

Мультикласова та мультизначна класифікації

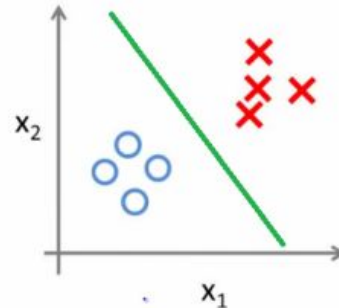
Binary vs Multiclass classifications

Бінарна класифікація:

- + У наборі даних присутні тільки 2 різних класи.
- + Потрібна тільки 1 модель класифікатора.
- + Матрицю невідповідностей легко отримати і зрозуміти.

Приклад: класифікація електронної пошти на спам/не спам

Binary classification:

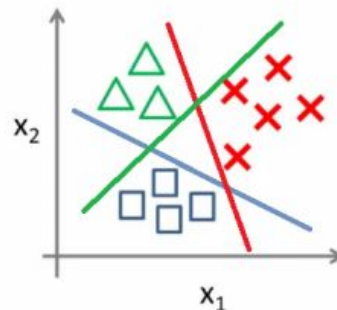


Мультикласова класифікація:

- + У наборі даних є **3 і більше** класів.
- + Кількість моделей класифікаторів залежить від застосовуваної нами методики класифікації.
- + Матрицю відповідностей легко отримати, але складно зрозуміти.

Приклад: класифікація інтересів клієнта на побутові товари, спортивні або вузько-спеціалізовані

Multi-class classification:



