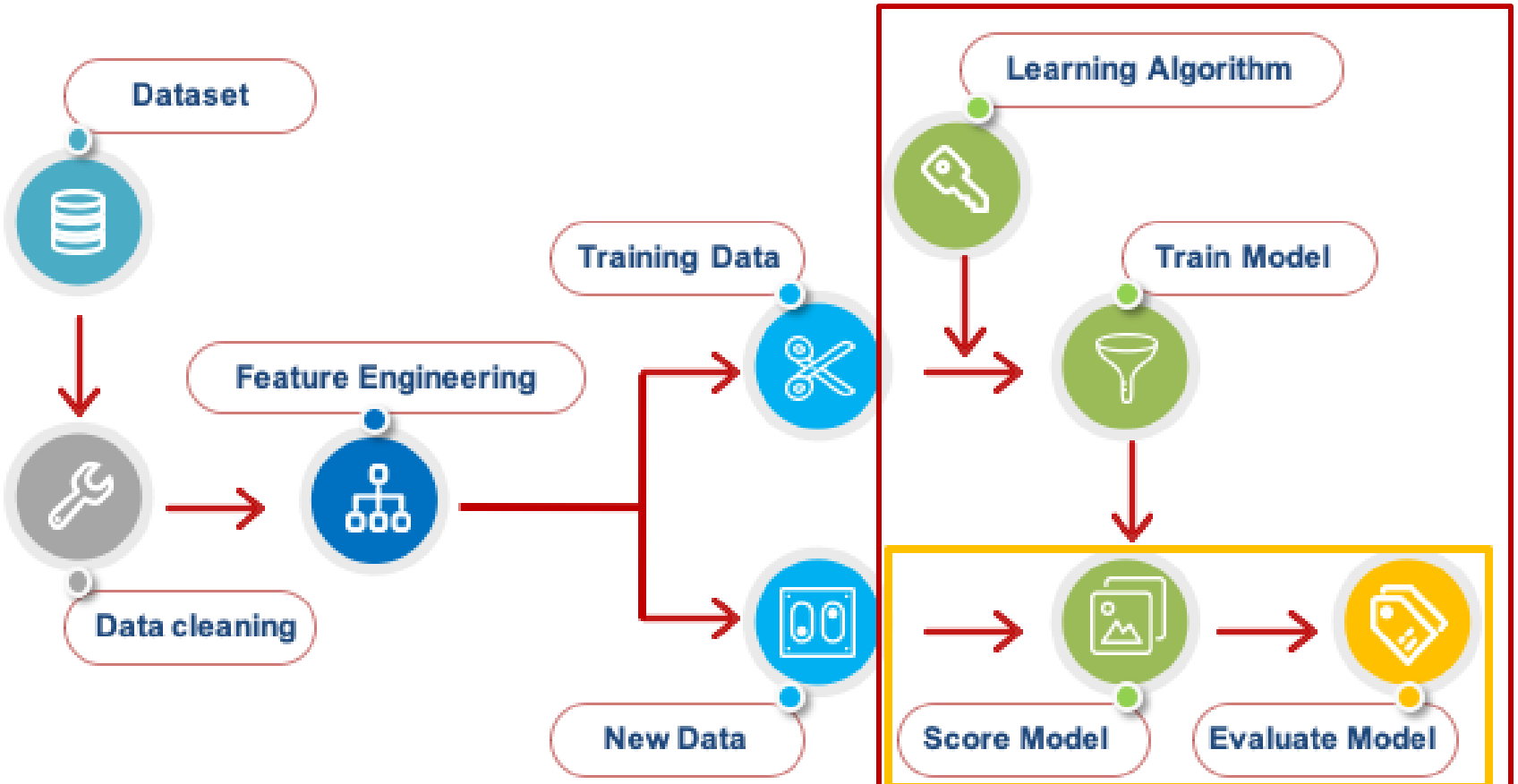
Machine Learning Performance Metrics

“Evaluating Model Performance Effectively”

Onoja Anthony, PhD

Machine Learning Models Require Evaluation To Ensure Their Effectiveness.

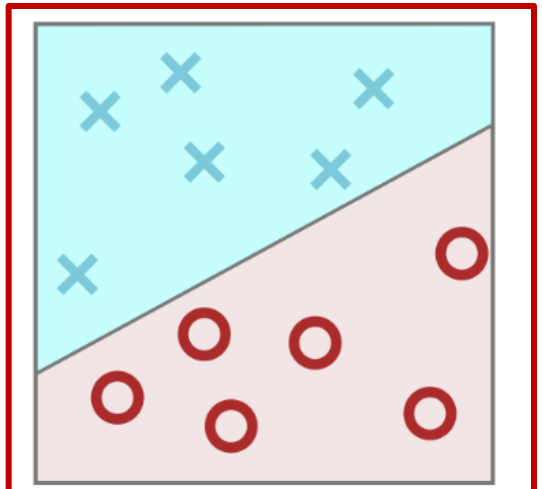


Performance metrics help determine how well a model performs.

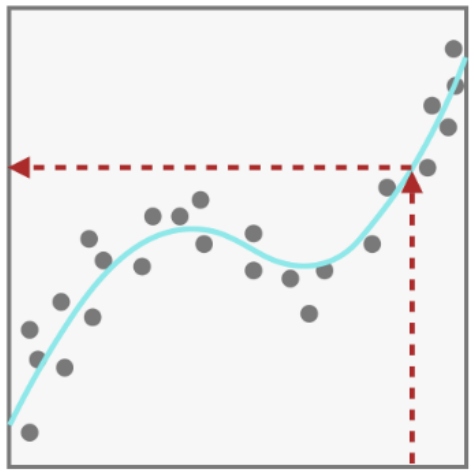
Different models require different metrics for evaluation.

