Evaluation Metrics

Evaluation Metrics for Supervised Algorithms:

Evaluation Metrics for Classification Algorithms

1. Accuracy
2. Confusion Matrix
3. Precision and Recall
4. F1 Score

Evaluation Metrics for Regression Algorithms

1. Mean Absolute Error (MAE) and Mean Squared Error (MSE)
2. R-squared (R^2)

Evaluation Metrics for Unsupervised Algorithms

1. Silhouette Score
2. Dunn Index

Evaluation Metrics for Classification Algorithms

1. Accuracy

Accuracy is a fundamental metric that tells you how often your model makes correct predictions.

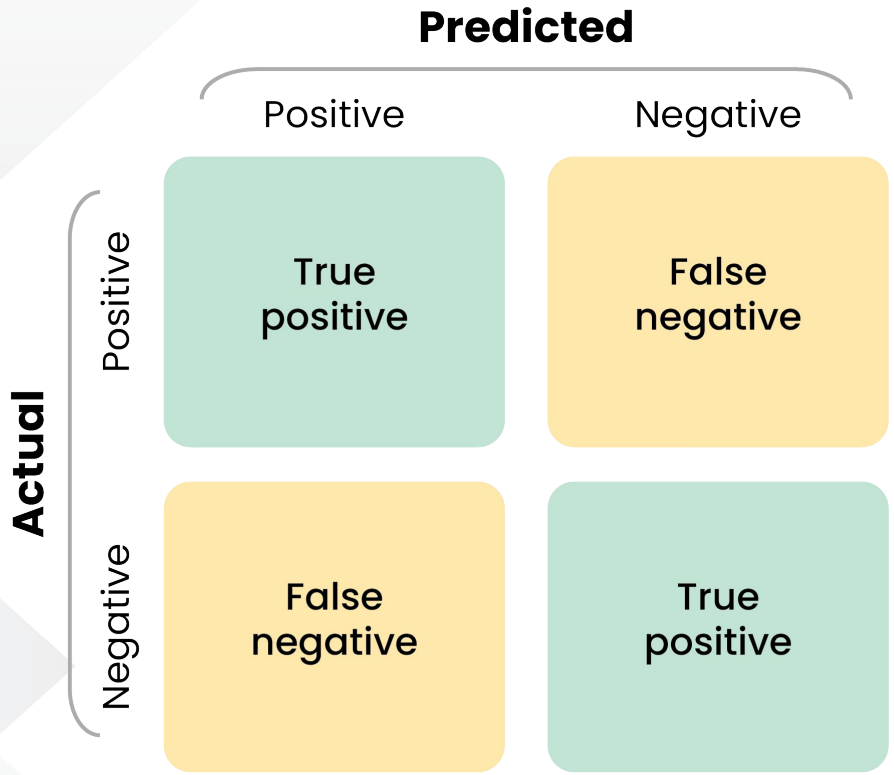
In a product recommendation system, accuracy would measure how often the suggested items align with the customer's preferences. For instance, if 90 out of 100 recommendations were relevant, the model's accuracy would be 90%.

It can be formulated as:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ number\ of\ predictions}$$

Alt text: *Product recommendation system*

2. Confusion Matrix



Alt text: *Confusion matrix*

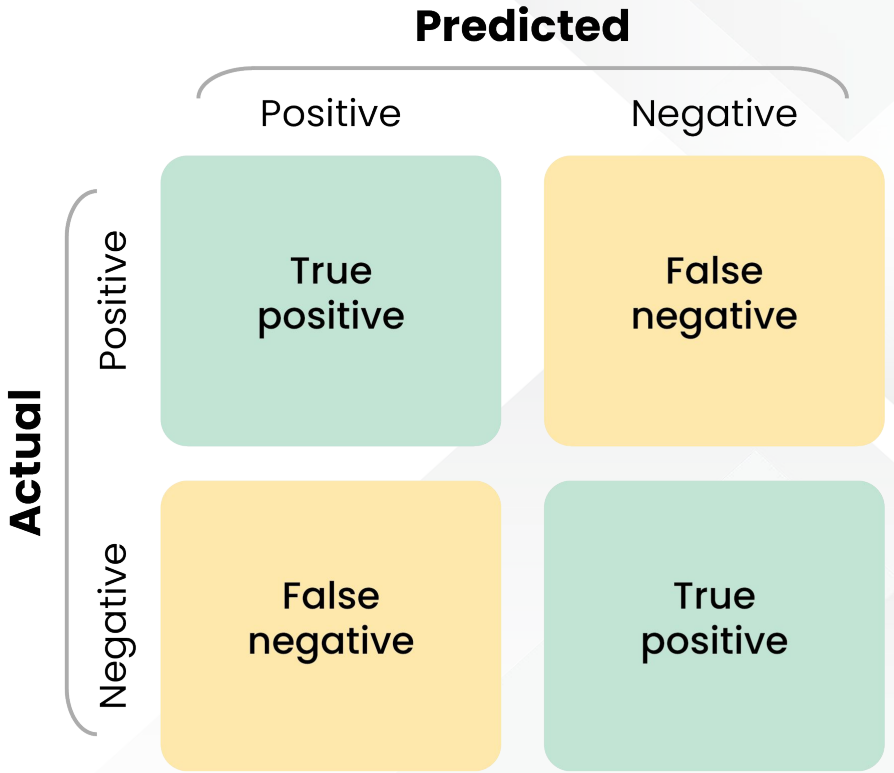
It is recommended not to use the Accuracy measure when the target variable majorly belongs to one class.

