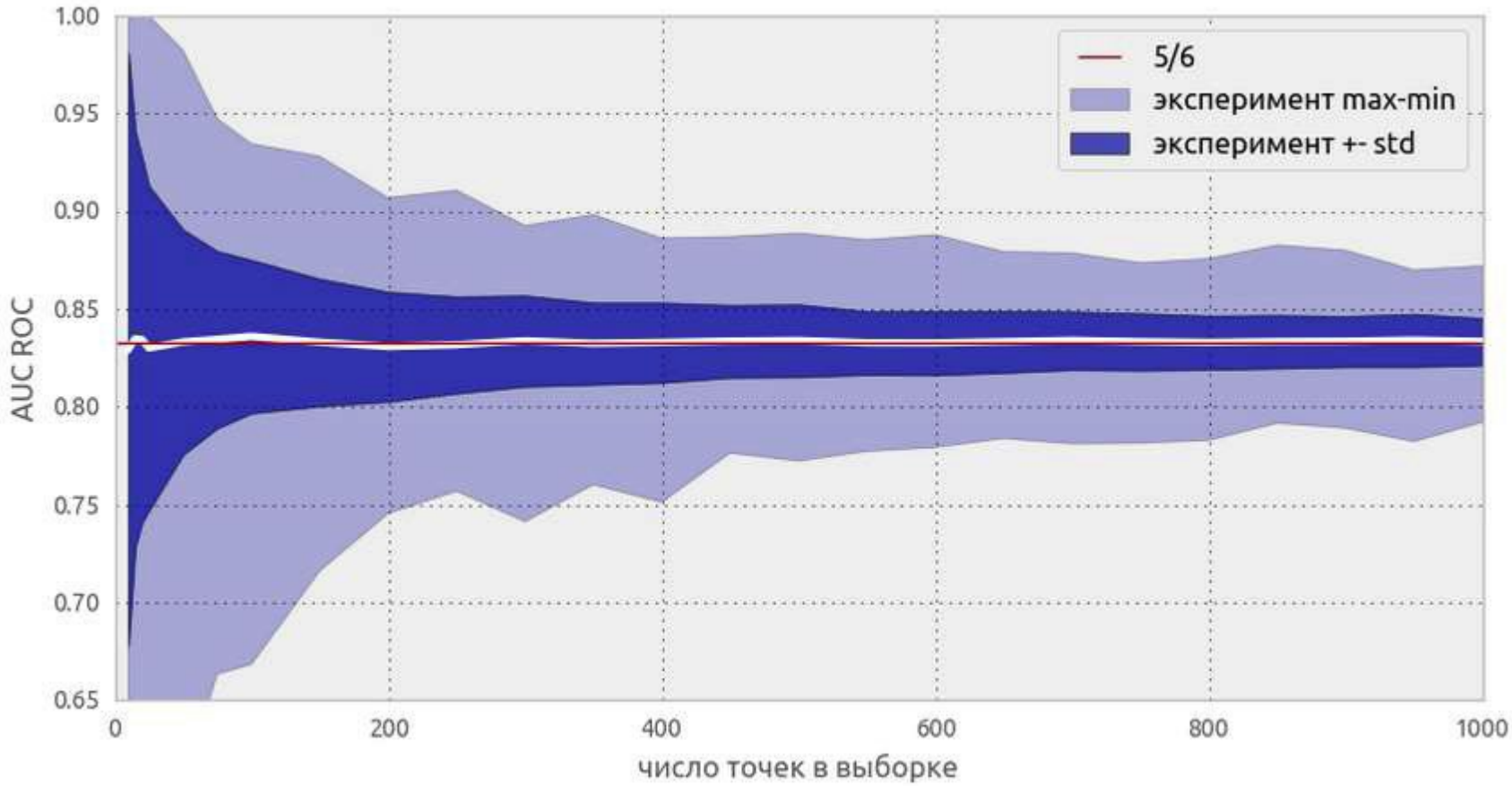


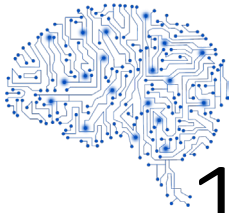
10. Area Under Curve (AUC)

- Площа під ROC-кривою AUG (Area Under Curve) є агрегованою характеристикою якості класифікатора
- Чим більше значення AUC, тим «краще» модель класифікації
- Даний показник часто використовується для порівняльного аналізу кількох моделей класифікації

Модельна задача

- Для оцінки AUC (ROC) вибірка у кілька сотень об'єктів мала!