Multiclass vs Multilable classification

Pick one

Label 1	✓
Label 2	

Binary

Pick one

Label 1	
Label 2	
Label 3	
Label 4	✓
...	
...	
Label L	

Multi-class

Pick all applicable

Label 1	
Label 2	✓
Label 3	
Label 4	✓
...	
...	
Label L	✓

Multi-label

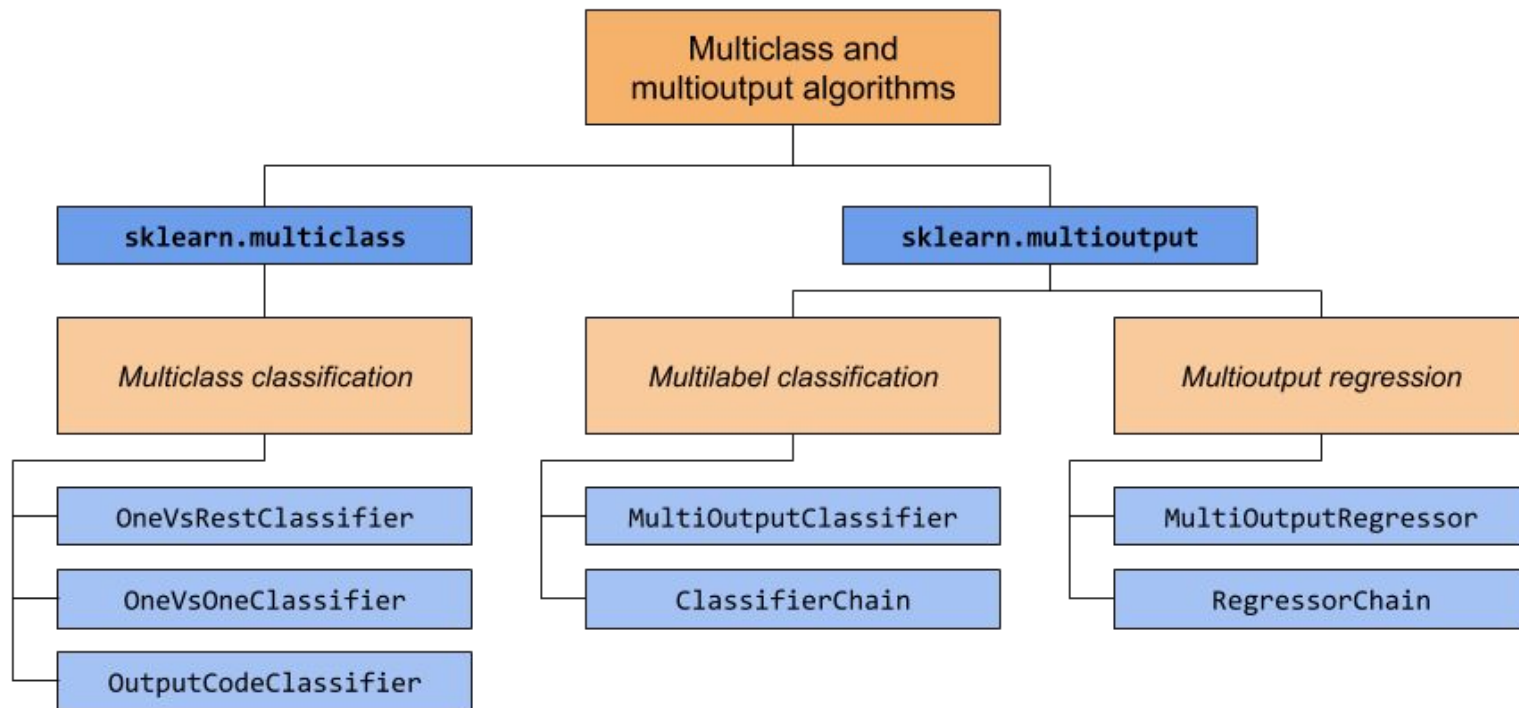
Мультилейбл класифікація

Multi-label classification має справу з випадками, які можуть одночасно бути позначені кількома мітками. Ця техніка ідеально підходить для завдань, де дані можуть належати до більш ніж одного класу одночасно.

Випадки використання:

- **Класифікація сцен на картинці:** Призначення кількох міток (наприклад, пляж, захід сонця, люди) до зображення.
- **Категоризація текстів:** Позначення статей кількома темами, які вони охоплюють.

З чого почати?



<https://scikit-learn.org/stable/modules/multiclass.html>

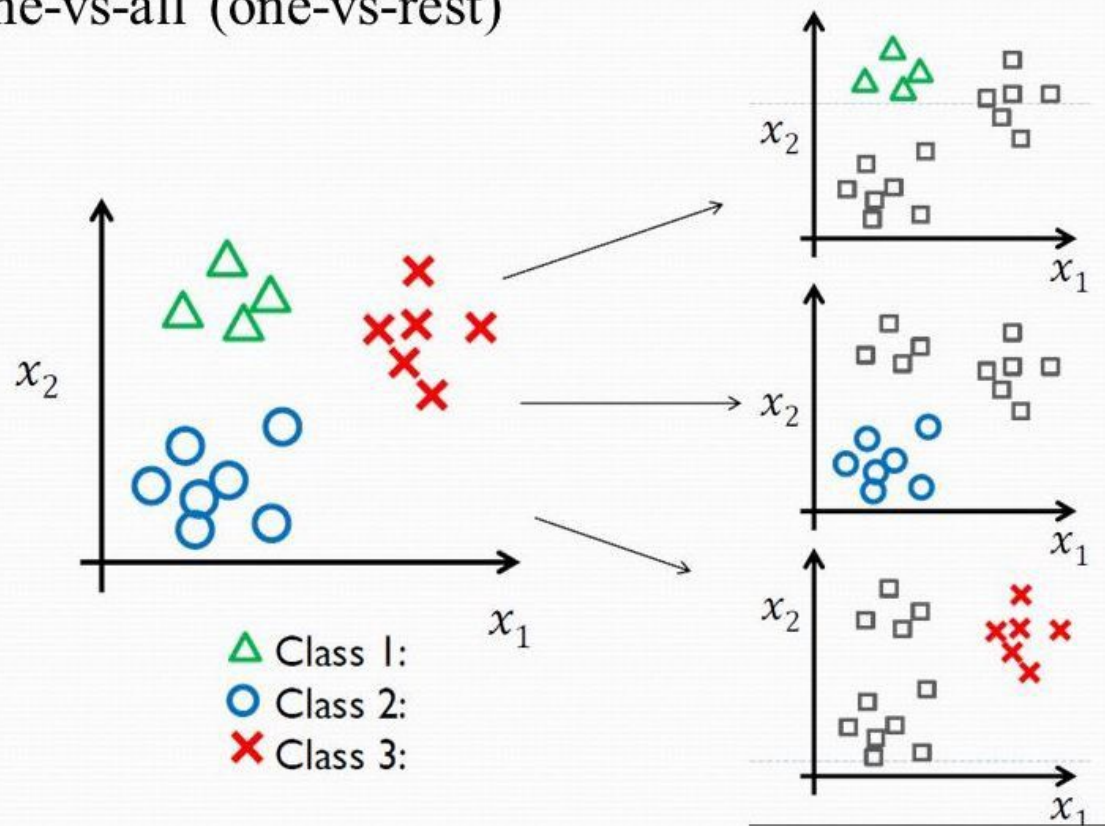
	Number of targets	Target cardinality	Valid <u>type_of_target</u>
Multiclass classification	1	>2	'multiclass'
Multilabel classification	>1	2 (0 or 1)	'multilabel-indicator'
Multiclass-multioutput classification	>1	>2	'multiclass-multioutput'
Multioutput regression	>1	Continuous	'continuous-multioutput'