To build a good model, think of yourself as a chef carefully mixing the right ingredients.

Classification Vs. Regression Metrics



Classification



Regression

- Classification: Used when predicting categorical labels (e.g., spam detection, disease diagnosis).
- Regression: Used when predicting continuous values (e.g., house prices, temperature prediction).

Each task requires specific evaluation metrics to assess performance effectively.

Classification Metrics Overview

Accuracy: Measures overall correctness.

Precision: Focuses on correctly identified positive cases.

Recall (Sensitivity): Measures how many actual positives were correctly predicted.

F1-score: Harmonic mean of precision and recall.

ROC-AUC Score: Evaluates the model's ability to distinguish between classes.

	Actual Positive	Actual Negative
Predicted Positive	True Positive(TP)	False Positive(FP) (Type 1 Error)
Predicted Negative	False Negative(FN) (Type 2 Error)	True Negative(TN)

$$\text{Accuracy} = \frac{\text{True Positive + True Negative}}{\text{Total Population}}$$

$$\text{Error Rate/Misclassification rate} = \frac{\text{False Positive + False Negative}}{\text{Total Population}}$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{Predicted Positive(TP+FP)}}$$

$$\text{Sensitivity/Recall} = \frac{\text{True Positive}}{\text{Actual Positive(TP+FN)}}$$

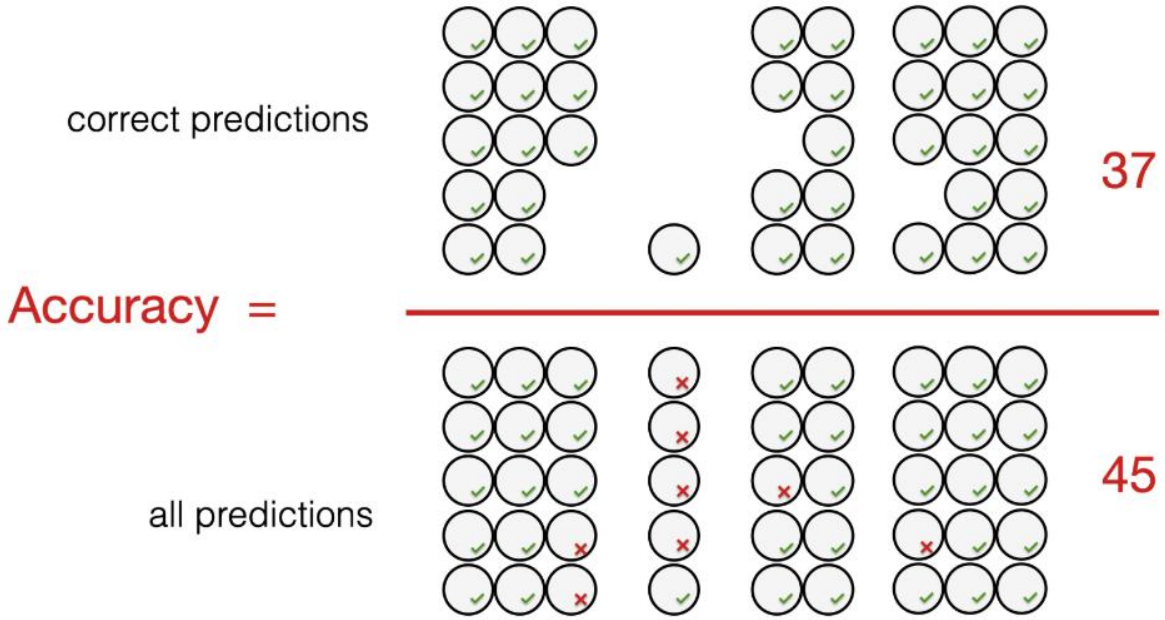
$$\text{Specificity} = \frac{\text{True Negative}}{\text{Actual Negative(FP+TN)}}$$

$$\text{F1 Score} = \frac{2 * (\text{Recall} * \text{Precision})}{\text{Recall + Precision}}$$

Confusion Matrix: Provides a breakdown of predictions into TP, TN, FP, and FN.

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

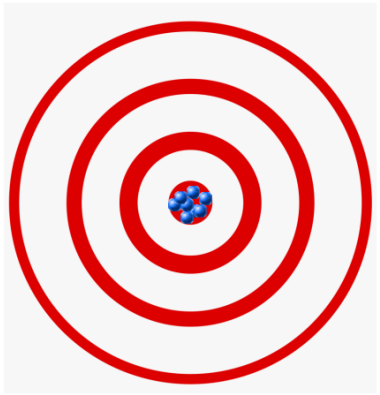
What is the Accuracy Score Metric?



$$Accuracy = \frac{Correct\ Predictions}{All\ Predictions} = \frac{TP + TN}{TP + TN + FP + FN}$$

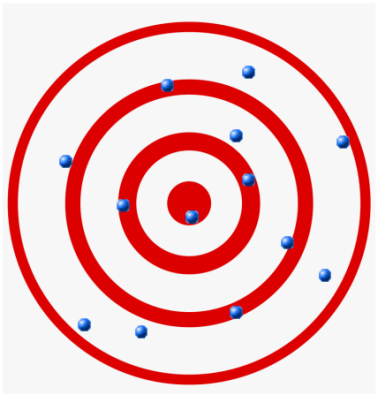
When to Use: Works well when classes are balanced.

Limitations: Can be misleading for imbalanced datasets.