11. Коефіцієнт Джині

- Коефіцієнт Джині — показник нерівності розподілу деякої величини, що приймає значення між 0 і 1, де 0 означає абсолютну рівність (величина приймає лише одне значення), а 1 позначає повну нерівність
- Найбільш відомим коефіцієнт є як міра нерівності доходів домогосподарств деякої країни чи регіону
- Коефіцієнт Джині для доходів домогосподарств є найпопулярнішим показником економічної нерівності в країні

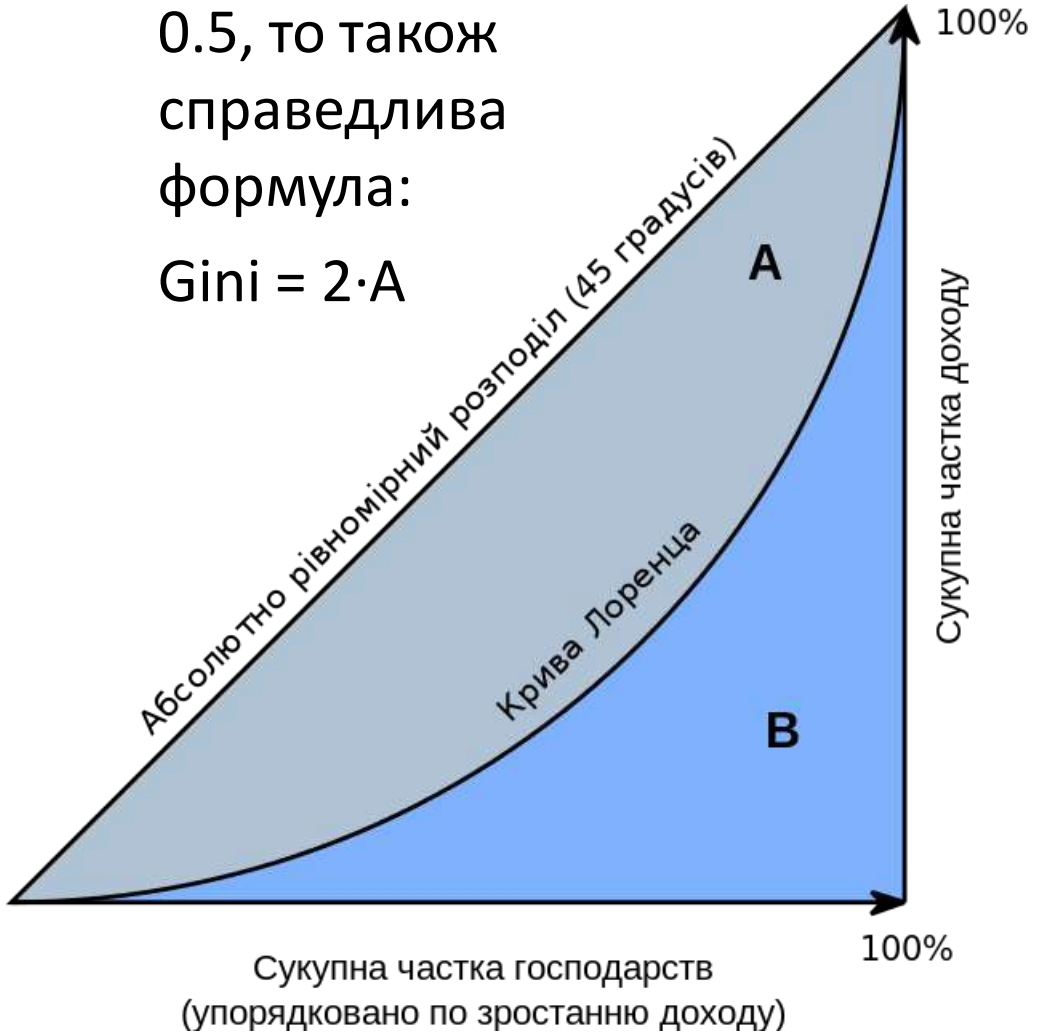
Розрахунок

- Індекс Джині найпростіше визначити за допомогою кривої Лоренца, що зображує частку величини y, що зосереджується на x% популяції з найменшим значенням цієї величини
- Наприклад для розподілу доходів точка (20%, 10%) буде лежати на кривій Лоренца, якщо сукупний дохід 20% найбідніших домогосподарств рівний 10% сукупного доходу усіх домогосподарств
- Gini = 2*AUC(ROC) – 1

Gini ≡ A/(A+B)