Стратегії розв'язку мультикласової класифікації

Стратегії розв'язку мультикласової класифікації

- Обираємо алгоритм, який підтримує багато класів (наприклад, дерева прийняття рішень)
- Використовуємо з бінарним класифікатором одну зі стратегій
 - a. Стратегія “Один проти всіх / **One-vs-all**”: для N-класової класифікації використовуємо N бінарних класифікаторів.
 - b. Стратегія “Один проти одного / **One-vs-one**”: для N-класової класифікації використовуємо $N * (N-1) / 2$ бінарних класифікаторів

One-vs-all (one-vs-rest)



One-vs-Rest мультикласова класифікація. Приклад

Сформуємо три навчальні набори даних із новими значеннями цільової змінної.

Main Dataset

Features			Classes
x1	x2	x3	G
x4	x5	x6	B
x7	x8	x9	R
x10	x11	x12	G
x13	x14	x15	B
x16	x17	x18	R

Class 1 :- Green Class 2 :- Blue Class 3 :- Red

Training Dataset 1
Class :- Green

Features			Green
x1	x2	x3	+1
x4	x5	x6	-1
x7	x8	x9	-1
x10	x11	x12	+1
x13	x14	x15	-1
x16	x17	x18	-1

Training Dataset 2
Class :- Blue

Features			Blue
x1	x2	x3	-1
x4	x5	x6	+1
x7	x8	x9	-1
x10	x11	x12	-1
x13	x14	x15	+1
x16	x17	x18	-1

Training Dataset 3
Class :- Red

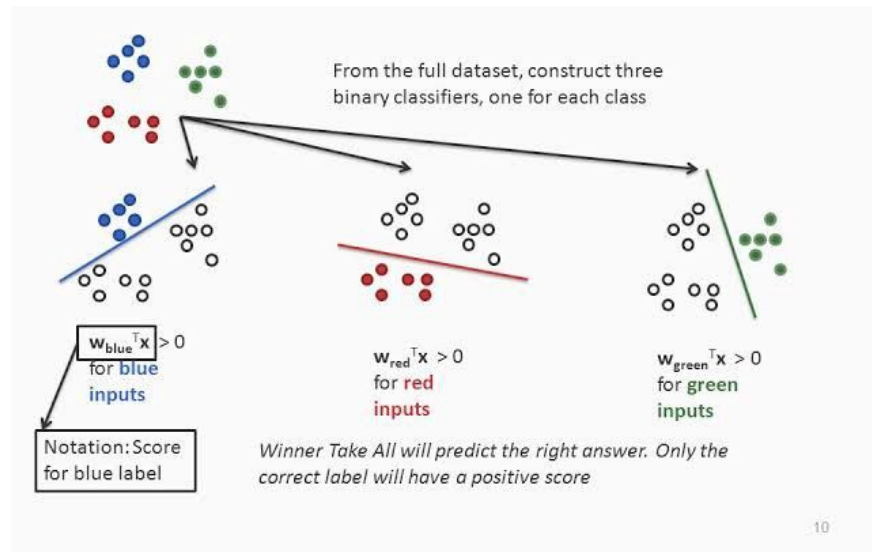
Features			Red
x1	x2	x3	-1
x4	x5	x6	-1
x7	x8	x9	+1
x10	x11	x12	-1
x13	x14	x15	-1
x16	x17	x18	+1

Після тренування трьох моделей беремо той клас, для приналежності до якого найвища ймовірність

Є три значення тестових функцій y_1 , y_2 і y_3 відповідно.