A: accurate and precise

B: precise, but not accurate



C: neither accurate nor precise

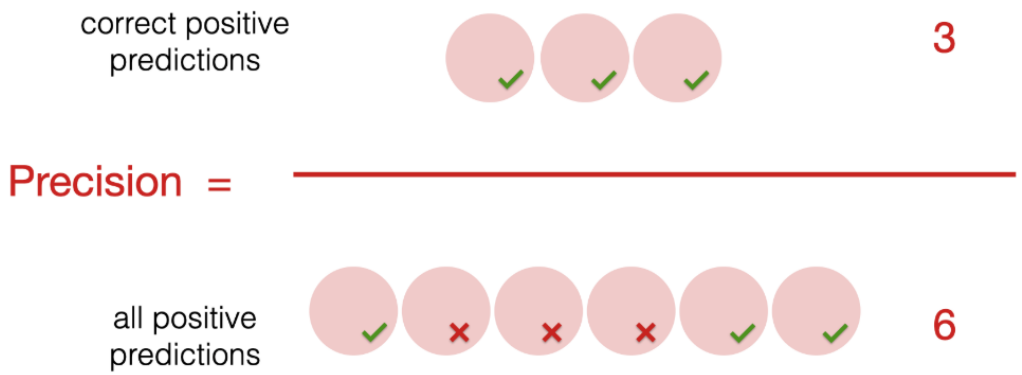
D: accurate, but not precise

Precision-Accuracy trade-off

An overview of the Precision and Recall Metrics

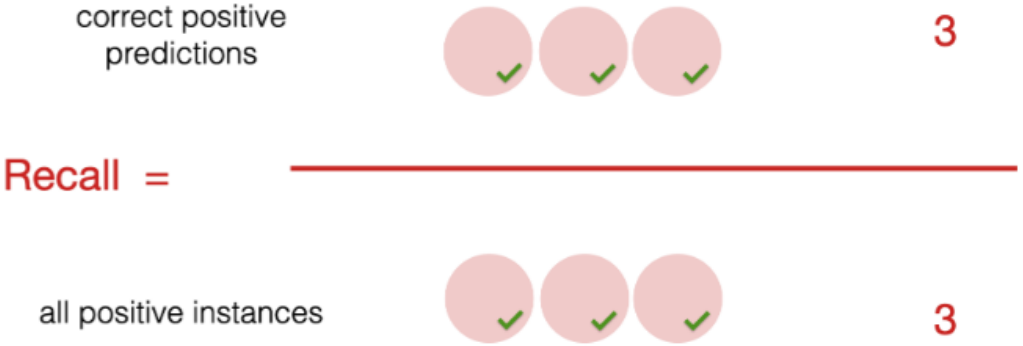
$$\text{Precision} = \frac{TP}{TP + FP}$$

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



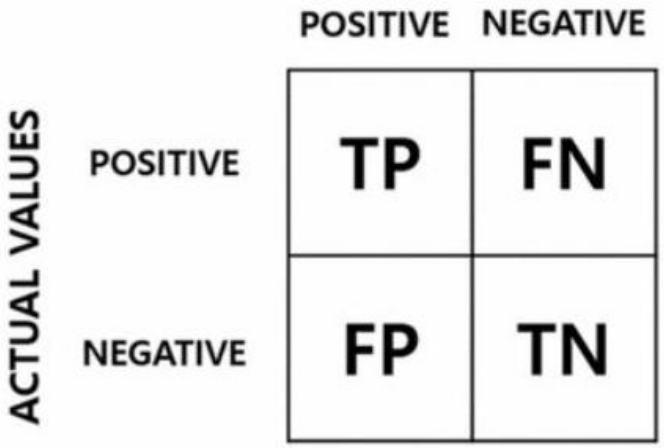
Trade-off: Increasing precision may lower recall and vice versa.

- *Pros: Precision is valuable for imbalanced data and when false positives are costly, ensuring accurate identification of the target class.*
- *Cons: Precision ignores false negatives, meaning it doesn't account for missed target events.*



- *Pros: Recall performs well with imbalanced classes by focusing on detecting target-class objects.*
- *Cons: Recall is that it does not account for the cost of these false positives.*

What is the F1-Score?



$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

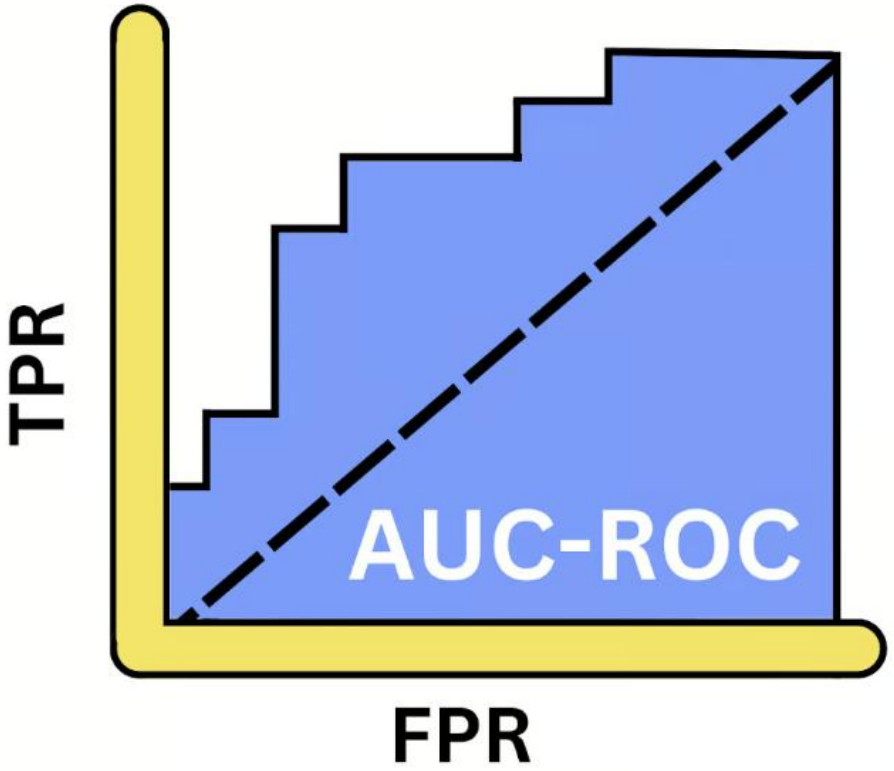
$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