A confusion matrix helps visualise the performance of classification models. It presents a breakdown of true positives, true negatives, false positives, and false negatives. This helps understand where the model struggles.

2. Confusion Matrix – Example

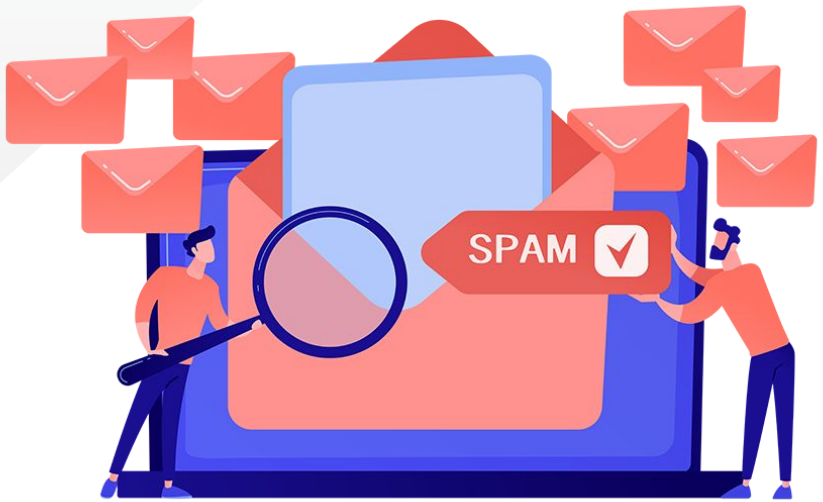
Imagine a machine learning model used in self-driving cars to detect pedestrians. A confusion matrix would show where the model fails to detect pedestrians or incorrectly identifies objects as pedestrians.

1. **True Positive(TP) is–** prediction outcome is true, and is true in reality, also.
2. **True Negative(TN) is –** prediction outcome is false, and is false in reality, also.
3. **False Positive(FP) –** prediction outcomes are true, but false in actuality.
4. **False Negative(FN):** predictions are false, and are true in actuality.



Alt text: *Confusion matrix*

3. Precision and Recall



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

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

Alt text: *Spam filtering system*

For tasks like fraud detection or spam filtering, precision and recall are crucial. Precision measures how many positive predictions are correct, while recall shows how many true positives were identified. In fraud detection, high precision means flagged transactions are truly fraudulent, and high recall means most fraud cases are caught.

Example: In a spam filter, high precision avoids mislabelling legitimate emails as spam, while high recall ensures most spam emails are detected.

4. F1 Score

The F1 score is particularly useful when precision and recall are equally important. It is the **harmonic mean** of precision and recall, giving a balanced view of the model's performance in tasks where both metrics need consideration.

In medical diagnosis, both false positives and false negatives can be costly, so F1 score would be an ideal metric to evaluate a model predicting disease outcomes.

It is calculated as:

$$F1 - score = 2 * \frac{precision * recall}{precision + recall}$$

Alt text: *F1 score*

Evaluation Metrics for Regression Algorithms

1. MAE and MSE

MAE and MSE are key metrics for regression tasks, where the goal is to predict a continuous outcome, such as house prices or stock market trends. MAE gives the average error in simple terms, while MSE penalises larger errors more harshly.

Suppose you are building a house price prediction model. If the model predicts $250,000 for a house that actually costs $300,000, both MAE and MSE will reflect the difference, helping you improve the model's predictions.

$$MAE = 1/N \sum |Y - Y'|$$

$$MSE = 1/N \sum (Y - Y')^2$$

Y –Actual outcome, Y'– predicted outcome, N – total number of data points.