Ми передали тестові дані в моделі класифікаторів. Ми отримали результат у $p(\text{Green})=0.9$
 $p(\text{Blue})=0.4$
 $p(\text{Red})=0.1$

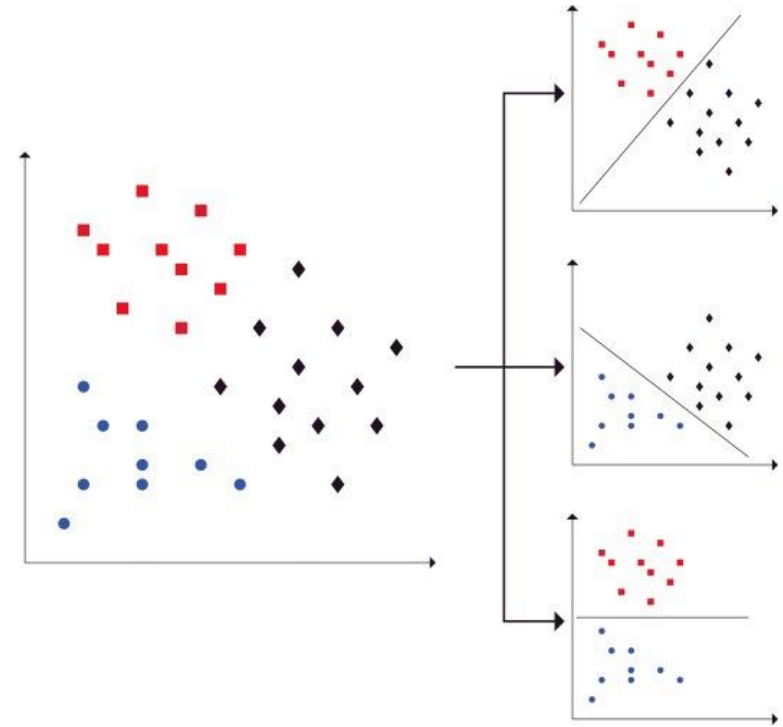
Отже, ми можемо сказати, що наші тестові дані належать до класу Green.



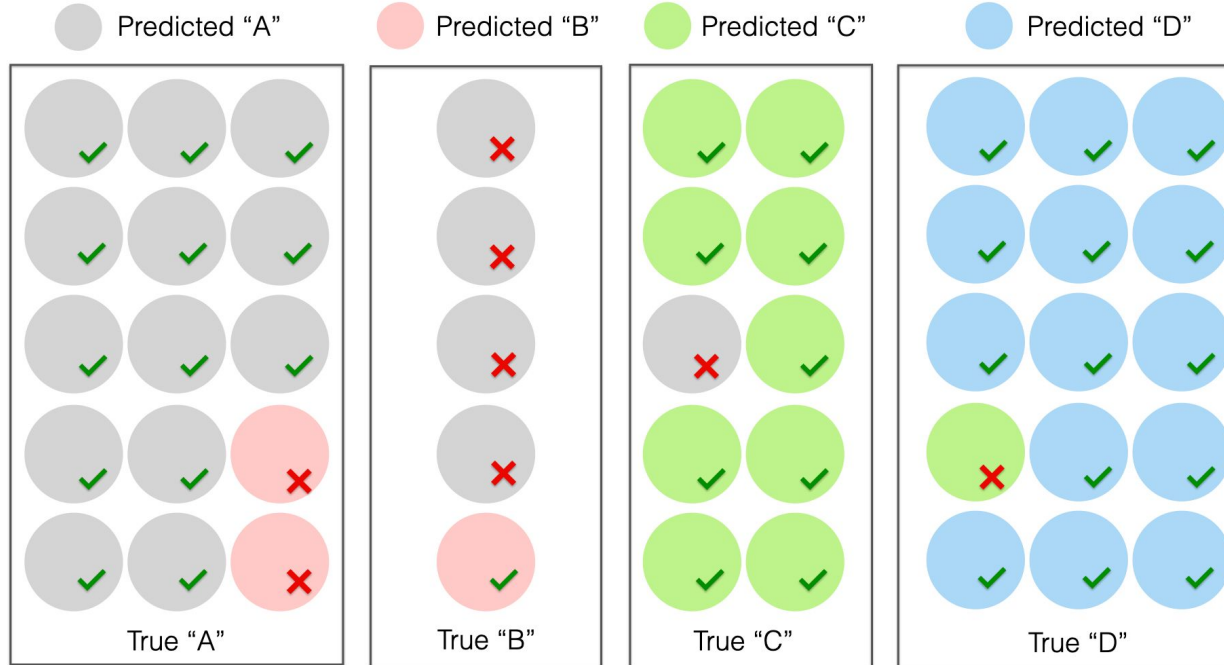
One vs one (OVO)