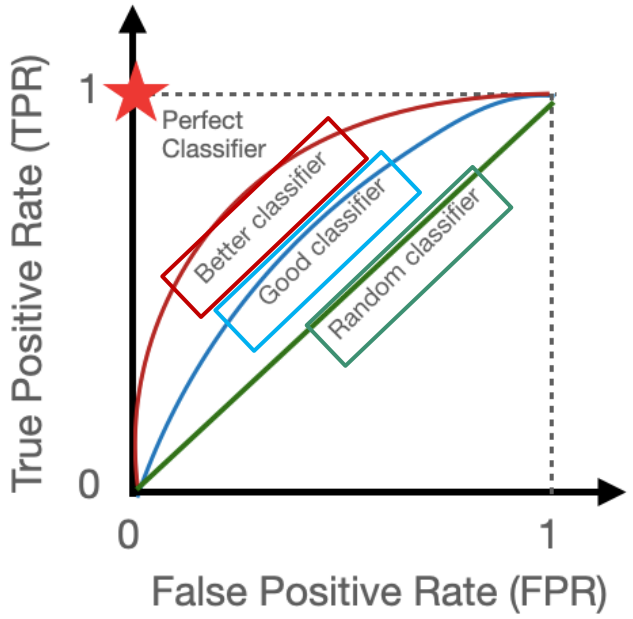
Balances precision and recall.

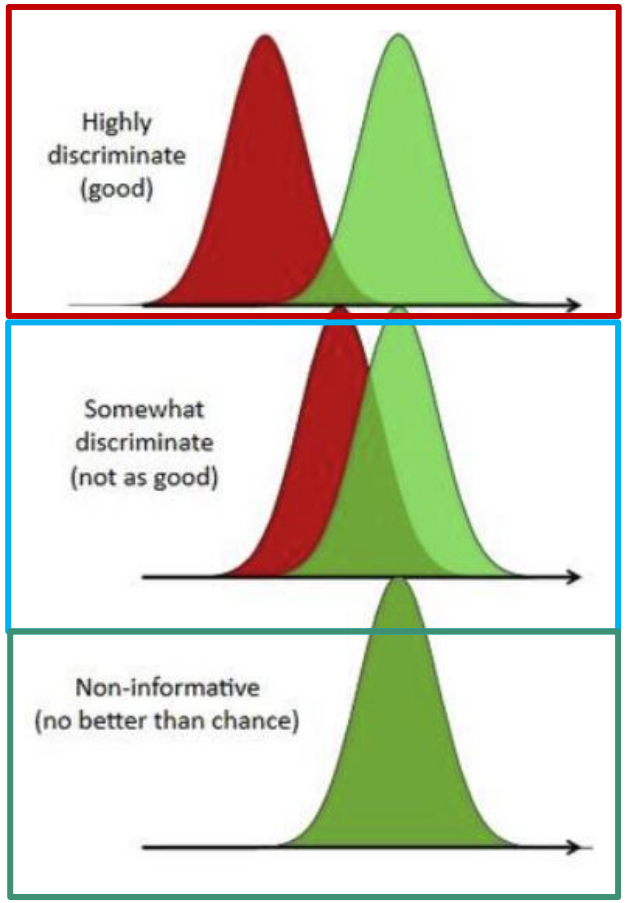
- It is useful when dealing with imbalanced datasets.
- Ideal for applications where both false positives and false negatives matter.

What is the ROC Curve & AUC?

It is the plots of TPR vs. FPR at different thresholds.