Alt text: *House price prediction system*

2. R-squared (R^2)

Alt text: *House pricing model*

R^2 indicates how well the model explains the variability of the target variable. A value closer to 1 suggests a better model fit.

Imagine in a house pricing model, R^2 would show how well the independent variables (e.g., number of rooms, location) explain the variability in house prices.

$$R^{2+} = 1 - \frac{MSE\ (Model)}{MSE\ (Baseline)}$$

Evaluation Metrics for Unsupervised Algorithms

1. Silhouette Score

Here is the formula to calculate the silhouette score:

$$\text{Formula: } s = \frac{b-a}{max(a,b)}$$

Where,

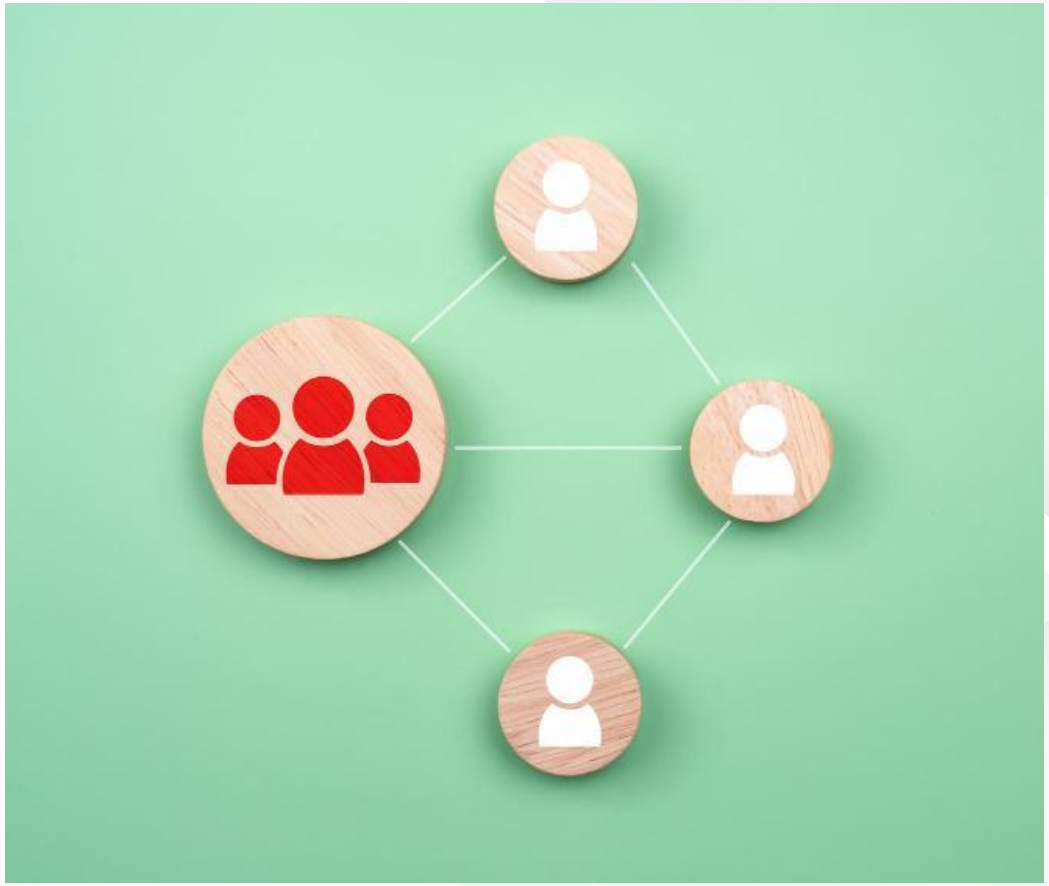
a: Average intra-cluster distance (how close a point is to its own cluster).

b: Average nearest-cluster distance (how far a point is from points in the nearest different cluster).

The silhouette score, ranging from -1 to 1, measures how well data points fit within their clusters compared to others, with higher scores indicating distinct clusters.

1. Silhouette Score – Example

For instance, a silhouette score of 0.75 for clustered customer purchase behaviours suggests that customers are well-separated and more similar within their group than to those in other groups.



Alt text: *Customer group*

2. Dunn Index

The Dunn Index is a metric used to evaluate clustering algorithms by measuring the ratio of the smallest inter-cluster distance to the largest intra-cluster distance. A higher Dunn Index indicates better clustering.

$$D = \frac{min(inter\text{-}cluster\ distance)}{max(intra\text{-}cluster\ distance)}$$

For example, in a dataset of customer segments, a higher Dunn Index indicates well-separated, compact clusters of similar customers.

Alt text: *Similar customers*