У мультикласовій класифікації «Один проти одного» ми розбиваємо первинний набір даних на один набір даних двійкової класифікації для кожної пари класів.

Кожен двійковий класифікатор передбачає одну мітку класу. Коли ми вводимо тестові дані в класифікатор, **результатом буде той клас, який «переміг» у максимальній кількості бінарних класифікацій.**



А як вимірювати якість?



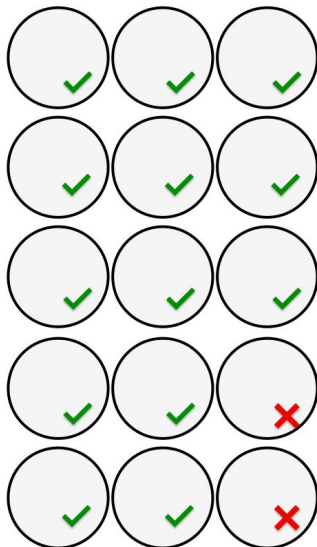
Найпростіше - Accuracy



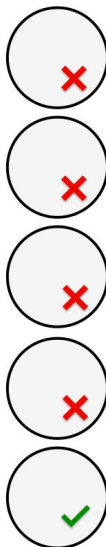
Correct predictions



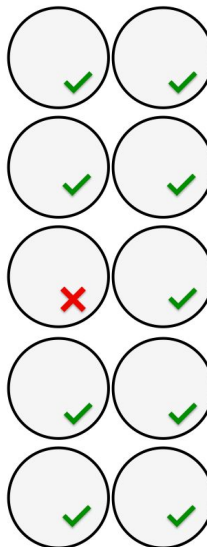
Wrong predictions



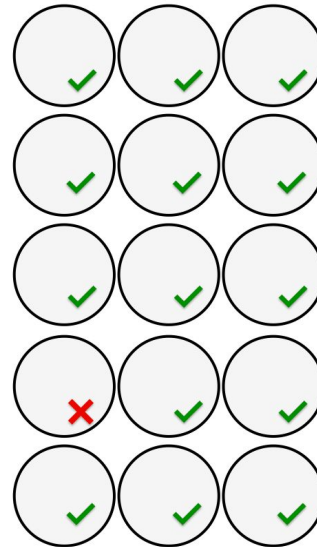
True "A"



True "B"



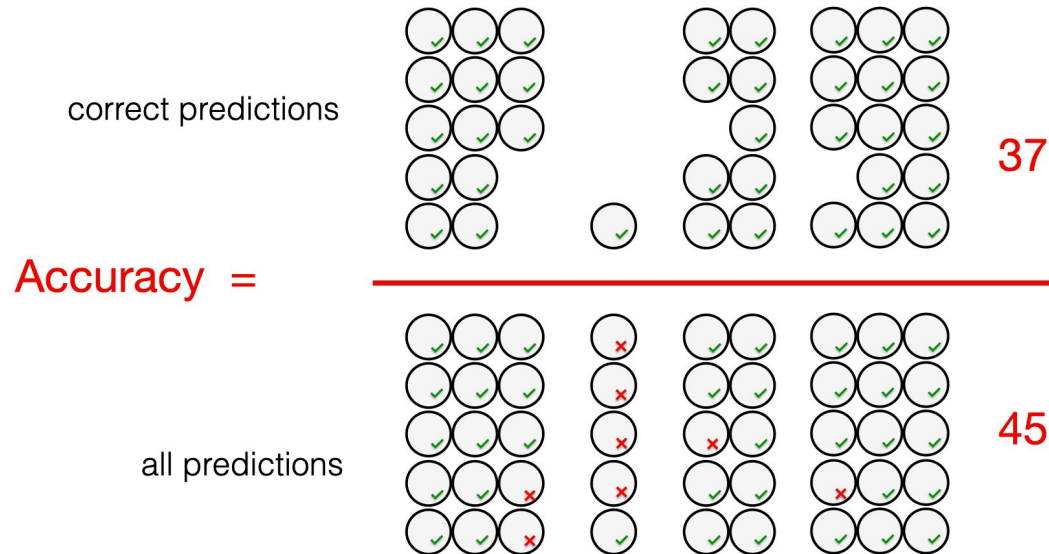
True "C"