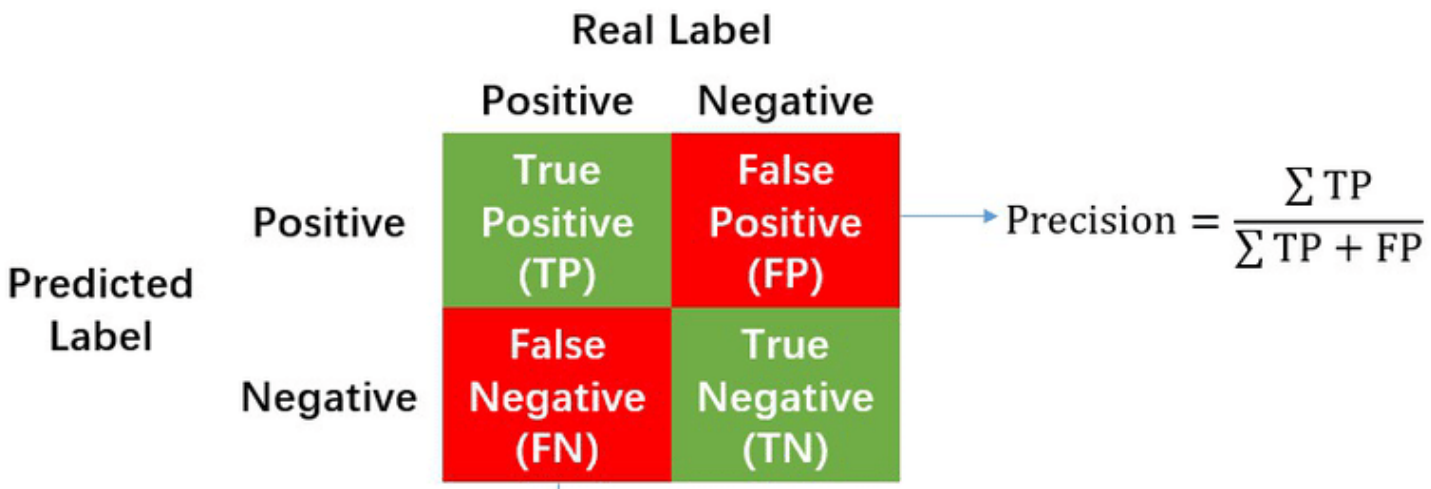


It is useful for evaluating binary classifiers.

AUC (Area Under Curve): Measures overall classification performance.

Confusion Matrix, what is it?

It is a table representation of classification performance.



True Positives (TP): Correctly predicted positives.

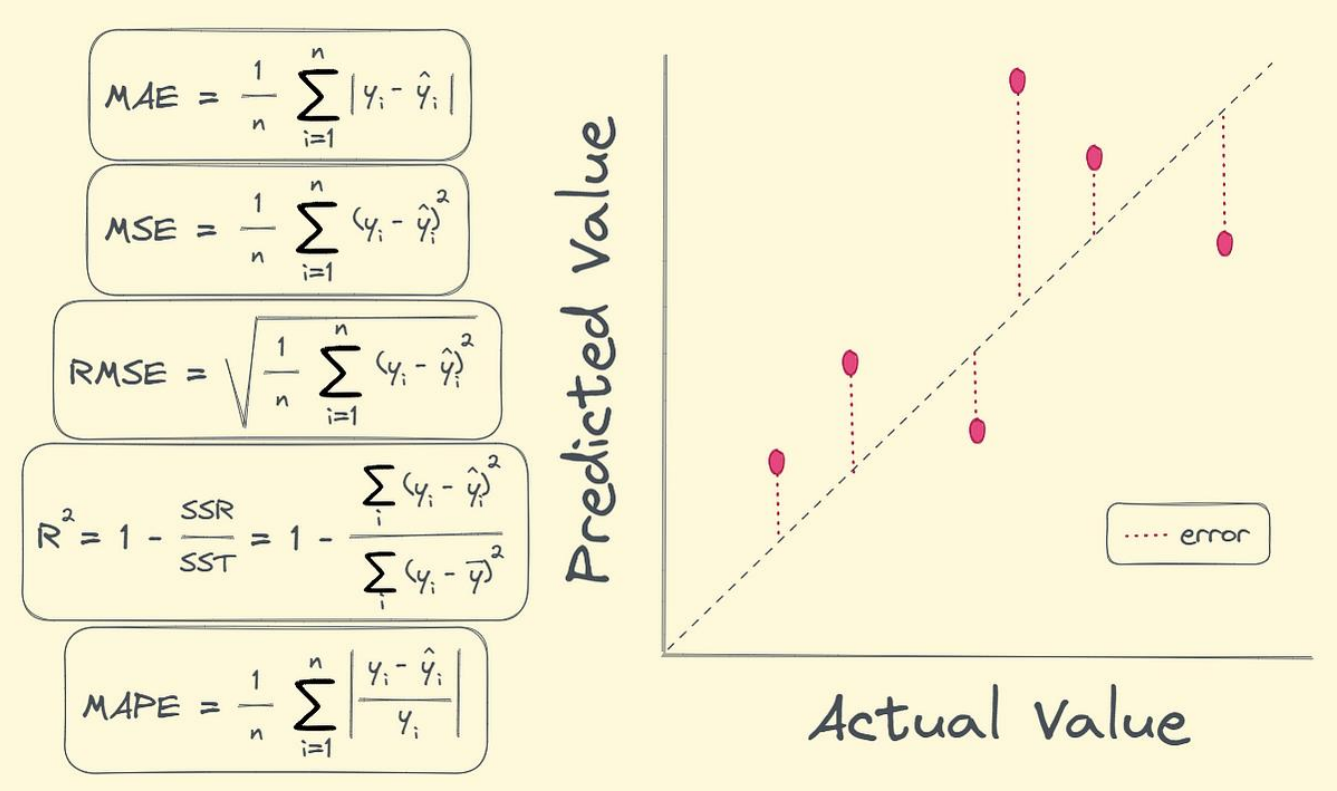
False Positives (FP): Incorrectly predicted positives.

True Negatives (TN): Correctly predicted negatives.

False Negatives (FN): Incorrectly predicted negatives.

It help to visualise model errors and biases.

Regression Metrics Overview



Mean Absolute Error (MAE): Measures average magnitude of errors.

Mean Squared Error (MSE): Penalises large errors.

Root Mean Squared Error (RMSE): Square root of MSE, interpretable in original units.