Оскільки A+B = 0.5, то також справедлива формула:

Gini = 2·A



12. Логістична функція втрат (log-loss)

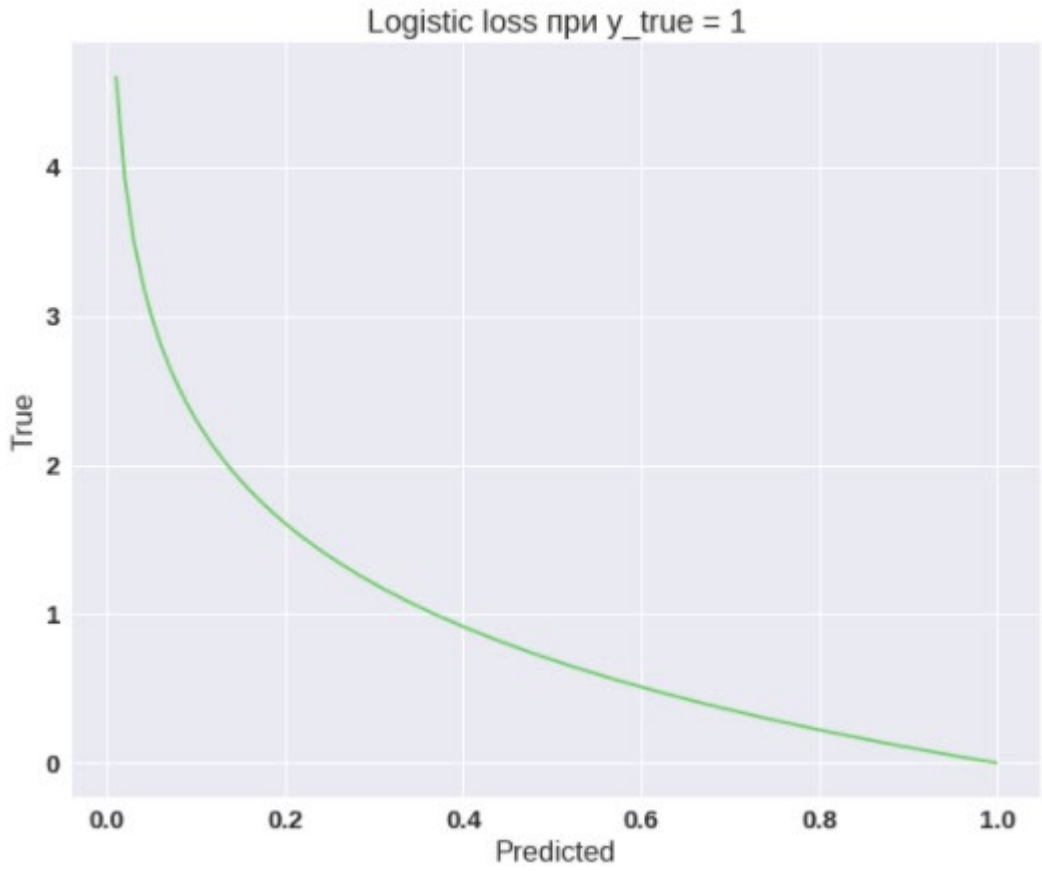
$$logloss = -\frac{1}{l} \cdot \sum_{i=1}^{l} (y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i))$$

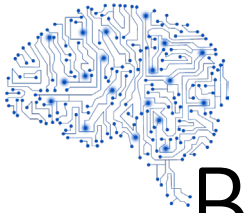
- $\hat{y}$ - це відповідь алгоритму на і-му об'єкті,
- y – справжня мітка класу на і-му об'єкті, а розмір вибірки.

- Інтуїтивно можна уявити мінімізацію log-loss як завдання максимізації accuracy шляхом штрафу за невірні прогнози. Однак необхідно відзначити, що log-loss вкрай сильно штрафує за впевненість класифікатора в невірній відповіді.

Властивість

- Видно, що чим ближче до нуля відповідь алгоритму при ground truth = 1, тим вище значення помилки і крутіше зростає крива.





Висновки

- У разі багатокласової класифікації потрібно уважно стежити за метриками кожного з класів і слідувати логіці рішення задачі, а не оптимізації метрики
- У разі нерівних класів (незбалансованих вибірок) потрібно підбирати баланс класів для навчання і метрику, яка буде коректно відображати якість класифікації
- Вибір метрики потрібно робити з фокусом на предметну область, попередньо обробляючи дані і, можливо, сегментацію (наприклад, з поділом на багатих і бідних клієнтів)

Питання?