True "D"



Найпростіше - Accuracy



Точність $37/45 = 82\%$

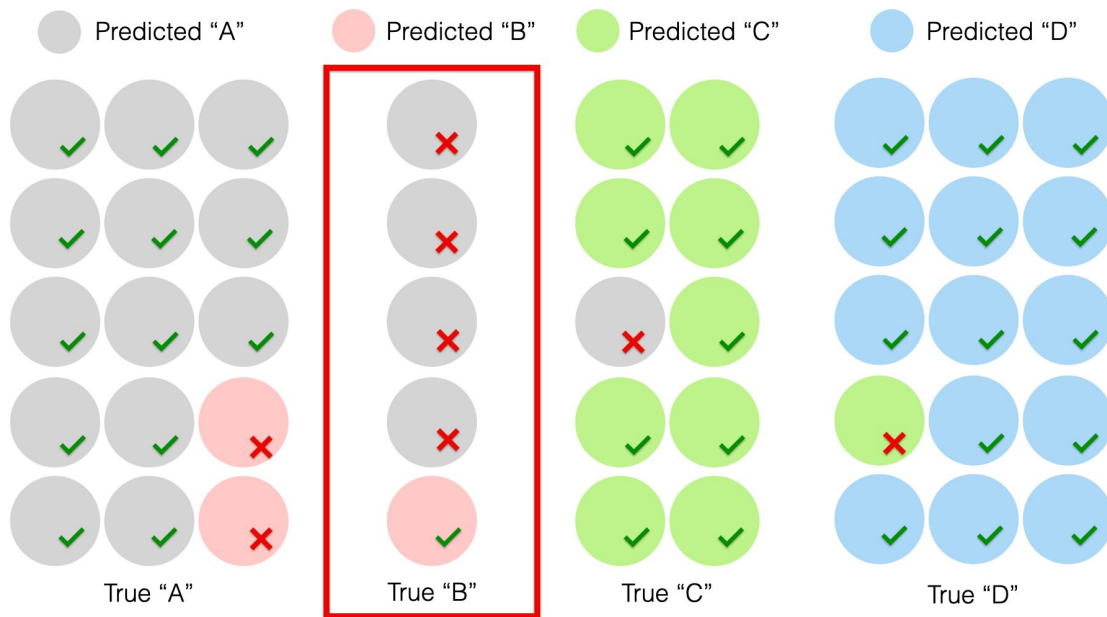
Плюси і недоліки Accurasy

Точність *легко інтерпретувати*. Ви створили модель, яка правильно класифікує 90 з 100 зразків? Точність становить 90%! Класифікувала 87? 87%!

Однак точність має свої недоліки. Хоча вона дає оцінку загальної якості моделі, *вона не враховує баланс класів і вартість різних помилок*.

Так само, як у випадку бінарної класифікації, при багатокласовій класифікації деякі класи можуть бути більш поширеними. Моделі може бути легше їх класифікувати – на шкоду меншим за кількістю класам. У такому випадку висока точність може бути оманливою.