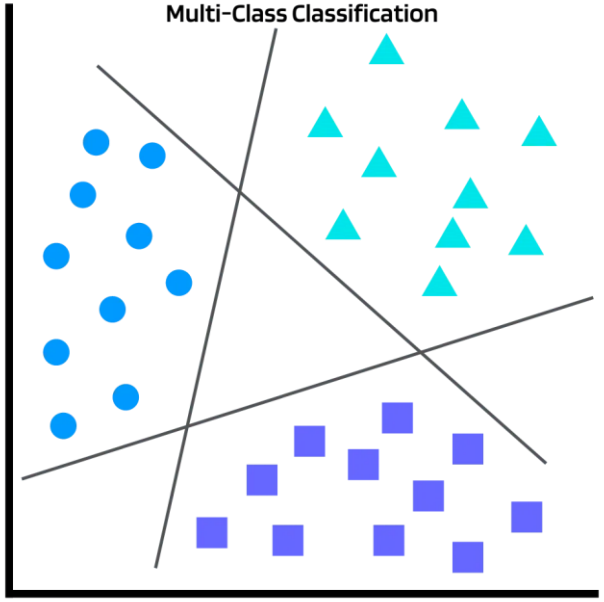
R-squared (R^2): Measures variance explained by the model. 1 means perfect fit, 0 no explain variance and negative implies worse performance than a simple mean predictor.

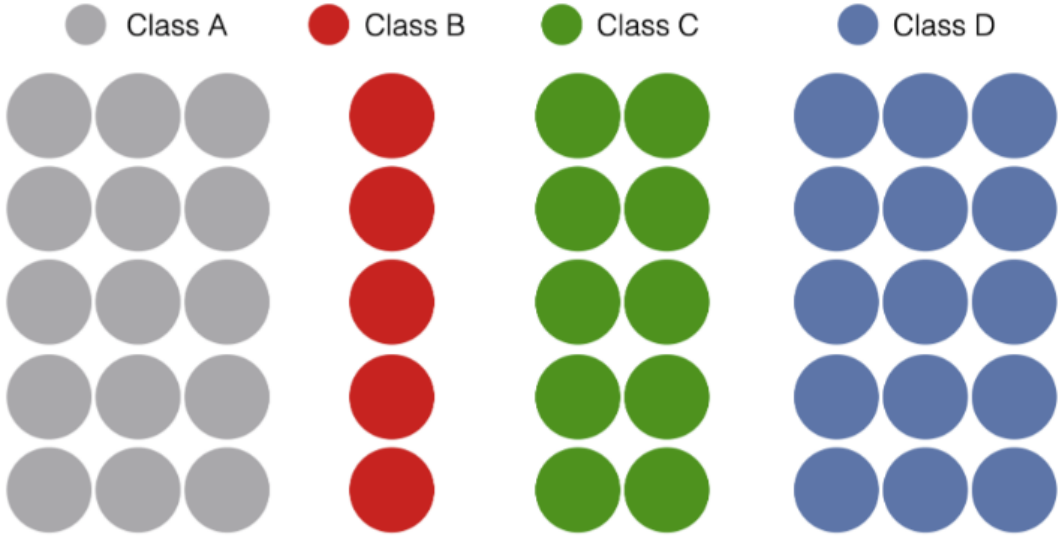
MAE: Easy to interpret, however, it treats all errors equally, ignoring their direction.

MSE: Penalises larger errors more than smaller ones. However, it is great for optimisation, and interpretability.

Accuracy, Precision, and Recall in Multi-class classification

- Multi-class classification assigns inputs to one of several categories, unlike binary classification, which deals with only two. It differs from multi-label classification. It involves predicting multiple categories..





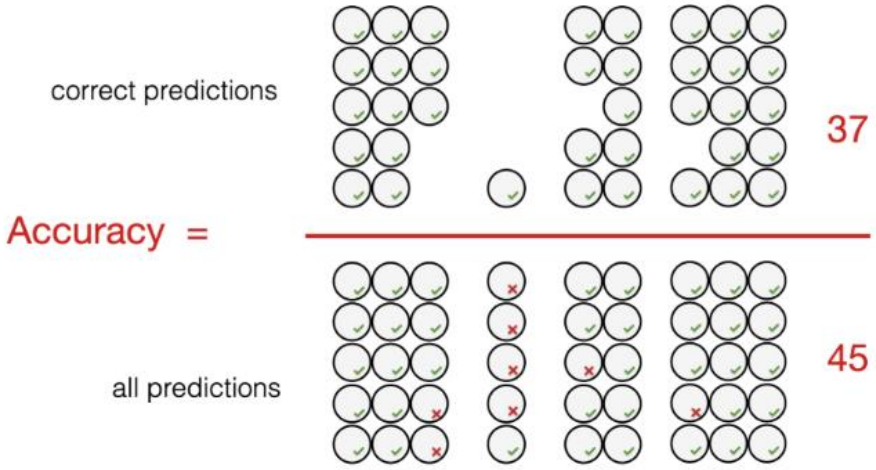
https://www.evidentlyai.com/classification-metrics/multi-class-metrics

Choosing the right approach depends on dataset characteristics.

Key metrics include:

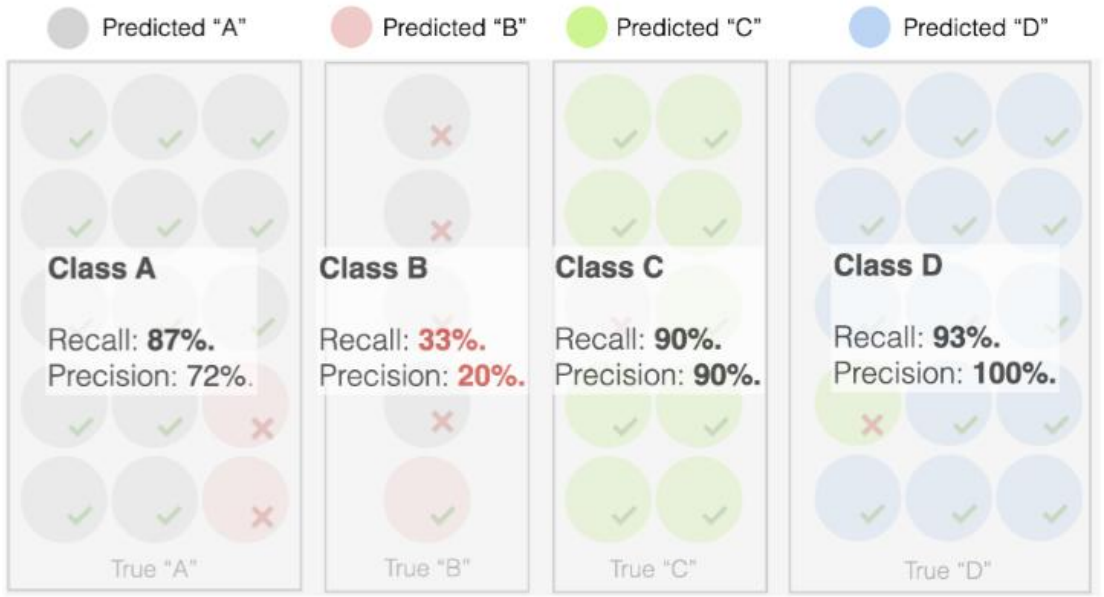
- Macro-Averaging: Averages metrics across all classes equally.
- Micro-Averaging: Aggregates predictions to compute overall performance.
- Weighted Averaging: Weighs metrics by class size, balancing imbalanced datasets.

When to Use Multi-Class Metrics



Accuracy provides an overall estimate of model quality but ignores class balance and error costs. High accuracy can be misleading in imbalanced datasets where the model favours majority classes at the expense of minority ones.

Calculating precision and recall by class is useful when specific classes require detailed evaluation, especially in imbalanced datasets where minority class performance matters. It helps assess how well a model distinguishes a particular class.

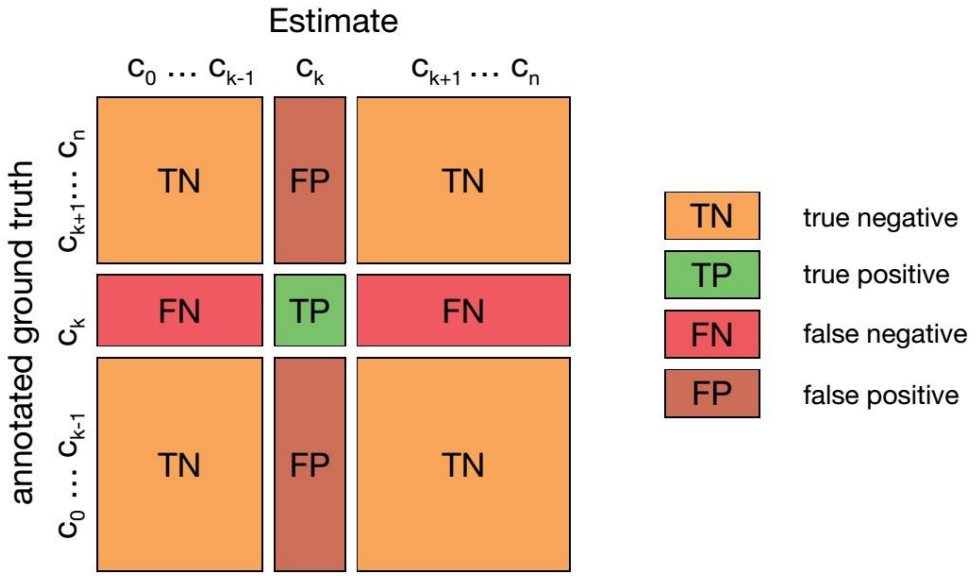


However, for a large number of classes, individual metrics can be overwhelming. In such cases, macro, micro, or weighted averaging provides a more concise summary.

When to Use Multi-Class Metrics

- Macro-Averaging: Best when all classes are equally important.
- Micro-Averaging: Ideal when total misclassifications matter most.
- Weighted Averaging: Useful for imbalanced datasets to avoid bias towards majority classes.
- Consider domain specific interest when selecting metrics.

What is the Macro-averaging?



- Calculate the number of true positives (TP), false positives (FP), and false negatives (FN) for each class.
- Compute precision and recall for each class as TP / (TP + FP) and TP / (TP + FN).

$$\text{Precision}_{\text{Macro-average}} = \frac{\text{Precision}_{\text{Class A}} + \text{Precision}_{\text{Class B}} + \dots \text{Precision}_{\text{Class N}}}{N}$$

$$\text{Recall}_{\text{Macro-average}} = \frac{\text{Recall}_{\text{Class A}} + \text{Recall}_{\text{Class B}} + \dots \text{Recall}_{\text{Class N}}}{N}$$

Average the precision and recall across all classes to get the final macro-averaged precision and recall scores.

What is the Micro-averaging?

First, calculate the total number of true positives (TP), false positives (FP), and false negatives (FN) across all classes.

- Total True Positive is the sum of true positive counts across all classes;
- Total False Positive is the sum of false positive counts across all classes;
- Total False Negative is the sum of false negative counts across all classes.