В нашому випадку
модель погано
впоралась з
класифікацією класу B



Precision, recall, F1 в multi-class задачах

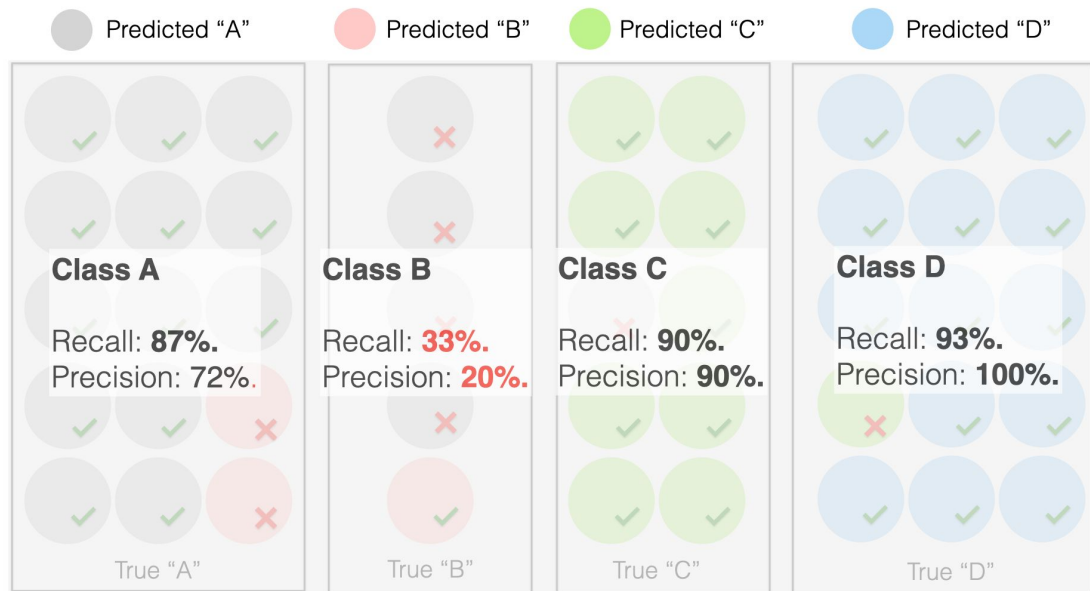
Precision та recall метрики також не обмежені лише бінарною класифікацією. Ви можете використовувати їх і в задачах багатокласової класифікації.

Однак, існують різні підходи до того, як це зробити, кожен з яких має свої переваги та недоліки.

- У першому випадку ви можете обчислити precision та recall для кожного класу окремо. Таким чином, ви отримаєте кілька метрик залежно від кількості класів у наборі даних.
- У другому випадку ви можете "усереднити" precision та recall по всіх класах, щоб отримати єдине число. Ви можете використовувати різні методи усереднення precision, такі як **macro-** або **micro-averaging**.

Розглянемо обидва підходи, починаючи з обчислення метрик по класах.

Обчислення по класах



Обчислення по класах

Коли варто обчислювати precision та recall по класах?

- Коли у вас є **певні класи, які вас цікавлять**. Це корисно, коли потрібно оцінити ефективність роботи класифікатора для конкретного класу та зрозуміти, наскільки добре він може відрізнити цей клас від інших.
- Коли ви маєте справу з **незбалансованими класами**. Обчислення precision та recall для менших класів чітко покаже будь-які проблеми.
- Коли у вас **невелика кількість класів**. У цьому випадку можна оцінити кожен метрику окремо і зрозуміти їх разом. Це часто найпростіше рішення.

Однак є й недоліки:

- Обчислення precision та recall для кожного класу може призвести до великої кількості метрик. Чим більше класів, тим більше метрик, що може ускладнити їхню інтерпретацію.
- Коли у вас багато класів, краще використовувати macro або micro averages. Вони надають більш стислий підсумок продуктивності.

Micro- та macro-average

Micro-average показник Precision і Recall розраховується на основі кількості істинно позитивних (TP), істинно негативних (TN), хибнопозитивних (FP) і хибнонегативних (FN) передбачень моделі.

Macro-average оцінка Precision і Recall розраховується як середнє арифметичне оцінок Precision і Recall окремих класів.

Macro-average значення F1-score розраховується як середнє арифметичне F1-score окремих класів.

Macro-average

Ось як обчислити macro-averaged precision та recall:

1. Обчисліть кількість справжніх позитивних (TP), хибних позитивних (FP) і хибних негативних (FN) для кожного класу.
2. Обчисліть precision та recall для кожного класу за формулами $TP / (TP + FP)$ та $TP / (TP + FN)$.
3. Усередніть precision та recall по всіх класах, щоб отримати остаточні значення macro-averaged precision та recall.

$$\begin{aligned} \text{Precision}_{\text{Macro-average}} &= \frac{\text{Precision}_{\text{Class A}} + \text{Precision}_{\text{Class B}} + \dots + \text{Precision}_{\text{Class N}}}{N} \\ \text{Recall}_{\text{Macro-average}} &= \frac{\text{Recall}_{\text{Class A}} + \text{Recall}_{\text{Class B}} + \dots + \text{Recall}_{\text{Class N}}}{N} \end{aligned}$$

Приклад обчислення macro-average

Class 1: Urgent

	true urgent	true not
system urgent	8	11
system not	8	340

$$\text{precision} = \frac{8}{8+11} = .42$$

Class 2: Normal

	true normal	true not
system normal	60	55
system not	40	212

$$\text{precision} = \frac{60}{60+55} = .52$$

Class 3: Spam

	true spam	true not
system spam	200	33
system not	51	83

$$\text{precision} = \frac{200}{200+33} = .86$$

Pooled

	true yes	true no
system yes	268	99
system no	99	635

$$\text{microaverage precision} = \frac{268}{268+99} = .73$$

$$\text{macroaverage precision} = \frac{.42+.52+.86}{3} = .60$$

Micro-average

Альтернативно, можна обчислити micro-average precision та recall.

Для цього спочатку необхідно підрахувати загальну кількість справжніх позитивних (TP), хибних позитивних (FP) та хибних негативних (FN) передбачень по всіх класах:

- Total True Positive - сума справжніх позитивних по всіх класах;
- Total False Positive - сума хибних позитивних по всіх класах;
- Total False Negative - сума хибних негативних по всіх класах.

Потім обчисліть precision та recall, використовуючи ці загальні значення.

- Щоб обчислити precision, поділіть загальну кількість True Positives на суму Total True Positives і False Positives.
- Щоб обчислити recall, поділіть загальну кількість True Positives на суму Total True Positives і False Negatives.

Приклад обчислення micro-average

Class 1: Urgent

	true urgent	true not
system urgent	8	11
system not	8	340

$$\text{precision} = \frac{8}{8+11} = .42$$

Class 2: Normal

	true normal	true not
system normal	60	55
system not	40	212

$$\text{precision} = \frac{60}{60+55} = .52$$

Class 3: Spam

	true spam	true not
system spam	200	33
system not	51	83

$$\text{precision} = \frac{200}{200+33} = .86$$

Pooled

	true yes	true no
system yes	268	99
system no	99	635

$$\text{microaverage precision} = \frac{268}{268+99} = .73$$

Коли що використовувати

Використовуємо оцінку **micro-average**, коли необхідно однаково зважити кожен екземпляр або прогноз.

Використовуємо оцінку **macro-average**, коли всі класи потрібно обробляти однаково, щоб оцінити загальну якість класифікатора щодо міток класів, які найчастіше зустрічаються.

Використовуйте зважену (weighted) оцінку **macro-average у разі дисбалансу класів** (різна кількість екземплярів, пов'язаних із різними мітками класів). Середньозважене макро-середнє значення розраховується шляхом зважування оцінки кожної мітки класу за кількістю істинних екземплярів під час обчислення середнього.

Micro-, macro-average в Sklearn

Classification report

	precision	recall	f1-score	support
Aeroplane	0.67	0.67	0.67	3
Boat	0.25	1.00	0.40	1
Car	1.00	0.50	0.67	6
accuracy			0.60	10
macro avg	0.64	0.72	0.58	10
weighted avg	0.82	0.60	0.64	10

Per-Class F1 scores

Average F1 scores

```
#  
# Average is assigned micro  
#  
precisionScore_sklearn_microavg = precision_score(y_test,  
y_pred, average='micro')  
#  
# Average is assigned macro  
#  
precisionScore_sklearn_macroavg = precision_score(y_test,  
y_pred, average='macro')
```