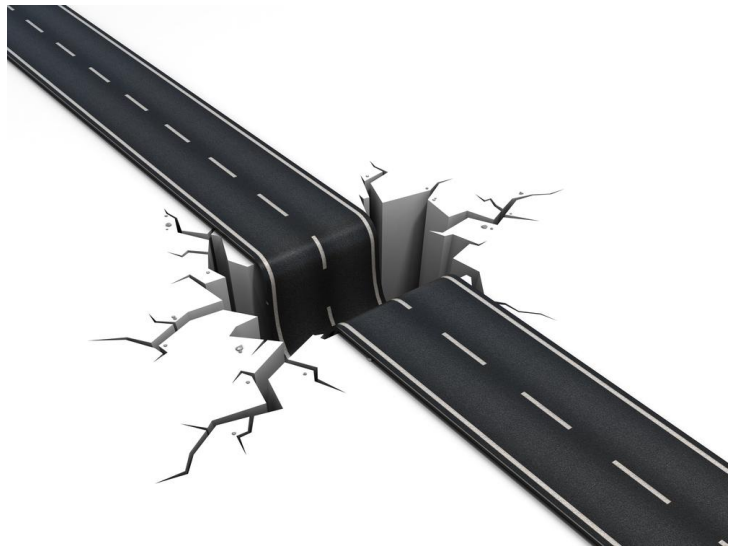
$$\text{Precision}_{\text{Micro-average}} = \frac{TP_A + TP_B + \dots TP_N}{TP_A + FP_A + TP_B + FP_B + \dots TP_N + FP_N}$$

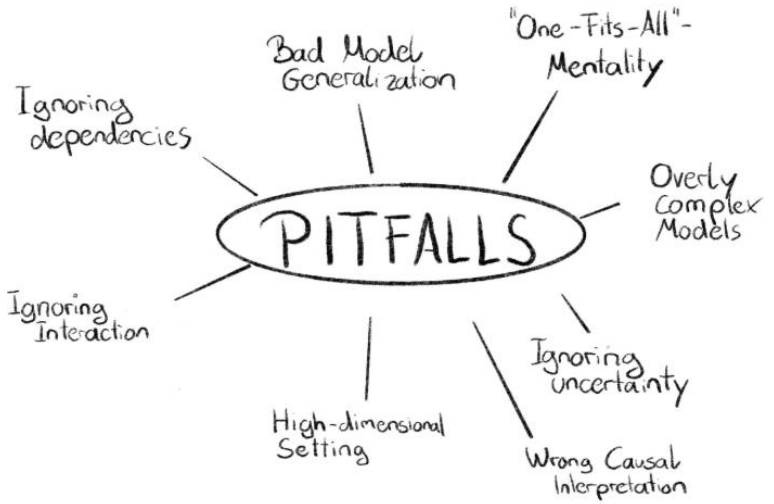
$$\text{Recall}_{\text{Micro-average}} = \frac{TP_A + TP_B + \dots TP_N}{TP_A + FN_A + TP_B + FN_B + \dots TP_N + FN_N}$$

To calculate precision, divide True Positives by the sum of True Positives and False Positives. For recall, divide True Positives by the sum of True Positives and False Negatives.

Common Pitfalls in Model Evaluation

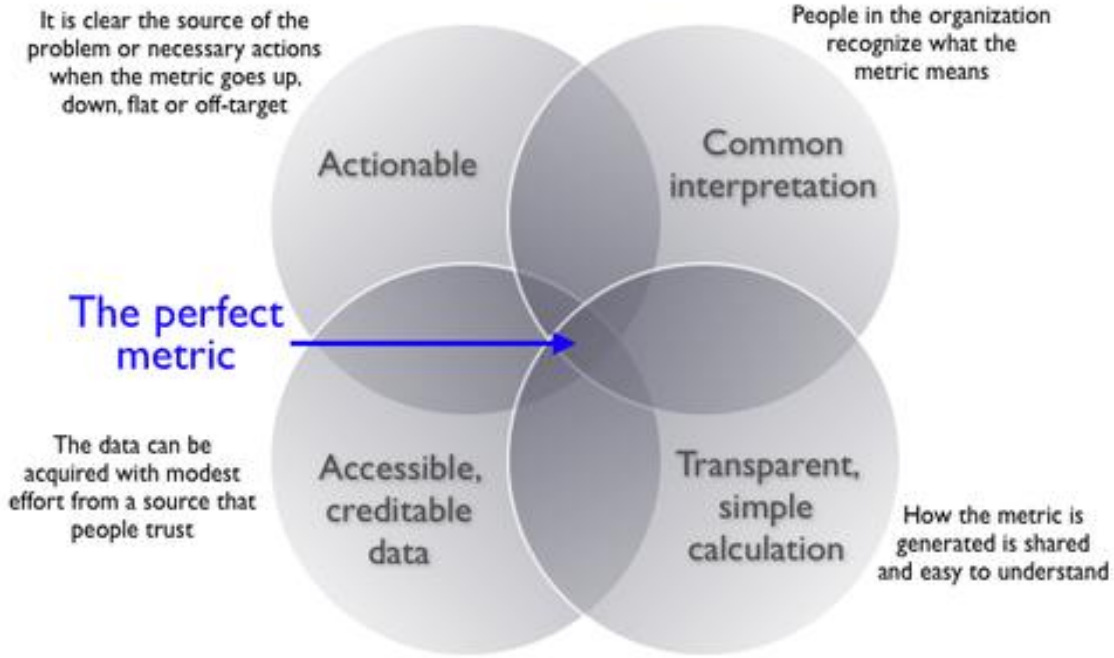
- ❑ Ignoring class imbalance when using accuracy.
- ❑ Misinterpreting precision-recall trade-offs.
- ❑ Over-relying on a single metric instead of a holistic approach.
- ❑ Ignoring domain specific question (hypotheses) in metric selection.





Choosing the Right Metric

Consider domain-specific requirements when selecting metrics.



Classification: Choose based on class balance (e.g., F1-score for imbalanced data).

Regression: Choose based on impact of large errors (e.g., RMSE for penalising large deviations).


Questions and
Answers